Exploring the Adversarial Vulnerabilities of Vision-Language-Action Models in Robotics

Supplementary Material

SUMMARY OF THE APPENDIX

This appendix contains additional experimental results and discussions of our work, organized as:

- §S1 includes additional related work in for robotic system safety.
- §S2 includes additional simulation results for **black-box attacks** with different VLA-based models.
- §S3 outlines the **experiment details**.
- §S4 provides **targeted attack supply results**, investigating TMA across various action magnitudes and patch sizes.
- §S5 explains tasks adopted for **real-world experiments**.
- §S6 outlines the parameter configurations used in **robustness evaluations**.
- §S7 studies failure cases, providing an in-depth analysis of the limitations of the proposed methods.
- §S8 supplies additional information on **diagnostic experiments**.
- §S9 visualizes **more qualitative results**.
- §S10 provides discussions on **future directions**, highlighting potential areas for further research.

S1. Human-Robot Interaction Safety

Current human-robot interaction risks include physical discomfort or injury [4, 5, 22]. Considering these risks, ensuring safety during interactions becomes the top priority as many literature outlined [24, 39, 67]. Existing robotic safety strategies have primarily concentrated on mitigating naturally occurring hazards [39]. These strategies can be generally categorized into pre-collision or post-collision interventions based on the timing of manage hazards [39]. Pre-collision approaches aim to prevent severe hazards by implementing physical constraints, such as controlling force and speed [28, 30], defining safety zones [68] and predicting human actions to avoid potentially dangerous robot movements [35, 36, 50]. Post-collision strategies, on the other hand, focus on detection and response after a harmful action has occurred [20, 48]. While there is substantial research on robotic safety, the integration of AI into robotic policies poses new challenges, where attackers can manipulate AI-driven robot and execute malicious actions. Such threats introduce additional layers of risk and complexity. In this paper, we primarily focus on adversarial threats.

S2. Black-box Attack Results

This section examines the black-box settings of our proposed attacks, focusing on evaluating the transferability of the proposed patch attack. To this end, we include another VLA-based model in our experiment: **LLaRA** [42].

Setup. For the LLaRA black-box experiment, we generate adversarial patch from simulation (*i.e.* openvla-7B-libero-long [34] model with LIBERO Long [44] dataset) and physcial (*i.e.* openvla-7B [34] model with BridgeData V2 [69] dataset). These patches are then pasted at the top-left corner of the image in the VIMA [32] scenario.

Table S1. **LLaRA [42] black-box results.** We report failure rates (%) in VIMA [32] scenario. Simulation and physical represent the same experiment setup as provided in §4.2. The FR (↑) is highlighted in **best** and second best for each difficulty.

			Task Difficulty			
Setup	Objective	DoF	L1	L2	L3	L4
LLaRA (D-inBC + Aux (D) + Oracle)			10.0	11.9	20.8	66.2
Simulation	Untargeted	DoF_1	15.8	20.8	**29.2**	67.5
	Untargeted	DoF_{1-3}	15.8	16.5	27.5	70.0
	UADA	DoF_1	15.0	18.8	27.1	71.3
	UPA	DoF_{1-3}	15.9	19.1	28.8	71.1
Physical	Untargeted	DoF_1	13.8	23.5	26.2	**76.4**
	Untargeted	DoF_{1-3}	14.2	26.4	27.6	74.7
	UADA	DoF_1	**16.3**	**27.2**	28.5	76.1
	UPA	DoF_{1-3}	12.7	25.6	28.7	72.1

Results. Across the four task difficulty tasks (L1-L4) in VIMA [32], adversarial patches significantly increase the failure rate (See Tab. S1) compared to the benign result of LLaRA (D-inBC + Aux (D) + Oracle). For L1, the UPA objective under the simulation setting increases the failure rate by 6.3% (16.3% $v.s.$ 10.0%). For L2, the adversarial patch generated with Untargeted objective under physical setup achieves the highest failure rate, increasing the faliure rate by 15.3% (27.2% $v.s.$ 11.9%). For L3, Untargeted Action Discrepancy Attack (UADA) under the physical setting demonstrates strong effectiveness, increasing the failure rate by 8.4% (29.2% $v.s.$ 20.8%). Finally, for L4, the physical setting with the Untargeted objective effectively increases the failure rate by 10.2% (76.4% $v.s.$ 66.2%).

