# CIVA: Critic-Induced Value-Subspace Attacks on Visual World-Model Agents

**Jiancheng Wang** **Mingli Zhu** **Tong Zhang** **Jiaqi Ruan**
Wei Wang Siyuan Liang* Dacheng Tao
Corresponding author:pandaliang521@gmail.com

## Abstract

Visual world-model agents such as DreamerV3 act through a recurrent latent state rather than a single observation, which weakens frame-wise observation attacks and makes their perturbations vary sharply over time under a strict per-frame perturbation constraint. We study white-box, causal, online attacks on such agents and propose Critic-Induced Value-Subspace Attacks (**CIVA**). Our key observation is that, along a rollout, critic-guided perturbations concentrate in a low-dimensional subspace induced by the victim's own critic. Based on this observation, CIVA first probes the frozen victim offline with critic-guided PGD and extracts a low-rank value-subspace by SVD. At test time, it optimizes only the subspace coefficients, smooths them with an exponential moving average (EMA), and maps them back to pixels. This design attacks value-sensitive recurrent dynamics while keeping the online optimization cheap and temporally coherent. Extensive experiments on DMC walker walk, Atari Pong, and Crafter show that CIVA consistently outperforms five recent methods; on DMC walker walk, it achieves the largest reward drop of 26.07% while keeping temporal variation low, with TempAbs of 0.646.

## 1 Introduction

Visual world models are becoming a mainstream paradigm for visual decision-making. Rather than reacting to the current frame alone, Dreamer-style agents write each observation into a recurrent latent state and act through learned dynamics in that internal space [14, 16–18]. Related lines on planning with learned models [52], transformer- [42, 51, 71] and diffusion-based world models [1], large-scale generative environments [4], scalable model-based control [19], and robot deployment [65] further reinforce recurrent latent dynamics as a core primitive in visual RL [6, 13, 66]. As these agents move closer to safety-critical use, understanding how to attack them becomes increasingly important.

Existing adversarial attacks on RL already show that deep policies are highly sensitive to small perturbations in visual or state observations [54, 11, 47, 44, 5, 41, 21, 26, 35, 48, 10, 29, 63, 30, 32, 31, 36, 58, 59]. Stronger variants have since been developed from several angles, including robust state-observation formulations [68, 69], real-time and universal observation attacks [56, 45, 61, 37], policy-aware attackers [53], detectability-aware stealthy attacks [9], and policy-distribution-based methods [7, 25, 2]. However, most of these methods still optimize perturbations directly in pixel space and treat each frame as an essentially independent target. That assumption is mismatched with visual world-model agents: what matters is not only whether one frame is perturbed, but whether the perturbation keeps steering the recurrent latent state under a realistic per-frame compute budget.

We study this setting in a white-box, causal, online attack regime against a frozen DreamerV3 victim. The attacker may modify only the current image observation and must produce the perturbation before the next environment step. Across DMC walker walk (DeepMind Control Suite [55]), Atari Pong, and Crafter, we observe two consistent phenomena. First, naive per-frame full-pixel projected

Preprint.

gradient descent (PGD) [41] is a weak primitive here: recurrent latent dynamics dilute isolated high-frequency perturbations, so greedy frame-wise attacks struggle to accumulate influence over long horizons. Second, effective attack directions do not spread uniformly across the full pixel space. When we stack critic-guided perturbations across time, their energy concentrates in a surprisingly low-dimensional subspace. This echoes subspace and low-frequency findings from supervised learning [67, 12, 45, 40, 23], but the structure here is induced by the victim's own value landscape rather than by input statistics.

These observations suggest a different attack strategy: identify value-sensitive directions once, then reuse them causally over time. We instantiate this idea as **CIVA**. CIVA first collects critic-guided perturbations offline and applies singular value decomposition (SVD) to extract a low-dimensional, victim-aware attack subspace. At deployment time, it no longer searches the full pixel space at every frame; instead, it optimizes only the subspace coefficients and smooths them with an exponential moving average before lifting them back to pixels. This directly addresses the core difficulty of attacking world-model agents under a tight $\ell_\infty$ budget: the perturbation must remain strong enough to influence recurrent latent dynamics while also staying computationally cheap, temporally coherent, and visually subtle [20, 70, 60, 24, 72]. Across continuous control, discrete control, and open-ended tasks, CIVA yields stronger and more stable degradation than representative observation-space baselines [68, 56, 53, 9, 7], while preserving favorable temporal smoothness and perceptual similarity. Ablations further show that the gains come from the combination of a critic-induced subspace, online optimization constrained to that subspace, and temporal smoothing on its coefficients.

Our contributions are threefold:

- We identify a key mismatch between conventional frame-wise observation attacks and visual world-model agents: because decisions depend on recurrent latent state, successful attacks must shape the victim's internal representation over time rather than only degrade individual frames [39, 38].
- We propose CIVA, a critic-induced value-subspace attack that extracts a low-rank victim-aware perturbation basis offline and performs efficient causal online optimization with temporal smoothing inside that subspace.
- We evaluate CIVA on DMC walker walk, Atari Pong, and Crafter across attack effectiveness, action-distribution shift, temporal variation, and perceptual similarity, showing that low-dimensional critic-aligned perturbations are more effective for world-model agents than direct full-pixel frame-wise optimization.

## 2 Related Work

**World-model reinforcement learning.** World models build policies around a recurrent latent dynamics model that summarizes past observations and supports multi-step prediction [14]. The Dreamer line demonstrates that this recipe scales across continuous control, Atari, and open-ended visual domains under a largely unified configuration [16–18]. Parallel efforts extend the same general direction to planning with learned models [52], transformer- [42, 51, 71] and diffusion-based world models [1], large-scale generative environments [4], scalable model-based control [19], and robotic deployment [65]; recent surveys summarize this broader trend [6, 13, 66]. For our purposes, the important point is not only that these agents are powerful, but that their decision process is mediated by a recurrent latent state. That makes their adversarial surface fundamentally temporal rather than frame-local.

**Adversarial attacks on deep policies.** Adversarial vulnerability in supervised learning [54, 11, 47, 44, 5, 41] carries over directly to RL, where small perturbations in observations can cause severe behavioral degradation [21, 26, 35, 48]. Follow-up work has strengthened this line through robust state-observation formulations [68, 69], real-time and universal perturbations [56, 61, 37], policy-aware attackers [53], detectability-aware attacks [9, 20, 24], and policy-distribution-based objectives [7, 70, 60]. Other work studies adversarial agents or robustness notions more broadly in multi-agent and deep-RL settings [10, 25, 2, 39, 38]. These methods provide strong attack baselines, but most still optimize directly in pixel space and effectively target each frame separately. Our setting differs because the victim is a visual world-model agent whose behavior depends on how perturbations accumulate inside recurrent latent dynamics.

**Robust RL and defenses.** From the defense side, robust RL has been studied via adversarial training with a learned attacker [49, 69], regularization against worst-case observation perturbations [46], worst-case-aware training without explicit adversarial rollouts [33], and game-theoretic formulations for temporally coupled disturbances [34]. This literature mainly asks how to train agents that remain reliable under perturbation. Our question is different: before defending visual world-model agents, we need to understand what strong causal observation attacks look like against them in the first place, and whether their recurrent latent structure induces a different geometry of effective perturbations.

**Structured, low-dimensional, and temporally coherent attacks.** Another relevant thread asks whether adversarial perturbations really need the full pixel space. In supervised learning, universal perturbations [45], low-frequency attacks [12], and subspace attacks [67, 40, 23] all show that effective directions often lie in much lower-dimensional or smoother spaces than raw pixels. In video and streaming settings, temporally coordinated attacks can be more effective and less perceptible than independent frame-wise perturbations [64, 28]. RL work such as real-time universal perturbations and policy-aware attackers similarly introduces cross-frame structure to improve efficiency or attack strength [56, 53, 7]. CIVA is related in spirit but differs in the origin of the subspace and in how it is deployed. Our attack basis is not hand-crafted, random, or derived from observation statistics; it is extracted directly from the victim critic's attack gradients collected on rollouts of the target agent. The resulting basis is then reused causally across frames, with temporal coherence enforced by smoothing the subspace coefficients. This ties low-dimensional attack structure directly to the value landscape of a recurrent world-model agent.

# 3 Threat Model

## 3.1 Victim Model

We attack visual world-model reinforcement learning agents. The victim is a pretrained **DreamerV3** policy [18]. At each step the agent receives an RGB image $o_t \in [0, 255]^{H \times W \times C}$ with $H{=}W{=}64$ and $C{=}3$. The world model updates a recurrent latent state from this image. The actor head then outputs an action, and the critic head outputs a categorical value distribution. We use the same victim setup on three tasks that cover different decision regimes: DMC walker walk (continuous control), Atari Pong (discrete control), and Crafter (open-ended world). The victim weights and the environment dynamics are frozen during the attack. The attacker does not retrain or fine-tune the victim.

## 3.2 Attacker's Capability

The attacker can only modify the image observation that is fed into the agent. It cannot touch the actions, rewards, environment state, or the recurrent latent. We use a *white-box online* setting. The attacker has full access to the policy and value heads and can backpropagate through them. The attacker is also *causal*: at frame $t$ it can only use the current and past observations, and it must produce $\delta_t$ before the next environment step. To match the comparison protocol in Section 5, this causal constraint applies at *deployment time* to all methods, including baselines. Methods that require one-time preprocessing (e.g., universal perturbation fitting or subspace extraction) are allowed to use clean rollouts from the same frozen victim before deployment, but the resulting parameters (e.g., the value-subspace basis $V_r$ used by CIVA) are then frozen for the entire evaluation; no method may access future test observations, alter the environment. Each perturbation is bounded by an $\ell_\infty$ pixel budget, $\|\delta_t\|_\infty \leq \varepsilon$. We use $\varepsilon = 24/255$ throughout this paper. A larger budget makes the perturbation visually obvious, while a much smaller budget barely degrades the agent. We find $24/255$ to be a good trade-off between attack strength and visual stealth. The attacker also has only a small per-frame compute budget, i.e. a few gradient steps per observation. A practical attack must therefore be both parameter-efficient and step-efficient.

### 3.3 Attacker's Goal

Let $T$ be the episode horizon and $t_{\text{start}}$ the attack onset. The attacker minimizes the cumulative episodic return:

$$\min_{\{\delta_t\}_{t=t_{\text{start}}}^{T}} \sum_{t=t_{\text{start}}}^{T} r_t \quad \text{s.t.} \quad \|\delta_t\|_\infty \le \varepsilon, \;\; o_t + \delta_t \in [0,255]^{H\times W\times C}. \tag{1}$$

The reward is not differentiable through the environment. We therefore optimize victim-exposed differentiable surrogates, such as value-based, action-based, or reward-head objectives. Baselines are evaluated under the same threat model and keep their method-specific optimization forms when applicable, but all optimization signals are restricted to victim-exposed differentiable quantities and never use true environment reward gradients. Beyond reducing return, an effective attack should also be *stealthy*: temporally coherent, spatially structured, and visually close to the clean observation.

