ACROSS VIEWPOINTS AND TIME: TRAJECTORY-GENERALIZABLE PHYSICAL ATTACKS ON WORLD MODELS AND VIDEO UNDERSTANDING MODELS

Anonymous authors
Paper under double-blind review

ABSTRACT

World models and video understanding models are increasingly deployed in dynamic visual environments, yet their robustness to physical adversarial threats remains largely underexplored. We study whether a single physical adversarial texture can mislead the victim model across different real-world scenarios where observing viewpoints vary along uncontrolled object–camera trajectories. We identify that heterogeneous video semantics and trajectories induce poorly aligned gradients for optimizing the adversarial texture. To address that, we propose the Trajectory-aware adversarial Alignment and Refinement Attack (TARA), which incorporates trajectory-aware optimization to capture the evolving geometric variations of a printed texture under real-world scenarios. To improve cross-instance optimization compatibility, TARA incorporates trajectory-aware contrastive alignment to encourage attacked representations from diverse video–trajectory instances to approach a shared adversarial region and then performs trajectory-aware task refinement to translate the aligned representation into task-level prediction errors. For reproducible evaluation, we develop a geometry-grounded tracking pipeline that provides physically plausible adversarial regions with temporally consistent trajectories on EPIC-KITCHENS-100 and Something-Something V2. We further collect real-world print-and-recapture videos to validate attack effectiveness under physical deployment conditions. Together, the annotated benchmarks and real-world captured data form a reproducible testbed for future evaluation. Notably, our results reveal that a single printable adversarial texture can persistently disrupt world models and video understanding models under physical conditions, highlighting the fundamental security challenge.

1 INTRODUCTION

World models (Assran et al., 2025; Wu et al., 2024; Zhou et al., 2025; Zhang et al., 2026b;a) and video foundation models (Tong et al., 2022; Wang et al., 2023; 2024; Zhao et al., 2024) are advancing visual understanding and prediction in dynamic environments. Their deployment in real-world systems raises concerns about vulnerability to physical adversarial inputs. Adversaries in video understanding and action anticipation have been extensively studied in the digital domain (Wei et al., 2019; 2022; Xia et al., 2024; Ding et al., 2026; Lu et al., 2026a). However, attacks applied directly to model inputs do not establish whether similar vulnerabilities exist under physical deployment (Brown et al., 2017; Athalye et al., 2018; Eykholt et al., 2018). In physical settings, an attacker can design and attach an adversarial texture but cannot control how it is subsequently observed. As object–camera trajectories evolve, the attached texture undergoes substantial variations in viewpoint, scale, and visibility, potentially limiting its attack effectiveness. This raises a critical concern about the practical vulnerability of video understanding and anticipation models to physical adversarial attacks under uncontrolled observation conditions in real-world deployments.

Previous physical attacks consider constraints such as fabrication, camera acquisition, and geometric variation (Brown et al., 2017; Athalye et al., 2018; Eykholt et al., 2018), with subsequent work extending to video recognition, tracking, and embodied systems (Pony et al., 2021; Du et al., 2022; Long et al., 2024; Chen et al., 2024; Li et al., 2025; Wang et al., 2025). While these studies increasingly account for variations in how adversarial patterns are observed, their effectiveness is often assessed under predefined conditions, which may not fully reflect the real-world variation (Hingun

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.



Figure 1: Illustration of physical adversarial texture on action anticipation under various trajectories.

et al., 2023). Specifically, the ability of a single physical adversarial texture to generalize jointly across heterogeneous videos and unseen object–camera trajectories remains insufficiently explored.

To investigate this setting, as shown in Figure 1, we consider a physical attack where a single printable texture is optimized before deployment and must remain effective across unseen object–camera trajectories beyond the attacker's control. However, jointly optimizing the texture across heterogeneous videos and observation trajectories can induce conflicting gradients, limiting the attack effectiveness. To solve that, we propose a Trajectory-aware adversarial Alignment and Refinement Attack(TARA), which optimizes a single texture across diverse video–trajectory instances to account for evolving physical observation conditions. TARA first employs trajectory-aware contrastive alignment to encourage attacked representations across heterogeneous instances to approach a shared adversarial region, improving cross-instance compatibility. The resulting texture is then refined along each video's trajectory to induce task-level prediction errors. Notably, evaluating physical attacks on video models requires temporally coherent surface trajectories that capture how a fixed texture is observed over time, which existing image-level benchmarks (Braunegg et al., 2020; Hingun et al., 2023) do not provide. To support that, we develop a geometry-grounded tracking pipeline that identifies feasible attachment regions and generates temporally consistent homographies and visibility masks for Something-Something V2 (Goyal et al., 2017) and EPIC-KITCHENS-100 (Damen et al., 2022) datasets. Meanwhile, we further collect real-world print-and-recapture videos, in which learned textures are attached to real objects and evaluated under physical conditions. We curate surface masks and trajectory annotations for both simulated and real-world captured videos, establishing a reproducible evaluation benchmark for future research.

We evaluate TARA across multiple world models and video understanding models on recognition and future prediction tasks. Our results demonstrate that TARA maintains its attack effectiveness across unseen video–trajectory instances and real-world print-and-recapture scenarios, highlighting the vulnerabilities of those models in practical deployment. Overall, our contributions are:

- We investigate a practical physical attack for world models and video understanding models, where a single printable texture must generalize across heterogeneous videos and object–camera trajectories after deployment.
- We identify conflicting optimization gradients across heterogeneous video–trajectory instances as a key challenge in learning a shared texture. To address this, we propose TARA, which integrates trajectory-aware optimization with cross-instance contrastive alignment and task-level refinement to improve effectiveness under dynamic physical conditions.
- We establish a geometry-grounded benchmark spanning Something-Something V2, EPIC-KITCHENS-100, and real-world print-and-recapture videos, providing temporally consistent trajectories with homographies and visibility masks for evaluating physical attacks.

2 Preliminaries

Attack setting. Denote F_θ as the attacked model and $\boldsymbol{X}_i = \{\boldsymbol{x}_{it}\}_{t=1}^{T_i}$ as the video clip with ground truth y_i. We assume a white-box adversary who optimizes a single printable adversarial texture $\boldsymbol{\delta}$ to disrupt model predictions across heterogeneous videos and observation trajectories. Practically, we assume the texture $\boldsymbol{\delta}$ is attached to a fixed object surface, while its image-space appearance evolves

with the underlying object–camera trajectory and scene visibility, which are beyond the adversary's control after deployment

Homography and visibility mask. For frame t of video i, we characterize the observation state of the attached texture by a homography $\boldsymbol{H}_{it}$ and a visibility mask $\boldsymbol{M}_{it}$. As illustrated in Figure 2, $\boldsymbol{H}_{it}$ maps the adversarial texture from its local coordinates to the image plane, accounting for changes in position, scale, orientation, and perspective distortion. The visibility mask $\boldsymbol{M}_{it} \in [0,1]^{h\times w}$ specifies the observable pixels of the projected texture after occlusion. We define the *observation trajectory* of video i as $\gamma_i = \{(\boldsymbol{H}_{it}, \boldsymbol{M}_{it})\}_{t=1}^{T_i}$, capturing the temporally coupled evolution of texture homography and visibility. Let $\mathcal{W}_{\boldsymbol{H}_{it}}$ be the differentiable projective matrix. Given the adversarial texture $\boldsymbol{\delta}$, the attacked frame–trajectory instance $\tilde{\boldsymbol{x}}_{it}^{\boldsymbol{\delta}}$ and video–trajectory instance $\tilde{\boldsymbol{X}}_i(\boldsymbol{\delta};\gamma_i)$ are constructed as:

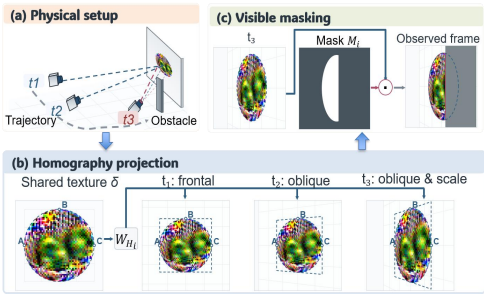


Figure 2: Illustration of the observation trajectory of adversarial texture.

$$\tilde{\boldsymbol{x}}_{it}^{\boldsymbol{\delta}} = (1-\boldsymbol{M}_{it})\odot \boldsymbol{x}_{it} + \boldsymbol{M}_{it}\odot \mathcal{W}_{\boldsymbol{H}_{it}}(\boldsymbol{\delta}), \qquad \tilde{\boldsymbol{X}}_i(\boldsymbol{\delta};\gamma_i) = \{\tilde{\boldsymbol{x}}_{it}^{\boldsymbol{\delta}}\}_{t=1}^{T_i}. \tag{1}$$

3 Methodology

3.1 Trajectory-Aware Adversarial Alignment and Refinement Attack

To account for the evolving physical observation conditions, we optimize a shared adversarial texture across diverse video–trajectory instances, where each instance captures the temporally coupled variations in texture homography and visibility. Given N training instances, a direct optimization aims to find the $\boldsymbol{\delta}$ that maximize a task-specific objective $\mathcal{L}_{\text{dir}}$ across all instances:

$$\boldsymbol{\delta}^{\star} = \arg\max_{\boldsymbol{\delta}} \frac{1}{N}\sum_{i=1}^{N} \mathcal{L}_{\text{dir}}\left(F_\theta\left(\tilde{\boldsymbol{X}}_i(\boldsymbol{\delta};\gamma_i)\right), y_i\right). \tag{2}$$

However, jointly optimizing the texture across heterogeneous video–trajectory instances can produce conflicting gradients, limiting the effectiveness of shared texture updates. Figure 3(a) and (b) report the average cross-instance update benefits and pairwise gradient similarities, reporting statistics averaged over all training clips and optimization iterations. The results reveal limited cross-instance update benefits and substantial gradient disagreement under direct optimization. These motivate us to propose Trajectory-aware adversarial Alignment and Refinement Attack (TARA), which first aligns adversarial representations across video–trajectory instances to improve optimization compatibility, followed by task refinement to induce prediction errors.

