CONTINUOUS SEARCH ALLOCATION: JOINTLY SHAPING PROBES AND UPDATES FOR QUERY-EFFICIENT BLACK-BOX VIDEO ATTACKS

Anonymous authors
Paper under double-blind review

ABSTRACT

Zeroth-order video attacks reduce query cost by structuring *where* the search is performed: they restrict the search space, select key frames or regions, or reuse a transformed perturbation across frames. We study another degree of freedom: *how strongly* each coordinate is explored. Continuous search allocation (CSA) derives a graded coordinate-wise allocation from the input before any victim query and applies it to both the zeroth-order probes and the update. A controlled factorial shows that probe weighting alone increases query cost, whereas joint probe–update weighting reduces it, with interactions of -950 and -806 queries on two victims. We instantiate CSA with codec motion vectors obtained without victim queries or additional model inference; the advantage persists when the motion-warped base probe is replaced by i.i.d. noise. Across five victims and two zeroth-order optimizers, CSA lowers CANQ, which charges failures the full query budget, in all ten comparisons against a control matched on probe amplitude and per-iteration update L_1, by 24.8% on Kinetics-400 and 9.9% on UCF101. At the default budget, larger savings are associated with more concentrated allocations. Under a tight L_∞ bound, controlled sweeps reverse this ordering as high-weight coordinates increasingly saturate under projection.

1 INTRODUCTION

Zeroth-order (ZO) optimization is the standard tool for attacking a model whose gradients are inaccessible: the optimizer probes the model with perturbed inputs and reconstructs a descent direction from the returned scores (Wierstra et al., 2014; Spall, 1992; Ilyas et al., 2018). Its weakness is dimensionality. A gradient estimate assembled from a handful of random probes carries information about only a vanishing fraction of the coordinates, so the queries needed to make progress grow with input size. Video makes this acute: a single clip carries millions of heavily correlated spatiotemporal variables, so an optimizer that treats them as exchangeable spends much of its budget re-measuring redundant directions. The temporal structure that makes uniform exploration wasteful is also readable from the input, a natural signal for allocating search magnitude.

Existing black-box video attacks inject structure into this search, some derived from the input: a restricted set of reachable directions (Guo et al., 2019; Andriushchenko et al., 2020), a selected support of frames or regions (Wei et al., 2020; 2022; 2023; Haghjooei & Rezghi, 2024), or one pattern reused across frames under a geometric or motion transform (Li et al., 2021; Zhang et al., 2020) (Section 2).

These methods decide where the search goes. We study how search effort is distributed across coordinates. Image-domain ZO methods already adapt coordinate-wise search magnitudes, but from the query history, by sampling coordinates in proportion to the perturbation found so far (Chen et al., 2017) or by rescaling steps with running moment estimates (Chen et al., 2019). We instead ask whether the input can fix, before the first victim query, a *continuous allocation*: one weight per coordinate that sets how strongly it is explored. Unlike binary support selection, the allocation assigns graded search magnitudes over its nonzero support (Figure 1).

The allocation pays off as a coupled search mechanism. The natural way to use an input-derived prior is to bias the probe distribution toward the coordinates it favors. In a 2×2 factorial, this alone

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.

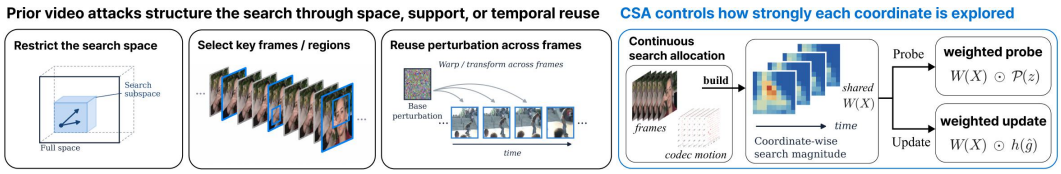


Figure 1: Left: three ways prior video attacks structure the search. Right: CSA computes one allocation $W(X)$ from the input and applies it to both the probe and the update.

raises query cost relative to the matched control. Applying the allocation to the update as well instead lowers it, with an interaction of -950 queries on I3D and -806 on X3D (Section 4.3.2). The gain persists after the motion-correlated probe is replaced by i.i.d. noise, which separates the allocation from the motion-warped probe of our default instantiation. Its size depends on two quantities fixed before the first query: the allocation's concentration and the L_∞ budget. Within-frame permutation preserves the query saving, indicating that CSA does not require pixel-exact alignment between the allocation and local motion (Section 4.3.2).

For video, the input already carries such a signal: a compressed clip stores blockwise motion vectors. The Change Magnitude Feature (CMF) turns their magnitude into a per-frame weight map $W(X)$, at no victim queries and no extra forward pass. ME-Sampler (Zhang et al., 2020) reads the same vectors but puts them to a different use: as a transformed-reuse method (Section 2) it correlates the perturbations of co-moving coordinates, whereas CMF uses the magnitude of motion to give individual coordinates different search magnitudes, a diagonal rather than an off-diagonal operator. Continuous search allocation (CSA) applies that same W to the update as well, so one allocation governs both where probes are spent and where perturbation accumulates (Section 3.4).

We evaluate on Kinetics-400 (Kay et al., 2017) across four video architectures and on UCF101 (Soomro et al., 2012) with a fifth, structurally different victim. Every matched comparison is against a constant weight that reproduces CMF's probe amplitude *and* its per-iteration update L_1 budget exactly and warps the same base probe, so it isolates how that L_1 budget is distributed; the L_2 difference it leaves is analyzed in Appendix C.1.2.

Our main contributions are as follows:

- *Continuous search allocation.* We identify the per-coordinate search magnitude as a design variable of zeroth-order video attacks. It is fixed from the input before the first victim query, unlike the query-history adaptation of image-domain ZO methods, and so it sets how search effort is distributed across coordinates before optimization begins.
- *Coupling between probe and update.* Controlled interventions on two victims show that the two roles of the allocation must act together: probe weighting alone raises query cost, joint weighting lowers it significantly on both, and the advantage widens as more probe pairs are averaged. The gain survives an i.i.d. base probe, and sweeps identify the allocation's concentration and the L_∞ bound as its operating factors, the advantage reversing under a tight bound.
- *A query-free video instantiation.* CMF turns codec motion vectors into such an allocation in one preprocessing pass. Against a uniform control matched on probe amplitude and per-iteration update L_1, it lowers CANQ in all ten victim–optimizer cells, by 24.8% pooled on Kinetics-400 and 9.9% on UCF101, and it tolerates coarse, blockwise motion estimates.

2 Related Work

We consider score-based attacks, which query the target and refine an adversarial example from its output scores (Jiang et al., 2019; Zhang et al., 2020; Wei et al., 2023). Transfer-based (Li et al., 2025; Xie et al., 2022), zero-query (Tang et al., 2026b), stylized (Tang et al., 2026a) and hard-label (Haghjooei & Rezghi, 2025) attacks each trade victim access or the norm bound for something else, so we position them here rather than comparing numbers. The earliest video attacks add no temporal structure, treating a clip as independent frames: V-BAD (Jiang et al., 2019) optimizes frames separately, and image attacks applied frame-by-frame (Ilyas et al., 2018) spread perturbation energy uniformly over time. Within the score-based setting, query-efficient video attacks largely

exploit temporal redundancy in three recurring ways: by restricting the search space, by selecting key frames or regions, or by reusing a transformed perturbation pattern across frames. CSA instead grades the search magnitude of each coordinate and shares that allocation between probing and updating. Appendix E places individual methods along these lines.

Restricting the search space. The first strategy confines the search to a lower-dimensional set of directions: a fixed orthonormal basis (Guo et al., 2019), localized square updates (Andriushchenko et al., 2020), a learned low-dimensional embedding (Tu et al., 2019), or an input-derived low-rank subspace from a tensor decomposition of the clip (Haghjooei & Rezghi, 2025); the last is a hard-label attack built on OPT-ATTACK and lies outside our score-based comparison.

Selecting key frames and regions. The second decides *where* perturbation may be placed. Heuristic (Wei et al., 2020), sparse (Wei et al., 2022) and reinforcement-learning (Wang et al., 2021; Yan & Wei, 2021) frame selection search with victim queries; AstFocus (Wei et al., 2023) picks key frames and key regions jointly with two agents, QEBB (Haghjooei & Rezghi, 2024) clusters the input's own frames and is query-free, and FDP (Jin et al., 2025) masks all but the top-k spectral regions. Both strategies remain uniform within the search space or support they retain.

Reusing a transformed perturbation across frames. The third strategy generates one base perturbation and reuses transformed copies of it over time, which ties coordinates together instead of removing them. Geo-TRAP (Li et al., 2021) samples a single static noise frame and warps it with a per-frame geometric transformation, so the search runs over the transformation parameters rather than over every pixel of every frame. Closest to us, ME-Sampler (Zhang et al., 2020) extracts motion from the video under attack and propagates shared noise values along motion trajectories, so co-moving coordinates are tied to a reused perturbation pattern. CSA reads the same codec motion vectors but makes a different operator out of them. Where the motion transform correlates coordinates, CMF instead turns the local *magnitude* of motion into a coordinate-wise scale and applies that same scale to the update, so unlike support selection the allocation is graded rather than binary; Section 3.4 formalizes the distinction.

Coordinate-wise and prior-guided ZO. Direction priors in zeroth-order attacks come from the query history, as in Bandits-TD (Ilyas et al., 2019), which adapts a time- and data-dependent prior online, or from a surrogate model, as in P-RGF (Cheng et al., 2019), which biases probes toward a transfer gradient. Per-coordinate effort has also been graded before, always from the query history: ZOO (Chen et al., 2017) samples coordinates for coordinate descent with probabilities proportional to the perturbation found so far, and ZO-AdaMM (Chen et al., 2019) rescales each coordinate's step with running moment estimates of past gradient estimates. Sign-based zeroth-order optimization has also been studied in ZO-signSGD (Liu et al., 2019); Section 3.4 instead analyzes one sign-update step under an input-derived, non-isotropic allocation. Outside attacks, shaping the sampling distribution or preconditioning the estimate is a standard lever on query complexity (Liu et al., 2020), and ZO-HessAware (Ye et al., 2018) preconditions with a structured Hessian approximation. Separable evolution strategies are the closest conceptual predecessor: separable NES (Wierstra et al., 2014) and separable CMA-ES (Ros & Hansen, 2008) likewise use coordinate-wise scales in both sampling and the mean update, but adapt those scales from queried fitness. CSA instead derives a fixed allocation from the attacked input before the first victim query, and studies its role in finite-difference attacks (Table 27).

3 METHODOLOGY

3.1 THREAT MODEL

We consider a score-based black-box attacker with query budget $Q_{\max}$ that obtains prediction scores but not the architecture, parameters, gradients or training data, operating in the digital domain against an inference API; physical-world constraints are out of scope. We write the output score vector as $f(X)$ and the predicted label as $F(X) = \arg\max_c f_c(X)$, and a clip as $X = \{x_t\}_{t=1}^{T}$ with $x_t \in \mathbb{R}^{H\times W\times C}$. Attacks are additive, $X_{\mathrm{adv}} = X + \Delta$ with $\Delta = \{\Delta_t\}_{t=1}^{T}$; the attacker builds Δ by query-based updates until $F(X_{\mathrm{adv}}) \neq y$ (untargeted) or $F(X_{\mathrm{adv}}) = y_{\mathrm{tar}}$ (targeted), or until $Q_{\max}$ is exhausted. Every run enforces an L_∞ bound ε_∞, stated in Section 4. All reported query counts are total model calls, probes and success checks included.

3.2 CONTINUOUS SEARCH ALLOCATION

CSA adds a control variable to an existing zeroth-order attack rather than replacing its estimator. Take any such attack: it draws a structured probe, queries the victim, and updates a perturbation. CSA inserts one input-derived quantity, a coordinate-wise allocation

$$W = W(X) \in [0,1]^d, \tag{1}$$

computed from the clean input alone, before the first victim query. Writing $D_W = \mathrm{diag}(W)$ and letting $P(z)$ be whatever base probe the underlying attack draws, W enters at exactly two points. It scales the probe, which sets each coordinate's amplitude when the loss is measured,

$$u = D_W\, P(z), \tag{2}$$

and the step, which sets how much perturbation each coordinate accumulates,

$$\Delta \leftarrow \Delta - \eta\, h(\hat{g}) \odot W, \tag{3}$$

where h is the optimizer's map from the gradient estimate to a direction; the estimator is unchanged.

With Π the projection onto the admissible set, Section 4.3.2 separates three ways of applying W:

$$\begin{aligned} \text{probe only:}\ & u = D_W P(z), && \Delta \leftarrow \Pi\big(\Delta - \eta\, h(\hat{g}_W)\big), \\ \text{update only:}\ & u = P(z), && \Delta \leftarrow \Pi\big(\Delta - \eta\, D_W h(\hat{g})\big), \\ CSA:\ & u = D_W P(z), && \Delta \leftarrow \Pi\big(\Delta - \eta\, D_W h(\hat{g}_W)\big), \end{aligned} \tag{4}$$

where $\hat{g}_W$ is the estimate formed from weighted probes. A switched-off side uses the matched constant: $\sqrt{\mathbb{E}[W^2]}$ at the probe, or weight 1 with step $\eta\,\mathbb{E}[W]$ at the update. With the matched control on both sides, these form the four arms of a 2×2 factorial; Section 4.3.2 evaluates them.

Properties of the allocation. CSA requires W to be graded and fixed from the clean input before any model call. Unlike binary support selection, nonzero-weight coordinates receive different search magnitudes, and $W_j{=}0$ leaves a coordinate unexplored (we impose no floor; Appendix B.2); unlike adaptive ZO methods, the magnitudes do not depend on query history or queried fitness. The same W is used for probing and updating, and weighting only one of Eq. 2 and Eq. 3 does not reproduce the gain (Section 4.3.2). Allocations also differ in how concentrated their mass is across coordinates; Section 5 quantifies this with effective dimensionality and relates it to when CSA helps. We next build $W(X)$ from codec motion (Section 3.3) and integrate it with NES and SPSA (Section 3.4).

3.3 A QUERY-FREE ALLOCATION FROM CODEC MOTION

The Change Magnitude Feature (CMF) provides a query-free video instantiation of $W(X)$: it converts the encoder's blockwise motion vectors, rather than a separately estimated optical flow, into a graded allocation profile, without victim queries or additional model inference (pipeline in Appendix A.2, Figure 2). Video that is already encoded carries these vectors; our clips are re-encoded to a fixed configuration first. Compressed video is organized into Groups of Pictures, each a short-range chain of motion vectors that can be composed into flows $\Phi_{t\to I}$ and $\Phi_{I\to t}$ tying every frame to its anchor. CMF consists of aggregation followed by framewise propagation. *Aggregation* produces the allocation: each P-frame's motion magnitude $A_t = \|\mathbf{V}_t\|_2$ is back-warped into anchor coordinates and accumulated, $W_I = \sum_t \mathrm{Warp}(A_t, \Phi_{t\to I})$, giving one map that integrates motion over the GOP. *Propagation* then renders it per frame along the motion trajectories, $W_{P_t} = \mathrm{Warp}(W_I, \Phi_{I\to t})$. The anchor map is normalized and smoothed before propagation, and the resulting profile is reshaped by an exponent γ ($\gamma{=}1$ throughout except in the γ sweep of Table 13), an efficiency–quality knob (Appendix C.1.2). Appendix A.1 gives the codec settings, the GOP structure and the multi-GOP handling, Appendix B.2 the normalization and smoothing, Appendix A.2 the procedure as Algorithm 1 with the per-GOP construction in Figure 3, and Appendix A.5 the measured preprocessing cost.

3.4 OPTIMIZATION WITH CSA

We instantiate CSA in a standard finite-difference attack, using CMF as $W(X)$. The equations below fully specify the method; pseudocode with the bookkeeping is deferred to Algorithm 2 in

Appendix A.3. Let $X = \{x_t\}_{t=1}^T$ denote the clean input and $\Delta = \{\Delta_t\}_{t=1}^T$ the accumulated perturbation; the adversarial input is $X_{\text{adv}} = X + \Delta$. We seek Δ minimizing a margin on $p = \text{softmax}(f(X+\Delta))$. Untargeted, $\mathcal{L} = p_{(1)} - p_{(2)}$, the gap between the two largest entries of p, which the attack drives toward zero; targeted with label t, $\mathcal{L} = \max\{0, \max_{c \neq t} p_c - p_t\}$, which reaches zero once t is ranked first. Eq. 7 descends on $\mathcal{L}$. Success is checked separately, by the predicted label, which is why the untargeted margin need not reference y.