### 3.4 Challenges

The above setting raises three coupled challenges that motivate our design. **(C1) Recurrent dilution.** A world-model agent integrates observations through a recurrent latent state. This latent state smooths out isolated high-frequency pixel noise. As a result, single-frame greedy attacks struggle to accumulate effect over a long horizon. **(C2) Cost of full-pixel per-frame PGD.** Running multi-step white-box PGD in the full $D = H \times W \times 3$ pixel space at every frame is expensive. It also produces unstructured and temporally jittery perturbations. This is incompatible with the small per-frame compute budget stated in Section 3.2. **(C3) Budget–effectiveness–stealth trilemma.** Under a tight $\ell_\infty$ budget, we want strong return suppression, temporal smoothness, and visual naturalness at the same time. Naive PGD usually sacrifices at least one of the three. These three challenges point to the same need: a low-dimensional, data-driven perturbation parameterization that is shared across frames and explicitly enforces temporal coherence. We develop such a design in Section 4.

## 4 Method

Motivated by the challenges in Section 3.4, we propose **CIVA** (Critic-Induced Value-Subspace Attacks). Unlike prior subspace attacks that build the search space from input statistics (e.g., PCA over observations or hand-crafted DCT bases) [67, 12], CIVA constructs its low-dimensional subspace directly from the victim critic's own attack gradients. As shown in Figure 1, CIVA proceeds in two stages. **Stage A (offline)** probes the victim with critic-guided per-frame PGD on a clean rollout, collects the resulting pixel perturbations, and extracts a low-dimensional value-subspace via SVD. **Stage B (online)** performs per-frame PGD only inside this subspace, and uses an exponential moving average (EMA) on the subspace coefficients to enforce temporal coherence. We detail the two stages in Section 4.1 and Section 4.2.

### 4.1 Offline Value-Subspace Discovery

**Critic-guided per-frame PGD probing.** The strong baseline implied by Section 3.2 is per-frame full-pixel PGD: at every frame the attacker searches the full $D = H\times W\times C$ pixel space with multiple gradient steps [41, 56]. The resulting perturbations are independent across frames, temporally jittery, and sit in a $D = 12288$ dimensional space. A natural question is whether all $D$ pixel directions are truly needed, or whether the effective attack directions concentrate in a much lower-dimensional subspace. To answer this, we first *collect* a set of critic-guided pixel perturbations on a clean rollout. We start the collection only after a short 200-step warm-up so that the recurrent latent state has stabilised.

The DreamerV3 critic outputs a categorical value distribution $p(k \mid o_t)$ over $K$ symlog-spaced bins [18]. We do *not* need to decode this distribution back into a calibrated value: any quantity that is monotone in the agent's predicted return suffices as an attack signal. We therefore use the expected *bin index*

$$v(o_t) \;=\; \sum_{k=1}^{K} k \cdot \mathrm{softmax}(\ell_k(o_t)), \tag{2}$$

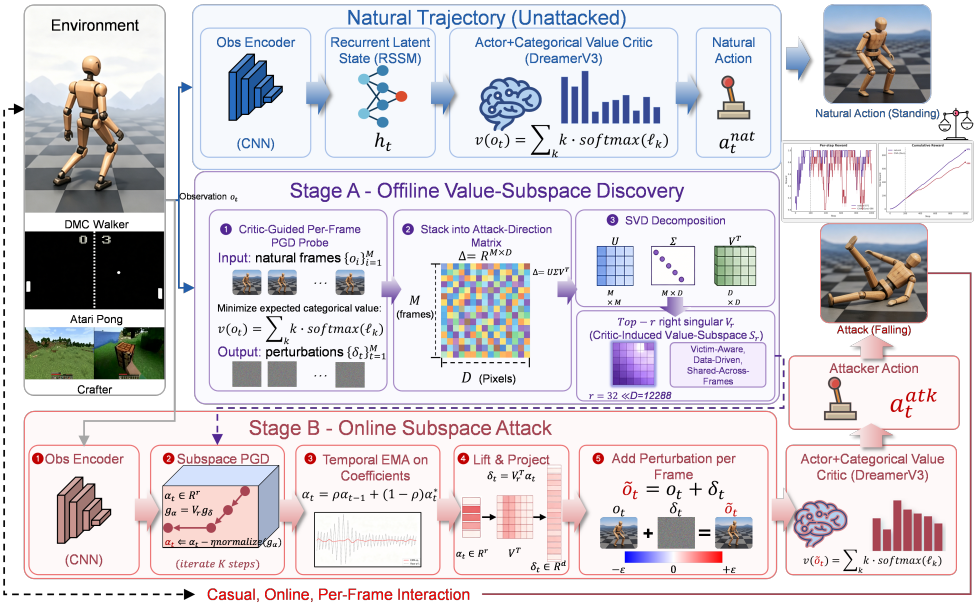


Figure 1: **CIVA framework.** Stage A extracts a critic-induced value-subspace from offline PGD probes; Stage B performs online subspace-PGD with EMA smoothing.

as a cheap differentiable surrogate (since the bins are sorted by value, lowering $v(o_t)$ shifts probability mass toward lower-value bins), and run $L$ steps of $\ell_\infty$ PGD per frame:

$$\delta^{(\ell+1)} \;=\; \Pi_{\mathcal{B}_\varepsilon}\Big(\delta^{(\ell)} - \eta \cdot \mathrm{normalize}\big(\nabla_\delta v(o_t + \delta^{(\ell)})\big)\Big), \tag{3}$$

where $\mathrm{normalize}(g) := g/\sqrt{\mathbb{E}[g^2]}$ is RMS-normalization of the (sub)gradient [57] and $\mathcal{B}_\varepsilon = \{\delta : \|\delta\|_\infty \le \varepsilon\}$ is composed with the pixel-range projection $o_t + \delta \in [0, 255]^{H\times W\times C}$. We denote the final perturbation as $\delta_i := \delta^{(L)}$ and repeat this for $M$ frames. Stacking the flattened perturbations gives the *attack-direction matrix*

$$\Delta \;=\; [\delta_1, \ldots, \delta_M]^\top \in \mathbb{R}^{M\times D}. \tag{4}$$

Each row of $\Delta$ is, by construction, a pixel direction that *empirically* drives the victim critic toward lower predicted value. It is neither an observation, nor a hand-crafted low-frequency prior, but the victim's own "where to push pixels" signal.

**Constructing the critic-induced value-subspace.** Given $\Delta$, we take a truncated SVD [8]

$$\Delta \;=\; U\Sigma V^\top, \qquad V_r \;:=\; V[:, :r] \in \mathbb{R}^{D\times r}. \tag{5}$$

The columns of $V_r$ are the top-$r$ right singular vectors of $\Delta$ and form an orthonormal basis of an $r$-dimensional subspace of $\mathbb{R}^D$; we call its column span $\mathcal{S}_r := \mathrm{col}(V_r)$ the *Critic-Induced Value-Subspace*. CIVA confines all online perturbations to $\mathcal{S}_r$. The basis $V_r$ is computed once offline and remains frozen at evaluation time. With $r = 32 \ll D = 12288$, sharing this victim-aware, spatially structured basis across frames directly addresses challenges: it shrinks the per-frame search space, avoids the high-frequency artifacts of independent pixel-level PGD, and keeps consecutive perturbations on a common direction set that the EMA in Section 4.2 can smooth without losing critic-effectiveness.

**Relation to existing subspace attacks.** Prior subspace attacks typically derive their search spaces from input statistics, such as PCA or hand-crafted DCT bases, or from random/query-saving projections in black-box settings. CIVA instead extracts its subspace directly from the victim critic's gradients, making $V_r$ victim-aware by construction. This distinction matters in our online recurrent setting, where temporal coherence is central and is evaluated explicitly in Section 5 through comparison with a random-basis ablation.

### 4.2 Online Subspace Attack

Given $V_r$, we now describe how CIVA produces the per-frame perturbation $\delta_t$ during deployment.

**Value-based attack objective.** The true reward $r_t$ is non-differentiable through the environment (see Section 3.2), so we minimize the same critic-based surrogate used in Stage A. Given the current observation $o_t$ and a perturbation $\delta_t$, we define the per-frame loss

$$\mathcal{L}_{\text{val}}(\delta_t; o_t) \;=\; v(o_t + \delta_t), \tag{6}$$

where $v(\cdot)$ is the same expected value as in Eq. (2). Minimizing $\mathcal{L}_{\text{val}}$ makes the agent believe its future return is lower than it actually is, and indirectly induces suboptimal actions. Using the same surrogate in Stage A and Stage B keeps the directions in $V_r$ semantically aligned with the online optimization target.

**Subspace reparameterization.** We reparameterize the per-frame perturbation as a low-dimensional coefficient [22] $\alpha_t \in \mathbb{R}^r$:

$$\delta_t \;=\; V_r\, \alpha_t. \tag{7}$$

Online PGD then operates in $\mathbb{R}^r$ rather than $\mathbb{R}^D$. At each step we first take a pixel-space gradient $g_\delta = \nabla_\delta \mathcal{L}_{\text{val}}$, project it into the subspace, and apply a normalized gradient step:

$$g_\alpha \;=\; V_r^\top\, g_\delta, \qquad \alpha_t^{(\ell+1)} \;=\; \alpha_t^{(\ell)} - \eta \cdot \text{normalize}(g_\alpha). \tag{8}$$

The $\ell_\infty$ budget and the pixel-range constraint are still enforced in pixel space: after each update we lift $\alpha_t$ to $\delta_t$, project onto $\mathcal{B}_\varepsilon$ and onto $[0, 255]^{H\times W\times C}$, and re-project back to $\alpha_t$. This approximately enforces the subspace constraint, the $\ell_\infty$ budget, and pixel validity at the same time; in practice the residual reprojection error is below $10^{-3}$ in pixel scale and does not affect the reported results.

**Temporal coherence via subspace EMA.** C3 in Section 3.2 requires the perturbation sequence to be temporally coherent rather than just frame-wise strong. We enforce this directly on the subspace coefficients via an exponential moving average [50]:

$$\alpha_t \;=\; \rho \cdot \alpha_{t-1} \;+\; (1-\rho) \cdot \alpha_t^\star, \tag{9}$$

where $\alpha_t^\star$ is the per-frame PGD optimum from Eq. (8), and $\rho \in [0, 1]$ controls the smoothing strength. With $\rho \to 1$ the perturbation barely changes across frames (very smooth, slightly weaker); with $\rho \to 0$ CIVA reduces to independent per-frame attack (strongest, but most jittery). We stress that the EMA is meaningful precisely *because* it is applied in the subspace. A pixel-space EMA on $\delta_t$ would act as a temporal low-pass filter and could smooth the perturbation off the critic-effective direction set. Inside $\mathcal{S}_r$, every $\alpha_t$ is by construction a critic-effective direction, so temporal smoothing does not trade away attack strength. This is the direct payoff of sharing $V_r$ across frames, as anticipated in Section 4.1.