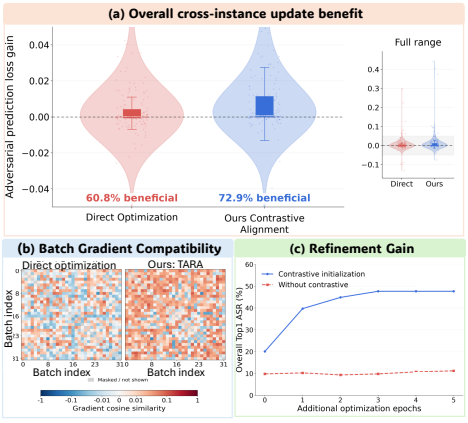


Figure 3: Analysis of trajectory-aware contrastive alignment.

Trajectory-Aware Contrastive Alignment. As shown in Figure 4(a) and (b), we pair each training video $\boldsymbol{X}_i$ with its observation trajectory γ_i to construct the attacked video–trajectory instance $\tilde{\boldsymbol{X}}_i(\boldsymbol{\delta};\gamma_i)$. We first extract its attacked representation $\boldsymbol{z}_i^a$ and the corresponding clean representation $\boldsymbol{z}_i^c$. To encourage compatible adversarial representations across heterogeneous video–trajectory instances, for a batch of B instances, we minimize the following contrastive loss $\mathcal{L}_{\text{con}}$:

$$\mathcal{L}_{\text{con}} = \frac{1}{B(B-1)}\sum_{i\neq j} \text{softplus}\left(\frac{s(\boldsymbol{z}_i^a, \boldsymbol{z}_i^c) - s(\boldsymbol{z}_i^a, \boldsymbol{z}_j^a)}{\tau}\right), \tag{3}$$

where $s(\cdot,\cdot)$ denotes cosine similarity and τ is a temperature parameter. Minimizing Equation 3 encourages attacked representations from different video–trajectory instances to approach a shared

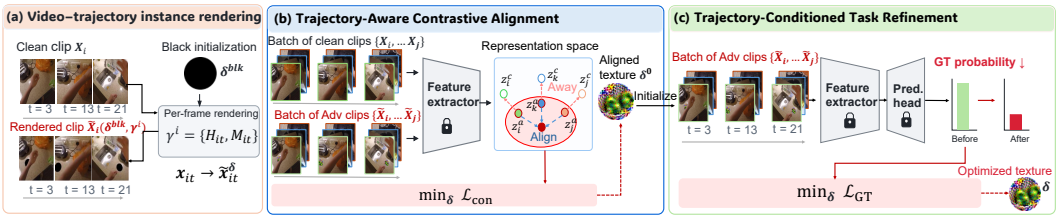


Figure 4: Overall framework of TARA. A black texture is rendered along the video-trajectory instance and optimized through trajectory-aware contrastive alignment and task refinement.

adversarial region while remaining separated from their clean representations. Optimization of $\mathcal{L}_{\text{con}}$ yields the aligned adversarial texture $\boldsymbol{\delta}^{(0)}$, which serves as the initialization for task refinement. Importantly, we find this alignment greatly improves cross-instance optimization compatibility. As indicated in Figure 3 (a) and (b), the contrastive alignment improves both average cross-instance update benefit and batch gradient agreement. Moreover, the resulting texture also provides a stronger initialization that brings great performance gain under the same refinement, shown in Figure 3 (c).

Trajectory-Aware Task Refinement. While contrastive alignment encourages shared adversarial representations across video–trajectory instances, it does not explicitly optimize task-level prediction errors. We therefore refine the aligned texture $\boldsymbol{\delta}^{(0)}$ along each training video's observation trajectory by minimizing:

$$\mathcal{L}_{\text{ref}} = \sum_{i=1}^{B} w_i \log p_\theta \Big(y_i \mid \tilde{\boldsymbol{X}}_i(\boldsymbol{\delta}; \gamma_i) \Big), \tag{4}$$

where w_i is a trajectory-aware weight inversely related to the projected visible texture area, computed from the frame-wise homographies and visibility masks. By assigning larger weights to instances with smaller observable texture regions, this weighting balances their contributions to shared texture optimization under heterogeneous observation conditions. More details of the weighting scheme are provided in the Appendix A.1.

3.2 Geometry-Grounded Trajectory Construction

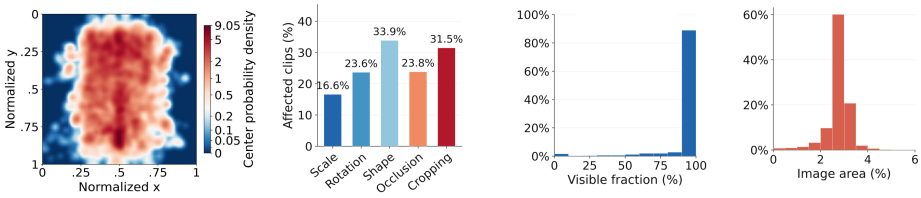
To support trajectory-aware optimization and reproducible evaluation, we construct geometry-grounded texture trajectories on Something-Something V2 (SSV2) (Goyal et al., 2017) and EPIC-KITCHENS-100 (EK100) (Damen et al., 2022). For each video clip, we manually identify a feasible texture attachment region on an approximately planar object surface. The region remains fixed on the object, while its image-space projection evolves with the underlying object–camera motion.

Adversarial texture homography and visibility. Starting from the selected attachment region in a reference frame, we sample anchor points within the region and track them across frames using TAPNext++ (Jung et al., 2026; Zholus et al., 2025). We estimate a reference-to-frame homography from the tracked correspondences using RANSAC (Fischler & Bolles, 1981), which is then composed with the fixed texture-to-reference mapping to obtain the texture-to-frame homography $\boldsymbol{H}_{it}$. When point tracking becomes unreliable, we leverage dense correspondences from RoMa (Edstedt et al., 2024) with nearby validated frames to recover geometrically consistent projections. The estimated homography determines the projected texture region in each frame, from which we further estimate its visible portion. Specifically, GroundingDINO (Liu et al., 2024) detects supporting surfaces and foreground occluders, such as hands and arms, while SAM2 (Ravi et al., 2025) converts the detections into pixel-level masks. We combine the projected support with image boundaries and surface masks, and remove foreground-occluded pixels to obtain the visibility mask $\boldsymbol{M}_{it}$. The final sequence $\gamma_i = \{(\boldsymbol{H}_{it}, \boldsymbol{M}_{it})\}_{t=1}^{T_i}$ defines the surface-observation trajectory and is further manually reviewed for consistency, drift, and occlusion quality. Further implementation and quality-control details are provided in the Appendix A.7.

Real-world print-and-recapture evaluation. We construct a real-world evaluation set by printing the optimized texture at a fixed physical size and attaching it to an approximately planar surface. For each sample, we record a clean reference video, an adversarial video with the optimized texture, and a calibration video with a uniform white patch of the same size and attachment location. The white-patch recordings provide a high-contrast reference for estimating frame-wise homographies

Table 1: Statistics of the geometry-grounded trajectory benchmarks.

Dataset	Task	Clips	Texture Train/Eval.	Annotated Frames	Homography Coverage	Mask Rate	Texture Area	Texture Motion
EK100	Anticipation	605	356 / 249	19,251	73.94%	7.56%	1.95%	20.65%
SSV2	Recognition	328	184 / 144	10,204	89.78%	9.70%	2.37%	7.66%
Real-world	Recog. / Antic.	348	0 / 348	8,645	81.27%	4.28%	2.97%	15.17%



(a) Trajectory center density (b) Texture observation variations (c) Texture visibility (d) Texture area

Figure 5: Distribution of surface-observation trajectories across the evaluation benchmarks.

and visibility masks. These annotations enable future studies to evaluate adversarial textures under identical real-world observation trajectories, ensuring reproducible comparisons. We further conduct an independent human evaluation to confirm that the underlying actions remain semantically interpretable. Together, the annotated benchmark videos and real-world captures establish a reusable and reproducible testbed for evaluating physical robustness under dynamic observation conditions.

Dataset statistics and reusable benchmark. Table 1 summarizes the overall characteristics of the constructed benchmark, comprising 605 clips (19,251 frames) from EK100, 328 clips (10,204 frames) from SSV2, and 348 real-world clips (8,645 frames). In Table 1, homography coverage measures the fraction of frames with valid surface projections; mask rate measures the frequency of occlusion within the projected attachment region; texture area denotes the average fraction of the image occupied by the projected texture; and motion range quantifies the spatial extent of its trajectory. Figure 5 further characterizes the benchmark at the frame level, illustrating the spatial distribution of projected texture centers (a), variations in texture geometry and visibility (b), and the distributions of visible texture fraction (c) and projected texture area (d). Together, these statistics characterize the diverse observation conditions captured by our benchmark. For reproducible comparison of different attack methods, by fixing γ_i, a new texture $\boldsymbol{\delta}'$ can be rendered to construct $\tilde{\boldsymbol{X}}_i(\boldsymbol{\delta}'; \gamma_i)$ under identical geometry and visibility. More details such as full benchmark construction and validation pipeline, the quality-control criteria, and metric definitions are provided in the Appendix A.7.

4 Experimental Results

4.1 Experimental Setup

Evaluation benchmarks and models. We evaluate our attack across three tasks using the geometry-grounded EK100 (Damen et al., 2022), SSV2 (Goyal et al., 2017) and our real-world captured benchmarks. The evaluated tasks include:

- **Future action anticipation** on EK100 using V-JEPA2-L (Assran et al., 2025), V-JEPA2.1-B (Mur-Labadia et al., 2026), and AVT-B (Girdhar & Grauman, 2021), and our real-world captured action anticipation set, using V-JEPA2-L and V-JEPA2.1-B.
- **Action recognition** on SSV2, using V-JEPA2-L, VideoMAE-B (Tong et al., 2022), and InternVideo2-dist-L (Wang et al., 2024), and our real-world captured action recognition set, using V-JEPA2-L and InternVideo2-dist-L.
- **Action-conditioned future frame generation** on SSV2 using DiLA (Zhang et al., 2026b).

Evaluation metrics and data splits. For action anticipation and recognition tasks, we report Top-1 attack success rate (ASR), defined as the percentage of initially correct predictions that become incorrect under attack. For future frame generation, we report PSNR and MSE against the clean target frames, where lower PSNR and higher MSE relative to clean reconstruction indicate greater attack-induced degradation. We partition the 605 EK100 clips into 356 training and 249 validation clips, and the 328 SSV2 clips into 184 and 144, respectively. The adversarial texture is optimized utilizing only the training videos and their associated surface trajectories. The held-out validation clips provide unseen samples for assessing the generalization of adversarial texture across different videos and object–camera trajectories.