Instantiating CSA on NES. We instantiate CSA on NES (Wierstra et al., 2014) with a motion-warped base probe. At each iteration we draw K Gaussian noise frames $z^{(i)} \sim \mathcal{N}(0, I)$, place each at the anchor of every group of pictures, and warp it along the codec motion vectors to the remaining frames, an operation we write T_m; CMF then scales the result coordinate-wise:

$$\widetilde{U}^{(i)} = D_W \, T_m z^{(i)}, \qquad D_W = \text{diag}\big(W(X)\big), \tag{5}$$

which is Eq. 2 with $P(z) = T_m z$. The base probe $T_m z^{(i)}$ is correlated along motion trajectories, as in motion-aware samplers (Section 2), and D_W sets each coordinate's probe amplitude according to its allocation weight. We then probe antithetic pairs and estimate the gradient:

$$\hat{g} = \frac{1}{2K\sigma} \sum_{i=1}^{K} \big(\mathcal{L}(f(X_+^{(i)})) - \mathcal{L}(f(X_-^{(i)}))\big)\widetilde{U}^{(i)}. \tag{6}$$

The update uses only the sign of $\hat{g}$. With the single antithetic pair we use (K=1), $\hat{g}$ is a scalar multiple of $\widetilde{U}$, and since $W \geq 0$, on every coordinate with $W_j > 0$ we have $\text{sign}(\hat{g})_j = s \cdot \text{sign}(T_m z)_j$, where s is the sign of the queried loss difference. At K=1, probe weighting changes the antithetic loss difference the optimizer observes, and update weighting sets the coordinate-wise step:

$$\Delta \leftarrow \Delta - \eta \cdot \text{sign}(\hat{g}) \odot W, \tag{7}$$

which is Eq. 3 with $h = \text{sign}$. The same W thus enters at two points: at sampling (Eq. 5) it shapes the perturbation at which the loss is queried, and at the update (Eq. 7) it sets how much each coordinate accumulates per step; Section 4.3.2 separates the two. **First-order behavior at K=1.** Take an i.i.d. Gaussian probe $u = D_a z$, $z \sim \mathcal{N}(0, I)$, with probe allocation a and update allocation b. For a fixed gradient g, the expected one-step decrease of the linearized loss has a closed form in the correlations $g_j a_j / \|g \odot a\|_2$, by the arcsine law for two Gaussian signs (Proposition 1). When no coordinate dominates that norm and the gradient energy is in addition roughly isotropic across coordinates, the matched arms of Eq. 4 therefore have relative first-order decreases $\mathcal{D}_{\text{CSA}}/\mathcal{D}_0 \simeq \rho$, $\mathcal{D}_{\text{update}}/\mathcal{D}_0 \simeq 1$ and $\mathcal{D}_{\text{probe}}/\mathcal{D}_0 \simeq \rho^{-1}$, where $\rho = \sqrt{\mathbb{E}[W^2]}/\mathbb{E}[W] = \sqrt{d/d_{\text{eff}}}$ and d_{eff} is the effective dimensionality of Section 5. The constant probe amplitude cancels, so the approximation recovers the factorial ordering and a larger joint-weighting advantage for more concentrated allocations. This is a one-step account for i.i.d. probes without projection; it predicts an ordering, not attack-level query counts.

This account is specific to the sign update at K=1. With more probe pairs the averaged estimator also keeps the covariance shaping that D_W induces. Under the same local linear approximation, $\hat{g} \approx (\nabla\mathcal{L}^\top \widetilde{U})\,\widetilde{U}$, whose expectation over the probe noise is $D_W C_P D_W \nabla\mathcal{L}(X)$, where $C_P = \mathbb{E}[P(z)P(z)^\top]$ is the covariance of whatever base probe is used ($C_P = T_m\,\mathbb{E}[zz^\top]\,T_m^\top$ for the motion-warped probe, and $C_P = I$ for an i.i.d. one). Probe weighting therefore changes the effective search geometry through a two-sided diagonal scaling rather than through an amplitude.

Remark 1 (A non-constant allocation is not a rescaling)**.** *Under this approximation a constant allocation $W = c\mathbf{1}$ gives $D_W C_P D_W = c^2 C_P$, which keeps the base probe's search geometry and only rescales it. A non-constant W instead applies a different factor on each side of C_P, and the result is in general not $c' C_P$ for any scalar c', so no step size or probe amplitude reproduces it.*

As K grows, $\hat{g}$ approaches this expectation and more of the shaping survives; the K sweep (Appendix C.2.1) measures the resulting CSA–control gap rather than isolating the probe side.

Projection. Every run enforces an L_∞ bound: we apply $\text{PROJECT}(\Delta_t, \varepsilon_\infty) := \text{clamp}(\Delta_t, -\varepsilon_\infty, \varepsilon_\infty)$ followed by clipping $X+\Delta$ to the valid input range $[0, 1]$. Because the update scales coordinate-wise with W, a heterogeneous allocation can interact asymmetrically with that

projection: high-weight coordinates receive larger increments and can therefore reach the boundary earlier than low-weight ones. Section 4 and Appendices C.4.1 and C.4.2 examine when this interaction becomes consequential.

Each iteration spends $2K$ probe queries plus one success check; we use $K{=}1$ and compute W once per clip.

The same probe scaling (Eq. 5) and update weighting apply to SPSA (Spall, 1992), with Rademacher noise drawn per group of pictures before warping (Section 4.3.1).

4 Experiments

4.1 Experimental Setup

On *Kinetics-400* (Kay et al., 2017) we attack four pretrained action recognizers from the PySlowFast zoo (Facebook AI Research, 2020), namely I3D (Carreira & Zisserman, 2017), SlowFast (SF) and SlowOnly (SO) (Feichtenhofer et al., 2019) and X3D (Feichtenhofer, 2020). On *UCF101* (Soomro et al., 2012) we attack an EfficientNet-B0 (Tan & Le, 2019) that classifies each frame and averages the per-frame logits over the clip, so the second benchmark also differs in victim architecture and not only in data. We compare against ME-Sampler and AstFocus, two query-based black-box video attacks, one from each of the reuse and support-selection families (Section 2): ME-Sampler (Zhang et al., 2020), which builds motion-correlated probe directions, and AstFocus (Wei et al., 2023), which localizes frames and regions by multi-agent reinforcement learning. All attacks run on $24{\times}128{\times}128$ clips under $Q_{\max}{=}5{,}000$. CSA and its controls use $\sigma{=}0.05$, $\eta{=}0.005$, $K{=}1$ antithetic pair and an L_∞ bound $\varepsilon_\infty{=}0.1$ on the $[0,1]$ pixel scale ($25.5/255$) unless stated otherwise; the baselines of Table 3 keep their own published bounds; checkpoints, hardware and hyperparameters are in Appendix B. Table 1 is the reference campaign; the diagnostics of Section 4.3.2 are separate paired reruns under the same protocol, so each comparison is read within its own campaign. The ablations and the prior-work comparison differ in clip population, not query budget: the ablations use 75 fixed clips over 3 seeds (5 on UCF101) to allow paired tests, while the prior-work comparison attacks one clip per correctly classified class in a single run (271–371 clips). Measured at $24{\times}128{\times}128$ on one clip per Kinetics-400 class (400 clips), clean top-1 accuracy is 39.25% for I3D, 52.75% for SlowFast, 35.25% for SlowOnly and 38.00% for X3D; the transformer victim of Appendix D.1 reaches 50.50% under the same protocol. These sit below the checkpoints' native-resolution accuracy because every attack here operates on $24{\times}128{\times}128$ inputs; all allocation comparisons are paired within one victim at that input.

4.2 Evaluation Metrics

Effectiveness is success rate (SR ↑, percentage of clips that reach the attack goal, misclassified or classified as the target, within budget) plus query cost; *stealthiness* is mean absolute perturbation (MAP ↓, i.e. $\|\Delta\|_1/d$ on the 0–255 scale), SSIM ↑ (Wang et al., 2004) and LPIPS ↓ (Zhang et al., 2018), computed over successful attacks only. Prior work reports ANQ ↓, the mean query count over successful attacks (Wei et al., 2022; Yan & Wei, 2021; Wei et al., 2023; Zhang et al., 2020; Jiang et al., 2023). Because it ignores failures, ANQ favors methods with low SR (Appendix D.2). We therefore also report comprehensive ANQ, $\text{CANQ} = (\text{SR}/100)\,\text{ANQ} + (1 - \text{SR}/100)\,Q_{\max}$, which charges each failure the full budget.

4.3 Evaluation on Video Models

4.3.1 Continuous Allocation Lowers Query Cost at a Matched Budget

The comparison changes how the exploration budget is distributed, not how much of it there is. At matched probe amplitude and update L_1, CSA reduces CANQ in all ten victim–optimizer–dataset cells, by 362 queries on average and up to 676 on X3D/SPSA (Table 1). A clip-clustered test of that overall effect, weighting cells equally and resampling each clip's optimizer pair as a unit, puts it at $[-476, -249]$ queries (95% CI, sign-flip $p < 10^{-4}$); cell by cell, eight differences reach $p < 0.05$ and four survive Holm correction (Appendix C.1.1). SR differs significantly in only one cell, so the dominant effect is lower query cost rather than higher attack success. Two controls rule out simpler

Table 1: Allocation ablation under NES and SPSA (Kinetics-400: n=75, 3 seeds; UCF101: n=75, 5 seeds). All settings use the same motion-warped base probe $T_m z$; *Uniform (L_1-matched)* matches CMF's probe RMS and per-iteration update L_1 budget. SR and CANQ: mean±s.d.; other metrics: seed means. Bold: best among the displayed L_1-budget settings, within each model and optimizer; Appendix C.1.2 compares against uniform arms at other step sizes.

		NES						SPSA					
Model	**Setting**	**SR(%)↑**	**ANQ↓**	**CANQ↓**	**MAP↓**	**SSIM↑**	**LPIPS↓**	**SR(%)↑**	**ANQ↓**	**CANQ↓**	**MAP↓**	**SSIM↑**	**LPIPS↓**
I3D	**CSA (CMF)**	**92.00±0.00**	**776**	**1114±14**	**2.53**	**0.9281**	0.0314	**92.00±1.33**	**821**	**1156±7**	**2.73**	0.9190	0.0341
	Uniform (L_1-matched)	91.11±1.54	1085	1433±15	2.99	0.9231	**0.0287**	89.33±0.00	1111	1526±11	3.10	**0.9199**	**0.0278**
SF	**CSA (CMF)**	82.22±0.77	**913**	**1640±33**	**2.63**	0.9301	0.0347	85.78±0.77	**910**	**1492±34**	**2.66**	**0.9247**	0.0357
	Uniform (L_1-matched)	**85.33±0.00**	1160	1723±15	2.86	**0.9373**	**0.0246**	**86.67±0.00**	1171	1682±31	3.12	0.9226	**0.0302**
SO	**CSA (CMF)**	**92.44±0.77**	**728**	**1051±30**	**1.99**	**0.9353**	0.0302	**93.33±1.33**	**859**	**1135±67**	**2.25**	**0.9257**	0.0346
	Uniform (L_1-matched)	91.56±0.77	1065	1398±18	2.51	0.9327	**0.0257**	87.56±0.77	1256	1722±27	2.94	0.9150	**0.0338**
X3D	**CSA (CMF)**	**95.56±0.77**	**557**	**754±10**	**2.31**	**0.9339**	0.0276	**94.67±0.00**	**539**	**777±15**	**2.35**	**0.9316**	0.0277
	Uniform (L_1-matched)	93.33±1.33	915	1188±15	2.76	0.9323	**0.0225**	88.00±1.33	968	1453±26	3.07	0.9196	**0.0273**
EffNet-B0 (UCF101)	**CSA (CMF)**	**73.33±1.63**	**1884**	**2715±67**	**3.33**	**0.8989**	0.0443	**67.47±2.23**	**1917**	**2922±52**	**3.21**	**0.9037**	**0.0385**
	Uniform (L_1-matched)	70.67±1.63	2143	2981±35	3.75	0.8771	**0.0384**	63.20±1.19	2266	3272±49	4.12	0.8596	0.0452

explanations. The *matched control*, *Uniform (L_1-matched)* (Appendix B.3), is a constant weight that reproduces CMF's probe RMS *and* its per-iteration update L_1 budget exactly on the same base probe $T_m z$; it rules out a gain from a larger probe or a larger per-iteration L_1 step. *Sparse Uniform* tests whether the gain comes from perturbing fewer coordinates. It places a random binary support covering a fraction $\mathbb{E}[W]$ of the coordinates and matches only the update L_1 (Appendix B.3). CSA beats it significantly in SR in 9 of 10 cells and in CANQ in 8 of 10. Distortion metrics are mixed: MAP favors CSA in all ten cells and SSIM in eight, while LPIPS favors the control in nine.

The advantage persists at matched MAP. Sweeping the step size of both allocations and comparing at equal MAP, CSA needs 346 fewer queries on I3D/NES (ΔCANQ 95% CI $[-612, -147]$), 787 on I3D/SPSA and 963 on X3D/NES (Appendix C.1.2).

The gain also persists with SPSA. SPSA (Spall, 1992) draws Rademacher probes, constant in magnitude before warping and weighting, so the weight alone shapes exploration. It reproduces the ordering (Table 1): the advantage over the matched control is larger under SPSA in every cell (mean -435 vs. -290).

4.3.2 How the Allocation Produces the Gain

We next isolate the two roles of W and test whether its spatial layout is necessary, on I3D/NES (Table 2) with everything else fixed.

(A) Probe and update weighting act jointly. CSA applies the same W twice: it scales the probe ($\widetilde{U} = D_W T_m z$) and gates the step (Eq. 7). At K=1 these act through distinct channels: probe weighting enters the scalar antithetic comparison that selects the signed base direction, while update weighting sets the coordinate-wise magnitude of the step that follows (Section 3.4). We run a 2×2 factorial on I3D over Eq. 4, replacing W with the matched constant at the probe, the update, both, or neither. Against weighting neither side, joint weighting lowers CANQ by 270 queries (p=0.012), while probe-only and update-only weighting raise it by 537 ($p<10^{-4}$) and 143 (p=0.15). The interaction is -950 queries (95% CI $[-1303, -635]$). The update-only arm already carries the larger L_2 step, so step energy alone is insufficient. The factorial replicates on X3D/NES (its own 75 clips, 3 seeds): the interaction is -806 queries (95% CI $[-1156, -486]$, $p=10^{-4}$), with probe-only again suffering most ($+542$ vs. $+537$ on I3D; Appendix C.2.2). This ordering is consistent with the first-order account of Section 3.4, which places only joint weighting above the control under approximately isotropic gradient energy. Separately, the CSA–control gap widens as more probe pairs are averaged, from 236 queries at K=1 to 654 at K=8 in a dedicated sweep, consistent with, but not isolating, the finite-difference reading of Section 3.4 (Appendix C.2.1).

(B) Separating allocation from motion-correlated probing. Every arm so far draws its base probe as $T_m z$, warped along the same motion the allocation is built from, so the two could be entangled. Replacing it with an i.i.d. probe, independent noise per frame and no warp, preserves the advantage over the matched control on I3D/NES (-455 queries, p=0.003) and X3D/NES (-577, p=0.0002),

Table 2: Where the allocation enters, and what its layout has to be (I3D/NES, 75 clips). The first block is the 2×2 over the two insertion points, against a matched-control rerun in the same campaign. The second keeps both positions weighted and varies the map, each row read against the CMF row above it in its own campaign. The i.i.d.-probe intervention of Section 4.3.2(B) is in Appendix C.2.3. All relaid-out maps carry CMF's exact value multiset. Paired statistics and campaign details are in Appendix C.3.1.

	Variant	SR (%)	CANQ ↓	SSIM ↑
Where W enters	probe ✓, update ✓ (CMF)	92.00	1122	0.9277
	probe ✓, update ✗	79.11	1929	0.9114
	probe ✗, update ✓	86.67	1535	0.9218
	probe ✗, update ✗ (matched control)	91.56	1392	0.9236
Placement, provenance and propagation	CMF	91.56±0.77	1133±13	0.9285±0.0017
	spatial permutation of W	91.56±0.77	1130±11	0.9175±0.0011
	smooth random layout	89.33±0.00	1243±22	0.9265±0.0032
	another clip's W	88.89±0.77	1359±21	0.9229±0.0009
	GOP propagation removed	90.67	1124	0.9241
	CMF	92.00±0.00	1122±11	0.9277±0.0022
	temporal permutation of W	92.44±0.63	987±22	0.9326±0.0010

Table 3: Comparison with prior black-box video attacks on Kinetics-400: one clip per correctly classified class, single run, $Q_{\max}$=5,000 (clip set differs from Table 1; Section 4). Baselines use their released implementations adapted to our victims (ME-Sampler ported from MXNet to PyTorch) and keep their own L_∞ bounds: 0.1 for CSA, 0.05 for AstFocus, 0.03 for ME-Sampler. All metrics here are thus under method-specific budgets; Table 1 is the L_1-matched comparison.