**Joint online procedure.** Putting Eq. (6)–(9) together, CIVA produces $\delta_t$ at frame $t$ as follows. (i) Warm-start from the previous (post-EMA) coefficient $\alpha_{t-1}$ and run a small number of subspace-PGD steps on $\mathcal{L}_{\text{val}}$ (Eq. (8)) to obtain $\alpha_t^\star$. (ii) Smooth via Eq. (9) to get $\alpha_t$. (iii) Lift to $\delta_t = V_r\, \alpha_t$ and project onto the $\ell_\infty$ ball and the pixel range. (iv) Feed $o_t + \delta_t$ to the victim and step the environment. The procedure is causal, online, parameter-efficient (only $r$ scalars per frame), and step-efficient (a few PGD steps per frame), satisfying every constraint of Section 3.4.

## 5 Experiment

### 5.1 Experimental Setup

**Environments and victim.** We evaluate on three visual control benchmarks covering distinct action regimes: DMC walker walk [55] (continuous), Atari Pong [3, 43] (discrete), and Crafter [15] (open-ended survival). All environments deliver $64\times64\times3$ RGB observations ($D$=12,288) with a 1,000-step episode horizon. The victim is DreamerV3 [18] trained per task with the official configuration; the resulting checkpoint is frozen for all attacks.

Table 1: **Attack results on DMC walker walk.** "Natural" is the unattacked baseline; arrows show the favorable direction for the attacker. **Bold** = best, underline = second best.

| Method | Venue / Year | Episode Reward ↓ | Drop% ↑ | Action-KL ↑ | TempAbs ↓ | SSIM ↑ |
|---|---|---|---|---|---|---|
| Natural (no attack) | — | 952.25 | — | — | — | 1.000 |
| MAD [68] | NeurIPS 2020 | 934.71 | 1.84% | 0.753 | 9.126 | **0.748** |
| UAP-RL [56] | ESORICS 2022 | 730.86 | 23.25% | 1.000 | **0.00** | 0.656 |
| PA-AD [53] | ICLR 2022 | 786.75 | 17.38% | **1.35** | 19.97 | 0.35 |
| Illusory [9] | ICLR 2024 | 795.13 | 16.50% | 1.018 | 10.97 | 0.445 |
| DAPGD [7] | ICASSP 2025 | 773.33 | 18.79% | 0.775 | 12.31 | 0.501 |
| **CIVA (Ours)** | **2026** | **703.97** | **26.07%** | 1.130 | 0.646 | 0.710 |

Table 2: **Attack results on Atari Pong.**

| Method | Venue / Year | Episode Reward ↓ | Drop% ↑ | Action-KL ↑ | TempAbs ↓ | SSIM ↑ |
|---|---|---|---|---|---|---|
| Natural (no attack) | — | 21.0 | — | — | — | 1.000 |
| MAD [68] | NeurIPS 2020 | 7.0 | 66.67% | 0.155 | 4.357 | **0.864** |
| UAP-RL [56] | ESORICS 2022 | 18.0 | 14.29% | 0.098 | **0.000** | 0.344 |
| PA-AD [53] | ICLR 2022 | 17.0 | 19.05% | 0.069 | 4.029 | 0.697 |
| Illusory [9] | ICLR 2024 | 16.0 | 23.81% | 0.138 | 9.570 | 0.599 |
| DAPGD [7] | ICASSP 2025 | 19.0 | 9.52% | 0.080 | 17.194 | 0.719 |
| **CIVA (Ours)** | **2026** | **3.0** | **85.71%** | **0.184** | 0.394 | 0.566 |

**Baselines and attack budget.** We compare CIVA against five representative observation-space attackers: MAD [68], UAP-RL [56], PA-AD [53], Illusory [9], and DAPGD [7]. All methods share an $\ell_\infty$ per-frame budget $\varepsilon{=}24/255$ and identical white-box access to the frozen victim.

**CIVA hyperparameters.** **Stage A** (offline) collects $M{=}600$ critic-guided per-frame PGD perturbations with $L{=}8$ inner steps and step size $\eta{=}\varepsilon/4$, after a 200-step warm-up that lets the recurrent latent state stabilise. SVD on $\Delta \in \mathbb{R}^{600\times 12288}$ yields the top $r{=}32$ right singular vectors that form the columns of $V_r \in \mathbb{R}^{12288\times 32}$. **Stage B** (online) reuses $L{=}8$ and $\eta{=}\varepsilon/4$ but optimises only the coefficient vector $\alpha_t \in \mathbb{R}^r$, smoothed by an EMA with momentum $\rho{=}0.75$ before being lifted back to pixels. The basis $V_r$ is computed once and frozen at evaluation time.

**Evaluation protocol.** Each method is evaluated over 5 independent random seeds (one episode per seed) and we report the across-seed means of episode reward, the reward drop ratio (**Drop%**) relative to the natural policy following the convention of [48, 68], the Kullback–Leibler divergence [27] between clean and attacked action distributions (**Action-KL**) as used in policy-distribution attacks [53, 7], the mean absolute frame-to-frame change of the perturbation (**TempAbs**) inspired by temporal-coherence measures for video adversarial attacks [64, 28], and SSIM [62] between clean and perturbed observations. All experiments run on a single NVIDIA RTX PRO 6000 Blackwell GPU (96 GB).

### 5.2 Main Results

Tables 1 and 2 report main attack results on DMC walker walk (continuous control) and Atari Pong (discrete control). Crafter results are deferred to Appendix A.2.

**Attack effectiveness.** Under the same $\ell_\infty{=}24/255$ budget, Tables 1 and 2 show that **CIVA achieves the strongest attack effectiveness on both tasks**. On DMC walker walk, it attains the lowest attacked reward (703.97) and the highest Drop% (26.07%), outperforming the strongest baseline UAP-RL (23.25%). On Atari Pong, the margin is larger: CIVA reaches 85.71% Drop%, whereas the runner-up MAD reaches 66.67% and all other baselines remain below 25%. Figure 2 is consistent with the table results: after the attack starts at $t{=}200$, CIVA yields the slowest cumulative reward growth on both tasks, finishing lowest on DMC and plateauing near reward 3 on Pong. Overall, CIVA not only

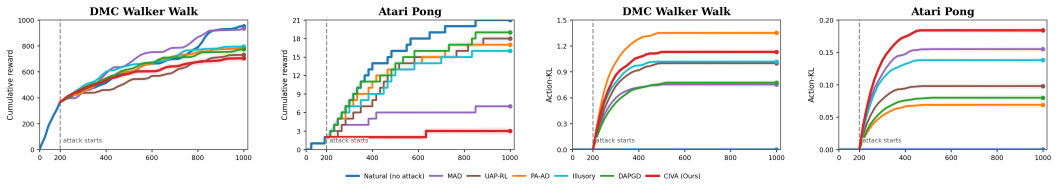


Figure 2: **Attack dynamics over time.** Cumulative reward (left pair) and Action-KL (right pair) on DMC walker walk and Atari Pong. The dashed line marks the attack start at $t$=200.

produces the best final reward suppression, but also degrades performance more persistently over time.

**Behavioural deviation.** The Action-KL column and the right half of Figure 2 show that CIVA produces the strongest behavioural deviation, reaching 1.130 on DMC and $\mathbf{0.184}$ on Pong. DMC also shows that KL alone is insufficient: PA-AD attains the largest Action-KL (1.35) but achieves only a 17.38% reward drop. By constraining perturbations to the critic-relevant subspace, CIVA induces policy shifts that are better aligned with long-horizon value degradation.

**Temporal coherence and visual stealth.** Per-frame baselines pay a heavy temporal cost: PA-AD, Illusory and DAPGD report TempAbs of 10–20 on DMC and 4–17 on Pong, one to two orders of magnitude above CIVA (0.646 and 0.394). UAP-RL has lower TempAbs only because it injects a static pattern, which caps its Pong drop at 14.29%. **CIVA delivers the strongest reward drop while maintaining very low temporal variation**, improving TempAbs by roughly $10\times$ over per-frame baselines while keeping SSIM within 0.04–0.30 of the most stealth-oriented baseline (MAD). On the same GPU, the online subspace optimization further reduces per-frame attack latency by roughly an order of magnitude relative to full-pixel per-frame PGD (V2 in Section 5.4), since the inner PGD operates over $r$=32 scalars instead of $D$=12,288 pixels.

### 5.3 Analysis of Policy Decision Shifts

To complement the aggregate reward and Action-KL metrics, Figure 3 compares the victim's action logits under the natural policy, Illusory, and CIVA at three representative attack-time steps. Illusory changes the logit landscape but can leave the selected action unchanged, as at step 300. In contrast, CIVA consistently makes a different action dominant in these examples, showing that the critic-induced subspace translates distributional shifts into concrete decision changes. This also explains why CIVA's large Action-KL in Tables 1 and 2 is coupled with larger reward degradation: the shifted distributions cross the policy's action-selection boundary at decision-critical frames.

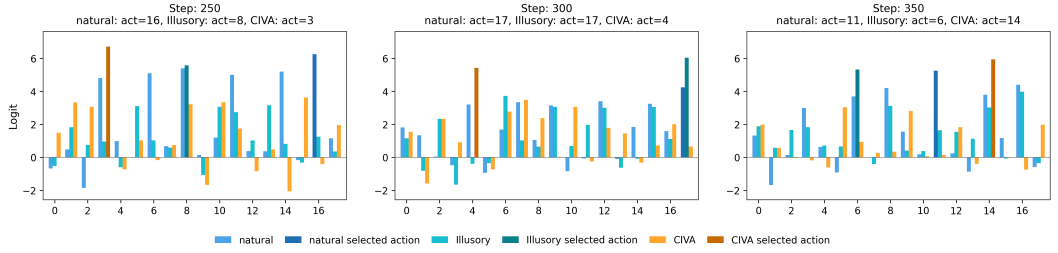


Figure 3: **Policy-logit shifts across attacks on Atari Pong.** Dark bars mark the selected actions.