Table 2: Comparison performance on EK100 future action anticipation. Top1 ASR (%, ↑) is reported on our split Train and Val data, with best and second-best results highlighted in bold and underlined.

Models	V-JEPA2-L		V-JEPA2.1-B		AVT-B		Overall	
Setting	Train	Val	Train	Val	Train	Val	Avg. Train	Avg. Val
Black texture	4.8	3.3	5.7	5.0	8.2	5.0	6.2	4.4
Random texture	5.5	4.7	8.9	5.5	7.4	6.1	7.3	5.4
EOT	<u>30.3</u>	<u>25.1</u>	<u>59.3</u>	<u>57.1</u>	32.9	25.0	<u>40.8</u>	<u>35.7</u>
Patch-Fool	6.5	9.0	46.9	27.9	28.2	18.3	27.2	18.4
UPA-RFAS	9.5	10.4	17.1	15.7	17.6	18.3	14.7	14.8
BTC-UAP	13.9	10.4	11.4	13.6	36.5	<u>28.3</u>	20.6	17.4
PhyCamo	13.9	13.7	58.3	29.3	**42.4**	**30.0**	38.2	24.3
TARA	**37.7**	**26.1**	**70.6**	**67.1**	<u>37.6</u>	**30.0**	**48.6**	**41.1**

Table 3: Comparison performance on SSV2 video action recognition. Top1 ASR (%, ↑) is reported on our split Train and Val data, with best and second-best results highlighted in bold and underlined.

Models	V-JEPA2-L		VideoMAE-B		InternVideo2		Overall	
Setting	Train	Val	Train	Val	Train	Val	Avg. Train	Avg. Val
Black texture	9.2	5.6	5.7	10.8	4.0	2.9	6.3	6.4
Random texture	7.4	4.9	1.1	1.8	2.1	0.3	3.5	2.3
EOT	<u>25.5</u>	<u>30.6</u>	14.9	<u>16.9</u>	15.8	13.2	18.7	<u>20.2</u>
Patch-Fool	12.5	8.3	13.7	<u>16.9</u>	5.1	2.2	10.4	9.1
UPA-RFAS	9.8	3.5	14.3	<u>16.9</u>	11.9	9.6	12.0	10.0
BTC-UAP	12.5	5.6	<u>15.4</u>	**17.7**	14.1	15.7	14.0	13.0
PhyCamo	17.9	12.5	<u>15.4</u>	<u>16.9</u>	<u>24.3</u>	<u>23.5</u>	<u>19.2</u>	17.6
TARA	**38.6**	**34.7**	**17.1**	**17.7**	**41.2**	**25.7**	**32.3**	**26.0**

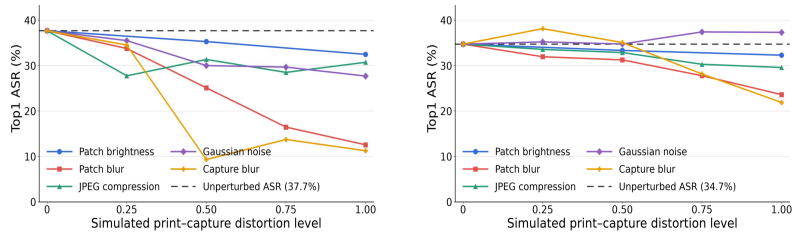
Compared methods and implementation details. We compare TARA with representative adversarial patch and physical attack methods. Specifically, we consider adversarial patch-based methods, including patch with expectation over transformation (EOT) (Brown et al., 2017), the attention-guided Patch-Fool (Fu et al., 2022), and PhyCamo (Zhang et al., 2025), which employs contrastive learning to improve the robustness of physical camouflage across viewpoints. We further include BTC-UAP (Kim et al., 2023), which focuses on temporal feature disruption, and UPA-RFAS (Lu et al., 2026b), which employs a visual feature-space objective to optimize transferable physical patches for vision-language-action models. For a fair comparison, all compared methods incorporate trajectory-aware optimization using the same annotated object–camera trajectories and geometric constraints described in Section 2. Black and random textures are included as non-optimized controls to distinguish adversarial optimization from the effects of ordinary texture insertion. For each victim model, all methods optimize a universal circular texture with a radius of 24, using the same training and evaluation splits. We preserve each model's native temporal sampling, spatial preprocessing, and prediction aggregation. The details of the model- and task-specific adaptations, optimization schedules and budgets, checkpoint-selection criteria, random seeds, and hardware configurations are provided in the Appendix A.6.

4.2 Main Results on geometry-grounded EK100 and SSV2

Performance on EK100 action anticipation. Table 2 compares the attack performance of different methods on EK100 future action anticipation across V-JEPA2-L, V-JEPA2.1-B, and AVT-B, with Top-1 ASR reported. In particular, TARA achieves the highest average ASR of 48.6% and 41.1% on the training and validation splits, outperforming the strongest baseline by 7.8% and 5.4%, respectively. Notably, V-JEPA2.1-B exhibits substantially higher adversarial vulnerability than V-JEPA2-L across several attack methods. Although V-JEPA2.1 incorporates dense feature modeling to capture finer-grained spatiotemporal information, its representations may also be more sensitive to persistent, localized adversarial perturbations. These findings suggest that richer spatiotemporal representations do not necessarily translate into greater robustness against persistent physical attacks. Overall, these results demonstrate TARA's effectiveness in disrupting future action anticipation across distinct video architectures and its generalization to unseen video instances under dynamic observation conditions.

Table 5: Performance on real-world captured data. The first six rows are digital projection results using recorded homographies and masks; the last two rows are physical print-and-recapture results.

Tasks	Action Anticipation		Action Recognition	
Models	V-JEPA2-L	V-JEPA2.1-B	V-JEPA2-L	InternVideo2
EOT	<u>29.7</u>	**58.8**	<u>21.2</u>	40.3
Patch-Fool	17.6	23.5	10.6	35.5
UPA-RFAS	18.9	17.6	11.8	<u>43.5</u>
BTC-UAP	14.9	17.6	12.9	24.2
PhyCamo	13.5	<u>29.4</u>	17.6	38.7
TARA	**32.4**	**58.8**	**25.8**	**50.0**
Physical EOT (printed)	28.6	\	23.1	\
Physical TARA (printed)	32.1	\	28.2	\



(a) EK100 future action anticipation (b) SSV2 action recognition

Figure 6: Performance of TARA under varying levels of simulated print-and-capture distortions.

Performance on SSV2 action recognition. Table 3 compares the attack performance of different methods on SSV2 action recognition across V-JEPA2-L, VideoMAE-B, and InternVideo2, with Top-1 ASR reported. It indicates that TARA achieves the highest average ASR of 32.3% and 26.0% on the training and validation splits, outperforming the strongest baseline by 13.1% and 5.8%, respectively. The improvements are particularly pronounced on InternVideo2, where TARA achieves 41.2% and 25.7% ASR, surpassing PhyCamo by 16.9% and 2.2%, respectively. In contrast, the performance gains on VideoMAE-B are relatively modest, with TARA achieving 17.1% and 17.7% ASR, suggesting that different video representation architectures exhibit varying susceptibility to persistent physical perturbations. Overall, TARA maintains the highest average ASR across both splits, indicating that its trajectory-aware optimization can effectively accommodate heterogeneous video representations while preserving attack effectiveness under dynamic observation conditions.

Performance on SSV2 future frame generation. Table 4 compares the attack performance of different methods on latent-action-conditioned future frame generation using DiLA, with reconstruction PSNR and MSE reported on both the training and validation splits. It indicates that TARA achieves the lowest PSNR of 11.38 and 12.66 dB and the highest MSE of 90.38 and 67.37 ($\times 10^{-3}$) on the training and validation splits, respectively, indicating substantial degradation in future frame reconstruction quality. Compared with the strongest baseline, UPA-RFAS, TARA further reduces PSNR by 3.94 and 2.92 dB while increasing MSE by 55.91 and 34.10 ($\times 10^{-3}$), respectively. These results indicate that TARA can substantially disrupt latent-action-conditioned future frame generation, extending its effectiveness beyond discriminative action recognition and anticipation to generative video prediction. Attacking objective and visualization for the generation task are provided in the Appendix A.2.

Table 4: Attack performance on SSV2 future frame generation using DiLA.

	Train		Val	
Method	PSNR↓	MSE($\times 10^{-3}$) ↑	PSNR↓	MSE($\times 10^{-3}$) ↑
Clean	17.37	24.42	17.19	24.33
Black patch	16.48	27.70	16.51	26.74
Random patch	17.10	24.88	17.12	24.37
EOT	16.30	29.96	16.25	29.03
Patch-Fool	15.94	33.83	16.41	28.48
UPA-RFAS	<u>15.32</u>	<u>34.47</u>	<u>15.58</u>	<u>33.27</u>
BTC-UAP	16.83	25.07	16.78	25.54
TARA	**11.38**	**90.38**	**12.66**	**67.37**

4.3 Real-world Results and Distortion Analysis

Performance on real-world captured data. Table 5 compares the attack performance of different methods on our real-world captured data. We first evaluate existing methods by digitally projecting their adversarial textures onto video-trajectory instances using our recorded homographies and

Table 6: Ablation study of our TARA on the EK100 dataset. Top-1 ASR (%, ↑) is reported. TA, CA, and TR denote trajectory awareness, contrastive alignment, and task refinement, respectively.