S3. Experiment Details

Attacking Details: Following [34], we apply scaling and normalization for data preprocessing. The initial learning rate is set to $2e^{-3}$, with the AdamW optimizer [47] with a cosine annealing scheduler, including 20 warm-up iterations. Training is conducted over $2e^3$ iterations with a batch size of 6 and 50 inner-loop steps (see Algorithm 1). The transformation parameters ϕ, ψ in Eq. 9 are set to 30 and 0.2 to simulate real-world perspective change. For all three objectives, considering the trade-off between stealth and performance observed in §S4, we set the patch size to 5%. For the hyperparameters α, β in Eq. 5, We conduct grid search and tune α from $[0.2, 0.4, 0.6, 0.8]$ and β from $[0.8, 0.6, 0.4, 0.2]$. We select $\alpha = 0.8$ and $\beta = 0.2$ based on the validation set performance. Ablation studies $w.r.t.$ α and β are presented in §S8.

Evaluation Details: To rigorously assess the effectiveness of attacks and minimize task failures arising from random placement of adversarial patches that obscure critical task-relevant objects, we identify base left-corner locations for patch placement specific to each of the four tasks in LIBERO [44]. For the Spatial task, the base point is set as $(120, 160)$. For the Object task, the base point is $(30, 150)$. Similarly, the base point is $(15, 158)$ for the Goal task and $(5, 160)$ for the Long task. Additionally, to enhance reproducibility by mitigating randomness, patches remain untransformed during evaluation. This approach ensures a reliable assessment of adversarial patches' impact on task performance.

S4. Targeted Attack supply Results

To further explore the action manipulation capabilities of TMA, we conduct experiments targeting varying magnitudes to assess its effectiveness across different scenarios.

Setup. We conduct experiments attack at DoF$_1$ and utilize various action magnitudes (*i.e.* $0.5, -0.5, 1.0, -1.0$) as the attack targets under simulation and physical settings.

Table S2. **Different Magnitude results.** Average failure rate (%) is reported in 4 LIBERO simulation tasks. The initial AFR across the four tasks is 23.5%.

Target Action	Metric	Action Magnitude 0.5	 -0.5	 1.0	 -1.0
Simulation	L1	0.072	0.049	0.132	0.107
	AFR	82.5	87.8	78.4	74.2
Physical	L1	0.055	0.071	0.103	0.121
	AFR	62.5	68.8	63.8	58.3

Results. The results are presented in Tab. S2. In the simulation setup, the highest AFR (87.8%) is achieved at a target magnitude of -0.5, representing a substantial increase of 64.3% compared to the initial AFR (87.8% $v.s.$ 23.5%). Similarly, in the physical setup, the target magnitude of -0.5 yields the highest AFR of 68.8%. These findings highlight the superior manipulation capability and disruptive effectiveness of TMA.

S5. Details of Real-world Experiments

We design four tasks to evaluate the impact of adversarial samples on task failure rates in real-world scenarios. The specific tasks include "put the carrot on the plate", "put the corn on the plate", "put the carrot into the bowl", and "flip the pot upright". To fairly evaluate the performance of our attack, we only consider tasks where the VLA model operates successfully, excluding failure cases. We then introduce the adversarial patch to assess its impact. This leads to an average task failure rate of 43%, underscoring the significant disruptive potential of adversarial samples on robotic tasks.

S6. More Robustness Evaluation Details

In this section, we provide a detailed parameter setting of conducted robustness evaluation in §4.5. Specifically, we assess the effectiveness of four commonly employed defense techniques, including JPEG compression [18], bit-depth reduction [78], median blur [78], and Gaussian noise [80]. **JPEG compression** applies compression algorithms to the input images prior to feeding them into the depth estimation network, aiming to disrupt adversarial patterns. We test compression quality levels ranging from 50 to 10, with lower quality levels corresponding to higher compression rates. **Gaussian noise** introduces zero-mean Gaussian noise to the input image, leveraging its randomness to counteract the structured nature of adversarial perturbations. The standard deviation of the noise varies from 0.01 to 0.1, with higher values introducing stronger noise. **Median blur** smooths the image by replacing each pixel value with the median of its surrounding pixels, using square kernel sizes from 3 to 9; larger kernel sizes produce stronger smoothing effects. **Bit-depth reduction** remaps the standard 8-bit depth of RGB channels to smaller bit depths, reducing the color space and potentially disrupting adversarial perturbations. We evaluate cases with bit depths ranging from 6 bits to 3 bits. These defense techniques are evaluated to understand the robustness of our attack (results see in Fig. 5).