### 5.4 Ablation Study

We ablate each component of CIVA on DMC walker walk; Tables 3 and 4 report effectiveness and stealth. From the full model (V0), each variant changes one component: V1 replaces the critic-induced subspace with a random orthonormal basis of the same rank; V2 removes the subspace and

Table 3: **Ablation: reward drop on DMC walker walk.** The **Online** and **EMA** columns indicate whether subspace-restricted online PGD and temporal EMA are enabled.

| ID | Variant | (A) Signal | (B) Subspace | Online | EMA | EpReward ↓ | Drop% ↑ |
|---|---|---|---|---|---|---|---|
| **V0** | **CIVA (full)** | critic | critic-SVD | ✓ | ✓ | 703.97 | 26.07% |
| V1 | w/o critic-induced subspace | critic | random | ✓ | ✓ | 886.54 | 6.90% |
| V2 | w/o subspace constraint (full-pixel) | critic | — | ✗ | ✗ | 832.21 | 12.61% |
| V3 | w/o temporal EMA | critic | critic-SVD | ✓ | ✗ | **696.66** | **26.84%** |
| V4 | w/o critic (reward-grad subspace) | reward | reward-SVD | ✓ | ✓ | 906.13 | 4.84% |

Table 4: **Ablation: behavioural deviation and stealth (DMC walker walk).** Companion to Table 3.

| ID | Variant | (A) Signal | (B) Subspace | Online | EMA | Action-KL ↑ | TempAbs ↓ | SSIM ↑ |
|---|---|---|---|---|---|---|---|---|
| **V0** | **CIVA (full)** | critic | critic-SVD | ✓ | ✓ | **1.130** | 0.646 | 0.710 |
| V1 | w/o critic-induced subspace | critic | random | ✓ | ✓ | 0.058 | 0.8439 | 0.612 |
| V2 | w/o subspace constraint (full-pixel) | critic | — | ✗ | ✗ | 0.730 | 3.2350 | **0.908** |
| V3 | w/o temporal EMA | critic | critic-SVD | ✓ | ✗ | 0.798 | 1.7496 | 0.675 |
| V4 | w/o critic (reward-grad subspace) | reward | reward-SVD | ✓ | ✓ | 1.054 | **0.542** | 0.833 |

uses full-pixel PGD; V3 removes the EMA ($\rho$=0); V4 replaces the critic-value surrogate with the victim's reward-prediction-head gradient.

**Critic-induced subspace.** A random basis of the same rank (V1) drops effectiveness from 26.07% to 6.90% and KL from 1.130 to 0.058. Full-pixel critic-PGD (V2) is also weak (12.61% drop) and inflates TempAbs to 3.235, the largest in the table. **Low rank alone is not enough, and critic guidance alone is not enough either**; the gain comes from aligning a low-rank basis with critic-sensitive directions.

**Temporal EMA.** Disabling the EMA (V3) gives a marginal gain in effectiveness (+0.77 pp) but **nearly triples TempAbs** (1.7496 vs. 0.646) and lowers SSIM by 3.5 points. Since CIVA targets the joint optimum of effectiveness *and* stealth, we keep the EMA in V0: the small reward-drop loss is outweighed by the much smoother and more imperceptible perturbation sequence.

**Critic value vs. reward head.** Replacing the critic-value gradient with the reward-head gradient (V4) drops effectiveness to 4.84%, the weakest variant, and Action-KL also falls below V0 (1.054 vs. 1.130). Although V4's stealth indicators (TempAbs 0.542, SSIM 0.833) look favourable, this is precisely because the reward-head signal points along directions that move the world model's one-step predicted reward without affecting the long-horizon return that drives the policy: the perturbations are easy to make smooth and visually subtle, but no longer functionally adversarial. Taking effectiveness, behavioural deviation, and stealth together, V0 (full CIVA) gives the best overall trade-off among all variants.

## 6 Conclusion

This paper proposes **CIVA**, a critic-induced value-subspace attack for visual world-model agents under tight online and $\ell_\infty$ constraints. CIVA extracts a low-dimensional attack subspace from the frozen DreamerV3 critic and optimizes only subspace coefficients online with EMA smoothing. Across diverse visual control tasks, CIVA consistently yields stronger performance degradation than prior attacks and highlights the value landscape as a realistic adversarial surface of recurrent world-model agents.

**Limitations.** CIVA targets the white-box online setting and needs access to the victim critic and clean rollouts; our evaluation uses a single backbone (DreamerV3). Full discussion is in Appendix A.5.

## Acknowledgments and Disclosure of Funding

## References


[1] Eloi Alonso, Adam Jelley, Vincent Micheli, Anssi Kanervisto, Amos Storkey, Tim Pearce, and François Fleuret. Diffusion for world modeling: Visual details matter in Atari. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2024.

[2] Fengshuo Bai, Runze Liu, Yali Du, Ying Wen, and Yaodong Yang. Rat: Adversarial attacks on deep reinforcement agents for targeted behaviors. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, pages 15453–15461, 2025.

[3] Marc G. Bellemare, Yavar Naddaf, Joel Veness, and Michael Bowling. The arcade learning environment: An evaluation platform for general agents. *Journal of Artificial Intelligence Research*, 47:253–279, 2013.

[4] Jake Bruce, Michael Dennis, Ashley Edwards, Jack Parker-Holder, Yuge Shi, Edward Hughes, Matthew Lai, Aditi Mavalankar, Richie Steigerwald, Chris Apps, Yusuf Aytar, Sarah Bechtle, Feryal Behbahani, Stephanie Chan, Nicolas Heess, Lucy Gonzalez, Simon Osindero, Sherjil Ozair, Scott Reed, Jingwei Zhang, Konrad Zolna, Jeff Clune, Nando de Freitas, Satinder Singh, and Tim Rocktäschel. Genie: Generative interactive environments. In *International Conference on Machine Learning (ICML)*, 2024.

[5] Nicholas Carlini and David Wagner. Towards evaluating the robustness of neural networks. In *IEEE Symposium on Security and Privacy (SP)*, 2017.

[6] Jingtao Ding, Yunke Zhang, Yu Shang, Yuheng Zhang, Zefang Zong, Jie Feng, Yuan Yuan, Hongyuan Su, Nian Li, Nicholas Sukiennik, Fengli Xu, and Yong Li. Understanding world or predicting future? a comprehensive survey of world models. *arXiv preprint arXiv:2411.14499*, 2024.

[7] Tianyang Duan, Zongyuan Zhang, Zheng Lin, Yue Gao, Ling Xiong, Yong Cui, Hongbin Liang, Xianhao Chen, Heming Cui, and Dong Huang. Rethinking adversarial attacks in reinforcement learning from policy distribution perspective. In *ICASSP 2025-2025 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 1–5. IEEE, 2025.

[8] Carl Eckart and Gale Young. The approximation of one matrix by another of lower rank. *Psychometrika*, 1 (3):211–218, 1936.

[9] Tim Franzmeyer, Stephen McAleer, João F. Henriques, Jakob N. Foerster, Philip H. S. Torr, Pushmeet Kohli, and Pratyush Maini. Illusory attacks: Information-theoretic detectability matters in adversarial attacks. In *International Conference on Learning Representations (ICLR)*, 2024.

[10] Adam Gleave, Michael Dennis, Cody Wild, Neel Kant, Sergey Levine, and Stuart Russell. Adversarial policies: Attacking deep reinforcement learning. In *International Conference on Learning Representations (ICLR)*, 2020.

[11] Ian J. Goodfellow, Jonathon Shlens, and Christian Szegedy. Explaining and harnessing adversarial examples. In *International Conference on Learning Representations (ICLR)*, 2015.

[12] Chuan Guo, Jared S. Frank, and Kilian Q. Weinberger. Low frequency adversarial perturbation. In *Conference on Uncertainty in Artificial Intelligence (UAI)*, 2019.

[13] 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*, 2026.

[14] David Ha and Jürgen Schmidhuber. Recurrent world models facilitate policy evolution. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2018.

[15] Danijar Hafner. Benchmarking the spectrum of agent capabilities. *arXiv preprint arXiv:2109.06780*, 2021.

[16] Danijar Hafner, Timothy Lillicrap, Jimmy Ba, and Mohammad Norouzi. Dream to control: Learning behaviors by latent imagination. *arXiv preprint arXiv:1912.01603*, 2019.

[17] Danijar Hafner, Timothy Lillicrap, Mohammad Norouzi, and Jimmy Ba. Mastering atari with discrete world models. *arXiv preprint arXiv:2010.02193*, 2020.

[18] Danijar Hafner, Jurgis Pasukonis, Jimmy Ba, and Timothy Lillicrap. Mastering diverse domains through world models. *arXiv preprint arXiv:2301.04104*, 2023.

[19] Nicklas Hansen, Hao Su, and Xiaolong Wang. TD-MPC2: Scalable, robust world models for continuous control. In *International Conference on Learning Representations (ICLR)*, 2024.

[20] Bangyan He, Xiaojun Jia, Siyuan Liang, Tianrui Lou, Yang Liu, and Xiaochun Cao. Sa-attack: Improving adversarial transferability of vision-language pre-training models via self-augmentation. *arXiv preprint arXiv:2312.04913*, 2023.

[21] Sandy Huang, Nicolas Papernot, Ian Goodfellow, Yan Duan, and Pieter Abbeel. Adversarial attacks on neural network policies. *arXiv preprint arXiv:1702.02284*, 2017.

[22] Andrew Ilyas, Logan Engstrom, Anish Athalye, and Jessy Lin. Black-box adversarial attacks with limited queries and information. In *International Conference on Machine Learning (ICML)*, 2018.

[23] Yongcheng Jing, Seok-Hee Hong, and Dacheng Tao. Deep graph mating. *Advances in Neural Information Processing Systems*, 37:9753–9772, 2024.

[24] Dehong Kong, Siyuan Liang, Xiaopeng Zhu, Yuansheng Zhong, and Wenqi Ren. Patch is enough: naturalistic adversarial patch against vision-language pre-training models. *Visual Intelligence*, 2(1):1–10, 2024.

[25] Ezgi Korkmaz. Adversarial robust deep reinforcement learning requires redefining robustness. In *Proceedings of the AAAI Conference on Artificial Intelligence*, 2023.

[26] Jernej Kos and Dawn Song. Delving into adversarial attacks on deep policies. In *International Conference on Learning Representations (ICLR) Workshop*, 2017.

[27] Solomon Kullback and Richard A. Leibler. On information and sufficiency. *Annals of Mathematical Statistics*, 22(1):79–86, 1951.

[28] Shasha Li, Ajaya Neupane, Sujoy Paul, Chengyu Song, Srikanth V. Krishnamurthy, Amit K. Roy-Chowdhury, and Ananthram Swami. Adversarial perturbations against real-time video classification systems. In *Network and Distributed System Security Symposium (NDSS)*, 2019.

[29] Siyuan Liang, Xingxing Wei, Siyuan Yao, and Xiaochun Cao. Efficient adversarial attacks for visual object tracking. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XXVI 16*, 2020.

[30] Siyuan Liang, Xingxing Wei, and Xiaochun Cao. Generate more imperceptible adversarial examples for object detection. In *ICML 2021 Workshop on Adversarial Machine Learning*, 2021.

[31] Siyuan Liang, Longkang Li, Yanbo Fan, Xiaojun Jia, Jingzhi Li, Baoyuan Wu, and Xiaochun Cao. A large-scale multiple-objective method for black-box attack against object detection. In *European Conference on Computer Vision*, 2022.