Components			V-JEPA2-L		V-JEPA2.1-B		AVT-B		Overall	
TA	CA	TR	Train	Val	Train	Val	Train	Val	Avg. Train	Avg. Val
×	✓	✓	10.4	10.9	8.0	5.7	9.4	3.3	9.3	6.6
✓	×	✓	32.9	22.7	68.6	58.3	**40.0**	28.3	47.2	36.4
✓	✓	×	29.0	22.7	64.0	65.0	36.5	**30.0**	43.2	39.2
✓	✓	✓	**37.7**	**26.1**	**70.6**	**67.1**	37.6	**30.0**	**48.6**	**41.1**

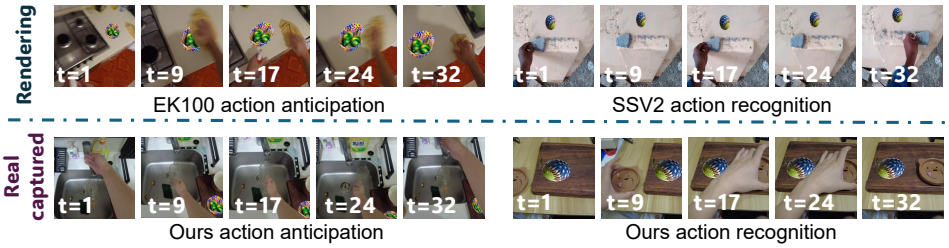


Figure 7: Visualization of TARA texture. The top row shows geometry-grounded texture rendering on EK100 and SSV2 videos, and the bottom row presents physical print-and-recapture examples.

masks. It indicates that TARA achieves the highest or joint-highest ASR across all evaluated models and tasks. In physical print-and-recapture experiments, TARA achieves 32.1% and 28.2% ASR on V-JEPA2-L for action anticipation and recognition, respectively, outperforming EOT by 3.5% and 5.1%. These results demonstrate that TARA maintains its attack effectiveness under uncontrolled object–camera trajectories and additional print-and-capture distortions, supporting its effectiveness under real-world physical deployment.

Robustness to simulated print–capture distortions. To further examine its sensitivity to print-and-capture distortions, Figure 6 reports the attack performance of our TARA texture under varying levels of distortions, including brightness changes, Gaussian noise, blur, and JPEG compression. TARA maintains relatively stable ASR under brightness variations, Gaussian noise, and JPEG compression, while stronger patch and capture blur lead to performance degradation, particularly on EK100 action anticipation. This degradation may result from the smoothing of local adversarial texture structures and the loss of fine-grained visual details, as further illustrated by the distorted texture visualizations in the Appendix A.5. Overall, these results complement our physical print-and-recapture evaluation, demonstrating that TARA maintains its attack effectiveness under a wide range of realistic imaging distortions within practical operating conditions.

4.4 Ablation Study and Analysis

Ablation study. Table 6 investigates the contributions of each component. To verify the contribution of trajectory awareness learning, we optimize a universal adversarial patch at a fixed image-center location without using the annotated video trajectories. The results highlight trajectory-aware optimization as a key component of TARA, contributing substantial improvements in attack performance. Meanwhile, both contrastive alignment and task refinement contribute non-trivial performance gains, further improving TARA's overall attack effectiveness. Together, these findings confirm the contributions of all three components to TARA's overall performance.

Visualization of adversarial textures. Figure 7 presents representative video frames of TARA under geometry-grounded rendering and physical print-and-recapture settings for both action anticipation and recognition tasks. The adversarial texture undergoes substantial variations in viewpoint, scale, position, and visibility along diverse object–camera trajectories, illustrating the dynamic observation conditions considered in our evaluation. The real-world captures further demonstrate how a fixed printed texture appears across different physical scenarios under uncontrolled observation conditions. Additional visualizations covering diverse video instances and observation trajectories are provided in the Appendix A.5 and A.8.

Generalization across content and trajectories. We further evaluate TARA by independently varying video content and observation trajectories, including unseen trajectories from independently captured real-world videos. We find that TARA retains its attack effectiveness under unseen tra-

jectories, even when applied to unseen video content, demonstrating its generalization beyond the training video–trajectory instances. The detailed results are in Appendix A.3.

Statistical Stability. We further evaluate the statistical stability of TARA across three random seeds, observing stable performance on both EK100 and SSV2. Details are shown in Appendix A.4.

5 Related Work

Video understanding models and world models. Video pretraining learns transferable spatiotemporal representations for understanding scenes and actions in videos. Early research utilizes reconstruction-based objectives, such as VideoMAE and VideoMAE V2 (Tong et al., 2022; Wang et al., 2023). Then, InternVideo2 and VideoPrism (Wang et al., 2024; Zhao et al., 2024) scale training data and supervision toward more general video understanding. Beyond interpreting observed events, AVT predicts future action labels from preceding observations (Girdhar & Grauman, 2021), while world models such as iVideoGPT and DINO-WM (Wu et al., 2024; Zhou et al., 2025) predict action-conditioned future observations or visual features to support planning. V-JEPA instead learns representations by predicting masked latent video features (Bardes et al., 2024) and V-JEPA 2 (Assran et al., 2025) connects these capabilities for video understanding and anticipation. Their reliance on visual observations motivates investigating their physical robustness to camera input.

Digital adversarial attacks on video and world models. Early video attacks exploit temporal sparsity and image-to-video feature transfer (Wei et al., 2019; 2022), while recent work targets pretrained video representations and transfers perturbations across downstream tasks (Xia et al., 2024; Lu et al., 2026a). World-model attacks further explore structured conditioning perturbations, automated attack search against closed-loop agents, and poisoning through generated data (Guo et al., 2026a;b; Rathbun et al., 2026). These approaches operate through digital access to inputs, conditioning signals, or training pipelines. Our setting instead constrains the adversarial input to a single printable texture in real-world scenarios, with its observation governed by object–camera trajectories. The texture must therefore generalize jointly across video semantics and evolving observations.

Physical adversarial attacks. Early work established printable adversarial patches and transformation-based optimization against fabrication and imaging variations (Brown et al., 2017; Athalye et al., 2018; Eykholt et al., 2018). Subsequent studies extended physical attacks to videos and tracking by modeling temporal and motion-induced variations (Pony et al., 2021; Du et al., 2022; Long et al., 2024), and improved robustness across poses and viewpoints through multi-view or geometry-aware optimization (Li et al., 2025; Zhang et al., 2025; Lou et al., 2025). Recent work further attacks embodied policies, including diffusion policies and VLA models, with efforts toward cross-model transferability and partial trajectory observability (Chen et al., 2024; Wang et al., 2025; Lu et al., 2026b; Wang et al., 2026). Tex3D and UniTexture optimize object-bound textures along manipulation trajectories and across multiple VLA tasks (Chen et al., 2026; Dai et al., 2026). These studies improve temporal persistence, multi-view robustness, transferability, or geometric consistency, but do not study joint generalization across heterogeneous video semantics and unseen object–camera trajectories.

6 Conclusion

In this work, we investigate a practical physical attack setting: a single printable adversarial texture across heterogeneous videos and uncontrolled object–camera trajectories. We identify conflicting optimization gradients across diverse video–trajectory instances as a key challenge to generating an effective adversarial texture. To address this challenge, we propose TARA, which employs trajectory-aware contrastive alignment to encourage shared adversarial representations across instances, followed by trajectory-aware task refinement to induce task-level prediction errors. We further establish a reproducible geometry-grounded benchmark with temporally consistent homography and visibility annotations, and conduct real-world print-and-recapture experiments to validate attack effectiveness under practical deployment conditions. Extensive evaluations in both simulated and real-world print-and-recapture experiments demonstrate that TARA effectively disrupts video understanding and anticipation across unseen video–trajectory instances, highlighting the security risks posed by persistent physical attacks in real-world deployments.

AI USE STATEMENT

In this work, we did not use generative AI tools for tasks requiring mandatory disclosure. We have not used generative AI tools to develop theoretical models, formulate mathematical claims, design experiments, or interpret results, and the remaining mandatory-disclosure tasks are not applicable to this work. Additionally, we used generative AI tools to create or edit software code and to proofread the manuscript for grammar, clarity, and readability. We have reviewed all AI-assisted work. The authors reviewed and tested AI-assisted code for correctness and verified that the suggested revisions preserved the technical meaning and accuracy of the manuscript. We take responsibility for the final content of this work, including text, claims, or artifacts produced with the aid of generative AI.

ETHICS STATEMENT

This work investigates the vulnerability of world models and video understanding models to persistent physical adversarial textures. Our objective is to characterize this security risk and support the development of more robust models. We evaluate the attack on established video benchmarks and through controlled print-and-recapture experiments, rather than deploying it in operational systems. Our findings highlight the need to assess model robustness under temporally varying physical observations, particularly before deploying video-based models in safety-critical applications.

REPRODUCIBILITY STATEMENT

We provide detailed descriptions of the attack formulation, optimization procedure, and evaluation protocols in the main paper, with additional implementation settings, trajectory-construction procedures, and quality-control details in the Appendix. To facilitate reproducibility, we will publicly release the code and data required to reproduce all experiments in this work, including our generated trajectory annotations and real-world captured videos. These resources will support the reproduction of our results and enable future research to evaluate adversarial textures under various video–trajectory conditions.

REFERENCES

Mido Assran, Adrien Bardes, David Fan, Quentin Garrido, Russell Howes, Matthew Muckley, Ammar Rizvi, Claire Roberts, Koustuv Sinha, Artem Zholus, et al. V-jepa 2: Self-supervised video models enable understanding, prediction and planning. *arXiv preprint arXiv:2506.09985*, 2025.

Anish Athalye, Logan Engstrom, Andrew Ilyas, and Kevin Kwok. Synthesizing robust adversarial examples. In *Int. Conf. Machine Learning*, 2018.

Adrien Bardes, Quentin Garrido, Jean Ponce, Xinlei Chen, Michael Rabbat, Yann LeCun, Mido Assran, and Nicolas Ballas. Revisiting feature prediction for learning visual representations from video. *Transactions on Machine Learning Research*, 2024.

Anneliese Braunegg, Amartya Chakraborty, Michael Krumdick, Nicole Lape, Sara Leary, Keith Manville, Elizabeth Merkhofer, Laura Strickhart, and Matthew Walmer. Apricot: A dataset of physical adversarial attacks on object detection. In *Eur. Conf. Comput. Vis.*, pp. 35–50. Springer, 2020.

Tom B Brown, Dandelion Mané, Aurko Roy, Martín Abadi, and Justin Gilmer. Adversarial patch. *arXiv preprint arXiv:1712.09665*, 2017.

Jiawei Chen, Simin Huang, Jiawei Du, Shuaihang Chen, Yu Tian, Mingjie Wei, Chao Yu, and Zhaoxia Yin. Tex3d: Objects as attack surfaces via adversarial 3d textures for vision-language-action models. *arXiv preprint arXiv:2604.01618*, 2026.