		Untargeted						Targeted					
Model	Method	SR ↑	ANQ ↓	CANQ ↓	MAP ↓	SSIM ↑	LPIPS ↓	SR ↑	ANQ ↓	CANQ ↓	MAP ↓	SSIM ↑	LPIPS ↓
I3D	ME-Sampler	81.97	375.97	1209.68	6.45	0.942	0.139	66.66	793.21	2195.75	6.74	0.915	0.198
	AstFocus	94.88	544.08	772.22	1.93	0.879	0.143	58.36	1897.35	3189.29	1.71	0.872	0.147
	CSA (CMF)	97.28	531.48	653.02	1.90	0.967	0.085	90.48	1070.98	1445.02	2.94	0.946	0.124
SF	ME-Sampler	81.67	398.67	1242.09	6.60	0.948	0.145	64.69	753.31	2252.82	6.87	0.917	0.207
	AstFocus	93.22	577.66	877.49	2.14	0.867	0.161	59.08	1469.76	2914.33	1.70	0.875	0.152
	CSA (CMF)	97.04	709.63	836.62	2.28	0.966	0.084	73.58	1213.98	2214.25	3.43	0.952	0.141
SO	ME-Sampler	87.54	342.66	922.96	6.50	0.945	0.148	74.16	685.89	1800.66	6.95	0.939	0.165
	AstFocus	94.43	519.39	768.96	1.80	0.869	0.147	69.29	1621.27	2658.88	1.70	0.882	0.146
	CSA (CMF)	99.09	559.72	600.13	2.03	0.968	0.085	90.88	1067.79	1426.41	3.31	0.943	0.147
X3D	ME-Sampler	82.29	440.54	1248.02	6.62	0.945	0.155	73.06	567.59	1761.68	6.62	0.943	0.157
	AstFocus	99.26	531.41	564.48	1.86	0.846	0.163	82.29	1643.90	2238.27	1.96	0.840	0.179
	CSA (CMF)	99.26	566.15	598.96	2.37	0.962	0.097	87.45	1094.68	1584.80	3.46	0.942	0.135

against -320 and -433 with the motion-correlated probe (Appendix C.2.3). On these two victims, the benefit does not require our default's motion-correlated probe.

(C) The allocation tolerates coarse spatial placement. Within-frame permutation preserves the query saving, indicating that CSA does not depend on pixel-level spatial precision and tolerates the coarse, blockwise precision of codec motion. Laying the same values out by another clip's map retains less of the advantage, but that control also changes the framewise allocation mass, so we do not read the difference as specifically about clip provenance (Appendix C.3.1). CMF's temporal ordering improves temporal coherence, while the CANQ difference under temporal shuffling is not statistically resolved (-135 queries in favor of shuffling, $[-318, +24]$, p=0.12); shuffling raises flow-compensated flicker E_{flow} by 19.1% (p=0.0001; Appendix C.3.2). Removing GOP propagation leaves both query cost and temporal coherence essentially unchanged (Table 2; Appendix C.3.2).

Raising W to a power γ moves CSA along its query–quality frontier; we use γ=1 throughout and report the sweep in Appendix C.1.2, Table 13.

4.3.3 Comparison with Prior Black-Box Video Attacks

Competitiveness at published operating points. Prior attacks use different L_∞ budgets, so Table 3 establishes practical competitiveness rather than controlled evidence for the allocation, which comes from Table 1 and Appendix C.1.2. At those operating points CSA attains the highest targeted SR on all four victims (by 5.2–23.8pp), the lowest CANQ in seven of the eight victim–setting pairs and the best untargeted SSIM and LPIPS on all four, and its MAP is below ME-Sampler's in every cell despite a looser bound. Our own sweeps show the advantage is budget-dependent (Appendices C.4.1 and C.4.2).

4.4 Defenses and Perceptual Quality

On UCF101 (Table 22), JPEG (q=75) (Guo et al., 2018; Das et al., 2018) is the strongest test-time defense, lowering CSA's SR to 44.74%; our CMF-aware randomized smoothing, following Cohen et al. (2019), lowers it to 61.90% at 95.45% benign accuracy. On SlowFast, CSA keeps above 94% SR under both JPEG and image smoothing (Table 24). In a descriptive user study (n=34), raters judged CSA's clips most similar to the originals on every backbone, against the two prior attacks at their bounds (Appendix D.5). Adversarial training, a second CMF-aware defense, temporal subsampling and full protocols are in Appendices D.3–D.5.

5 Discussion

At the default budget, CSA's query saving is largest among clips with concentrated allocations. Across the Kinetics-400 instances behind Table 1, the query saving falls from $1{,}240$ queries per instance in the most concentrated quintile of $d_{\text{eff}}(X) = (\sum_j W_j)^2 / \sum_j W_j^2$ to essentially zero in the most diffuse (Appendix C.3.3). Effective dimensionality is computed from the input alone, so it gives a query-free indication of where it is likely to matter; because $d_{\text{eff}}/d = 1/\rho^2$ ties it to the norm geometry of Appendix C.1.2, we treat the association as descriptive, not causal.

Concentration has opposite effects across perturbation regimes: at the default bound it is where the saving is largest, and under a tight bound it is most exposed to projection. CSA leads on both swept victims at $25.5/255$ and on X3D at $16/255$, while at $8/255$ the matched control is more query-efficient on both (Appendix C.4.1). At $8/255$, varying only the heterogeneity of the allocation with $\mathbb{E}[W]$ and the probe energy fixed, query efficiency falls monotonically as concentration rises (Appendix C.4.2). The projection clips the high-weight coordinates selectively: at the most concentrated setting 5.3% of the top quintile reaches the bound and none of the bottom quintile does, whereas a constant allocation clips both equally. These results are consistent with projection-induced truncation contributing to the low-budget crossover.

Scope of the evidence. Our controlled comparisons match probe amplitude and update L_1, the budget CSA redistributes; matched under L_2-family criteria, the two allocations are not separated (Appendix C.1.2), consistent with the larger L_2 a graded allocation carries at fixed L_1, though L_2 alone is insufficient: the update-only arm carries it and does not reproduce the gain. Evidence beyond convolutional victims is preliminary: MViTv2-S shows a directionally consistent but inconclusive query reduction (Appendix D.1). The defense study re-runs the attack against each defended pipeline without EOT or BPDA, and the subsampling test is a weaker post-hoc transfer test (Appendices D.3 and D.4). All attacks use $24\times128\times128$ inputs; native resolutions remain future work.

6 Conclusion

Prior black-box video attacks structure where the search goes: its space, its support, or its reuse across frames. We show that how search effort is distributed across coordinates is a design variable in its own right. In controlled factorials, applying the same input-derived allocation to both the probes and the update produces the query reduction we observe on both tested victims. Its size depends on the allocation's concentration and the perturbation regime, which clarifies where CSA helps and where projection limits it. Codec motion is one query-free realization; the design variable is the magnitude itself.

AI USE STATEMENT

We used generative AI tools for writing and coding assistance, including editing and condensing manuscript prose, drafting analysis and plotting scripts, scripting experiment orchestration, and assisting with the refinement of experimental analyses and controls. All AI-assisted suggestions were evaluated and verified by the authors, who made the final decisions on the methodology, experiments, interpretation, and manuscript content. Reported numbers were recomputed from the raw result files using author-inspected scripts, and all claims and citations were independently checked by the authors. We take responsibility for the final content of this work.

ETHICS STATEMENT

This paper studies an attack on video recognition models, so its findings are dual-use. We report it because the vulnerability it exposes is a property of how zeroth-order search is allocated rather than of any one deployed system, and because a defender can compute the same input-derived allocation, which motivates the input-conditioned defenses we evaluate. To that end we evaluate CSA against several test-time defenses and adversarial training, and we construct and report two CMF-aware defenses of our own (Section 4.4); one of them reduces success to 61.90% at 95.45% benign accuracy, without retraining.

All experiments were run offline against locally held checkpoints. No deployed or third-party service was queried, and no attack artifact was released against any live system. The datasets (Kinetics-400, UCF101) are public research benchmarks used under their original terms; we release no new video data.

The user study involved 34 adult participants who rated video pairs for visual similarity. It collected no personally identifying information, no demographic data and no sensitive attributes, and participation was voluntary and revocable. The study was reviewed by the relevant institutional review body and determined to be exempt. The clips shown were drawn from the public benchmarks above.

REPRODUCIBILITY STATEMENT

An anonymized implementation and the experiment scripts behind the reported tables are included in the supplementary material, with a README mapping tables to their entry points; victim checkpoints and result directories are not redistributed. All experiments use publicly available datasets (Kinetics-400, UCF101) and, for Kinetics-400, publicly released pretrained checkpoints, identified by name in Appendix B. That appendix also records the hardware and software versions, the victim architectures including how the UCF101 model aggregates frames, and every attack hyperparameter. The CMF construction is stated as Algorithm 1 and the optimization procedure as Algorithm 2, both in Appendix A, with the probe and update equations in Section 3.4; both procedures are deterministic given the input, the random seed and the execution configuration, and every seed and configuration we report is recorded. The ablation protocol, including how the matched control and *Sparse Uniform* are constructed and how their step sizes and probe amplitudes are equalized, is given in Section 4.3.1 and Appendix B.3, Table 6. Query counts reported throughout include probe queries and the per-iteration success check, so they can be compared directly against other work.

REFERENCES

Maksym Andriushchenko, Francesco Croce, Nicolas Flammarion, and Matthias Hein. Square attack: A query-efficient black-box adversarial attack via random search. In *European Conference on Computer Vision*, 2020.

Joao Carreira and Andrew Zisserman. Quo vadis, action recognition? a new model and the kinetics dataset. In *proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 6299–6308, 2017.

Pin-Yu Chen, Huan Zhang, Yash Sharma, Jinfeng Yi, and Cho-Jui Hsieh. Zoo: Zeroth order optimization based black-box attacks to deep neural networks without training substitute models. In *Proceedings of the 10th ACM workshop on artificial intelligence and security*, pp. 15–26, 2017.

Xiangyi Chen, Sijia Liu, Kaidi Xu, Xingguo Li, Xue Lin, Mingyi Hong, and David Cox. Zo-adamm: Zeroth-order adaptive momentum method for black-box optimization. *Advances in neural information processing systems*, 32, 2019.

Shuyu Cheng, Yinpeng Dong, Tianyu Pang, Hang Su, and Jun Zhu. Improving black-box adversarial attacks with a transfer-based prior. *Advances in neural information processing systems*, 32, 2019.

Jeremy M. Cohen, Elan Rosenfeld, and J. Zico Kolter. Certified adversarial robustness via randomized smoothing. In *Proceedings of the 36th International Conference on Machine Learning*, 2019.

Nilaksh Das, Madhuri Shanbhogue, Shang-Tse Chen, Fred Hohman, Siwei Li, Li Chen, Michael E. Kounavis, and Duen Horng Chau. Shield: Fast, practical defense and vaccination for deep learning using jpeg compression. *arXiv preprint arXiv:1802.06816*, 2018.

Facebook AI Research. SlowFast Model Zoo - GitHub Repository. `https://github.com/facebookresearch/SlowFast/blob/main/MODEL_ZOO.md`, 2020. Accessed: 2025-04-06.

Christoph Feichtenhofer. X3d: Expanding architectures for efficient video recognition. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 203–213, 2020.

Christoph Feichtenhofer, Haoqi Fan, Jitendra Malik, and Kaiming He. Slowfast networks for video recognition. In *Proceedings of the IEEE/CVF international conference on computer vision*, pp. 6202–6211, 2019.

Chuan Guo, Mayank Rana, Moustapha Cisse, and Laurens van der Maaten. Countering adversarial images using input transformations. In *International Conference on Learning Representations*, 2018.

Chuan Guo, Jacob R. Gardner, Yurong You, Andrew Gordon Wilson, and Kilian Q. Weinberger. Simple black-box adversarial attacks. In *International Conference on Machine Learning*, 2019.

Kimia Haghjooei and Mansoor Rezghi. Qebb: A query-efficient black-box adversarial attack on video recognition models based on unsupervised key frame selection. In *ICPRAM*, pp. 288–295, 2024.

Kimia Haghjooei and Mansoor Rezghi. Tenad: A tensor-based low-rank black box adversarial attack for video classification. *arXiv preprint arXiv:2504.01228*, 2025.

Andrew Ilyas, Logan Engstrom, Anish Athalye, and Jessy Lin. Black-box adversarial attacks with limited queries and information. In *International conference on machine learning*, pp. 2137–2146. PMLR, 2018.

Andrew Ilyas, Logan Engstrom, and Aleksander Madry. Prior convictions: Black-box adversarial attacks with bandits and priors. In *International Conference on Learning Representations*, 2019.

Kaixun Jiang, Zhaoyu Chen, Hao Huang, Jiafeng Wang, Dingkang Yang, Bo Li, Yan Wang, and Wenqiang Zhang. Efficient decision-based black-box patch attacks on video recognition. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 4356–4366, 2023. doi: 10.1109/ICCV51070.2023.00404.

Linxi Jiang, Xingjun Ma, Shaoxiang Chen, James Bailey, and Yu-Gang Jiang. Black-box adversarial attacks on video recognition models. In *Proceedings of the 27th ACM International Conference on Multimedia*, pp. 864–872, 2019.

Teng Jin, Ziwen He, Zhangjie Fu, Songping Wang, Yueming Lyu, and Yufei Shi. Frequency domain distributed perturbations: Towards query-efficient black-box adversarial video attack. In *Proceedings of the 33rd ACM International Conference on Multimedia*, pp. 8360–8368, 2025.

Will Kay, Joao Carreira, Karen Simonyan, Brian Zhang, Chloe Hillier, Sudheendra Vijayanarasimhan, Fabio Viola, Tim Green, Trevor Back, Paul Natsev, et al. The kinetics human action video dataset. *arXiv preprint arXiv:1705.06950*, 2017.

Didier Le Gall. Mpeg: A video compression standard for multimedia applications. *Communications of the ACM*, 34(4):46–58, 1991.

Ping Li, Jianan Ni, and Bo Pang. Temporal consistency constrained transferable adversarial attacks with background mixup for action recognition. *arXiv preprint arXiv:2505.17807*, 2025.

Shasha Li, Abhishek Aich, Shitong Zhu, Salman Asif, Chengyu Song, Amit Roy-Chowdhury, and Srikanth Krishnamurthy. Adversarial attacks on black box video classifiers: Leveraging the power of geometric transformations. *Advances in Neural Information Processing Systems*, 34:2085–2096, 2021.

Yanghao Li, Chao-Yuan Wu, Haoqi Fan, Karttikeya Mangalam, Bo Xiong, Jitendra Malik, and Christoph Feichtenhofer. Mvitv2: Improved multiscale vision transformers for classification and detection. In *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 4794–4804. IEEE, 2022.

Ming Liou. Overview of the p$\times$ 64 kbit/s video coding standard. *Communications of the ACM*, 34 (4):59–63, 1991.

Sijia Liu, Pin-Yu Chen, Xiangyi Chen, and Mingyi Hong. signsgd via zeroth-order oracle. In *International conference on learning representations*, 2019.

Sijia Liu, Pin-Yu Chen, Bhavya Kailkhura, Gaoyuan Zhang, Alfred O Hero III, and Pramod K Varshney. A primer on zeroth-order optimization in signal processing and machine learning: Principals, recent advances, and applications. *IEEE Signal Processing Magazine*, 37(5):43–54, 2020.

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*, 2018.

Raymond Ros and Nikolaus Hansen. A simple modification in cma-es achieving linear time and space complexity. In *International conference on parallel problem solving from nature*, pp. 296–305. Springer, 2008.

Heyuan Shi, Binqi Zeng, Yu Zhan, Rongkai Liu, Yulin Yang, Li Chen, Chao Hu, and Ying Fu. Cubeagent: Efficient query-based video adversarial examples generation through deep reinforcement learning. *Journal of Systems and Software*, 226:112437, 2025.

Jianxin Song, Dan Yu, Hongfei Teng, and Yongle Chen. RLVS: A reinforcement learning-based sparse adversarial attack method for black-box video recognition. *Electronics*, 14(2):245, 2025. doi: 10.3390/electronics14020245.

Khurram Soomro, Amir Roshan Zamir, and Mubarak Shah. Ucf101: A dataset of 101 human actions classes from videos in the wild. *arXiv preprint arXiv:1212.0402*, 2012.

James C. Spall. Multivariate stochastic approximation using a simultaneous perturbation gradient approximation. *IEEE Transactions on Automatic Control*, 37(3):332–341, 1992.

Mingxing Tan and Quoc Le. Efficientnet: Rethinking model scaling for convolutional neural networks. In *International conference on machine learning*, pp. 6105–6114. PMLR, 2019.

Duoxun Tang, Yuxin Cao, Xi Xiao, Derui Wang, Sheng Wen, and Tianqing Zhu. Query-efficient video adversarial attack with stylized logo on service computing. *IEEE Transactions on Dependable and Secure Computing*, 2026a.

Duoxun Tang, Xi Xiao, Guangwu Hu, Kangkang Sun, Xiao Yang, Dongyang Chen, Qing Li, Yongjie Yin, and Jiyao Wang. Featurefool: Zero-query fooling of video models via feature map. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 42268–42279, 2026b.

Chun-Chen Tu, Paishun Ting, Pin-Yu Chen, Sijia Liu, Huan Zhang, Jinfeng Yi, Cho-Jui Hsieh, and Shin-Ming Cheng. AutoZOOM: Autoencoder-based zeroth order optimization method for attacking black-box neural networks. *Proceedings of the AAAI Conference on Artificial Intelligence*, 33 (01):742–749, 2019. doi: 10.1609/aaai.v33i01.3301742.