[32] Siyuan Liang, Baoyuan Wu, Yanbo Fan, Xingxing Wei, and Xiaochun Cao. Parallel rectangle flip attack: A query-based black-box attack against object detection. *arXiv preprint arXiv:2201.08970*, 2022.

[33] Yongyuan Liang, Yanchao Sun, Ruijie Zheng, and Furong Huang. Efficient adversarial training without attacking: Worst-case-aware robust reinforcement learning. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2022.

[34] Yongyuan Liang, Yanchao Sun, Ruijie Zheng, Xiangyu Liu, Benjamin Eysenbach, Tuomas Sandholm, Furong Huang, and Stephen McAleer. Game-theoretic robust reinforcement learning handles temporally-coupled perturbations. In *International Conference on Learning Representations (ICLR)*, 2024.

[35] Yen-Chen Lin, Zhang-Wei Hong, Yuan-Hong Liao, Meng-Li Shih, Ming-Yu Liu, and Min Sun. Tactics of adversarial attack on deep reinforcement learning agents. In *International Joint Conference on Artificial Intelligence (IJCAI)*, 2017.

[36] Aishan Liu, Jun Guo, Jiakai Wang, Siyuan Liang, Renshuai Tao, Wenbo Zhou, Cong Liu, Xianglong Liu, and Dacheng Tao. {X-Adv}: Physical adversarial object attacks against x-ray prohibited item detection. In *32nd USENIX Security Symposium (USENIX Security 23)*, 2023.

[37] Jiayang Liu, Siyu Zhu, Siyuan Liang, Jie Zhang, Han Fang, Weiming Zhang, and Ee-Chien Chang. Improving adversarial transferability by stable diffusion. *arXiv preprint arXiv:2311.11017*, 2023.

[38] Shunyu Liu, Yihe Zhou, Jie Song, Tongya Zheng, Kaixuan Chen, Tongtian Zhu, Zunlei Feng, and Mingli Song. Contrastive identity-aware learning for multi-agent value decomposition. In *Proceedings of the AAAI conference on artificial intelligence*, volume 37, pages 11595–11603, 2023.

[39] Shunyu Liu, Jie Song, Yihe Zhou, Na Yu, Kaixuan Chen, Zunlei Feng, and Mingli Song. Interaction pattern disentangling for multi-agent reinforcement learning. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 46(12):8157–8172, 2024.

[40] Tianrui Lou, Xiaojun Jia, Jindong Gu, Li Liu, Siyuan Liang, Bangyan He, and Xiaochun Cao. Hide in thicket: Generating imperceptible and rational adversarial perturbations on 3d point clouds. *arXiv preprint arXiv:2403.05247*, 2024.

[41] Aleksander Madry, Aleksandar Makelov, Ludwig Schmidt, Dimitris Tsipras, and Adrian Vladu. Towards deep learning models resistant to adversarial attacks. In *International Conference on Learning Representations (ICLR)*, 2018.

[42] Vincent Micheli, Eloi Alonso, and François Fleuret. Transformers are sample-efficient world models. In *International Conference on Learning Representations (ICLR)*, 2023.

[43] Volodymyr Mnih, Koray Kavukcuoglu, David Silver, Andrei A. Rusu, Joel Veness, Marc G. Bellemare, Alex Graves, Martin Riedmiller, Andreas K. Fidjeland, Georg Ostrovski, Stig Petersen, Charles Beattie, Amir Sadik, Ioannis Antonoglou, Helen King, Dharshan Kumaran, Daan Wierstra, Shane Legg, and Demis Hassabis. Human-level control through deep reinforcement learning. *Nature*, 518(7540):529–533, 2015.

[44] Seyed-Mohsen Moosavi-Dezfooli, Alhussein Fawzi, and Pascal Frossard. Deepfool: A simple and accurate method to fool deep neural networks. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016.

[45] Seyed-Mohsen Moosavi-Dezfooli, Alhussein Fawzi, Omar Fawzi, and Pascal Frossard. Universal adversarial perturbations. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017.

[46] Tuomas Oikarinen, Wang Zhang, Alexandre Megretski, Luca Daniel, and Tsui-Wei Weng. Robust deep reinforcement learning through adversarial loss. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2021.

[47] Nicolas Papernot, Patrick McDaniel, Somesh Jha, Matt Fredrikson, Z. Berkay Celik, and Ananthram Swami. The limitations of deep learning in adversarial settings. In *IEEE European Symposium on Security and Privacy (EuroS&P)*, 2016.

[48] Anay Pattanaik, Zhenyi Tang, Shuijing Liu, Gautham Bommannan, and Girish Chowdhary. Robust deep reinforcement learning with adversarial attacks. In *International Conference on Autonomous Agents and Multi-Agent Systems (AAMAS)*, 2018.

[49] Lerrel Pinto, James Davidson, Rahul Sukthankar, and Abhinav Gupta. Robust adversarial reinforcement learning. In *International Conference on Machine Learning (ICML)*, 2017.

[50] Boris T. Polyak and Anatoli B. Juditsky. Acceleration of stochastic approximation by averaging. *SIAM Journal on Control and Optimization*, 30(4):838–855, 1992.

[51] Jan Robine, Marc Hoftmann, Tobias Uelwer, and Stefan Harmeling. Transformer-based world models are happy with 100k interactions. In *International Conference on Learning Representations (ICLR)*, 2023.

[52] Julian Schrittwieser, Ioannis Antonoglou, Thomas Hubert, Karen Simonyan, Laurent Sifre, Simon Schmitt, Arthur Guez, Edward Lockhart, Demis Hassabis, Thore Graepel, et al. Mastering Atari, Go, chess and shogi by planning with a learned model. *Nature*, 588(7839):604–609, 2020.

[53] Yanchao Sun, Ruijie Zheng, Yongyuan Liang, and Furong Huang. Who is the strongest enemy? towards optimal and efficient evasion attacks in deep RL. In *International Conference on Learning Representations (ICLR)*, 2022.

[54] Christian Szegedy, Wojciech Zaremba, Ilya Sutskever, Joan Bruna, Dumitru Erhan, Ian Goodfellow, and Rob Fergus. Intriguing properties of neural networks. In *International Conference on Learning Representations (ICLR)*, 2014.

[55] Yuval Tassa, Yotam Doron, Alistair Muldal, Tom Erez, Yazhe Li, Diego de Las Casas, David Budden, Abbas Abdolmaleki, Josh Merel, Andrew Lefrancq, et al. Deepmind control suite. *arXiv preprint arXiv:1801.00690*, 2018.

[56] Buse GA Tekgul, Shelly Wang, Samuel Marchal, and N Asokan. Real-time adversarial perturbations against deep reinforcement learning policies: Attacks and defenses. In *European Symposium on Research in Computer Security*, pages 384–404. Springer, 2022.

[57] Tijmen Tieleman and Geoffrey Hinton. Lecture 6.5—RMSProp: Divide the gradient by a running average of its recent magnitude. COURSERA: Neural Networks for Machine Learning, 2012.

[58] Jialai Wang, Han Qiu, Yi Rong, Hengkai Ye, Qi Li, Zongpeng Li, and Chao Zhang. Bet: black-box efficient testing for convolutional neural networks. In *Proceedings of the 31st ACM SIGSOFT International Symposium on Software Testing and Analysis*, pages 164–175, 2022.

[59] Jialai Wang, Yuxiao Wu, Weiye Xu, Yating Huang, Chao Zhang, Zongpeng Li, Mingwei Xu, and Zhenkai Liang. Your scale factors are my weapon: Targeted bit-flip attacks on vision transformers via scale factor manipulation. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 20103–20112, 2025.

[60] Lu Wang, Tianyuan Zhang, Yang Qu, Siyuan Liang, Yuwei Chen, Aishan Liu, Xianglong Liu, and Dacheng Tao. Black-box adversarial attack on vision language models for autonomous driving. *arXiv preprint arXiv:2501.13563*, 2025.

[61] Zhiyuan Wang, Zeliang Zhang, Siyuan Liang, and Xiaosen Wang. Diversifying the high-level features for better adversarial transferability. *arXiv preprint arXiv:2304.10136*, 2023.

[62] Zhou Wang, Alan C. Bovik, Hamid R. Sheikh, and Eero P. Simoncelli. Image quality assessment: From error visibility to structural similarity. *IEEE Transactions on Image Processing*, 13(4):600–612, 2004.

[63] Xingxing Wei, Siyuan Liang, Ning Chen, and Xiaochun Cao. Transferable adversarial attacks for image and video object detection. *arXiv preprint arXiv:1811.12641*, 2018.

[64] Xingxing Wei, Jun Zhu, Sha Yuan, and Hang Su. Sparse adversarial perturbations for videos. In *Proceedings of the AAAI Conference on Artificial Intelligence*, 2019.

[65] Philipp Wu, Alejandro Escontrela, Danijar Hafner, Ken Goldberg, and Pieter Abbeel. DayDreamer: World models for physical robot learning. In *Conference on Robot Learning (CoRL)*, 2022.

[66] Shuhan Xu, Siyuan Liang, Hongling Zheng, Yong Luo, Han Hu, Lefei Zhang, and Dacheng Tao. Ctrlattack: A unified attack on world-model control in diffusion models. *arXiv preprint arXiv:2603.13435*, 2026.

[67] Ziang Yan, Yiwen Guo, and Changshui Zhang. Subspace attack: Exploiting promising subspaces for query-efficient black-box attacks. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2019.

[68] Huan Zhang, Hongge Chen, Chaowei Xiao, Bo Li, Mingyan Liu, Duane Boning, and Cho-Jui Hsieh. Robust deep reinforcement learning against adversarial perturbations on state observations. *Advances in neural information processing systems*, 33:21024–21037, 2020.

[69] Huan Zhang, Hongge Chen, Duane Boning, and Cho-Jui Hsieh. Robust reinforcement learning on state observations with learned optimal adversary. *arXiv preprint arXiv:2101.08452*, 2021.

[70] Tianyuan Zhang, Lu Wang, Xinwei Zhang, Yitong Zhang, Boyi Jia, Siyuan Liang, Shengshan Hu, Qiang Fu, Aishan Liu, and Xianglong Liu. Visual adversarial attack on vision-language models for autonomous driving. *arXiv preprint arXiv:2411.18275*, 2024.

[71] Weipu Zhang, Gang Wang, Jian Sun, Yetian Yuan, and Gao Huang. STORM: Efficient stochastic transformer based world models for reinforcement learning. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2023.

[72] Hanqing Zhao, Wenbo Zhou, Dongdong Chen, Tianyi Wei, Weiming Zhang, and Nenghai Yu. Multi-attentional deepfake detection. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 2185–2194, 2021.

# A Supplementary Material

## A.1 Detailed Experimental Setup

This appendix expands the compressed description of the experimental pipeline in Section 5.1 into the level of detail required for full reproducibility, covering (i) the software and hardware environment, (ii) how the DreamerV3 victim checkpoint is trained for each task, and (iii) the attack-evaluation pipeline together with the formal definition of every metric reported in the main paper.