Yipu Chen, Haotian Xue, and Yongxin Chen. Diffusion policy attacker: Crafting adversarial attacks for diffusion-based policies. *Adv. Neural Inform. Process. Syst.*, 2024.

Yukun Dai, Mingzhe Dai, Tianshi Wang, Fengling Li, Jingjing Li, and Lei Zhu. Unitexture: Cross-task universal adversarial textures for vision-language-action models. *arXiv preprint arXiv:2608.13453*, 2026.

Dima Damen, Hazel Doughty, Giovanni Maria Farinella, Antonino Furnari, Evangelos Kazakos, Jian Ma, Davide Moltisanti, Jonathan Munro, Toby Perrett, Will Price, et al. Rescaling egocentric vision: Collection, pipeline and challenges for epic-kitchens-100. *Int. J. Comput. Vis.*, pp. 33–55, 2022.

Meiwen Ding, Song Xia, Yi Yu, Shuting He, and Xudong Jiang. Transferable adversarial attack on referring video object segmentation. *IEEE Transactions on Information Forensics and Security*, 2026.

Zhenyu Du, Xingxing Wei, Weiming Zhang, Fangzheng Liu, Huanyu Bian, and Jiayang Liu. A motional but temporally consistent physical video examples. *Journal of Information Security and Applications*, 2022.

Johan Edstedt, Qiyu Sun, Georg Bökman, Mårten Wadenbäck, and Michael Felsberg. Roma: Robust dense feature matching. In *IEEE Conf. Comput. Vis. Pattern Recog.*, 2024.

Kevin Eykholt, Ivan Evtimov, Earlence Fernandes, Bo Li, Amir Rahmati, Chaowei Xiao, Atul Prakash, Tadayoshi Kohno, and Dawn Song. Robust physical-world attacks on deep learning visual classification. In *IEEE Conf. Comput. Vis. Pattern Recog.*, 2018.

Martin A Fischler and Robert C Bolles. Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography. *Communications of the ACM*, 1981.

Yonggan Fu, Shunyao Zhang, Shang Wu, Cheng Wan, and Yingyan Lin. Patch-fool: Are vision transformers always robust against adversarial perturbations? In *Int. Conf. Learn. Represent.*, 2022.

Rohit Girdhar and Kristen Grauman. Anticipative video transformer. In *Int. Conf. Comput. Vis.*, 2021.

Raghav Goyal, Samira Ebrahimi Kahou, Vincent Michalski, Joanna Materzynska, Susanne Westphal, Heuna Kim, Valentin Haenel, Ingo Fruend, Peter Yianilos, Moritz Mueller-Freitag, et al. The "something something" video database for learning and evaluating visual common sense. In *Int. Conf. Comput. Vis.*, 2017.

Zhixiang Guo, Siyuan Liang, Andras Balogh, Noah Lunberry, Rong-Cheng Tu, Mark Jelasity, and Dacheng Tao. When world models dream wrong: Physical-conditioned adversarial attacks against world models. *arXiv preprint arXiv:2602.18739*, 2026a.

Zhixiang Guo, Siyuan Liang, Shi Fu, Cheng Guo, Andras Balogh, Mark Jelasity, and Dacheng Tao. Wmattack: Automated attack search for adversarial evaluation of world-model agents. *arXiv preprint arXiv:2605.23220*, 2026b.

Nabeel Hingun, Chawin Sitawarin, Jerry Li, and David Wagner. Reap: A large-scale realistic adversarial patch benchmark. In *Int. Conf. Comput. Vis.*, 2023.

Sebastian Jung, Artem Zholus, Martin Sundermeyer, Carl Doersch, Ross Goroshin, David Joseph Tan, Sarath Chandar, Rudolph Triebel, and Federico Tombari. Tapnext++: What's next for tracking any point (tap)? In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) Findings*, 2026.

Hee-Seon Kim, Minji Son, Minbeom Kim, Myung-Joon Kwon, and Changick Kim. Breaking temporal consistency: Generating video universal adversarial perturbations using image models. In *Int. Conf. Comput. Vis.*, 2023.

Yanjie Li, Kaisheng Liang, and Bin Xiao. Uv-attack: Physical-world adversarial attacks on person detection via dynamic-nerf-based uv mapping. In *Int. Conf. Learn. Represent.*, 2025.

Shilong Liu, Zhaoyang Zeng, Tianhe Ren, Feng Li, Hao Zhang, Jie Yang, Qing Jiang, Chunyuan Li, Jianwei Yang, Hang Su, et al. Grounding dino: Marrying dino with grounded pre-training for open-set object detection. In *Eur. Conf. Comput. Vis.*, 2024.

Jiahuan Long, Tingsong Jiang, Wen Yao, Shuai Jia, Weijia Zhang, Weien Zhou, Chao Ma, and Xiaoqian Chen. Papmot: Exploring adversarial patch attack against multiple object tracking. In *Eur. Conf. Comput. Vis.*, 2024.

Tianrui Lou, Xiaojun Jia, Siyuan Liang, Jiawei Liang, Ming Zhang, Yanjun Xiao, and Xiaochun Cao. 3d gaussian splatting driven multi-view robust physical adversarial camouflage generation. In *Int. Conf. Comput. Vis.*, 2025.

Hui Lu, Yi Yu, Song Xia, Yiming Yang, Deepu Rajan, Boon Poh Ng, Alex Kot, and Xudong Jiang. From pretrain to pain: Adversarial vulnerability of video foundation models without task knowledge. In *AAAI*, 2026a.

Hui Lu, Yi Yu, Yiming Yang, Chenyu Yi, Qixin Zhang, Bingquan Shen, Alex C Kot, and Xudong Jiang. When robots obey the patch: Universal transferable patch attacks on vision-language-action models. In *IEEE Conf. Comput. Vis. Pattern Recog.*, 2026b.

Lorenzo Mur-Labadia, Matthew Muckley, Amir Bar, Mido Assran, Koustuv Sinha, Mike Rabbat, Yann LeCun, and Nicolas Ballas. V-jepa 2.1: Unlocking dense features in video self-supervised learning. In *Eur. Conf. Comput. Vis.*, 2026.

Roi Pony, Itay Naeh, and Shie Mannor. Over-the-air adversarial flickering attacks against video recognition networks. In *IEEE Conf. Comput. Vis. Pattern Recog.*, 2021.

Ethan Rathbun, Ahmed Agha, Saaduddin Mahmud, Christopher Amato, Alina Oprea, and Eugene Bagdasarian. Targeting world models to compromise robot learning pipelines. *arXiv preprint arXiv:2606.09499*, 2026.

Nikhila Ravi, Valentin Gabeur, Yuan-Ting Hu, Ronghang Hu, Chaitanya Ryali, Tengyu Ma, Haitham Khedr, Roman Rädle, Chloe Rolland, Laura Gustafson, et al. Sam 2: Segment anything in images and videos. In *Int. Conf. Learn. Represent.*, 2025.

Zhan Tong, Yibing Song, Jue Wang, and Limin Wang. Videomae: Masked autoencoders are data-efficient learners for self-supervised video pre-training. *Adv. Neural Inform. Process. Syst.*, 2022.

Limin Wang, Bingkun Huang, Zhiyu Zhao, Zhan Tong, Yinan He, Yi Wang, Yali Wang, and Yu Qiao. Videomae v2: Scaling video masked autoencoders with dual masking. In *IEEE Conf. Comput. Vis. Pattern Recog.*, 2023.

Taowen Wang, Cheng Han, James Liang, Wenhao Yang, Dongfang Liu, Luna Xinyu Zhang, Qifan Wang, Jiebo Luo, and Ruixiang Tang. Exploring the adversarial vulnerabilities of vision-language-action models in robotics. In *Int. Conf. Comput. Vis.*, 2025.

Xiaofei Wang, Mingliang Han, Tianyu Hao, Yi Yang, Yun-Bo Zhao, and Keke Tang. Partially observable adversarial patch attacks on vision-language-action models in robotics. *IEEE Robotics and Automation Letters*, 2026.

Yi Wang, Kunchang Li, Xinhao Li, Jiashuo Yu, Yinan He, Guo Chen, Baoqi Pei, Rongkun Zheng, Zun Wang, Yansong Shi, et al. Internvideo2: Scaling foundation models for multimodal video understanding. In *Eur. Conf. Comput. Vis.*, 2024.

Xingxing Wei, Jun Zhu, Sha Yuan, and Hang Su. Sparse adversarial perturbations for videos. In *AAAI*, 2019.

Zhipeng Wei, Jingjing Chen, Zuxuan Wu, and Yu-Gang Jiang. Cross-modal transferable adversarial attacks from images to videos. In *IEEE Conf. Comput. Vis. Pattern Recog.*, 2022.

Jialong Wu, Shaofeng Yin, Ningya Feng, Xu He, Dong Li, Jianye Hao, and Mingsheng Long. ivideogpt: Interactive videogpts are scalable world models. *Adv. Neural Inform. Process. Syst.*, 2024.

Song Xia, Wenhan Yang, Yi Yu, Xun Lin, Henghui Ding, Lingyu Duan, and Xudong Jiang. Transferable adversarial attacks on sam and its downstream models. *Adv. Neural Inform. Process. Syst.*, 2024.

Tianqiu Zhang, Muyang Lyu, Xiao Liu, and Si Wu. Structure abstraction and generalization in a hippocampal-entorhinal inspired world model. In *Int. Conf. Machine Learning*, 2026a.

Tianqiu Zhang, Muyang Lyu, Yufan Zhang, Fang Fang, and Si Wu. DiLA: Disentangled latent action world models. In *Int. Conf. Machine Learning*, 2026b.

Ximin Zhang, Jinyin Chen, Haibin Zheng, and Zhenguang Liu. Phycamo: A robust physical camouflage via contrastive learning for multi-view physical adversarial attack. In *AAAI*, 2025.

Long Zhao, Nitesh B Gundavarapu, Liangzhe Yuan, Hao Zhou, Shen Yan, Jennifer J Sun, Luke Friedman, Rui Qian, Tobias Weyand, Yue Zhao, et al. Videoprism: A foundational visual encoder for video understanding. *Int. Conf. Machine Learning*, 2024.

Artem Zholus, Carl Doersch, Yi Yang, Skanda Koppula, Viorica Patraucean, Xu Owen He, Ignacio Rocco, Mehdi SM Sajjadi, Sarath Chandar, and Ross Goroshin. Tapnext: Tracking any point (tap) as next token prediction. In *Int. Conf. Comput. Vis.*, 2025.