Zeyuan Wang, Chaofeng Sha, and Su Yang. Reinforcement learning based sparse black-box adversarial attack on video recognition models. *arXiv preprint arXiv:2108.13872*, 2021.

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.

Xingxing Wei, Huanqian Yan, and Bo Li. Sparse black-box video attack with reinforcement learning. *International Journal of Computer Vision*, 130(6):1459–1473, 2022.

Xingxing Wei, Songping Wang, and Huanqian Yan. Efficient robustness assessment via adversarial spatial-temporal focus on videos. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2023.

Zhipeng Wei, Jingjing Chen, Xingxing Wei, Linxi Jiang, Tat-Seng Chua, Fengfeng Zhou, and Yu-Gang Jiang. Heuristic black-box adversarial attacks on video recognition models. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pp. 12338–12345, 2020.

Daan Wierstra, Tom Schaul, Tobias Glasmachers, Yi Sun, Jan Peters, and Jürgen Schmidhuber. Natural evolution strategies. *The Journal of Machine Learning Research*, 15(1):949–980, 2014.

Chao-Yuan Wu, Manzil Zaheer, Hexiang Hu, R Manmatha, Alexander J Smola, and Philipp Krähenbühl. Compressed video action recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 6026–6035, 2018.

Shangyu Xie, Han Wang, Yu Kong, and Yuan Hong. Universal 3-dimensional perturbations for black-box attacks on video recognition systems. In *2022 IEEE Symposium on Security and Privacy (SP)*, pp. 1390–1407. IEEE, 2022.

Huanqian Yan and Xingxing Wei. Efficient sparse attacks on videos using reinforcement learning. In *Proceedings of the 29th ACM International Conference on Multimedia*, pp. 2326–2334, 2021.

Haishan Ye, Zhichao Huang, Cong Fang, Chris Junchi Li, and Tong Zhang. Hessian-aware zeroth-order optimization for black-box adversarial attack. *arXiv preprint arXiv:1812.11377*, 2018.

Hu Zhang, Linchao Zhu, Yi Zhu, and Yi Yang. Motion-excited sampler: Video adversarial attack with sparked prior. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XX 16*, pp. 240–256. Springer, 2020.

Richard Zhang, Phillip Isola, Alexei A Efros, Eli Shechtman, and Oliver Wang. The unreasonable effectiveness of deep features as a perceptual metric. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 586–595, 2018.

A Method Details

Roadmap. Appendix A details the method: how the codec's motion vectors are read, how CMF is built from them, and the optimization procedure. Appendix B gives the experimental setup, including the definitions of every control arm. Appendix C contains the allocation analysis behind Sections 4.3.1, 4.3.2 and 5. Appendix D holds the remaining evaluations (a transformer victim, defenses, subsampling, user study and qualitative examples) and Appendix E the extended related work.

A.1 Codec Motion Vectors and Encoding

In standard compressed video, frames are grouped into Groups of Pictures (GOPs). A GOP begins with an I-frame (intra; a self-contained full image) and is followed by P-frames (predictive) that are encoded relative to a reference frame (Le Gall, 1991). Concretely, each P-frame P_t is reconstructed from a reference I/P frame R (typically the immediately preceding frame in our setting) using blockwise motion compensation plus a transform-coded residual. Motion compensation uses variable-size, sub-pixel motion vectors to indicate where a block in R should be sampled to predict

its location in P_t; any remaining mismatch is captured by the residual. Some blocks may still be intra-coded (i.e., predicted without motion), depending on encoder decisions.

IPPP chain. We adopt a simple IPPP configuration in which each P-frame references the immediately previous I/P frame, forming a linear reference chain (Liou, 1991):

$$\text{GOP} = \{I, P_1, P_2, \ldots, P_L\}.$$

This avoids bidirectional dependencies (no B-frames) and yields deterministic forward/backward warps, which is convenient for our CMF construction.

Motion-vector field. Let $\mathbf{V}_t \in \mathbb{R}^{H\times W\times 2}$ denote the dense motion-vector (MV) field, indexed in P_t's coordinates and giving for each pixel of P_t the displacement back to the reference frame it was predicted from, which is the backward convention the reconstruction below uses. We obtain $\mathbf{V}_t$ by upsampling the encoder's blockwise vectors (variable block sizes, sub-pixel precision) to per-pixel resolution (bilinear interpolation); blocks that are intra-coded are treated as zero-motion. We write the standard warping model as

$$P_t(\mathbf{x}) \;\approx\; R\big(\mathbf{x} - \mathbf{V}_t(\mathbf{x})\big) \;+\; \text{residual}_t(\mathbf{x}),$$

where R is the reference frame, $\mathbf{x}$ is a pixel location, and the residual accounts for appearance not explained by motion. The per-pixel MV magnitude,

$$\|\mathbf{V}_t(\mathbf{x})\|_2 \;\; \text{(pixels/frame)},$$

serves as a proxy for the local amount of inter-frame change; aggregations of this quantity are the basis of our video CMF.

GOP length and our choice. Every clip is re-encoded before its vectors are read: `ffmpeg` with the `mpeg4` encoder at a 1000 kb/s target, and the vectors are then read with the CoViAR loader (Wu et al., 2018). That encoder's default group of pictures is 12 frames with no B-frames, which we verified by probing the produced streams: the picture types run $\texttt{I}\,\texttt{P}^{11}\,\texttt{I}\,\texttt{P}^{11}\cdots$. In this work we therefore use a GOP size of 12 (1 I + 11 P), following CoViAR's default and a common MPEG-style configuration (Le Gall, 1991). This choice balances temporal coverage and compute: for our 24-frame clips, two GOPs cover the full sequence; the length is sufficient to summarize motion over a meaningful window while keeping the number of warps and the memory footprint modest. Practically, the IPPP chain also simplifies composing long-range flows (for back-/forward-warping) when we align and aggregate motion into the anchor map. A clip longer than one GOP is processed one GOP at a time: each GOP's anchor map is built, normalized and propagated from its own I-frame, and the per-frame maps are concatenated in time.

A.2 CMF Construction

Figure 2 shows the pipeline end to end, Figure 3 the per-GOP construction on one clip, and Algorithm 1 states the procedure.

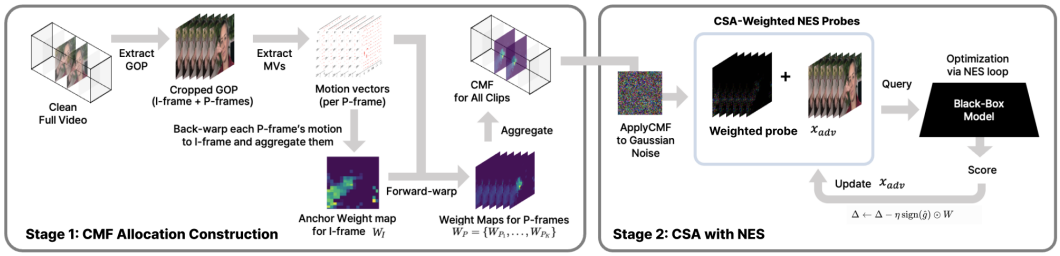


Figure 2: Codec-motion instantiation of CSA. A CMF allocation $W(X)$ is computed from codec motion before the first victim query and shared by the zeroth-order probe and the update.

A.3 Optimization

Algorithm 2 gives the search loop with the bookkeeping that Section 3.4 omits; the probe and update equations are stated there.

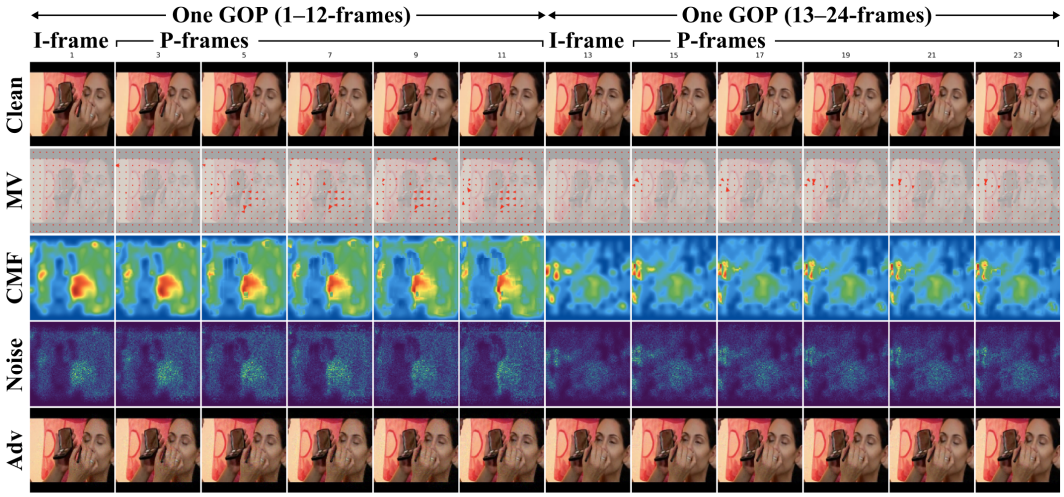


Figure 3: Per-GOP construction on one clip (two GOPs, 24 frames). Rows: clean frames (*Clean*), codec motion vectors (*MV*), CMF maps (*CMF*), the CMF-weighted random pattern (*Noise*) and adversarial frames (*Adv*). Maps are max-normalized per GOP, so intensities are comparable within a GOP but not across the two.

Algorithm 1: CMF Generation for a Single GOP

Input: GOP $\mathcal{G} = \{I, P_1, \ldots, P_{11}\}$, per-P-frame MV fields $\{\mathbf{V}_t\}_{t=1}^{11}$ at pixel resolution, warping operator $\mathrm{Warp}(A, \Phi)$
Output: Per-frame CMF maps $W = \{W_I, W_{P_1}, \ldots, W_{P_{11}}\}$

for $t = 1, \ldots, 11$ **do**
  $A_t \leftarrow \|\mathbf{V}_t\|_2$ // MV magnitude per P-frame
for $t = 1, \ldots, 11$ **do**
  $\tilde{A}_t \leftarrow \mathrm{Warp}(A_t, \Phi_{t \to I})$ // back-warp to I-frame
$W_I \leftarrow \sum_{t=1}^{11} \tilde{A}_t$ // aggregate motion at I-frame
$W_I \leftarrow G_{\sigma_s} * (W_I / \max W_I)$ // normalize and smooth (App. B.2)
for $t = 1, \ldots, 11$ **do**
  $W_{P_t} \leftarrow \mathrm{Warp}(W_I, \Phi_{I \to t})$ // forward-warp to each P-frame
return W

A.4 First-Order Analysis at $K=1$

This appendix gives the closed form and the reduction quoted in Section 3.4, and checks both numerically on the CMF maps of the ablation clips.

Proposition 1 (One-step descent at $K=1$)**.** *Let $u = D_a z$ with $z \sim \mathcal{N}(0, I)$ and $a, b \geq 0$, and let the update be $-\eta\, b \odot \mathrm{sign}(\hat{g})$ with $\hat{g}$ formed from one antithetic pair. Under a first-order expansion of the loss at an iterate with fixed gradient g, the expected decrease $\mathcal{D}(a, b) = -g^\top \delta X$ is*

$$\mathbb{E}_z\big[\mathcal{D}(a,b) \,\big|\, g\big] = \frac{2\eta}{\pi} \sum_{j\,:\,a_j > 0} g_j b_j \arcsin\left(\frac{g_j a_j}{\|g \odot a\|_2}\right).$$

Proof. To first order in σ the antithetic difference is $\mathcal{L}(X+\sigma u) - \mathcal{L}(X-\sigma u) = 2\sigma\, g^\top u + O(\sigma^3)$, so $\hat{g} \propto (g^\top u)\, u$ with a positive factor. Hence for $a_j > 0$, $\mathrm{sign}(\hat{g})_j = \mathrm{sign}(g^\top u)\,\mathrm{sign}(z_j)$, while $a_j = 0$ gives $\hat{g}_j = 0$ and no step. Writing $Y = g^\top u = \sum_k g_k a_k z_k$, the pair (Y, z_j) is zero-mean jointly Gaussian with correlation $r_j = g_j a_j / \|g \odot a\|_2$, and the arcsine law gives $\mathbb{E}[\mathrm{sign}(Y)\mathrm{sign}(z_j)] = \frac{2}{\pi} \arcsin r_j$. Summing $\eta\, g_j b_j\, \mathbb{E}[\mathrm{sign}(Y)\mathrm{sign}(z_j)]$ over j gives the statement. □

Corollary 1 (High-dimensional reduction)**.** *Suppose no coordinate dominates $\|g \odot a\|_2$, so $\arcsin r_j \simeq r_j$, and that the weighted gradient energies concentrate around second moments s_j^2.*

Algorithm 2: CSA with NES and a CMF Allocation

Input: $X = \{x_t\}_{t=1}^{T}$ (clean clip), y (ground-truth) or y_{tar} (target), F (black-box model), $\{W_t\}$ (CMF maps), $\{\mathbf{V}_t\}$ (codec motion vectors), σ (NES probe), K (pairs per iter; default 1), η (step size) ε_∞ (projection bound), $Q_{\max}$

Output: $X_{\text{adv}} = X + \{\Delta_t\},\ \text{success},\ Q$

initialize: $\Delta_t \leftarrow 0\ (\forall t)$, $X_{\text{adv}} \leftarrow X$, $Q \leftarrow 0$, success $\leftarrow$ False
while $Q < Q_{\max}\ \wedge\ \neg$*success* **do**
  sample noise frames $z^{(i)} \sim \mathcal{N}(0, I)$ of size $3\times H\times W$ for $i = 1, \ldots, K$
  $Z^{(i)} \leftarrow T_m(z^{(i)})$: place $z^{(i)}$ at every GOP anchor and warp it along $\{\mathbf{V}_t\}$
  $\widetilde{U}_t^{(i)} \leftarrow W_t \odot Z_t^{(i)} \quad (\forall i, t)$
  for $i = 1, \ldots, K$ **do**
    $X^{(i)\pm} \leftarrow X_{\text{adv}} \pm \sigma\, \widetilde{U}^{(i)}$
    $\mathcal{L}^{(i)\pm} \leftarrow \mathcal{L}\big(f(X^{(i)\pm})\big)$
  $Q \leftarrow Q + 2K$
  $\hat{g} \leftarrow \dfrac{1}{2K\sigma} \sum_{i=1}^{K} \big(\mathcal{L}^{(i)+} - \mathcal{L}^{(i)-}\big)\, \widetilde{U}^{(i)}$
  $\Delta_t \leftarrow \Delta_t - \eta \cdot \text{sign}(\hat{g}_t) \odot W_t \quad (\forall t)$
  $\Delta_t \leftarrow \text{PROJECT}(\Delta_t, \varepsilon_\infty) \quad (\forall t)$
  $X_{\text{adv}} \leftarrow X + \{\Delta_t\}$
  $Q \leftarrow Q + 1$
  success $\leftarrow (F(X_{\text{adv}}) = y_{\text{tar}})$ if targeted, else $(F(X_{\text{adv}}) \neq y)$
return $X_{\text{adv}},\ \textit{success},\ Q$

With $\mu_s(f) = \sum_j s_j^2 f_j / \sum_j s_j^2$, the four arms of Eq. 4 satisfy

$$\frac{\mathcal{D}_{\text{CSA}}}{\mathcal{D}_0} \simeq \frac{\sqrt{\mu_s(W^2)}}{\mathbb{E}[W]}, \qquad \frac{\mathcal{D}_{\text{update}}}{\mathcal{D}_0} \simeq \frac{\mu_s(W)}{\mathbb{E}[W]}, \qquad \frac{\mathcal{D}_{\text{probe}}}{\mathcal{D}_0} \simeq \frac{\mu_s(W)}{\sqrt{\mu_s(W^2)}},$$

where $\mathcal{D}_0$ is the matched control ($a = \sqrt{\mathbb{E}[W^2]}\,\mathbf{1}$, $b = \mathbb{E}[W]\,\mathbf{1}$). The constant probe amplitude cancels. For isotropic gradient energy the three ratios become ρ, 1 and ρ^{-1} with $\rho = \sqrt{\mathbb{E}[W^2]}/\mathbb{E}[W]$.

Proof. Under the stated approximation $\mathbb{E}_z[\mathcal{D}(a, b) \mid g] \simeq \frac{2\eta}{\pi} \sum_j g_j^2 a_j b_j / \|g \odot a\|_2$, and replacing $\sum_j g_j^2 f_j$ by $\sum_j s_j^2 f_j$ gives $\mathcal{D}(a, b) \propto \sum_j s_j^2 a_j b_j / (\sum_j s_j^2 a_j^2)^{1/2}$. Substituting the four (a, b) pairs and dividing by the control gives the ratios; the constant a of the control and of the probe-only arm appears in numerator and denominator alike. □

Two consequences follow under the approximation of Corollary 1, both by Cauchy–Schwarz. Probe weighting alone cannot exceed the matched control, since $\mu_s(W) \leq \sqrt{\mu_s(W^2)}$. And the two sides are super-additive on the multiplicative scale, $(\mathcal{D}_{\text{CSA}}/\mathcal{D}_0) \,/\, [(\mathcal{D}_{\text{update}}/\mathcal{D}_0)(\mathcal{D}_{\text{probe}}/\mathcal{D}_0)] = \mu_s(W^2)/\mu_s(W)^2 \geq 1$, which is a positive coupling between the probe and update sides rather than a prediction of the query-cost interaction the factorial measures.