S7. Failure Case Analysis

This section analyzes the attack failure cases to gain deeper insights into robot manipulation and highlight the critical role of DoF targets on task failure.

We analyze most of attack failure cases and find that certain DoFs are redundant in task execution. In Fig. S4, the adversarial patch targeting DoF$_4$ with a value of 0 fails to disrupt the execution of the task. This failure can be attributed to the fact that DoF$_4$ controls the orientation along the x-axis, which is redundant for grasping objects such as bowls in the context of this task. As a result, attacks targeting **redundant DoF of the task** [38] are less likely to disrupt successful execution effectively. This observation

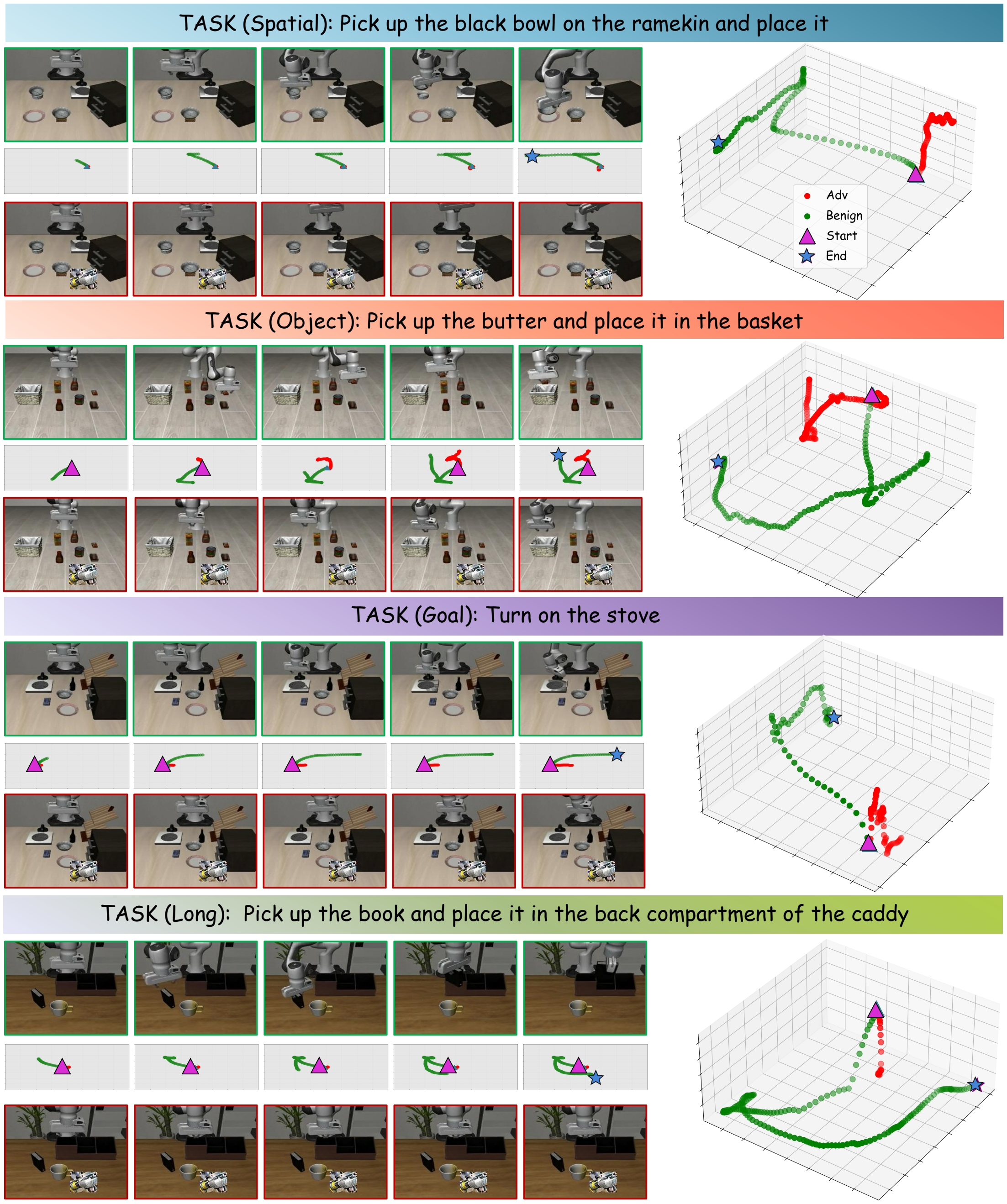

Figure S1. **UADA Qualitative Results** The figure illustrates the 3D and 2D trajectories for both benign ● and adversarial ● scenarios, highlighting the impact of the adversarial patch at each time step. We visualize the start point, marked as ▲, and the endpoint, marked as ★.

underscores the importance of task-specific considerations when designing adversarial attacks on robotic systems.