Gaoyue Zhou, Hengkai Pan, Yann LeCun, and Lerrel Pinto. Dino-wm: World models on pre-trained visual features enable zero-shot planning. *Int. Conf. Machine Learning*, 2025.

A Appendix

A.1 Trajectory-Aware Weighting

To balance the contributions of video–trajectory instances with different levels of texture exposure, we introduce trajectory-aware weighting during task refinement. Let $\alpha_{itv}(p) \in [0, 1]$ denote the visible texture alpha at pixel p in frame t and inference view v, obtained from the frame-wise homography and visibility mask. The total observable texture area of instance i is $a_i = \sum_{v,t,p} \alpha_{itv}(p) > 0$, with a single view for models that do not use multi-view inference. For each minibatch B, we set $a_{\text{ref}} = \text{median}_{j \in \mathcal{B}}\, a_j$ and compute the instance weights as:

$$\tilde{w}_i = \text{clip}\left(\sqrt{\frac{a_{\text{ref}}}{a_i}},\, 0.5,\, 2\right), \qquad w_i = \frac{\tilde{w}_i}{\sum_{j \in B} \tilde{w}_j}. \tag{5}$$

This scheme increases the relative contribution of instances with smaller observable texture areas, while clipping limits extreme weights and normalization ensures that $\sum_{i \in \mathcal{B}} w_i = 1$. As the observation trajectories are fixed, the weights are computed independently of the adversarial texture and remain constant during optimization.

A.2 Attack details and visualization for DILA generation task

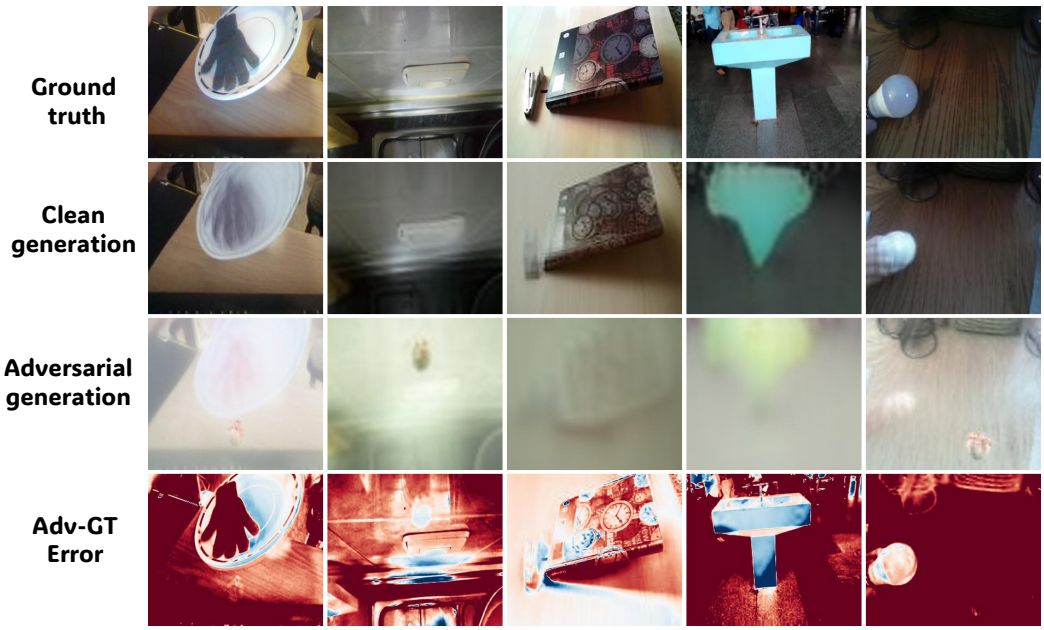


Figure 8: Visualization of DiLA reconstruction results under TARA. Each column is a different validation clip.

Attack details. For DiLA reconstruction, we use 155 Train and 82 Val clips, with 16 attacked source frames. Contrastive alignment optimizes the mean future spatiotemporal embedding for 10 epochs using Adam ($0.03 \rightarrow 0.003$). The subsequent five-epoch refinement uses Adam ($0.03 \rightarrow 0.005$) to maximize outside-patch RGB MSE relative to the clean reconstruction. Both stages use a logical batch of 32 clips and 24 updates per epoch; latent actions are recomputed from attacked source frames.

Visualization of the attacked generation. Figure 8 shows five validation clips selected for high MSE gain outside the patch region. The first row shows the ground truth observed frame, followed by clean reconstruction, adversarial reconstruction, and the change in per-pixel squared error. The adversarial reconstructions exhibit appearance changes well beyond the attached texture, demonstrating that our attack remains effective at disrupting conditional reconstruction.

A.3 Generalization Across Content and Trajectories

Table 7: Generalization across video content and observation trajectories on EK100 with V-JEPA2-L. Top-1 ASR (%, ↑) is reported on 74 eligible clip instances per condition.

	Seen Content		Unseen Content	
Method	Seen Traj.	Unseen Traj.	Seen Traj.	Unseen Traj.
Static center	17.6	17.6	13.5	9.5
EOT	32.4	24.3	32.4	16.2
PhyCamo	17.6	20.3	14.9	12.2
Patch-Fool	12.2	13.5	10.8	10.8
UPA-RFAS	13.5	13.5	9.5	5.4
BTC-UAP	21.6	20.3	18.9	13.5
TARA	**40.5**	**31.5**	**43.2**	**28.8**

Experimental setup. We evaluate frozen adversarial textures under four combinations of video content and observation trajectories. The seen-content cohort contains 74 EK100 training clips; the unseen-content cohort contains 74 clip instances from 35 independently captured practical recordings. For crossed conditions, we remap the homographies to the V-JEPA2-L input viewport while retaining temporal visibility. The static-center texture is also rendered along each target trajectory, which is trained without trajectory-aware optimization as a baseline reference.

Results. Table 7 shows that TARA achieves the highest ASR in all four conditions. On seen content with unseen trajectories, TARA reaches 31.5% ASR, compared with 24.3% for the strongest baseline, EOT. When both content and trajectory are unseen, TARA achieves 28.8%, exceeding EOT by 12.6 percentage points and static center by 19.3 points. This demonstrates the effectiveness of TARA texture under the joint shift in content and trajectory.

A.4 Statistical Stability

Table 8: Random-seed stability of TARA. Final-epoch Top-1 ASR (%). Mean and sample SD are computed across all three seeds; seed 0 is also reported in the main results.

Dataset / Model	Seed	Train	Val	Overall
EK100 / V-JEPA2-L	0 (reported)	37.7	26.1	32.1
	1	37.9	26.1	32.2
	2	36.8	24.6	31.0
	Mean ± SD	37.5 ± 0.6	25.6 ± 0.9	31.8 ± 0.7
SSV2 / V-JEPA2-L	0 (reported)	38.6	34.7	36.9
	1	38.4	34.0	36.5
	2	37.0	34.0	35.7
	Mean ± SD	38.0 ± 0.9	34.2 ± 0.4	36.4 ± 0.6

Randomness test. To assess the sensitivity of TARA to random initialization, we repeat the experiments with three random seeds on EK100 and SSV2 using V-JEPA2-L. We keep the model, data split, attack objective, and training schedule fixed, and vary the random sampling of training batches and trajectories. As shown in Table 8, TARA exhibits consistent performance across seeds, with small standard deviations for both Train and Val splits. The resulting Overall ASR varies only moderately across runs, indicating that the reported performance is robust to random-seed variation.

A.5 More visualization

Visualization under simulated capture-and-print distortions. Figure 9 compares the original adversarial textures with their distorted counterparts on EK100 and SSV2. Patch blur and capture blur visibly smooth local texture structures and suppress fine-grained details, providing a qualitative

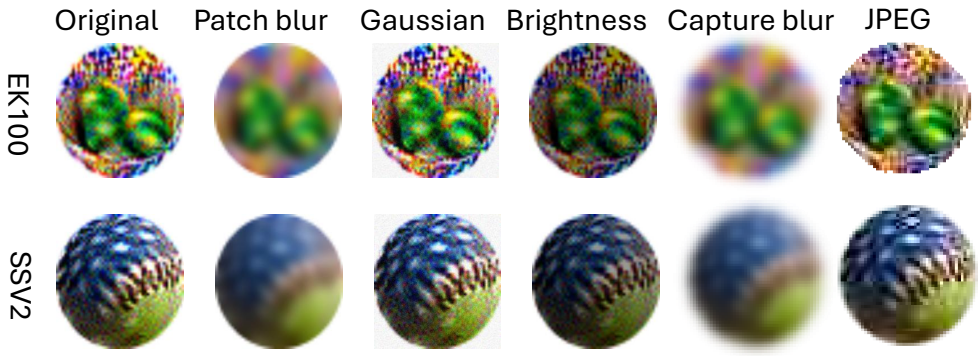


Figure 9: Visualization of adversarial textures under different imaging distortions on EK100 and SSV2. Patch and capture blur substantially suppress fine-grained texture details.

illustration of the performance degradation observed under severe blur. The remaining distortions alter texture appearance without comparably suppressing its local structures.