Numerical check. Table 4 evaluates the three expressions on CMF maps drawn from the ablation clips, over 200,000 coordinates each and 3,000 probe draws per arm. The exact expression of Proposition 1 and the reduction of Corollary 1 agree throughout. The Monte Carlo estimate tracks them to within its sampling noise except for the update-only arm on the most concentrated allocation, which is the one cell where the estimator itself is inefficient: the probe is uniform there, so every coordinate contributes a correlation of order $d^{-1/2}$ to the mean, while the update weights are concentrated, so a few coordinates carry most of the variance. Its relative standard error is $(\pi/2)\rho/\sqrt{N}$, or 29% at $\rho{\approx}10$ and $N{=}3{,}000$, and the entry lies about 1.7 standard errors from the exact value; a longer run at $N{=}120{,}000$ agrees with the exact value to within 0.1 standard errors.

When the gradient energy is instead correlated with W, the general form of Corollary 1 still matches the simulation while the isotropic reduction does not: on the same clips the update-only ratio rises above 1 and the CSA ratio above ρ, since μ_s then weights the high-W coordinates more heavily than $\mathbb{E}$ does. The isotropic case is therefore the conservative one for the ordering we report.

Table 4: Relative one-step decrease against the matched control, on CMF maps from four ablation clips spanning the range of ρ, with isotropic gradient energy. *MC*: Monte Carlo over probe draws; *exact*: Proposition 1; *approx*: Corollary 1; *isotropic*: ρ, 1, ρ^{-1}.

ρ	Arm	MC	exact	approx	isotropic
1.13	CSA	1.138	1.133	1.133	1.130
	update-only	0.960	1.003	1.003	1.000
	probe-only	0.878	0.885	0.885	0.885
1.59	CSA	1.649	1.587	1.587	1.586
	update-only	0.969	1.001	1.001	1.000
	probe-only	0.638	0.631	0.631	0.630
2.72	CSA	2.696	2.739	2.739	2.724
	update-only	1.082	1.004	1.004	1.000
	probe-only	0.375	0.367	0.367	0.367
10.01	CSA	9.560	9.780	9.777	10.005
	update-only	0.510	0.974	0.974	1.000
	probe-only	0.105	0.100	0.100	0.100

A.5 Preprocessing Cost

CMF is built once per clip, before the first victim query, and then held fixed: during optimization W scales the probe $T_m z$ and the sign step, so the search loop costs no more per iteration than the uniform controls it is compared against.

Measured cost. We time the two preprocessing calls that run once per clip before the search loop, on a 128×128 clip, five repeats per length (Table 5).

Table 5: CMF preprocessing time per clip (128×128, five repeats): motion-vector parsing and map construction only, re-encoding excluded.

T (frames)	8	16	24	32	48	64	96	128
mean (ms)	49.3	96.4	164.1	211.9	364.9	440.5	670.5	963.0
s.d. (ms)	6.3	9.1	12.9	16.3	54.1	28.6	98.4	149.7

The cost is linear in clip length (7.5 ms per frame, R^2=0.996). At our setting (T=24) CMF construction takes 164 ms once the clip is encoded, under 0.2% of the wall-clock of a 5,000-query attack; the re-encoding pass itself is not included in that figure.

This timing covers parsing the motion vectors and building W, not encoding: video that already arrives in the 12-frame IPPP form carries its vectors, while raw frame sequences must be encoded first. All experiments use the fixed encoder configuration of Appendix A.1.

B Experimental Setup

B.1 Victim Models and Environment

Target models. Kinetics-400 checkpoints are taken from the PySlowFast model zoo (Facebook AI Research, 2020): I3D (I3D_8x8_R50; 73.3% top-1) (Carreira & Zisserman, 2017), SlowFast (SLOWFAST_8x8_R50; 74.6%) (Feichtenhofer et al., 2019), SlowOnly (SLOW_8x8_R50; 76.8%) (Feichtenhofer et al., 2019), and X3D (X3D_S; 71.3%) (Feichtenhofer, 2020). These top-1 figures are the model zoo's, at each checkpoint's native input; the clean accuracy of these victims at the $24\times128\times128$ input we attack is reported in Section 4. For UCF101 we train a TensorFlow EfficientNet-B0 (Tan & Le, 2019) applied per frame (a `TimeDistributed` backbone) whose per-frame logits are averaged over the clip to give one clip-level prediction; the attack queries that clip-level output. The adversarially trained variant of this victim is used only for the adversarial-

training result reported in the note to Table 22, and is a separately trained model at a different input resolution, so its numbers are not directly comparable with the rest of the paper.

Environment. All experiments were carried out on an AMD Ryzen Threadripper 3990X with 128 GB RAM and a single NVIDIA RTX 4090 on Ubuntu 20.04. The attack pipeline and the Kinetics-400 victims run in PyTorch 2.3.0+cu118; the UCF101 EfficientNet-B0 victim is served through ONNX Runtime from its TensorFlow training graph.

B.2 Attack Configuration

Attack hyperparameters. Probe scale σ=0.05, step size η=0.005, L_∞ bound ε_∞=0.1, K=1 antithetic pair per iteration, and $Q_{\max}$=5,000 unless stated otherwise. The matched-uniform and dense-uniform baselines use $\eta\,\mathbb{E}[W]$ and the sparse-uniform baseline η, so that all four arms share the per-iteration L_1 budget; the matched-uniform arm additionally sets its constant probe weight to $\sqrt{\mathbb{E}[W^2]}$, computed per clip, so that its probe RMS equals CMF's (Table 6). CMF anchor maps are max-normalized per GOP and smoothed with a 15×15 Gaussian kernel (σ_s=3); the degeneracy blending and weight floor available in the implementation are disabled throughout (τ=0, $w_{\min}$=0) so that what is evaluated is the input-derived allocation itself and not a hand-designed fallback; clips with $W \equiv 0$ are retained and counted as failures (Appendix B.4). γ=1 except in the scale sweep. CMF maps are computed once per GOP and reused across iterations; all reported query counts include both probe queries and success-check queries.

B.3 Control and Ablation Definitions

Table 6 states how each arm in Table 1 is constructed and what it holds fixed. Two quantities must be matched, and they are not the same: the per-iteration update L_1, which scales with $\mathbb{E}[W]$, and the probe amplitude, whose scale-free measure is the RMS $\sqrt{\mathbb{E}[W^2]}$. A binary mask cannot match both: fixing its density to $\mathbb{E}[W]$ fixes the budget but forces its nonzero entries to 1, while the dense control matches the budget through the step size and probes at full amplitude everywhere. Only *Uniform (L_1-matched)* equalizes both. The other two arms are reported here rather than in the main text, *Sparse Uniform* in Table 7 and *Dense Uniform* in Table 8; the dense arm is retained because it is the unmatched uniform control closest to plain NES and because it shows the direction of the confound. Figure 4 shows the three allocations on one clip.

Table 6: Control arms. All arms weight the same motion-warped base probe $T_m z$; $\mathbb{E}[W]$ is the mean CMF weight. Probe RMS is reported relative to the CMF-weighted probe, pooled over the 1,800 Kinetics-400 attack instances. Only *Uniform (L_1-matched)* matches both the per-iteration update L_1 and the probe RMS; it is the control of Table 1.

Setting	Probe weight	Update step	Nonzero support	Description	Probe RMS
CMF	$W \in [0,1]$	η	73% mean	Graded by motion magnitude	1.00×
Uniform (L_1-matched)	$\sqrt{\mathbb{E}[W^2]}\cdot\mathbf{1}$	$\eta\cdot\mathbb{E}[W]$	100%	Constant, L_1-matched	1.00×
Sparse Uniform	$\{0,1\}$, density $\mathbb{E}[W]$	η	$\mathbb{E}[W]$	Sparse, random locations	1.52×
Dense Uniform	$\mathbf{1}$	$\eta\cdot\mathbb{E}[W]$	100%	Uniform, budget-equalized only	3.83×

Table 7: *Sparse Uniform* control (Table 6), on the clips, seeds and protocol of Table 1. It matches only the update L_1 and probes at 1.52× CMF's RMS on a random support. Paired differences against it are in Table 9.

		NES						SPSA					
Model	Setting	SR(%) ↑	ANQ ↓	CANQ ↓	MAP ↓	SSIM ↑	LPIPS ↓	SR(%) ↑	ANQ ↓	CANQ ↓	MAP ↓	SSIM ↑	LPIPS ↓
Kinetics-400 (*n*=75, 3 seeds)													
I3D	Sparse Uniform	75.11±1.54	525	1639±38	1.72	0.9089	0.0318	63.11±2.04	513	2170±42	1.66	0.9080	0.0336
SF	Sparse Uniform	72.00±1.33	481	1747±17	1.77	0.9124	0.0339	67.11±2.04	660	2090±51	1.93	0.8987	0.0402
SO	Sparse Uniform	82.67±1.33	598	1361±38	1.59	0.9083	0.0371	76.44±2.78	642	1669±122	1.61	0.9017	0.0378
X3D	Sparse Uniform	92.00±1.33	460	823±98	1.79	0.9103	0.0323	84.44±0.77	593	1278±87	1.93	0.8993	0.0393
UCF101 (*n*=75, 5 seeds)													
EffNet-B0	Sparse Uniform	25.07±1.12	393	3845±66	1.72	0.8896	0.0295	25.60±1.12	639	3884±60	1.95	0.8687	0.0387

Table 8: *Dense Uniform* control (Table 6), on the clips, seeds and protocol of Table 1. It matches only the update L_1 and probes at $3.83\times$ CMF's RMS, so its deficit mixes allocation shape with probe scale.

		NES						SPSA					
Model	**Setting**	**SR(%)↑**	**ANQ↓**	**CANQ↓**	**MAP↓**	**SSIM↑**	**LPIPS↓**	**SR(%)↑**	**ANQ↓**	**CANQ↓**	**MAP↓**	**SSIM↑**	**LPIPS↓**
Kinetics-400 (*n*=75, 3 seeds)													
I3D	Dense Uniform	73.78±1.54	1344	2303±43	4.09	0.8852	0.0435	60.89±2.04	1835	3074±31	4.85	0.8475	0.0609
SF	Dense Uniform	69.78±0.77	1620	2641±48	4.11	0.8831	0.0492	45.33±1.33	1635	3475±30	5.00	0.8480	0.0676
SO	Dense Uniform	68.44±0.77	1512	2613±47	3.89	0.8758	0.0513	56.89±1.54	2006	3298±46	4.88	0.8364	0.0765
X3D	Dense Uniform	76.44±2.04	1424	2267±52	3.98	0.8869	0.0446	47.56±1.54	1602	3385±36	5.63	0.8261	0.0803
UCF101 (*n*=75, 5 seeds)													
EffNet-B0	Dense Uniform	58.40±1.74	2225	3379±41	4.34	0.8463	0.0484	46.93±4.36	2141	3663±64	4.59	0.8249	0.0580

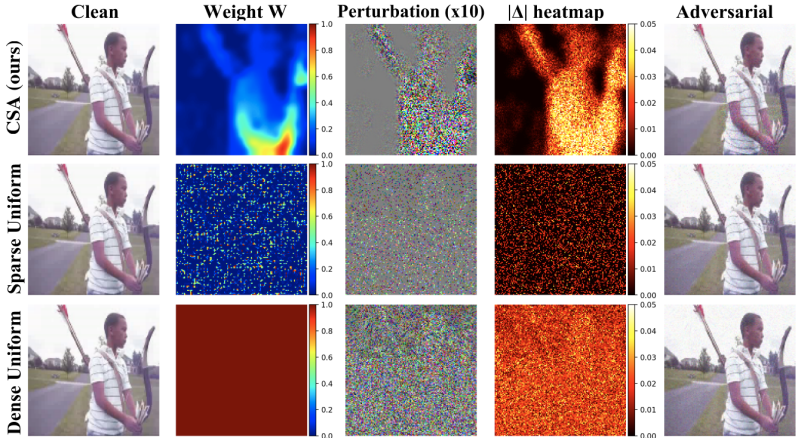


Figure 4: Weight map, perturbation ($\times 10$) and $|\Delta|$ for CSA and the two unmatched controls (UCF101, *Archery*).

B.4 Degenerate Allocations ($W \equiv 0$)

Clips with no motion vectors. Some ablation clips carry no motion vectors in any group of pictures, so $W \equiv 0$: two of the 75 clips for I3D, SF and SO and one for X3D, 42 of the 1,800 Kinetics-400 attack instances. CMF cannot perturb them and they enter every CANQ as failures at $Q_{\max}$; they are not dropped. The L_1-matched control fails on them identically, since its constant step is scaled by $\mathbb{E}[W]$, so they contribute nothing to the paired differences and cost both arms the same 2.67 percentage points of success rate (1.33 on X3D). Only the ρ and d_{eff} statistics exclude them, both being undefined at $W \equiv 0$.

C Allocation Analysis

This appendix expands Sections 4.3.1, 4.3.2 and 5, in four parts. Appendix C.1 examines the matched control and its paired statistics (C.1.1) and the query–distortion geometry (C.1.2). Appendix C.2 analyzes probe–update coupling: estimator strength (C.2.1), the factorial on a second victim (C.2.2) and the motion-independent probe (C.2.3). Appendix C.3 studies the allocation's structure: placement and provenance (C.3.1), temporal organization (C.3.2), the concentration stratification of Section 5 (C.3.3) and a qualitative example (C.3.4). Appendix C.4 characterizes the reversal under tight bounds: the sensitivity to the perturbation budget (C.4.1) and the mechanism at a tight bound (C.4.2).

Conventions. Unless stated otherwise, every comparison in this appendix is paired by clip, with seeds averaged within a clip; differences are CSA minus the comparison arm, so negative ΔCANQ and positive ΔSR favor CSA; 95% intervals come from 10,000 clip-level bootstrap resamples; and p is a paired sign-flip permutation test. Comparisons read off an interpolated curve instead use the two-sided bootstrap p stated in their captions. A failure enters CANQ at $Q_{\max}$.

C.1 Matched Controls and Distortion Geometry

C.1.1 Controls and Paired Statistics

The allocation ablation against the matched control and *Sparse Uniform* is reported in Section 4.3.1 (Table 1), covering both Kinetics-400 and UCF101; the arms are defined in Appendix B.3. Table 9 gives the paired clip-level differences behind that table, with bootstrap intervals and sign-flip permutation tests.

Table 9: Paired clip-level differences behind Table 1: CSA against *Sparse Uniform* (*vs. Sparse*) and against the matched control (*vs. Matched*). †: $p \geq 0.05$, the criterion we report significance by, so an interval that just excludes zero can still carry † (SO/SPSA ΔSR, p=0.061). SR in percentage points, CANQ in queries. Kinetics-400: 75 clips $\times$ 3 seeds; UCF101: 75 clips $\times$ 5 seeds.

		ΔSR, CSA $-$ baseline (pp)		ΔCANQ, CSA $-$ baseline	
Model	Opt.	vs. Sparse	vs. Matched	vs. Sparse	vs. Matched
I3D	NES	$+16.9$ [$+8.4$, $+25.8$]	$+0.9$ [-4.0, $+5.8$]†	-525 [-843, -208]	-320 [-546, -101]
SF	NES	$+10.2$ [$+2.7$, $+18.2$]	-3.1 [-8.9, $+1.8$]†	-107 [-455, $+234$]†	-83 [-285, $+130$]†
SO	NES	$+9.8$ [$+3.6$, $+16.4$]	$+0.9$ [-4.4, $+6.2$]†	-311 [-578, -48]	-347 [-607, -83]
X3D	NES	$+3.6$ [$+0.0$, $+7.6$]†	$+2.2$ [-0.4, $+5.8$]†	-68 [-305, $+146$]†	-433 [-672, -220]
I3D	SPSA	$+28.9$ [$+19.6$, $+38.7$]	$+2.7$ [-3.1, $+8.9$]†	-1015 [-1446, -603]	-371 [-698, -56]
SF	SPSA	$+18.7$ [$+10.2$, $+27.6$]	-0.9 [-4.9, $+3.1$]†	-598 [-991, -230]	-190 [-375, -0.1]
SO	SPSA	$+16.9$ [$+10.2$, $+24.0$]	$+5.8$ [$+0.9$, $+11.6$]†	-534 [-818, -268]	-587 [-925, -272]
X3D	SPSA	$+10.2$ [$+4.4$, $+17.3$]	$+6.7$ [$+1.8$, $+12.4$]	-502 [-830, -206]	-676 [-994, -393]
EffNet-B0	NES	$+48.3$ [$+36.5$, $+60.0$]	$+2.7$ [-6.4, $+11.7$]†	-1130 [-1567, -683]	-266 [-601, $+70$]†
EffNet-B0	SPSA	$+41.9$ [$+30.1$, $+53.1$]	$+4.3$ [-4.3, $+13.1$]†	-962 [-1394, -525]	-350 [-685, -8]

Per-cell significance. All ten ΔCANQ point estimates against the matched control favor CSA; eight of the ten sign-flip tests are nominally significant at $p < 0.05$, and four survive Holm step-down correction over the ten CANQ comparisons (X3D/SPSA $p<10^{-4}$, X3D/NES 0.0001, SO/SPSA 0.0005, I3D/NES 0.0046; the remaining raw values are SO/NES 0.0098, I3D/SPSA 0.0277, UCF101/SPSA 0.0430, SF/SPSA 0.0479, UCF101/NES 0.1197, SF/NES 0.4464).