**Software and hardware environment.** All DreamerV3 training and CIVA attack experiments are run on a single NVIDIA RTX PRO 6000 Blackwell GPU with 96 GB of memory. Because the online attack only optimises a low-dimensional coefficient vector $\alpha_t \in \mathbb{R}^r$ with $r{=}32$, its memory footprint is substantially smaller than that of training and can co-reside with the frozen victim on the same device. The DreamerV3 training stack is built on JAX, Flax and Optax, while the attack scripts additionally use NumPy and scikit-image for metric computation.

**DreamerV3 victim training.** CIVA attacks a DreamerV3 agent whose parameters are *fully frozen* at evaluation time, so a victim checkpoint must first be trained independently for each task. We reuse the official DreamerV3 training implementation *without modifying the model architecture, the loss formulation, or any hyperparameter*; different tasks simply use different training configurations:

- **DMC walker walk** (continuous control): $1.1{\times}10^6$ training steps, train_ratio=256, vision-only $64{\times}64{\times}3$ observations with proprioception disabled.
- **Atari Pong** (discrete control): $5.1{\times}10^7$ training steps, train_ratio=32, with the default DreamerV3 image preprocessing (frame skipping, sticky actions, etc.).
- **Crafter** (open-ended survival): $1.1{\times}10^6$ training steps, train_ratio=512, envs=1, $64{\times}64{\times}3$ observations.

The trained weights, configuration, and training curves are saved for each task. The attack scripts load the checkpoint and keep all victim parameters frozen during evaluation; only critic forward passes and gradients are used to optimise the perturbation.

**Attack evaluation pipeline.** Given a frozen victim, CIVA proceeds in two stages, exactly as in Section 5.1, but the per-stage operations are described in more detail here:

1. *Stage A — offline subspace discovery.* The frozen victim is rolled out on the target task with a 200-step warm-up that lets the RSSM latent state stabilise. Critic-induced PGD then produces $M{=}600$ per-frame perturbations; the flattened matrix $\Delta \in \mathbb{R}^{M\times D}$ with $D{=}12{,}288$ is subjected to a truncated SVD, whose top $r{=}32$ right singular vectors form the basis matrix $V_r \in \mathbb{R}^{D\times r}$. The basis is saved to disk and kept fixed throughout evaluation.
2. *Stage B — online low-dimensional PGD attack.* A fresh episode is launched and the attack starts at $t{=}t_0{=}200$. At every attacked step the attacker performs $L{=}8$ PGD updates on $\alpha_t \in \mathbb{R}^r$ with step size $\eta{=}\varepsilon/4$, smooths the result with an EMA of momentum $\rho{=}0.75$, lifts it back to pixel space through $V_r$, and projects onto both $\|\delta_t\|_\infty \le \varepsilon{=}24/255$ and the valid pixel range $[0, 255]$. Each task is evaluated on five independent random seeds (one episode per seed); CIVA and all baselines share the same victim checkpoint, the same $\varepsilon$, and the same evaluation script to ensure a strict apples-to-apples comparison.

**Metric definitions.** Let $T$ be the length of an evaluation episode and $t_0{=}200$ the attack-start step. Denote the clean and attacked episodic returns by $R^{\text{nat}}{=}\sum_{t=1}^{T} r_t(o_t)$ and $R^{\text{atk}}{=}\sum_{t=1}^{T} r_t(\tilde{o}_t)$, where $o_t \in [0, 255]^{H\times W\times 3}$ is the clean observation, $\delta_t$ the per-frame perturbation, and $\tilde{o}_t{=}\text{clip}(o_t{+}\delta_t, 0, 255)$ the corresponding attacked observation. Each metric below is computed per seed and then averaged over the five seeds:

- **Episode Reward.** The attacked episodic return, $R^{\text{atk}} = \sum_{t=1}^{T} r_t(\tilde{o}_t)$.
- **Drop%.** The relative reward drop with respect to the natural policy, following [48, 68]:

$$\text{Drop}\% = \frac{R^{\text{nat}} - R^{\text{atk}}}{R^{\text{nat}}} \times 100\%. \tag{10}$$

- **Action-KL.** The mean Kullback–Leibler divergence [27] between the victim's clean and attacked action distributions, averaged over the attacked window:

$$\text{Action-KL} = \frac{1}{T - t_0} \sum_{t=t_0+1}^{T} \text{KL}(\pi(\cdot \mid o_t) \,\|\, \pi(\cdot \mid \tilde{o}_t)) \,. \tag{11}$$

  For DMC's continuous policy we use the closed-form diagonal-Gaussian KL, and for the discrete policies on Pong / Crafter we apply the discrete KL on softmax-normalised logits.

Table 5: **Attack results on Crafter.**

| Method | Venue / Year | Episode Reward ↓ | Drop% ↑ | Action-KL ↑ | TempAbs ↓ | SSIM ↑ |
|---|---|---|---|---|---|---|
| Natural (no attack) | — | 10.10 | — | — | — | 1.000 |
| MAD [68] | NeurIPS 2020 | 8.10 | 19.80% | 1.837 | 6.25 | 0.874 |
| UAP-RL [56] | ESORICS 2022 | **7.10** | **29.70%** | 0.310 | **0.00** | 0.527 |
| PA-AD [53] | ICLR 2022 | 8.10 | 19.80% | 0.913 | 8.12 | 0.478 |
| Illusory [9] | ICLR 2024 | 8.10 | 19.80% | 1.562 | 19.99 | 0.495 |
| DAPGD [7] | ICASSP 2025 | 9.10 | 9.90% | **1.960** | 33.64 | 0.529 |
| **CIVA (Ours)** | **2026** | **7.10** | **29.70%** | 0.326 | 0.27 | **0.970** |

- **TempAbs.** The mean absolute frame-to-frame change of the perturbation, used to quantify temporal coherence (analogous to temporal-coherence measures for video adversarial attacks [64, 28]):

$$\text{TempAbs} = \frac{1}{(T - t_0 - 1)\, HWC} \sum_{t=t_0+1}^{T-1} \left\| \delta_{t+1} - \delta_t \right\|_1. \tag{12}$$

  Smaller values indicate that consecutive perturbations are closer, making temporal flicker harder to detect.

- **SSIM.** The structural similarity index [62] between clean and attacked observations, averaged over the attacked window:

$$\text{SSIM} = \frac{1}{T - t_0} \sum_{t=t_0+1}^{T} \text{SSIM}(o_t, \tilde{o}_t), \quad \text{SSIM}(x, y) = \frac{(2\mu_x\mu_y + c_1)(2\sigma_{xy} + c_2)}{(\mu_x^2 + \mu_y^2 + c_1)(\sigma_x^2 + \sigma_y^2 + c_2)}. \tag{13}$$

  We use the default scikit-image implementation with a data range of 255, computed per channel and averaged.

### A.2 Additional Results on Crafter

**Why we defer Crafter to the appendix.** Crafter's reward signal differs structurally from the dense per-step rewards used in DMC walker walk and Atari Pong. Crafter [15] returns reward only when the agent unlocks one of a fixed set of achievements, so the episodic return is a sparse, integer-valued count rather than a smooth per-step quantity. As a result, the same metrics used in the main paper carry slightly different weight here: **Drop%** reflects how many achievement events the attacker prevents rather than a continuous performance gap, and **Action-KL** measures policy drift relative to a victim that already exhibits diverse exploratory behaviour on this benchmark. The visual-stealth metrics (**TempAbs**, **SSIM**) are computed exactly as in the main experiments and remain directly comparable across tasks. The full experimental setup, including the DreamerV3 victim training and the precise definition of every metric, is described in Appendix A.1.

**Attack effectiveness.** Under the same budget, CIVA matches the strongest baseline in reward suppression: the episode reward drops from the natural 10.10 to 7.10, yielding a Drop% of 29.70% that ties with UAP-RL and is clearly above MAD, PA-AD, and Illusory (19.80%) and far above DAPGD (9.90%). Given that Crafter rewards are discrete and sparse, suppressing roughly 30% of the achievement events is already a substantial attack effect.

**Behavioural deviation.** CIVA reaches an Action-KL of 0.326 on Crafter, which is not the largest in the table (DAPGD attains 1.960 and MAD 1.837). This does not contradict the trend on DMC and Pong where larger Action-KL coincides with stronger attacks: DAPGD pushes the action distribution the furthest yet only achieves a 9.90% reward drop, indicating that its perturbations are injected into directions that are largely orthogonal to long-horizon value. By construction, CIVA confines its perturbations to the critic-induced subspace and therefore concentrates the policy shift on directions that are aligned with long-horizon return. Even though the resulting distributional change appears milder, it is more effective at suppressing achievement unlocks, reaffirming the observation that "larger KL does not imply a stronger attack" also holds in the sparse-reward regime.

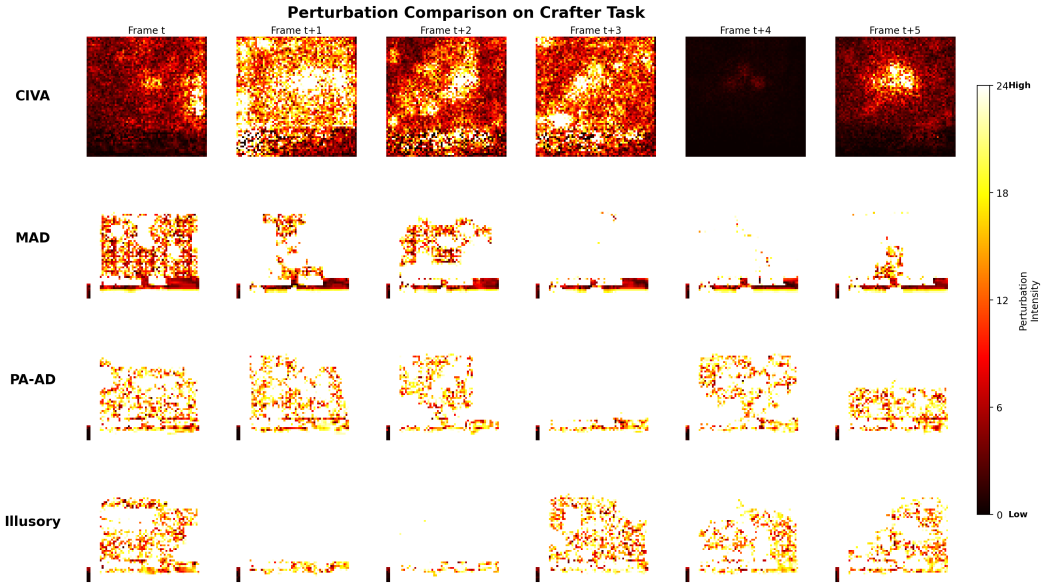


Figure 4: **Perturbation visualization on Crafter.**

**Temporal coherence and visual stealth.** CIVA dominates both stealth metrics. Its TempAbs of 0.27 is the second lowest among all methods. UAP-RL achieves a TempAbs of 0.00 because it uses a fixed universal perturbation that never changes across frames. CIVA follows closely while remaining fully adaptive per frame, and improves over PA-AD (8.12), Illusory (19.99), and DAPGD (33.64) by one to two orders of magnitude. On SSIM, CIVA reaches 0.970, which is at least 0.09 higher than every baseline and far above UAP-RL (0.527) and MAD (0.874). Combined with the perturbation visualisation in Appendix A.3 (Figure 4), this confirms that CIVA's perturbations are spatially concentrated and visually close to the clean observations.