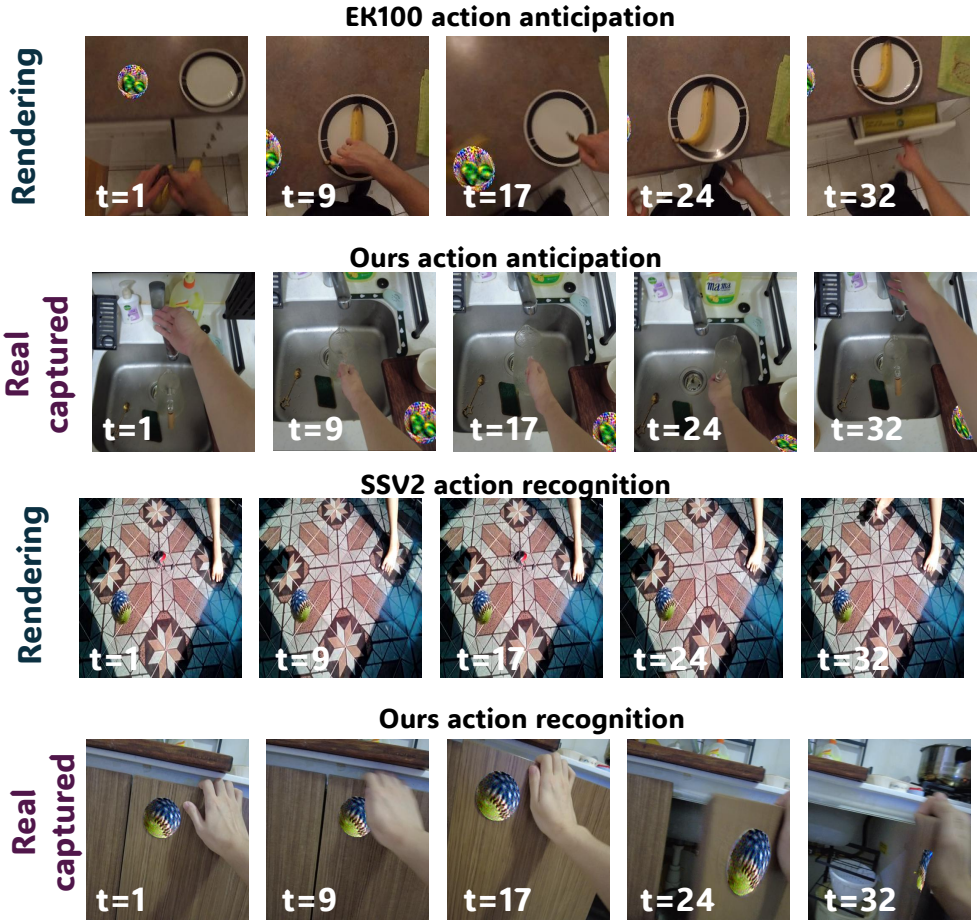


Figure 10: Additional trajectory visualizations across digital and physical settings. The first and third rows show digitally projected textures on EK100 and SSV2, respectively, while the second and fourth rows show real-world print-and-recapture recordings with physically attached textures.

Additional trajectory visualizations. Figure 10 presents representative examples from two evaluation settings: digitally projected adversarial textures on EK100 and SSV2 videos, and real-world

Table 9: Model-specific TARA optimization schedules and update budgets.

Model / task	Train / Val	Alignment feature	Optimization schedule	Updates
V-JEPA2-L / EK100	356 / 249	Action query	F10 Adam ($0.03 \rightarrow 0.003$) FT5 sign-PGD ($0.01 \rightarrow 0.005$, step)	360
V-JEPA2.1-B / EK100	175 / 140	Action + token mean	F10 Adam ($0.03 \rightarrow 0.003$) FT5 sign-PGD ($0.01 \rightarrow 0.005$, cosine)	360
AVT-B / EK100	85 / 60	Future aggregate	F10 Adam ($0.03 \rightarrow 0.003$, cosine) FT5 sign-PGD ($0.005 \rightarrow 0.001$, cosine)	360
V-JEPA2-L / SSV2	184 / 144	Probe prelinear	F10 Adam ($0.03 \rightarrow 0.003$) FT5 sign-PGD ($0.01 \rightarrow 0.005$, step)	360
VideoMAE-B / SSV2	184 / 144	Classifier input F_0	F10 Adam ($0.03 \rightarrow 0.003$) FT5 sign-PGD ($0.03 \rightarrow 0.005$, cosine)	360
InternVideo2-dist-L / SSV2	184 / 144	Layer-24 token mean	F10 Adam ($0.03 \rightarrow 0.003$) FT5 sign-PGD ($0.005 \rightarrow 0.001$, cosine)	360

print-and-recapture recordings in which the textures are physically printed and attached to objects. For each video, we show sampled frames along its observation trajectory to illustrate how a fixed texture is observed under evolving object–camera motion. Across different video instances, the texture undergoes substantial changes in image location, apparent scale, and visibility, including partial occlusion and movement toward image boundaries. These examples illustrate the diverse, temporally varying observation conditions considered in both our digital benchmark and physical evaluation.

A.6 Implementations details

Model-specific attack configurations. For EK100 anticipation, V-JEPA2-L uses a 32-frame, 256×256 context and aligns its action-query representation. V-JEPA2.1-B uses a 32-frame, 384×384 mixed-window input and aligns its pooled action representation concatenated with the final-token mean. For AVT-B, it uses its native 10-frame input with three crops and horizontal flips, aligning the aggregated future representation. For SSV2 recognition, V-JEPA2-L aligns the prelinear probe representation using two 16-frame segments and three crops; VideoMAE-B aligns its classifier-input feature F_0 using two 16-frame samples and three crops; and InternVideo2-dist-L aligns the mean layer-24 token representation using four eight-frame offsets and three crops. Task refinement uses each model's final prediction after native view aggregation, rather than optimizing separate predictions for individual views.

Training details and checkpoint selection. All classification alignment stages use temperature $\tau = 0.1$ and Adam optimizer. Refinement minimizes the ground-truth log-probability of the model's aggregated task prediction using sign-PGD, with texture values projected to $[0, 1]$ after each update. The training details are shown in 9. For a fair comparison, all compared attacks use the same optimizer, learning-rate schedule, and number of texture updates as TARA under the corresponding model and task setting, while retaining their respective attack objectives. All models use the final checkpoint of their alignment stage.

Real-World Digital and Physical Evaluation. For the real-world digital comparison, we utilize the same homography and mask to generate the adversarial video for a fair comparison. For the physical print-and-recapture evaluation, we attach printed EOT and TARA textures to physical objects and compare the resulting adversarial recordings (ADV) against matched recordings with white textures at the same attachment locations (WHITE), under corresponding actions and recording conditions.

Random seeds and hardware. All reported results use seed 0 for texture initialization and batch/trajectory sampling and represent single-seed runs. Experiments were conducted on four NVIDIA GeForce RTX A6000 GPUs (48 GiB each), with each model-specific attack job running on one GPU. Micro-batching and gradient replay are used where necessary without changing the logical batch size or texture-update budgets.

Table 10: Geometry-estimation settings for newly annotated trajectories.

Component	Setting
Reference-point sampling	12×12 grid; 2-pixel erosion
Minimum correspondences	≥ 20 fitting; ≥ 8 held-out
RANSAC	2-pixel threshold; 0.999 confidence; 10,000 iterations
Inlier screening	≥ 20 inliers; ratio ≥ 0.60
Spatial screening	Inliers in ≥ 3 local quadrants
Temporal screening	Projection-area continuity

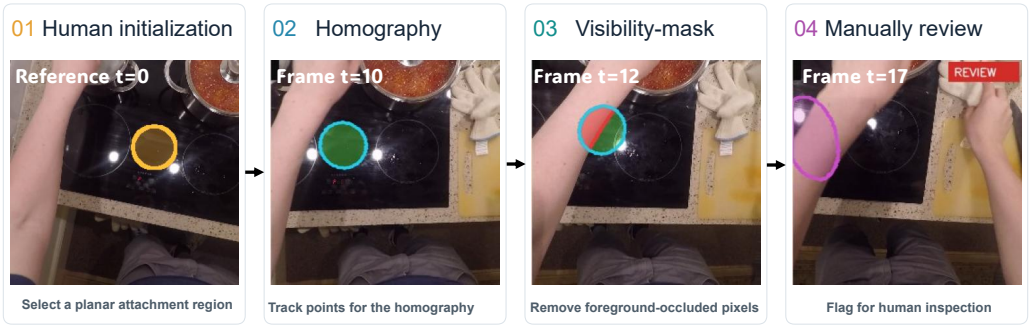


Figure 11: Human-guided surface-trajectory construction in an EK100 clip. (1) Manual attachment-region selection. (2) Homography estimation from tracked points. (3) Visible-support estimation under foreground occlusion. (4) Flagging of an inconsistent projection for human review.

A.7 Surface-Trajectory Construction and Quality Control

Trajectory construction. For each newly annotated clip, an annotator selects a locally planar object surface and a feasible texture attachment region in a reference frame. We sample anchor points within this region and track them across frames using TAPNext++ (Jung et al., 2026; Zholus et al., 2025). For frame t of clip i, we estimate a reference-to-frame homography $\boldsymbol{G}_{it}$ from the tracked correspondences using RANSAC (Fischler & Bolles, 1981). The texture-to-frame homography is $\boldsymbol{H}_{it} = \boldsymbol{G}_{it}\boldsymbol{A}_i$, where $\boldsymbol{A}_i$ maps the canonical texture to the reference frame. Candidate projections are validated using inlier support, held-out reprojection error, spatial coverage, and temporal scale consistency. For spatial validation, the bounding box of the source fitting points is divided into four quadrants, of which at least three must contain accepted inliers. When sparse tracking is insufficient, RoMa (Edstedt et al., 2024) provides dense correspondences with nearby validated frames for geometric recovery. Table 10 summarizes the geometry-estimation and screening settings used for newly annotated trajectories. GroundingDINO (Liu et al., 2024) and SAM2 (Ravi et al., 2025) provide candidate supporting-surface and foreground-occluder masks for estimating visible texture support.

Human review and quality control. Human review guides the initial texture placement and the correction of uncertain geometric and visibility annotations. Reviewers inspect temporal overlays for surface mismatch, projection drift, abrupt scale changes, and incorrect occlusion. As illustrated in Figure 11, they may revise the fixed attachment region for an entire clip, reselect anchor points and rerun tracking, verify a recovered projection, or correct its visible support. Tracking failure on a still-visible surface triggers geometric review rather than an automatic assignment of zero visibility. Zero visibility is assigned only when the texture is confirmed to be fully occluded or outside the image. Unresolved frames retain an explicit unknown status rather than being treated as physically invisible.

Benchmark composition and rendering. The digital benchmark contains 605 EK100 clips (356 Train/249 Val) and 328 SSV2 clips (184 Train/144 Val), with trajectory records for 19,251 and 10,204 distinct raw frames, respectively. These counts include frames with unresolved geometry. SSV2 clips were selected based on V-JEPA2-L clean Top-1 correctness and geometric eligibility; attack success is computed separately using each evaluated model's clean-correct examples. Clip

identities, splits, and trajectory annotations are fixed across attack methods, and only Train clips contribute texture gradients. The stored frame-wise annotations include a validity status in addition to the homography and visibility mask defining γ_i. We render validated projections using their recorded geometry and visibility; confirmed invisibility and unresolved geometry both leave the input frame unchanged. The latter remains labeled unknown rather than physically invisible, while valid projections in other frames of the same clip are retained. For each evaluated model, the annotations are aligned with its sampled raw frames and transformed consistently with its resizing and cropping operations.