A clip-clustered test of the overall effect. The Holm correction in Table 9 answers whether each cell is individually significant. The claim the paper makes is the weaker one that the allocation lowers query cost across cells, so we also test it directly. The unit of resampling is a clip, not a clip–cell row: within a victim, NES and SPSA attack the same clips, so a clip's two optimizer differences are resampled and sign-flipped together, while clip sets are disjoint across victims because each victim keeps its own correctly classified classes. Every cell enters with equal weight, so no cell dominates through its clip count. Since a clip's cost is its query count on success and $Q_{\max}$ on failure, the clip-level mean of that cost is exactly CANQ, and the clip-level differences decompose ΔCANQ without further assumptions. The cell-averaged reduction is 362 queries, with a 95% clustered bootstrap interval of $[-476, -249]$ over 10,000 resamples, and a clustered sign-flip permutation test places the observed value outside all 10,000 permutations ($p < 10^{-4}$). We report both: the clustered test speaks to the overall effect, the Holm-corrected column to per-cell claims.

C.1.2 Query–Distortion Geometry

Why a single matched point is not enough. *Uniform (L_1-matched)* in Table 1 equalizes CMF's probe RMS and its per-iteration update L_1. It does not equalize the update L_2, and it cannot: for a constant vector $c\mathbf{1}$ the ratio $\|\cdot\|_2/\|\cdot\|_1$ equals $1/\sqrt{d}$, which is the minimum over all vectors, while any W with $\mathbb{E}[W^2] > \mathbb{E}[W]^2$ lies strictly above it. At equal L_1, CMF therefore carries

$$\rho \;=\; \sqrt{\mathbb{E}[W^2]}\,/\,\mathbb{E}[W]$$

times the constant arm's update L_2; pooled over the 1,758 Kinetics-400 attack instances where it is defined (42 of 1,800 have $W \equiv 0$) $\rho = 1.55$, with a per-clip range of 1.04 to 10.07. No constant arm can match both norms, so two constants bracket CMF rather than either one being *the* control: matching L_1 leaves CMF more L_2, and matching L_2 (setting $c = \sqrt{\mathbb{E}[W^2]}$) hands the constant arm

ρ times more L_1. Measured on I3D/NES, the L_2-matched arm reaches CANQ 828 at MAP 3.36 against CMF's 1,114 at MAP 2.53: cheaper, but at more perturbation.

Comparing at matched distortion instead. We therefore stop matching norms and sweep the step of both allocations, holding the probe fixed within each family (CMF probes with W; every constant arm probes with $\sqrt{\mathbb{E}[W^2]}$), so each family is a one-parameter curve in its step norm. CMF is swept over $\eta \in \{0.00125, 0.005, 0.01, 0.025\}$ and the constant arm over $c/\sqrt{\mathbb{E}[W^2]} \in \{0.161, 0.322, 0.50, 0.644, 1, 2, 4\}$, where 0.644 is the pooled $\mathbb{E}[W]/\sqrt{\mathbb{E}[W^2]} = 1/\rho$ and so a fixed ratio across clips, unlike the per-clip L_1-matched control of Table 1, which sets $c{=}\mathbb{E}[W]$ for each clip separately, all on the same 75 clips and 3 seeds. The value 0.50 was chosen, before running it, as the one the rest of the sweep predicts would land nearest CMF's own MAP; it realizes MAP 2.42 against CMF's 2.53, so the region where the two families are compared is measured rather than interpolated across a gap. Figure 5 plots the two curves against each of the three distortion measures. Reading a curve at the other family's distortion gives the query cost each allocation needs to reach the same perturbation.

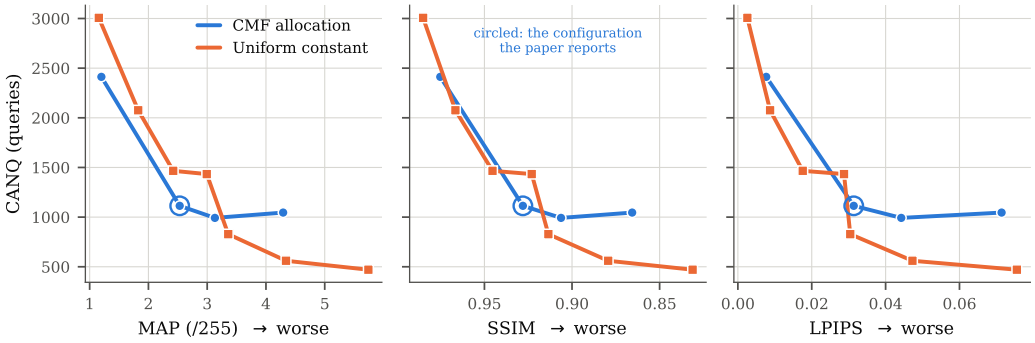


Figure 5: Query–distortion curves on I3D/NES (75 clips × 3 seeds). The probe is fixed within each family, so only the step norm moves along a curve; rightward is worse on every panel. Circled: the operating point used elsewhere (η=0.005). Readings at matched distortion are in Table 10.

Table 10: Query cost at matched distortion. The constant arm's curve is read at CMF's own distortion by linear interpolation; intervals resample curve and target together, so they include the interpolation uncertainty, and p is the two-sided bootstrap value. "–": CMF lies outside the constant curve's span and is not extrapolated. I3D/NES uses the seven-point sweep, the other cells three constant points around the operating region.

Victim / optimizer	Matched on	ΔCANQ	95% CI	p
I3D / NES	MAP	−346.1	[−612.2, −147.3]	0.0012
	SSIM	−327.0	[−524.8, +97.8]	0.10
	LPIPS	+300.3	[−447.5, +522.6]	0.55
I3D / SPSA	MAP	−787.1	[−1365.0, −234.9]	0.0044
	SSIM	−331.1	[−875.1, +235.8]	0.28
	LPIPS	–		
X3D / NES	MAP	−962.6	[−1457.6, −534.5]	$< 10^{-4}$
	SSIM	−488.7	[−938.3, −95.9]	0.0120
	LPIPS	–		

Conditioning on success. MAP, SSIM and LPIPS are only defined for a successful attack, so every distortion number in this paper averages over that arm's *own* successes. When two arms solve different clips, those averages describe different populations. CANQ is not exposed to this: a failure enters it at $Q_{\max}$ rather than being dropped. To bound the effect on the distortion axis we recomputed every cell of Table 1 on the intersection, the clips both arms solved, with no new attack runs. The MAP advantage keeps its sign and its significance in all ten cells (Table 11). Re-conditioning never moves the estimate away from zero, and moves it by 0.121 MAP on average; the

move is small on Kinetics-400, where the two arms' success sets differ by a handful of clips, and largest on UCF101/SPSA, the cell with the most asymmetric success sets (65 clips solved only by CMF against 49 only by the control), where it takes -0.913 to -0.480. We report the L_1-criterion advantage as robust to this conditioning.

Table 11: ΔMAP against the matched control, over each arm's own successes ("own", as elsewhere) and over clips both arms solved ("both"; interval and p on the intersection).

	clips			ΔMAP		
cell	both	CSA only	ctrl only	own	both	p
I3D/NES	200	7	5	-0.461	-0.358 $[-0.501, -0.228]$	$< 10^{-4}$
SF/NES	182	3	10	-0.236	-0.236 $[-0.345, -0.124]$	$< 10^{-4}$
SO/NES	199	9	7	-0.521	-0.399 $[-0.543, -0.250]$	$< 10^{-4}$
X3D/NES	209	6	1	-0.453	-0.410 $[-0.550, -0.280]$	$< 10^{-4}$
I3D/SPSA	195	12	6	-0.370	-0.319 $[-0.463, -0.173]$	$< 10^{-4}$
SF/SPSA	190	3	5	-0.457	-0.447 $[-0.590, -0.313]$	$< 10^{-4}$
SO/SPSA	195	15	2	-0.695	-0.536 $[-0.714, -0.363]$	$< 10^{-4}$
X3D/SPSA	198	15	0	-0.717	-0.573 $[-0.777, -0.393]$	$< 10^{-4}$
UCF101/NES	236	39	29	-0.420	-0.274 $[-0.434, -0.109]$	0.0007
UCF101/SPSA	188	65	49	-0.913	-0.480 $[-0.736, -0.219]$	0.0003

Reading the curves. At matched mean absolute perturbation the CMF allocation is significantly cheaper in all three cells, across two architectures and two zeroth-order optimizers, and not specific to one victim or one probe family. **MAP is itself an L_1-family measure**, $\|\Delta\|_1/d$, so this moves the comparison from a per-iteration L_1 convention to a final-perturbation one rather than escaping the L_1 family: at equal MAP the two arms need not carry the same $\|\Delta\|_2$, and a graded perturbation is expected to carry more. The result therefore does not establish the ordering under an L_2-type criterion, and the SSIM and LPIPS rows are consistent with the advantage narrowing as the criterion moves in that direction. Note also that only I3D/NES carries the dense seven-point uniform sweep; the other two cells use three constant points bracketing the operating region, so their interpolated magnitudes rest on wider gaps.

Testing the L_2 criterion directly. We therefore repeat the comparison under two L_2-family measures, RMS and PSNR. We re-ran CMF at its operating point and four constant arms on I3D/NES under one protocol, logging two L_2-family companions to MAP on the final clipped perturbation: the root-mean-square $\mathrm{RMS} = \|\Delta\|_2/\sqrt{d}$, on MAP's 0–255 scale, and PSNR. Reading the constant curve at CMF's own value of each criterion gives Table 12.

Table 12: Query advantage under one L_1 and two L_2-family criteria (I3D/NES, 75 clips $\times$ 3 seeds, one protocol for all arms). The constant curve, $c/\sqrt{\mathbb{E}[W^2]} \in \{0.50, 0.644, 1, 2\}$, is read at CMF's own value of each criterion. The RMS and PSNR rows are interpolations with a two-sided bootstrap p. CMF's MAP falls just outside the curve's span, so the MAP row is the direct paired comparison with c=0.50.

Criterion	**CMF at**	**ΔCANQ**	**95% CI**	p
MAP $= \|\Delta\|_1/d$	2.427	-364.2	$[-584.3, -147.7]$	0.0010
RMS $= \|\Delta\|_2/\sqrt{d}$	4.068	$+182.5$	$[-73.2, +433.0]$	0.157
PSNR	37.70	$+206.0$	$[-30.4, +443.9]$	0.084

Under L_2-family matching the two allocations are not statistically separated on I3D/NES: neither interval excludes zero. The mechanism is visible in the arms themselves: the constant arm at c=0.50 reaches essentially CMF's MAP (2.432 against 2.427) while carrying 25% less $\|\Delta\|_2$ (3.050 against 4.068); equivalently, CMF carries 33% more. The ρ inequality that holds per iteration therefore survives into the final perturbation. The larger L_2 energy that a concentrated allocation is permitted at fixed L_1 is therefore a plausible contributor to the MAP-based advantage, though this experiment does not isolate it. The update-side L_2 difference alone is insufficient to reproduce the gain: the update-only arm of Section 4.3.2(A) carries CSA's per-iteration update L_1 and L_2 and is not sig-

nificantly cheaper than the matched control on either victim. We report the query advantage as a statement about mean absolute perturbation throughout, and MAP is the stealthiness measure prior video attacks report (Zhang et al., 2020; Wei et al., 2023; 2022), which is why the comparison is anchored there. At matched SSIM the point estimate favors CMF in all three cells but is significant in one. LPIPS matching is inconclusive: CMF falls outside the measured constant-arm range in two cells, and in the third the interval includes zero. These results do not establish a consistent perceptual ordering between the two allocations, matching the mixed metric ordering in Table 1. The direct MAP comparison (-364.2) closely agrees with the interpolated -346.1 of Table 10.

Disagreement among distortion measures. Against the matched control, CMF realizes *less* mean absolute perturbation in all five NES cells of Table 1 (2.53 against 2.99 on I3D/NES, and lower in each of the others) and higher SSIM in four of them, yet worse LPIPS in all five. We therefore report all three rather than treating any one of them as a complete account of perceptual quality. The present experiments do not isolate the source of this LPIPS reversal. What they do show is that these quantities are not interchangeable. MAP counts how much the pixels move; SSIM and LPIPS compare frames independently and so cannot see flicker at all. Appendix C.3.2 demonstrates the gap directly rather than by argument: permuting W along time leaves success rate, CANQ and every per-frame measure essentially unchanged while raising E_{flow} by 19.1% (p=0.0001). A perturbation can therefore be reorganized into something measurably less temporally coherent without any of the per-frame measures registering it. The user study of Appendix D.5 adds a fourth axis, what a viewer notices, but covers only the comparison against prior attacks and so does not settle the LPIPS ordering against the matched control.

The magnitude profile W^γ. Raising W to a power γ before use keeps the coordinate ordering but changes both the spread of the values and their mean, and query cost responds strongly to it; Section 4.3.2 defers the sweep here. It is not a free improvement: because γ moves $\mathbb{E}[W]$, a flatter profile spends more perturbation and a more concentrated one spends less. Table 13 gives the three points. A flatter allocation (γ=0.5) reaches CANQ 824 against CMF's 1,122, but at MAP 3.56 against 2.53, or 41% more perturbation, while a more concentrated one (γ=2) halves MAP and nearly doubles the query cost. The three points trace the same query–distortion trade-off as the step-size sweep above, which is why γ=1 is used throughout rather than the setting with the lowest CANQ.

Table 13: Magnitude profile W^γ (I3D/NES, 75 clips $\times$ 3 seeds). γ preserves the ordering of W but changes both its spread and its mean, so rows differ in budget as well as in concentration. γ=1 is CMF as used elsewhere.

	SR (%) ↑	**CANQ** ↓	**MAP** ↓	**SSIM** ↑	**LPIPS** ↓
γ=0.5 (flatter)	**92.89**	**824**	3.556	0.8922	0.0467
γ=1 (CMF)	92.00	1122	2.529	0.9277	0.0318
γ=2 (more concentrated)	83.56	1778	**1.452**	**0.9627**	**0.0159**

C.2 Probe–Update Coupling and Estimator Strength

C.2.1 Joint Weighting and Estimator Strength

Every result elsewhere in this paper uses a single antithetic pair per iteration (K=1), where the step direction is $\pm\text{sign}(T_m z)$ and the victim contributes one bit per iteration. If the allocation only compensated for a very noisy estimator, its advantage should shrink as K grows. Instead, averaging K pairs before taking the sign, at $1{+}2K$ queries an iteration out of the same $Q_{\max}$, the advantage over the control grows monotonically from -236 queries at K=1 to -654 at K=8, significant at every K (Table 14). What moves is mostly the control: its success rate falls from 91.11% to 83.11% as the budget buys fewer, better steps, while CSA's stays between 91.6% and 94.2%. The persistence across K argues against a gain that depends on single-pair estimation. Absolute CANQ is not comparable across rows.

A second victim. The same comparison on X3D/NES moves the same way: the advantage is -430.6 queries at K=1 ($[-581.9, -291.1]$) and -765.2 at K=8 ($[-950.5, -588.0]$), both $p < 10^{-4}$ over 225 clip–seed pairs, with the control's success rate falling from 92.44% to 84.00% while

CSA's holds at 94.67% and 93.78%. Both rows are produced under one execution configuration. As a check on that configuration, the stored campaigns give -433.4 ($[-574.9, -298.5]$) at $K{=}1$, within three queries of the value above.

Independent campaigns on the same clip–seed pairs differ by tens of queries, because their execution configurations consume different pseudorandom streams; every within-row comparison here uses one configuration, and the $K{=}1$ row is smaller than the corresponding cell of the separate campaign in Table 9 for that reason.

Table 14: Estimator strength (I3D/NES, 75 clips $\times$ 3 seeds). Each iteration averages K antithetic pairs at $1{+}2K$ queries; the matched control is re-run at each K with the same accounting. Compare ΔCANQ within a row only; absolute CANQ is not comparable across rows.

K	queries/iter	SR CSA	SR unif.	CANQ CSA	CANQ unif.	ΔCANQ	p
1	3	91.56	91.11	1133	1369	-235.9 $[-363.9, -111.9]$	0.0001
2	5	94.22	92.44	1054	1440	-386.3 $[-523.0, -257.4]$	$< 10^{-4}$
4	9	94.22	88.89	1140	1690	-549.8 $[-705.7, -402.3]$	$< 10^{-4}$
8	17	93.78	83.11	1324	1978	-653.5 $[-820.9, -493.4]$	$< 10^{-4}$

C.2.2 The Factorial on a Second Victim

The $2{\times}2$ of Section 4.3.2(A) runs on I3D/NES. To check that the coupling is not a property of that cell we repeated it on X3D/NES, over that victim's own 75 clips (built the same way as everywhere else, one clip per correctly classified class, so not the I3D set) and 3 seeds, with all four arms in one campaign so they share the clip set and the random-number stream within a seed and the interaction can be formed clip by clip. The criteria were fixed before the runs: whether the interaction $I = Q_{\text{joint}} - Q_{\text{probe}} - Q_{\text{update}} + Q_{\text{neither}}$ is negative, and whether the joint arm beats the arm with neither side weighted. The position of the update-only arm was deliberately not a criterion, since it was not separated on I3D either.