S8. More Diagnostic Experiment Details

We perform diagnostic experiments to assess the influence of the hyperparameters α and β in UPA. As shown in Tab. S3, when $\alpha = 0.8$ and $\beta = 0.2$, UPA achieves the highest average score of 93.4 across the four LIBERO [44] tasks among all tested configurations. This result indicates that

Table S3. **Parameter Diagnostic Results.** Four distinct α and β combination results. We report the average failure rate of four tasks in LIBERO [44].

Setup	α	β	AFR (%)
Simulation	0.2	0.8	86.5
	0.4	0.6	90.5
	0.6	0.4	89.4
	0.8	0.2	93.4

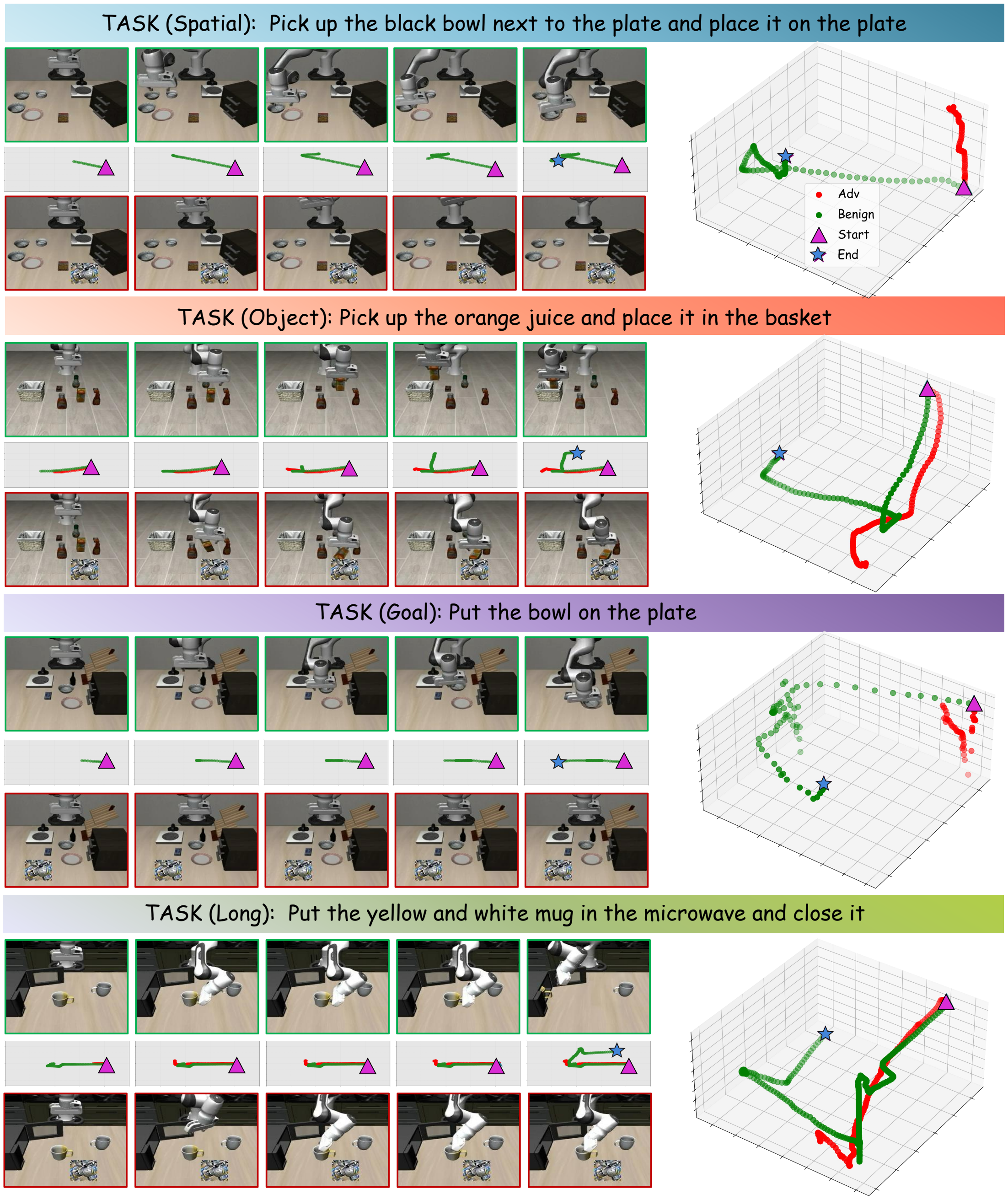

Figure S2. **UPA Qualitative Results** The figure illustrates the 3D and 2D trajectories for both benign ● and adversarial ● scenarios, highlighting the impact of the adversarial patch at each time step. We visualize the start point, marked as ▲, and the endpoint, marked as ★

this specific combination optimally balances the trade-offs between direction and magnitude adjustments governed by α and β, establishing it as the most effective choice.

S9. More Qualitative Results