Statistical definitions. The digital statistics in Table 1 use a fixed sampling and center-crop configuration for each dataset, whereas annotated-frame counts refer to distinct raw frames with trajectory records. Homography coverage is the fraction of sampled positions with a validated homography and a nonempty projected texture region within the crop. Mask rate is the fraction of projected positions with assessable foreground annotations for which an occluder overlaps the projected region; unknown occlusion states are excluded. Texture area measures the pooled fraction of image pixels covered by the recorded visible texture support:

$$\mathrm{TextureArea} = 100 \frac{\sum_{(i,t)\in\mathcal{S}} |\boldsymbol{M}_{it}|}{\sum_{(i,t)\in\mathcal{S}} |\boldsymbol{I}_{it}|}, \tag{6}$$

where $\mathcal{S}$ denotes all sampled positions and $\boldsymbol{I}_{it}$ is the corresponding model input crop. Unresolved positions contribute zero rendered area but remain in the denominator; thus, this metric measures annotation-supported rendering coverage rather than independently verified physical visibility. Texture motion is the median across clips of the maximum pairwise projected-center displacement normalized by the crop diagonal, computed for clips with at least two distinct valid raw-frame projections. Figure 5 additionally reports frame-level visible fraction, $|\boldsymbol{M}_{it}|/|\boldsymbol{P}_{it}|$, for resolved visibility annotations, and projected area, $100|\boldsymbol{P}_{it}|/|\boldsymbol{I}_{it}|$, for valid nonempty projections, where $\boldsymbol{P}_{it}$ denotes the projected texture region before foreground removal.

A.8 Examples of rendered and real-world captured clips.

Figures 12 and 13 show complete 32-frame model inputs for action anticipation and recognition, respectively. In each figure, the first two clips use textures digitally rendered along annotated trajectories, while the last two show physically printed textures captured in real-world videos. These examples illustrate how the same surface-attached texture is observed across changing viewpoints, scales, and occlusions.

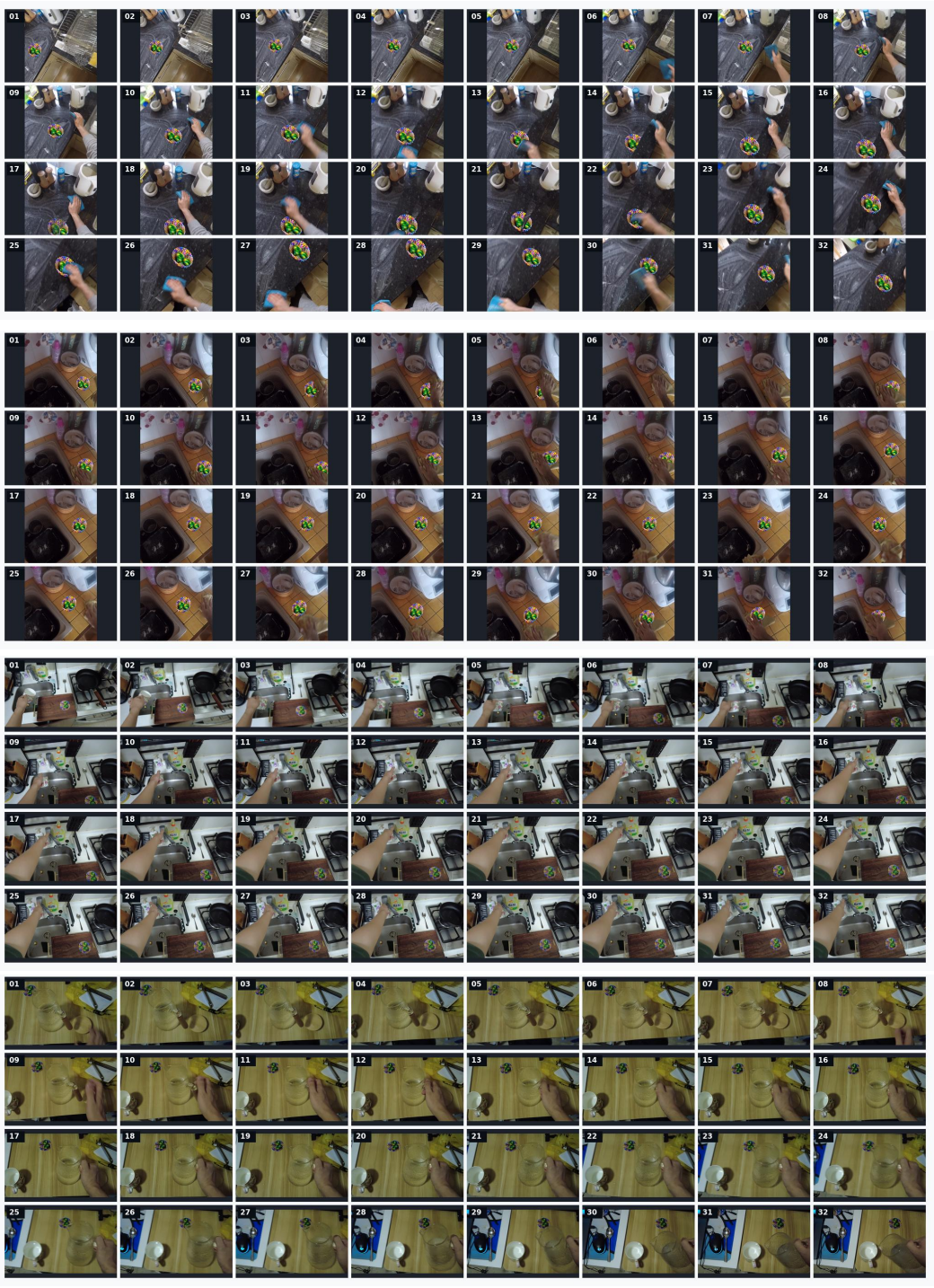


Figure 12: Visualization of the clips in action anticipation. The first two samples are rendered from the EK100 dataset, and the last two samples are our real-world print-and-capture data.

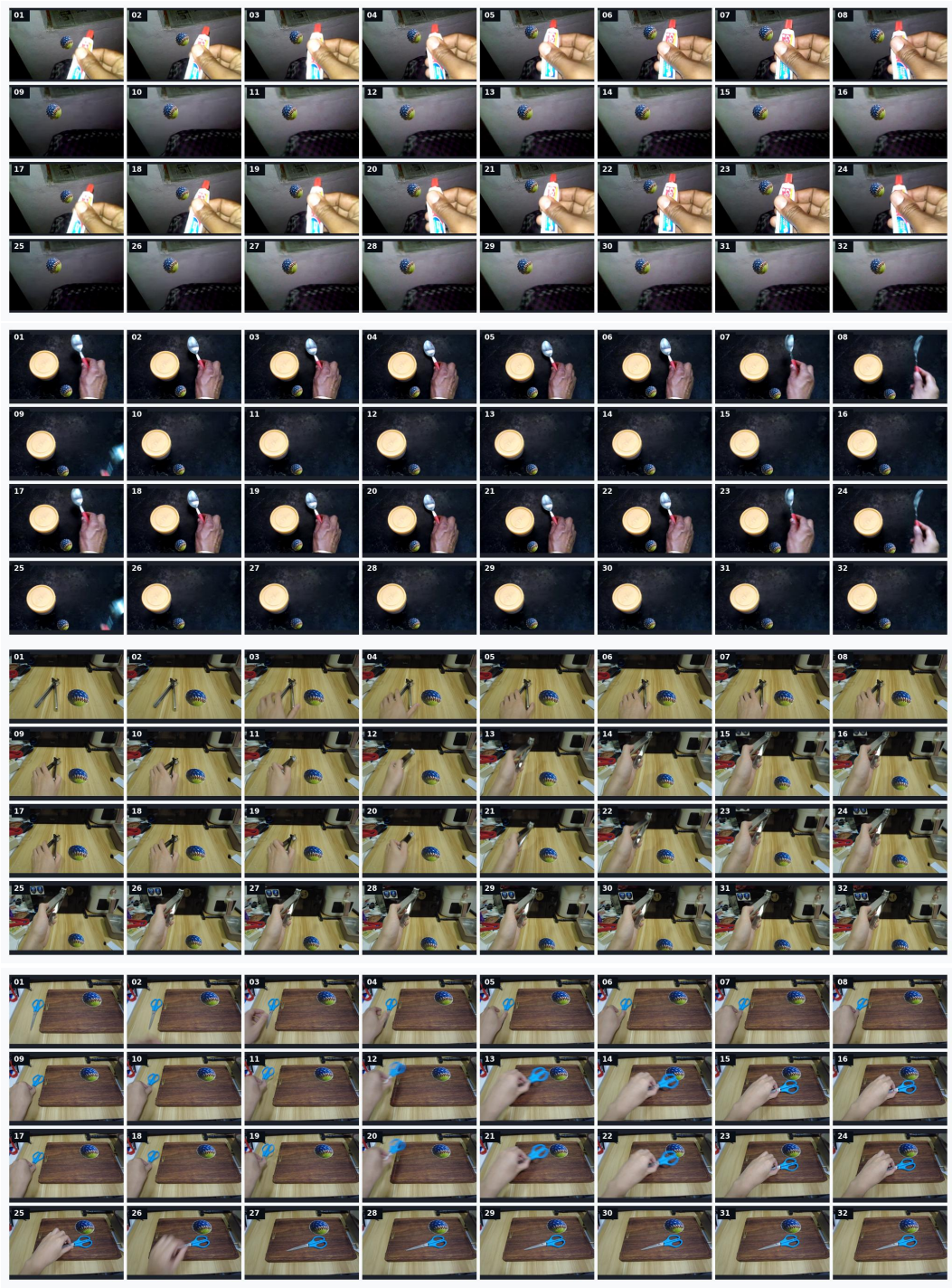


Figure 13: Visualization of the clips in action recognition. The first two samples are rendered from the SSV2 dataset, and the last two samples are our real-world print-and-capture data.