Both criteria are met (Table 15). The interaction is -806 queries (95% CI $[-1156, -486]$, sign-flip $p{=}10^{-4}$) against -950 on I3D, and the joint arm is 425 queries below the unweighted arm. The penalty for weighting the probe alone is $+542$ queries, against $+537$ on I3D, so the arm that suffers most is the same one and by almost the same amount. The update-only arm is again not separated (-162, $p{=}0.16$; it was $+143$, $p{=}0.15$ on I3D), which is why its position was not part of the test. The coupling therefore replicates on both victims we have tested; we do not claim it is necessary in general.

Table 15: Probe/update factorial on X3D/NES (75 clips, 3 seeds), replicating Section 4.3.2(A). ΔCANQ is against the arm with neither side weighted. SR and CANQ: mean $\pm$ s.d. over seeds.

probe	update	**SR** (%) $\uparrow$	**CANQ** $\downarrow$	ΔCANQ vs. neither
✓	✓	94.67 ± 0.00	$\mathbf{763.2 \pm 4.7}$	-425 $[-670, -199]$, $p{=}0.0003$
✓	✗	83.11 ± 0.77	1730.9 ± 25.2	$+542$ $[+276, +826]$, $p{=}0.0001$
✗	✓	92.89 ± 0.77	1026.8 ± 29.4	-162 $[-394, +54]$, $p{=}0.16$
✗	✗	92.89 ± 0.77	1188.5 ± 42.8	—
interaction I (per seed -779, -723, -916)				-806 $[-1156, -486]$, $p{=}10^{-4}$

C.2.3 Allocation Under i.i.d. Probes

Every arm elsewhere in this paper draws its base probe as $T_m z$, warped along the same codec motion that W is built from, so the scale the allocation applies and the direction structure of the probe could be entangled. We reran both arms with that structure removed, drawing independent noise on every frame with no warp, so the allocation is the only structure left in the probe. The matched control still sets its constant probe weight to $\sqrt{\mathbb{E}[W^2]}$ and its step to $\eta\,\mathbb{E}[W]$, so probe RMS and per-iteration update L_1 remain matched as in Table 1. The single criterion, fixed before the runs, was whether the advantage keeps its sign.

The sign holds on both victims and the advantage grows: -455 queries on I3D/NES (95% CI $[-781, -165]$, p=0.003) and -577 on X3D/NES ($[-867, -302]$, p=0.0002), over 75 clips and 3 seeds, against -320 and -433 with the motion-correlated probe. In absolute terms, CANQ with the i.i.d. probe is 1,141 for CSA and 1,596 for the control on I3D/NES and 712 and 1,289 on X3D/NES, against 1,114, 1,433, 754 and 1,188 with the motion-correlated probe in the separate campaign of Table 1. In these two cells the benefit of continuous allocation does not require the probe structure our default video instantiation pairs it with. We tested NES only; cross-optimizer generality is covered separately in Section 4.3.1.

C.3 Allocation Structure

C.3.1 Placement and Provenance

Section 4.3.2(C) permutes W within a frame. That preserves the value multiset but destroys the allocation's smoothness along with its placement, so it cannot tell an allocation that must come from this clip apart from any graded profile of the same shape. Two further surrogates separate the two. Both take the clip's own CMF values and lay them out in the rank order of a different structure, so the value multiset is preserved *exactly*: $\mathbb{E}[W]$, $\mathbb{E}[W^2]$, ρ and d_{eff} agree with CMF's to every digit, verified per clip, and only the origin of the layout changes. *randsmooth* uses a random field that never saw the input; *crossclip* uses another clip's CMF map, drawn from a bank of all 75 clips at a fixed half-list rotation so a clip never donates to itself.

The surrogates are as smooth as what they replace. Holding the multiset fixed is not sufficient on its own: a surrogate that is markedly rougher than CMF could lose for that reason rather than for its provenance. *randsmooth*'s spatial blur and temporal smoothing are therefore fitted per clip to CMF's own lag-1 autocorrelations, a fit that reads only W and costs no queries. Table 16 reports the resulting arrangement. The autocorrelation columns are the fitted ones and track CMF closely; the total-variation columns are not fitted and are reported as they fall, both surrogates retaining somewhat more frame-to-frame variation than CMF. The surrogates are not identical to CMF in arrangement, but they sit in its regime and far from the i.i.d. permutation.

Table 16: Arrangement of each layout over the 73 clips of Table 17 with $W \not\equiv 0$. All rows share CMF's value multiset, so $\mathbb{E}[W]$, $\mathbb{E}[W^2]$, ρ and d_{eff} are identical and only the layout differs. TV is normalized by $\mathbb{E}[W]$; AC(1) is the lag-1 autocorrelation, fitted to CMF's for *randsmooth* only. Mean $\pm$ s.d. over clips.

	TV / $\mathbb{E}[W]$		AC(1)	
	spatial	temporal	spatial	temporal
CMF	0.105 ± 0.058	0.085 ± 0.038	0.994 ± 0.003	0.944 ± 0.029
randsmooth	0.096 ± 0.062	0.210 ± 0.084	0.997 ± 0.003	0.943 ± 0.032
crossclip	0.111 ± 0.091	0.168 ± 0.127	0.991 ± 0.006	0.908 ± 0.068
spatial permutation	2.132 ± 1.067	1.071 ± 0.526	0.131 ± 0.186	0.118 ± 0.171

Result. Table 17 measures each arm against the same control Table 1 uses, the matched control; since every arm shares CMF's $\mathbb{E}[W]$ and $\mathbb{E}[W^2]$, that one control is the correct comparison for all of them. A failure enters CANQ at $Q_{\max}$, so this axis is not conditioned on success. Spatial permutation retains the whole advantage, reproducing Section 4.3.2(C). The two arrangement-matched surrogates do not: *randsmooth* keeps 63% of it and remains significant, while *crossclip* keeps 25% and its interval covers zero. Within-frame permutation preserves the query saving, so CSA does not require pixel-exact alignment with local motion; replacing the layout with another clip's raises query cost, although that intervention also changes the framewise weight profile. Because both surrogates reuse the target clip's CMF values and randomize only where those values sit, these controls isolate layout provenance rather than the origin of the value profile, which in every arm here is still the clip under attack. Since every arm carries CMF's exact value multiset, $\mathbb{E}[W]$, $\mathbb{E}[W^2]$, ρ and the global L_1 and L_2 geometry of the allocation are identical across rows, so the differences between them cannot be attributed to ρ alone.

What the rank matching does not hold fixed. The multiset is matched over the whole clip, not per frame: the layout is laid out in the rank order of the donor structure after flattening, so a surrogate can place more of the total weight on some frames than CMF does. Measured over the weight bank, the per-frame $\sum_j W_j$ of a relaid layout departs from CMF's by 31% on average and the per-frame $\sum_j W_j^2$ by 49%. Part of what separates the surrogates from CMF may therefore be how the budget is divided between frames rather than layout provenance as such, and we do not separate the two here.

Table 17: Layout provenance (I3D/NES, 75 clips $\times$ 3 seeds). All arms carry CMF's value multiset. "vs. control": ΔCANQ against the matched control (SR 91.11%, CANQ $1{,}433$). "retained": share of CSA's own advantage over that control.

	SR (%)	CANQ	vs. control	p	retained
CMF	91.56	1133	$-300.3\ [-439.4, -166.7]$	0.0001	–
spatial permutation	91.56	1130	$-303.3\ [-439.1, -169.3]$	$< 10^{-4}$	101%
randsmooth	89.33	1243	$-190.3\ [-303.3,\ -81.1]$	0.0006	63%
crossclip	88.89	1359	$-74.0\ [-191.0,\ +39.8]$	0.21	25%

C.3.2 Temporal Organization and Consistency

SSIM and LPIPS are per-frame and cannot see flicker, so we measure the residual frame-to-frame change of the perturbation directly. E_{raw} is the mean absolute difference between consecutive perturbation frames, normalized by the mean perturbation magnitude; E_{flow} first warps the previous frame's perturbation by the optical flow of the *clean* clip, estimated independently of the attack with Farnebäck's method, so that motion the perturbation merely follows is not counted as flicker. Lower is smoother. Table 18 reports both on the clips of block (C) of Table 2: the I3D/NES runs at seed 0, over the 23 of 25 saved clips whose perturbation is non-degenerate. The two excluded clips are the ones with $W \equiv 0$; the clean frames, stored at 8 bits, are used only to estimate the flow, which the metric would quantize anyway, while the perturbations are read from the full-precision arrays.

Temporal permutation raises E_{flow} by 19.1% (Δ=$+0.163$, 95% CI $[+0.108, +0.231]$, paired sign-flip p=0.0001), while leaving SR and the per-frame metrics alone. Its effect on query cost is not resolved at this sample size: the paired difference is -134.8 queries with a 95% interval of $[-318.1, +24.1]$ (p=0.12, 75 clips $\times$ 3 seeds), so the point estimate favors the permuted arm but the interval covers zero. Removing GOP propagation leaves E_{flow} essentially unchanged (0.851 versus 0.853), so the propagation step is not what keeps the perturbation temporally coherent.

Table 18: Temporal consistency of the perturbation for the placement and propagation arms of Table 2 (I3D/NES, one seed, n=23; mean $\pm$ s.d. over clips; lower is smoother). E_{flow} is not reported for the spatial permutation, which redraws the allocation independently in each frame, so any flicker it induces would reflect the control rather than placement.

Arm	n	$E_{\text{raw}} \downarrow$	$E_{\text{flow}} \downarrow$
CMF	23	0.708 ± 0.28	0.853 ± 0.22
temporal permutation of W	23	0.969 ± 0.20	1.016 ± 0.15
GOP propagation removed	23	0.713 ± 0.29	0.851 ± 0.23
spatial permutation of W	23	1.311 ± 0.34	–

C.3.3 Concentration Analysis

Section 5 reads CSA's advantage against how concentrated the allocation is. The effective dimensionality $d_{\text{eff}}(X) = (\sum_j W_j)^2 / \sum_j W_j^2$ is scale-invariant, lower values meaning a more concentrated map. Table 19 stratifies by it the Kinetics-400 attack instances behind Table 1 (4 victims $\times$ 2 optimizers $\times$ 75 clips $\times$ 3 seeds) and reports, within each quintile, the mean per-instance cost of CSA and of the matched control. Cost is the CANQ summand: queries on success, $Q_{\max}$ on failure, so it is not conditioned on success. Following Appendix B.4, the 42 instances with $W \equiv 0$

are excluded here, d_{eff} being undefined for them, leaving 1,758; they remain in every attack statistic reported elsewhere, where they are genuine failures.

The saving falls monotonically across quintiles, from 1,240 queries per instance in the most concentrated to 9 in the most diffuse, and the success-rate gain sits in the first quintile (+10.2 percentage points; the other four are within 1.5 points of zero). This is an observational stratification on a quantity that is not manipulated: $d_{\text{eff}}/d = \mathbb{E}[W]^2/\mathbb{E}[W^2] = 1/\rho^2$, so concentration and the norm geometry of Appendix C.1.2 are not separated by it. We use d_{eff} as a query-free diagnostic of where CSA is likely to help, not as a causal account of why.

Table 19: CSA against the matched control by quintile of d_{eff}/d (Q_1 most concentrated), over the 1,758 Kinetics-400 instances of Table 1 with $W \not\equiv 0$ (Appendix B.4). Cost is averaged per instance. The stratifying variable is observed, not manipulated.

			Cost (queries)			**Success rate** (%)		
Quintile	d_{eff}/d	n	CSA	Control	Δ	CSA	Control	Δ
Q_1 (most concentrated)	0.088	352	1630.0	2870.0	−1240.0	82.67	72.44	+10.23
Q_2	0.242	352	1232.1	1580.7	−348.6	90.62	89.49	+1.14
Q_3	0.451	352	725.9	909.9	−183.9	99.43	100.00	−0.57
Q_4	0.605	351	737.7	879.4	−141.7	99.15	98.86	+0.28
Q_5 (most diffuse)	0.825	351	910.4	919.1	−8.7	94.02	95.44	−1.42

C.3.4 Qualitative Allocation Structure

Figure 6 shows what an input-derived allocation looks like on a real clip. Its role here is illustrative: whether this placement is what produces the query saving is tested separately in Appendix C.3.1, and within-frame permutation preserves the saving.

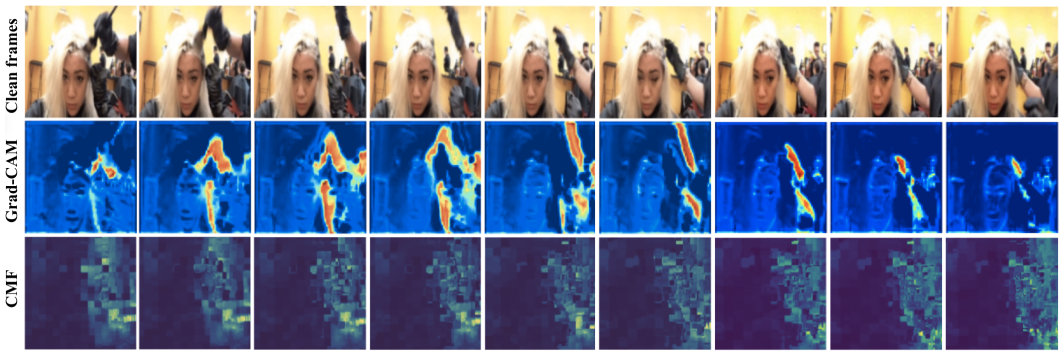


Figure 6: The CMF allocation on a Kinetics-400 clip (*dyeing hair*), computed without victim-model access; Grad-CAM (middle row) is shown as scene context only. Its role in the query saving is tested in Appendix C.3.1.

C.4 Reversal Under Tight Perturbation Budgets

C.4.1 Perturbation-Budget Sensitivity

Every result elsewhere in this paper uses the L_∞ bound ε_∞=0.1 (25.5/255). Because a graded allocation asks for more perturbation on some coordinates than a constant one does, the bound is a plausible moderator of the advantage, so we re-ran both arms at two tighter bounds under one protocol. Table 20 reports all three.

The advantage is not uniform across budgets. At 8/255 the matched control is *more* query-efficient than CSA on both victims, by 693 and 447 queries, and also more successful by 20.0 and 15.1 percentage points. At 16/255 the two are statistically indistinguishable on I3D and CSA is ahead on X3D. At the default budget CSA is ahead on both. Across the three tested budgets CSA's relative

query advantage therefore increases as the constraint is relaxed, with the crossover falling between $8/255$ and $16/255$ on X3D and between $16/255$ and $25.5/255$ on I3D. We do not extrapolate this ordering beyond the budgets measured.

Table 20: CSA against the matched control at three L_∞ bounds (I3D and X3D, NES, 75 clips $\times$ 3 seeds). Both arms are re-run at each bound under one execution configuration; $25.5/255$ (ε_∞=0.1) is the bound used elsewhere.

		SR (%)		**CANQ**				
Victim	ε_∞	CSA	Control	CSA	Control	ΔCANQ	95% CI	p
I3D	$8/255$	63.11	**83.11**	2519.3	**1825.9**	$+693.4$	$[+510.5, +874.6]$	$< 10^{-4}$
	$16/255$	83.56	88.89	1483.2	1453.3	$+29.9$	$[-106.9, +166.7]$	0.67
	$25.5/255$	**92.00**	91.11	**1113.7**	1433.3	-319.6	$[-453.2, -189.1]$	$< 10^{-4}$
X3D	$8/255$	71.11	**86.22**	2002.2	**1554.9**	$+447.4$	$[+262.0, +631.4]$	$< 10^{-4}$
	$16/255$	**93.33**	92.89	**974.0**	1184.5	-210.4	$[-338.9, -82.3]$	0.0016
	$25.5/255$	**95.56**	93.33	**754.2**	1187.6	-433.4	$[-576.7, -299.4]$	$< 10^{-4}$

C.4.2 Allocation Concentration and Projection

The crossover of Appendix C.4.1 raises a question the budget sweep alone cannot answer: is it concentration of the allocation that hurts at a tight bound, or something else that changes with ε_∞? To separate the two we hold the allocation's budget fixed and vary only its heterogeneity. Motion supplies a percentile rank r_j and the allocation is $W_j = b\,(\delta + r_j)^\kappa / \mathbb{E}[(\delta + r)^\kappa]$, so $\mathbb{E}[W] = b$ for every κ: the update L_1 budget is constant and κ alone sets the spread. We additionally hold the probe energy fixed by renormalizing the probe allocation to a common RMS, so the finite difference sees the same L_2 energy at every κ. What cannot be held fixed is the update L_2, which rises with heterogeneity by the same ρ inequality as in Appendix C.1.2; that is the quantity under study.