This section presents additional qualitative results for each of the three attack objectives, complementing results in §4.3 and offering deeper insights into the effectiveness of our adversarial attacks. Specifically, the results of UADA, UPA, and TMA are shown in Fig. S1, S2, and S3, respectively. These qualitative results reveal significant deviations in the adversarial trajectories of the proposed methods across all four LIBERO tasks compared to the benign trajectories. Specifically, **UADA** induces substantial action discrepancies while diverging sharply from the benign trajectory, highlighting its disruptive impact. For **UPA**, a significant position deviation is observed on the spatial task [44], where the adversarial trajectory deviates significantly from the benign trajectory,

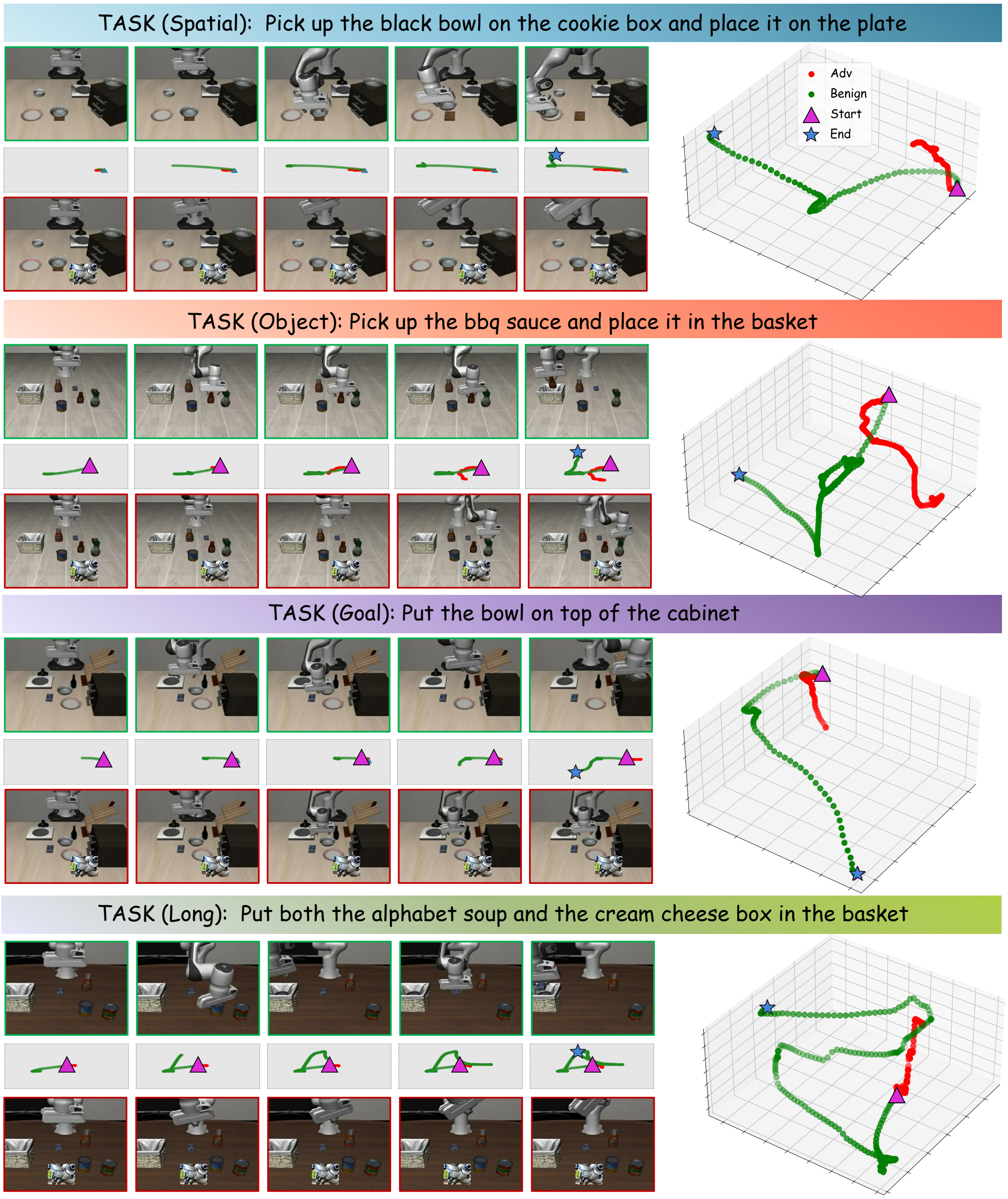
TASK (Spatial): Pick up the black bowl on the cookie box and place it on the plate
Adv
Benign
Start
End
TASK (Object): Pick up the bbq sauce and place it in the basket
TASK (Goal): Put the bowl on top of the cabinet
TASK (Long): Put both the alphabet soup and the cream cheese box in the basket