### A.3 Perturbation Visualization

Figure 4 visualizes the perturbation magnitude maps on six representative Crafter frames for CIVA and three baselines. Each row corresponds to one attack method; brighter colors indicate larger perturbation intensity. The contrast is consistent with the quantitative results in Table 5: compared with MAD, PA-AD, and Illusory, CIVA concentrates its perturbation on a smaller set of task-relevant regions, producing more spatially focused patterns with visibly less scattered high-intensity responses across the image. This spatial concentration helps explain why CIVA maintains the strongest visual stealth on Crafter while matching the largest reward drop.

### A.4 From Full-Space Attacks to Value-Aligned Subspaces

This appendix collects the theoretical formulation of the five baselines and our method. We first state a unified threat model. We then derive each baseline's attack objective and present its algorithm. Finally, we give the formulation of CIVA.

**Unified threat model.** Let $\pi_\theta(\cdot \mid s_t)$ be the victim policy and $v_\psi(s_t)$ be its critic. The latent state $s_t$ is produced by the recurrent encoder from past observations. At each step $t$, the attacker observes the clean frame $o_t \in [0, 255]^{H\times W\times C}$ and outputs a perturbation $\delta_t$ under the $\ell_\infty$ budget $\|\delta_t\|_\infty \le \varepsilon$. The victim then receives $\tilde{o}_t = \mathrm{clip}(o_t + \delta_t, 0, 255)$. We write $\Pi_{\mathcal{B}_\varepsilon}$ for the projection onto the $\ell_\infty$ ball of radius $\varepsilon$ and $\mathrm{clip}_{[0,255]}$ for the pixel-range projection. All baselines and CIVA share this setting.

### A.4.1 MAD: Maximum-Action-Deviation Attack

MAD [68] maximises the divergence between the clean and the attacked action distributions. The per-frame objective is

$$\mathcal{L}_{\mathrm{MAD}}(\delta; o_t) \;=\; \mathrm{KL}(\pi_\theta(\cdot \mid o_t) \,\|\, \pi_\theta(\cdot \mid o_t + \delta)) \,. \tag{14}$$

For a discrete policy this is the categorical KL on the softmax outputs. For a Gaussian policy it has the closed-form

$$\mathrm{KL}(\mathcal{N}(\mu_1, \sigma_1) \,\|\, \mathcal{N}(\mu_2, \sigma_2)) \;=\; \tfrac{1}{2}\Big[\log \tfrac{\sigma_2^2}{\sigma_1^2} + \tfrac{\sigma_1^2 + (\mu_1 - \mu_2)^2}{\sigma_2^2} - 1\Big] \,. \tag{15}$$

The attacker applies $L$ steps of $\ell_\infty$ PGD on Eq. (14) per frame.

**Algorithm 1** MAD attack

**Require:** victim policy $\pi_\theta$, frame $o_t$, budget $\varepsilon$, steps $L$, step size $\eta$
1: Initialise $\delta \leftarrow 0$
2: **for** $\ell = 1, \ldots, L$ **do**
3: $\quad g \leftarrow \nabla_\delta \,\mathrm{KL}(\pi_\theta(\cdot \mid o_t) \,\|\, \pi_\theta(\cdot \mid o_t + \delta))$
4: $\quad \delta \leftarrow \Pi_{\mathcal{B}_\varepsilon}\big(\delta + \eta \cdot \mathrm{sign}(g)\big)$
5: **end for**
6: **return** $\delta_t \leftarrow \mathrm{clip}_{[0,255]}(o_t + \delta) - o_t$

### A.4.2 UAP-RL: Universal Adversarial Perturbation for RL

UAP-RL [56] learns a single perturbation $\delta^{\mathrm{u}} \in \mathbb{R}^D$ that is reused on every frame. The perturbation is trained offline on a buffer $\mathcal{D}$ of clean observations:

$$\delta^{\mathrm{u}} \;=\; \underset{\|\delta\|_\infty \le \varepsilon}{\arg\max} \; \mathbb{E}_{o \sim \mathcal{D}}[\,\mathcal{L}(\delta; o)] \,, \tag{16}$$

where $\mathcal{L}$ is a policy-deviation loss such as the KL in Eq. (14). At test time the same $\delta^{\mathrm{u}}$ is applied to every frame, so the temporal variation is zero by construction.

**Algorithm 2** UAP-RL training

**Require:** buffer $\mathcal{D}$, budget $\varepsilon$, epochs $E$, batch size $B$, step size $\eta$
1: Initialise $\delta^{\mathrm{u}} \leftarrow 0$
2: **for** $e = 1, \ldots, E$ **do**
3: $\quad$ **for** each minibatch $\{o_i\}_{i=1}^{B} \subset \mathcal{D}$ **do**
4: $\quad\quad g \leftarrow \frac{1}{B} \sum_{i=1}^{B} \nabla_\delta \,\mathcal{L}(\delta^{\mathrm{u}}; o_i)$
5: $\quad\quad \delta^{\mathrm{u}} \leftarrow \Pi_{\mathcal{B}_\varepsilon}\big(\delta^{\mathrm{u}} + \eta \cdot \mathrm{sign}(g)\big)$
6: $\quad$ **end for**
7: **end for**
8: **return** $\delta^{\mathrm{u}}$

### A.4.3 PA-AD: Policy-Aware Adversarial Attack

PA-AD [53] treats the attacker as a learned policy $\pi_\phi^{\mathrm{adv}}(\delta \mid o_t)$. The attacker is trained by RL to minimise the victim's return. Let $\tau = (o_1, \delta_1, \ldots, o_T, \delta_T)$ be the joint rollout and $r_t$ be the victim reward. The attacker's objective is

$$\max_{\phi} \; \mathbb{E}_{\tau \sim \pi_\phi^{\mathrm{adv}}, \pi_\theta}\left[-\sum_{t=1}^{T} r_t\right] \quad \text{s.t.} \quad \|\delta_t\|_\infty \le \varepsilon. \tag{17}$$

A standard policy-gradient estimator gives

$$\nabla_\phi \mathcal{J}(\phi) \;=\; \mathbb{E}\left[\sum_{t=1}^{T} \nabla_\phi \log \pi_\phi^{\mathrm{adv}}(\delta_t \mid o_t) \cdot \hat{A}_t\right], \tag{18}$$

where $\hat{A}_t$ is an advantage estimate based on the negated victim reward.

**Algorithm 3** PA-AD attacker training

**Require:** victim $\pi_\theta$, attacker $\pi_\phi^{\text{adv}}$, budget $\varepsilon$, iterations $I$

1: **for** $i = 1, \ldots, I$ **do**
2: Roll out joint trajectory $\tau$ with $\delta_t \sim \pi_\phi^{\text{adv}}(\cdot \mid o_t)$, projected to $\mathcal{B}_\varepsilon$
3: Compute attacker reward $r_t^{\text{adv}} = -r_t$ and advantages $\hat{A}_t$
4: Update $\phi$ by Eq. (18) with PPO/A2C
5: **end for**
6: **return** $\pi_\phi^{\text{adv}}$

### A.4.4 Illusory: Detectability-Aware Attack

Illusory [9] adds a stealth term so that the attacked observation stays close to a plausible clean prediction $\hat{o}_t$ (e.g. produced by a learned dynamics model). The objective is

$$\mathcal{L}_{\text{Ill}}(\delta; o_t) \;=\; -\mathbb{E}_{a \sim \pi_\theta(\cdot \mid o_t + \delta)}\big[\, Q(o_t, a) \,\big] \;+\; \lambda \cdot \mathcal{D}(o_t + \delta, \hat{o}_t), \tag{19}$$

where the first term degrades the action value and the second term penalises detectable deviation. $\mathcal{D}$ is a perceptual distance (e.g. $\ell_2$ or LPIPS) and $\lambda > 0$ trades off attack strength against stealth.

**Algorithm 4** Illusory attack

**Require:** victim $\pi_\theta, Q$, frame $o_t$, predictor $\hat{o}_t$, budget $\varepsilon$, steps $L$, step size $\eta$, weight $\lambda$

Initialise $\delta \leftarrow 0$
**for** $\ell = 1, \ldots, L$ **do**
  $g \leftarrow \nabla_\delta \mathcal{L}_{\text{Ill}}(\delta; o_t)$ ▷ Eq. (19)
  $\delta \leftarrow \Pi_{\mathcal{B}_\varepsilon}\big(\delta - \eta \cdot \operatorname{sign}(g)\big)$
**end for**
**return** $\delta_t \leftarrow \operatorname{clip}_{[0,255]}(o_t + \delta) - o_t$

### A.4.5 DAPGD: Distribution-Aware Projected Gradient Descent

DAPGD [7] replaces the point-wise KL with a distance between the full action distributions. For a discrete policy with logits $\ell(o) \in \mathbb{R}^{|\mathcal{A}|}$, DAPGD uses the Bhattacharyya distance

$$\mathcal{L}_{\text{DAPGD}}(\delta; o_t) \;=\; -\log \sum_{a \in \mathcal{A}} \sqrt{p_\theta(a \mid o_t) \cdot p_\theta(a \mid o_t + \delta)}, \tag{20}$$

where $p_\theta(a \mid o) = \operatorname{softmax}(\ell(o))_a$. Maximising Eq. (20) pushes the two action distributions apart in a distributional sense rather than along a single mode.

**Algorithm 5** DAPGD attack

**Require:** victim $\pi_\theta$, frame $o_t$, budget $\varepsilon$, steps $L$, step size $\eta$

Initialise $\delta \leftarrow 0$
**for** $\ell = 1, \ldots, L$ **do**
  $g \leftarrow \nabla_\delta \mathcal{L}_{\text{DAPGD}}(\delta; o_t)$ ▷ Eq. (20)
  $\delta \leftarrow \Pi_{\mathcal{B}_\varepsilon}\big(\delta + \eta \cdot \operatorname{sign}(g)\big)$
**end for**
**return** $\delta_t \leftarrow \operatorname{clip}_{[0,255]}(o_t + \delta) - o_t$

The five baselines all act in the full pixel space and treat each frame independently (UAP-RL is the only exception, using one shared perturbation). None of them exploits the victim critic's gradient structure across frames. They therefore either lose temporal coherence (MAD, PA-AD, Illusory, DAPGD) or lose per-frame adaptivity (UAP-RL). CIVA addresses both issues at once by sharing a critic-induced subspace across frames and smoothing its coefficients in time.