Figure S3. **TMA Qualitative Results** The figure illustrates the 3D and 2D trajectories for both benign ● and adversarial ● scenarios, highlighting the impact of the adversarial patch at each time step. We visualize the start point, marked as ▲, and the endpoint, marked as ★

ultimately causing task failure. In contrast, the **TMA** exhibits a smaller action amplitude, successfully manipulating the robot's behavior and resulting in task failure. These findings highlight the distinct impacts and underlying mechanisms of the three attack methods.

S10. Future direction

Future research on attacks against VLA-based models can potentially be conducted on two key objectives: enhancing camouflage and ensuring practical feasibility. Improving camouflage involves reducing detection probability by generating adversarial patches that seamlessly integrate with the environment, leveraging natural patterns and context-aware design. Furthermore, future efforts should avoid targeting redundant DoF and instead concentrate on critical components that are most likely to disrupt task performance. This can be achieved by leveraging task-specific knowledge and employing advanced optimization techniques to maximize the effectiveness of such attacks while aligning with real-world physical constraints.

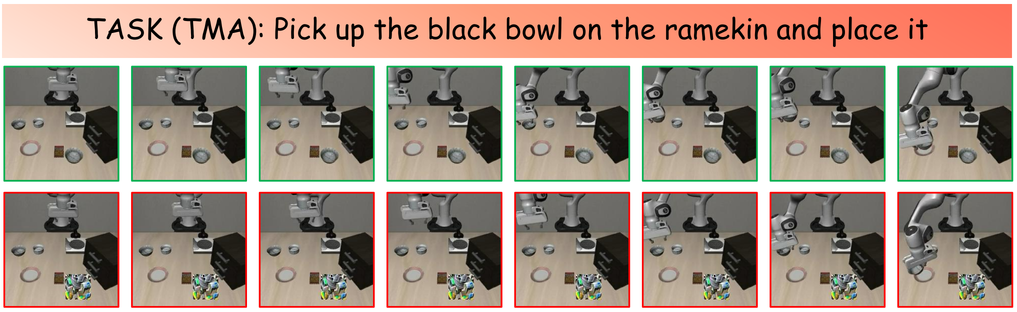
TASK (TMA): Pick up the black bowl on the ramekin and place it

Figure S4. **Redundant DoF in a Failure Case.** The first row is the benign scenario, while the second row illustrates the adversarial scenario. In the adversarial scenario, the adversarial patch is generated by targeting DoF_4 within the simulation setup. In this task, orientation (*i.e*. DoF_{4-6}) is identified as a redundant DoF since the task completion does not require any changes in orientation. As a result, the DoF_4 attack, which focuses on orientation, fails to disrupt the task's execution.