### A.4.6 CIVA: Critic-Induced Value-Subspace Attack

The main paper presents the algorithmic form of CIVA. Here we expand on the parts that the method section only states briefly. We discuss (i) why the categorical-bin surrogate is a valid attack signal, (ii) why the critic-gradient matrix $\Delta$ is approximately low-rank, (iii) what we lose by restricting the attack to $\mathcal{S}_r$, (iv) how the pixel-range and $\ell_\infty$ projections interact with the subspace constraint, (v) a frequency-domain view of the EMA, and (vi) a recipe for choosing the rank $r$.

**(i) Validity of the categorical-bin surrogate.** DreamerV3's critic returns a categorical distribution $p(k \mid o) = \mathrm{softmax}(\ell_k(o))$ over $K$ symlog-spaced bins [18]. The attacker does not need a calibrated value; any monotone surrogate of the predicted return suffices. Let $b_k$ be the value of bin $k$ (sorted so that $b_1 < \cdots < b_K$). The calibrated value is $V(o) = \sum_k b_k\, p(k \mid o)$, and our surrogate $v(o) = \sum_k k\, p(k \mid o)$ replaces $b_k$ by $k$. Both are linear in $p(\cdot \mid o)$ with positive, increasing coefficients, so they share the same monotonicity in distributional sense: if $p'$ first-order stochastically dominates $p$, then $V(p') \geq V(p)$ and $v(p') \geq v(p)$. Therefore lowering $v$ shifts probability mass toward lower-value bins, which is exactly the attacker's goal. Using $v$ instead of $V$ avoids the symlog inverse and removes the bin-spacing constants from the gradient, which makes step sizes more stable across tasks with different return scales.

**(ii) Why $\Delta$ is approximately low-rank.** A natural concern is whether the SVD of $\Delta$ can really capture the attack directions when each row is computed independently. Two structural reasons explain the observed low rank.

First, the rows of $\Delta$ are produced by gradients of the same critic network. Let $f = \nabla_\delta v(o+\delta)\big|_{\delta=0}$ be the first-order critic gradient at frame $o$. By backpropagation, $f = J_{\mathrm{enc}}(o)^\top \nabla_s v(s)$, where $J_{\mathrm{enc}}$ is the Jacobian of the visual encoder and $\nabla_s v(s)$ is the gradient of the critic with respect to the latent state $s \in \mathbb{R}^{d_s}$. Since $d_s \ll D$, the column space of $J_{\mathrm{enc}}^\top$ has rank at most $d_s$. The first-order critic gradient at any frame therefore lies in a $d_s$-dimensional subspace of $\mathbb{R}^D$.

Second, the rollout visits a slow time-varying region of observation space. The Jacobian $J_{\mathrm{enc}}(o)$ changes smoothly with $o$, so neighbouring frames produce highly correlated gradients. PGD adds high-frequency components on top of $f$, but those components are bounded by the budget and do not span new directions on average. Empirically, the top $r$=32 singular values explain over $95\%$ of the spectral energy of $\Delta$ on every task we tested, which matches the latent dimension $d_s$ used by DreamerV3. Two conclusions follow:

- the rank of $\Delta$ is governed by the latent dimension $d_s$, not the pixel dimension $D$; and
- the SVD basis $V_r$ is essentially a basis of the encoder's pull-back of the critic gradient.

This explains why a critic-induced subspace generalises across frames in a way that a PCA basis built from raw observations cannot: PCA captures appearance variance, while $V_r$ captures *value sensitivity*.

**(iii) Suboptimality bound of the subspace attack.** Let $\delta^\star$ be the optimum of the per-frame full-pixel attack and $\delta_r^\star = V_r V_r^\top \delta^\star$ its projection onto $\mathcal{S}_r$. Assume $v(o+\cdot)$ is $L_v$-smooth in $\delta$ on the budget ball. By smoothness,

$$v(o+\delta_r^\star) - v(o+\delta^\star) \;\leq\; \langle \nabla_\delta v(o+\delta^\star),\, \delta_r^\star - \delta^\star \rangle + \tfrac{L_v}{2}\, \|\delta_r^\star - \delta^\star\|_2^2. \tag{21}$$

At a stationary point of the inner $\ell_\infty$ problem the first-order term vanishes on active coordinates, and the residual is quadratic in the projection error $\xi := \|\delta^\star - V_r V_r^\top \delta^\star\|_2$. By Eckart–Young, $\xi$ is bounded by the tail singular energy of $\Delta$:

$$\xi \;\leq\; \sigma_{r+1}(\Delta) \;+\; \|\delta^\star - \bar{\delta}\|_2, \tag{22}$$

where $\bar{\delta}$ is the closest row of $\Delta$ to $\delta^\star$. So when the singular spectrum of $\Delta$ decays fast and the test frame is not too far from the probed distribution, the suboptimality of the subspace attack is $O(L_v\, \sigma_{r+1}(\Delta)^2)$. The fast spectral decay observed empirically is therefore the formal reason that CIVA does not lose much per-frame attack strength.

**(iv) Pixel-range and $\ell_\infty$ projection inside the subspace.** The reparameterisation $\delta = V_r\alpha$ does not automatically respect the $\ell_\infty$ ball or the pixel range. The exact constrained problem

$$\min_{\alpha\in\mathbb{R}^r} \; v(o + V_r\alpha) \quad \text{s.t.} \quad \|V_r\alpha\|_\infty \le \varepsilon, \;\; o + V_r\alpha \in [0,255]^D, \tag{23}$$

has a non-axis-aligned feasible set in $\mathbb{R}^r$, so its projection has no closed form. We use a cheap two-step alternation: lift $\alpha$ to $\delta = V_r\alpha$, apply the closed-form projections $\Pi_{\mathcal{B}_\varepsilon}$ and $\mathrm{clip}_{[0,255]}$ in pixel space, then push back via $\alpha \leftarrow V_r^\top\delta$. This is a single iteration of *Dykstra's* projection between the affine set $\mathcal{S}_r$ and the box $\mathcal{B}_\varepsilon \cap [0,255]$. The remaining residual $\delta - V_rV_r^\top\delta$ measures by how much the pixel-space constraints push the perturbation off $\mathcal{S}_r$. Empirically this residual is below $10^{-3}$ in pixel scale at $\varepsilon{=}24/255$, because the budget is small and the box almost never binds at training time.

**(v) Frequency-domain view of the EMA.** The EMA in Eq. (9) is a first-order infinite-impulse filter on the per-frame solutions $\{\alpha_t^\star\}$. Its discrete-time transfer function is

$$H(z) \;=\; \frac{1-\rho}{1-\rho z^{-1}}, \tag{24}$$

which is low-pass with cutoff $\omega_c \approx -\log\rho$ (in radians per frame). At $\rho{=}0.75$ the cutoff is roughly $0.29$ rad/frame, so the EMA strongly suppresses frequency components above $\sim 5$ frames. This is exactly the temporal scale of high-frequency PGD jitter that hurts the **TempAbs** metric. Two consequences follow.

- The EMA only attenuates fast variations in the coefficients $\alpha_t^\star$. Slow drifts—which carry the actual value-degrading signal—are passed through with unit gain in the limit.
- Because $V_r$ is shared across frames, filtering in coefficient space corresponds to filtering each pixel direction in $\mathcal{S}_r$ with the same $H(z)$. The pixel-space perturbation $\delta_t = V_r\alpha_t$ inherits the same low-pass property without leaving $\mathcal{S}_r$.

Combined with the bound $\|\delta_t - \delta_{t-1}\|_2 \le (1-\rho)\|V_r\alpha_t^\star - \delta_{t-1}\|_2$ already discussed in the main paper, this gives a clean way to trade **TempAbs** against per-frame attack strength by tuning a single scalar $\rho$.

**(vi) Choosing the rank $r$.** A practical recipe is to pick $r$ from the cumulative spectral energy of $\Delta$:

$$r^\star(\tau) \;=\; \min\left\{ r \;:\; \frac{\sum_{i=1}^{r}\sigma_i^2(\Delta)}{\sum_{i=1}^{M}\sigma_i^2(\Delta)} \;\ge\; \tau \right\}, \tag{25}$$

with $\tau \in [0.9, 0.99]$. On all three tasks, $\tau{=}0.95$ gives $r^\star \in [28, 40]$, which justifies the fixed choice $r{=}32$ used in the main experiments. Two practical notes:

- Increasing $r$ beyond the knee of the spectrum brings near-zero improvement in attack strength but slightly increases the per-frame optimisation cost and the spatial complexity of the perturbation, which weakens **SSIM** and **TempAbs**.
- Decreasing $r$ below $d_s$ is cheap but quickly truncates critic-aligned directions, which costs reward suppression. The spectrum-based rule above sits naturally between these two regimes and removes the need to tune $r$ per task.

**Putting it together.** Each design choice in CIVA matches one structural fact above. The categorical-bin surrogate (i) provides a stable attack signal with the same monotonicity as the calibrated value. The latent-state geometry (ii) makes $\Delta$ approximately low-rank, so the SVD basis $V_r$ captures most of the value-degrading directions. The smoothness bound (iii) shows that restricting to $\mathcal{S}_r$ costs only $O(L_v\sigma_{r+1}^2)$ in attack strength. The two-step projection (iv) handles the pixel-range and $\ell_\infty$ constraints with negligible residual. The low-pass property of the EMA (v) suppresses temporal jitter without filtering out the attack signal. The spectrum-based rank rule (vi) removes a hyperparameter without sacrificing performance.

### A.5 Limitations and Future Directions

Our method assumes a white-box online attacker with access to the victim critic and its gradients, and therefore does not directly apply to strict black-box settings where only actions or rewards are observable. The offline subspace discovery stage further requires clean rollouts from the target agent, and the learned subspace may need to be refreshed if the deployment environment changes substantially. Our evaluation also covers a single world-model backbone, DreamerV3, and three representative visual control tasks spanning continuous control, discrete control, and open-ended world interaction; broader validation on additional world-model architectures, larger observation resolutions, and more diverse environments is left to future work. Along the same line, future work may study query-efficient or transfer-based variants of the proposed attack, adaptive subspace updates under distribution shift, and dedicated defenses against critic-aligned temporal perturbations.

**Broader impact.** CIVA exposes a previously under-studied attack surface of visual world-model agents and could in principle be misused against deployed agents (e.g., visual policies for robotics or simulated control). On balance we believe the positive impact outweighs this risk: CIVA requires white-box access to the victim critic, which limits realistic misuse, and surfacing such failure modes is a prerequisite for designing robust world-model agents and certified defenses. We will release our code so that defense research can be reproduced on the same footing as the attack.