This is a post-hoc diagnostic, designed after observing the reversal at $8/255$, and it is run on one victim–optimizer cell (I3D/NES, 75 clips $\times$ 3 seeds, ε_∞=$8/255$).

Table 21 reports the sweep. Query efficiency falls monotonically as the allocation concentrates, and the clipping diagnostic shows where the cost is paid: the fraction of coordinates that end at the L_∞ boundary is equal in the top and bottom weight quintiles for a constant allocation, but at κ=2 the top quintile saturates at 5.25% while the bottom quintile never saturates at all. Projection therefore disproportionately truncates the high-weight coordinates on which a concentrated allocation places most of its update mass. We read this as linking the crossover to projection-induced truncation rather than as isolating it from the accompanying change in update L_2.

Table 21: Allocation heterogeneity at ε_∞=$8/255$ (I3D/NES, 75 clips $\times$ 3 seeds). κ sets the spread of a rank-calibrated allocation at fixed $\mathbb{E}[W]$ and probe RMS; κ=0 is constant. Saturation: share of coordinates whose final perturbation reaches the bound, in the highest- and lowest-weight quintiles of W.

				Saturated coordinates (%)	
κ	$\mathbb{E}[W]$	**SR** (%) ↑	**CANQ** ↓	top quintile	bottom quintile
0 (constant)	0.188	**79.56**	**1937.2**	1.51	1.51
1	0.188	75.56	2085.7	3.69	0.11
2 (concentrated)	0.186	67.26	2466.0	5.25	0.00

Step size does not reproduce this mechanism. Increasing the update step η also makes CANQ non-monotonic, and it is natural to ask whether projection explains that too. It does not. Rerunning I3D/NES at $\eta \in \{0.005, 0.01, 0.025, 0.05\}$ with one seed at the default ε_∞=0.1, high-weight boundary occupancy rises from 0.50% to 10.40% as η grows and stays 22 to 66 times that of the low-weight quintile, yet within each step size it is uncorrelated with whether the attack succeeds ($|r| \le 0.061$, all $p > 0.60$). Larger steps do saturate high-weight coordinates selectively, but that saturation is not what decides which clips fail, which is consistent with a large-step optimization

effect rather than a saturation-driven failure mode. The mechanism above is therefore specific to a tight budget and does not carry over to the step-size axis.

D Additional Evaluation

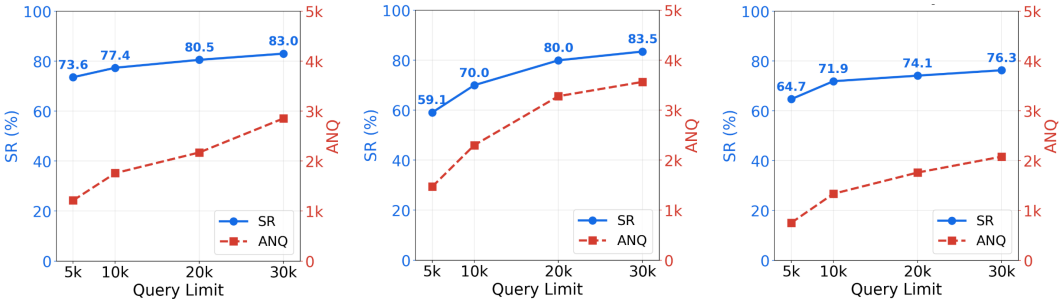
D.1 A Transformer Victim

The four Kinetics-400 victims of Table 1 are convolutional. As a check on whether the allocation behaves differently against a different inductive bias, we ran the same paired comparison on MViTv2-S (Li et al., 2022), a video transformer, at the same $24\times128\times128$ input and the same hyperparameters as the convolutional victims. Under the clean-accuracy protocol of Section 4 it reaches 50.50% top-1, comparable to SlowFast's 52.75% and well above I3D's 39.25% and X3D's 38.00%.

CSA shows the same directional reduction in CANQ as on the convolutional victims, 1758.4 against the control's 1965.2, a difference of -206.9 queries. The paired difference is not statistically significant at this sample size: the 95% bootstrap interval is $[-469.9, +49.8]$ and the sign-flip permutation test gives p=0.13 over 75 clips and 3 seeds. Success rate moves the same way and is likewise not separated (81.78% against 78.22%, p=0.48), while MAP is lower for CSA (2.68 against 2.93) and LPIPS higher (0.039 against 0.029), the same mixed perceptual ordering the convolutional cells show. The point estimate for query cost is directionally consistent with the convolutional results, but its confidence interval spans zero; we therefore treat this as a supplementary generalization check rather than as evidence of a transformer-wide advantage. It is a single victim under one optimizer, run with hyperparameters chosen for the convolutional victims and not retuned.

D.2 Success–Query Trade-off

Figure 7 sweeps the query budget from 5k to 30k. CSA converts extra budget into harder samples at moderate query growth, and ME-Sampler stays below 77% SR up to 30k queries. This plateau is why ANQ alone misleads: at 30k, ME-Sampler's ANQ is ≈2,100 but its CANQ is 8,796, because 24% of samples are never solved. We report the strict 5k budget throughout, instead of the 30k–100k budgets common in prior work, because that is the regime where this distinction matters.



(a) CSA (b) AstFocus (c) ME-Sampler

Figure 7: Success-query trade-off on Kinetics-400 (SF, targeted) under increasing query budgets (5k–30k). Each method runs at its own L_∞ bound (Table 3), so the curves are not matched in perturbation bound.

D.3 Defense Evaluation

Each defense is placed inside the victim query loop and CSA is re-run against the defended pipeline, so these are not transferred adversarial examples. The attack is evaluated non-adaptively, without EOT or BPDA; adaptive evaluation may therefore recover additional attack success. Table 22 reports UCF101 and Table 24 Kinetics-400.

Table 23 reports the clean-video quality trade-off each defense makes; clean frames under each defense are in Figure 9.

Table 22: Test-time defenses on UCF101 (EfficientNet-B0, $Q_{\max}$=5,000), evaluated non-adaptively, over 44 clips the undefended attack fools. n: clips whose benign prediction the defense preserves; BA = $n/44$ (%); attack metrics are over those n. Temporal (k): each frame replaced by the mean of frames $t-k$ to $t+k$. CMF-G / CMF-R: CMF-guided Gaussian / randomized smoothing.

Defense	n	BA↑	SR(%)↑	ANQ↓	CANQ↓	MAP↓	SSIM↑	LPIPS↓
JPEG (q=75)	38	86.36	44.74	1956.8	3638.6	4.29	0.8584	0.0622
Smoothing (r=0.7)	34	77.27	58.82	1323.1	2837.1	2.82	0.9044	0.0416
Temporal (k=1)	40	90.91	92.50	1845.8	2082.3	3.43	0.8777	0.0476
Temporal (k=2)	38	86.36	92.11	1741.4	1998.7	3.44	0.8791	0.0478
CMF-G (σ=1.0)	41	93.18	97.56	1804.5	1882.5	3.53	0.8778	0.0494
CMF-G (σ=1.5)	40	90.91	95.00	1883.4	2039.3	3.52	0.8779	0.0508
CMF-R (σ=0.05)	44	100.00	77.27	1904.4	2608.0	4.02	0.8615	0.0594
CMF-R (σ=0.10)	42	95.45	61.90	2316.5	3338.8	4.21	0.8588	0.0561

Adversarially trained victim (Madry et al., 2018) (PGD, $\varepsilon=2/255$, $\alpha=1/255$, 3 steps), a separate model at $10\times224\times224$ on a different clip set: CSA retains 50.67% SR, not comparable with the rows above.

Table 23: Clean-video degradation under each defense (no attack). Higher SSIM and PSNR (dB) and lower LPIPS mean less degradation.

(a) Existing defenses				(b) CMF-aware defenses			
Defense	**SSIM↑**	**LPIPS↓**	**PSNR↑**	**Defense**	**SSIM↑**	**LPIPS↓**	**PSNR↑**
JPEG (q=75)	0.9519	0.0131	33.60	CMF-G (σ=1.0)	0.9909	0.0183	41.33
Smoothing (r=0.7)	0.9492	0.0995	30.47	CMF-G (σ=1.5)	0.9836	0.0265	38.80
Temporal (k=1)	0.9636	0.0357	34.47	CMF-R (σ=0.05)	0.9595	0.0097	41.06
Temporal (k=2)	0.9329	0.0625	31.33	CMF-R (σ=0.10)	0.9004	0.0377	35.13

Table 24 shows attack performance under JPEG compression and image smoothing on Kinetics-400 (SF model). At the method-specific L_∞ bounds of Table 3, SR under JPEG is 94.94% for CSA, 78.48% for ME-Sampler and 14.09% for AstFocus; under smoothing it is 94.29%, 71.43% and 71.00%. The bounds differ across methods, so these are single operating points rather than a matched robustness comparison.

Table 24: Comparison of attack performance under test-time defenses (JPEG compression and image smoothing) on the SF model (Kinetics-400). Metrics are computed only over clips whose benign prediction is preserved by the defense. This is a selected population that differs from the undefended runs of Table 3, so absolute query counts are not comparable with that table. Each method keeps its own L_∞ bound, as in Table 3.

Defense	Attack Method	SR (%) ↑	ANQ ↓	CANQ ↓	MAP ↓	SSIM ↑	LPIPS ↓
JPEG	CSA	**94.94**	377.46	**611.51**	**1.485**	0.9294	0.1121
	AstFocus	14.09	**117.13**	4312.00	2.978	0.8892	**0.0986**
	ME-Sampler	78.48	407.66	1395.93	6.618	**0.9578**	0.1951
Smoothing	CSA	**94.29**	434.82	**695.69**	**1.721**	0.9160	0.1505
	AstFocus	71.00	**62.93**	1494.68	3.494	0.8881	**0.1159**
	ME-Sampler	71.43	595.06	1853.55	6.585	**0.9650**	0.1840

D.4 Temporal Subsampling

Because CMF is read from frame-to-frame transitions, a preprocessing step that changes which transitions exist is a more direct threat to it than a spatial filter. We therefore keep every s-th frame, drop the rest, and resample back to the original T=24 length by nearest-neighbor temporal upsampling, which preserves clip length and frame rate at the model's input while destroying the alignment between the perturbation and the motion it was built on. Table 25 reports the fraction of clips still

misclassified afterward, on adversarial examples that fool the undefended victim in every case. Success falls on all four victims, to 39.7% on average, and there is no consistent trend in s: dropping more frames is not reliably worse, which is consistent with retention depending on which frames carry the perturbation rather than on its total energy. This is a post-hoc transfer test: subsampling is applied after the attack, which never observes the subsampling operator.

Table 25: Retention of CSA adversarial examples under temporal subsampling (Kinetics-400): every s-th frame is kept and the clip is resampled to its original length. All examples fool the undefended victim, so each value is a retention rate (%).

Victim	n	s=2	s=3	s=4	**mean**
I3D	23	30.43	34.78	34.78	33.33
SF	21	52.38	52.38	71.43	58.73
SO	24	37.50	37.50	37.50	37.50
X3D	24	29.17	37.50	20.83	29.17
mean		37.37	40.54	41.13	**39.68**

D.5 User Study and Qualitative Examples

Protocol. Each participant saw four video sets, one per victim backbone. A set played the original clip and its three attacked versions *simultaneously* in a 2×2 grid, the original in the top-left cell and the three attacks labeled *Number 1–Number 3*, so the method behind each cell was hidden from raters; participants were asked to view each set full-screen before answering. For every set they rated each attacked version against the original on a six-point similarity scale anchored at 1 ("very low similarity") and 6 ("identical"), so a higher score means harder to tell apart from the original, and equal ratings were explicitly permitted. This gives twelve ratings per participant, one for each backbone and attack; a thirteenth item was an attention check instructing raters to select a fixed response. The exact question wording was "How similar is Number N to the Original in the [backbone] video?". The assignment of the three attacks to the *Number* labels was randomized when the stimuli were constructed, and one clip was drawn at random for each backbone; because each set is a single pre-rendered video, every participant saw the same clips under the same label assignment.

The study was submitted to the relevant institutional review body and determined to be exempt; the institution and the determination number are withheld here for anonymous review and will be restored in the camera-ready version. Participants were adults who gave informed consent before taking part, through an explicit consent item that had to be accepted before any stimulus was shown; the study was administered partly online and partly in person, and participants received compensation of roughly US$3. No personally identifying information, demographic data or sensitive attributes were collected, and participation was revocable at any point. Predefined exclusion criteria removed responses that failed the attention check or that met the study's prespecified response-quality criteria for exclusion; n=34 is the final number of participants retained for analysis. Table 26 reports, for each backbone, the mean rating over the 34 participants, and the *Average* column the mean of those four backbone means.

Scope. The study covers the three attacks of Table 3 on one clip per backbone, so it does not measure variation across clips, and the per-response data were not retained, so no dispersion or significance test can be reported for Table 26. Viewing conditions were not uniform, the study having been run both online and in person. It does not include the matched control of Table 1 or any allocation ablation. We therefore read it as evidence about CSA against the two prior video attacks it shows, and not as a resolution of the metric disagreements discussed in Appendix C.1.2.

Qualitative examples. Figure 8 shows representative adversarial frames from the three attacks, and Figure 9 shows clean frames under each defense of Appendix D.3. Quantitative perceptual comparisons are in Tables 3 and 26; the comparison against the matched control is in Table 1.

Table 26: User study: mean similarity to the original on a six-point scale (6 = identical), n=34 participants, one clip per backbone, each attack at its own L_∞ bound (Table 3). No matched control is included, and no dispersion is reported (see *Scope* in Appendix D.5). *Average*: mean of the four backbone means.

Method	I3D	SF	SO	X3D	Average
CSA (ours)	**4.39**	**4.39**	**4.69**	**3.88**	**4.34**
AstFocus	3.83	3.84	3.89	3.78	3.84
ME-Sampler	2.62	2.66	2.98	2.54	2.70

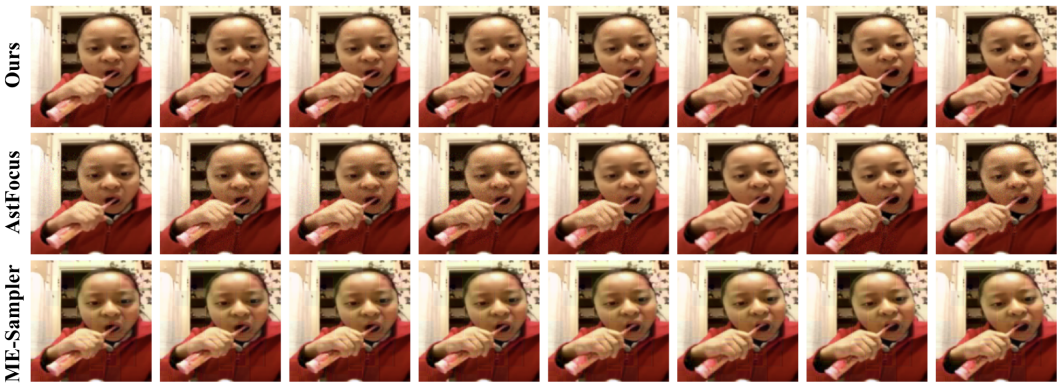


Figure 8: Representative adversarial frames from CSA (top), AstFocus (middle) and ME-Sampler (bottom), each run at its own L_∞ bound (Table 3). The bounds differ, so these panels are not a matched perceptual comparison.

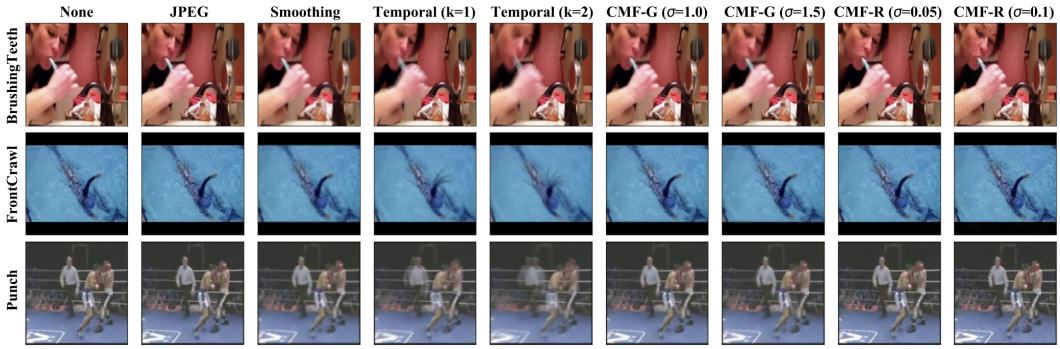


Figure 9: Clean frames under each defense (no attack).

E Extended Related Work

Table 27 places CSA against prior black-box video attacks and the image-domain zeroth-order methods that adapt per-coordinate effort, along the axes Section 2 uses: what structure each method injects into the search, where that structure comes from, and how it enters the optimizer.

Two recent video attacks would sit in the support row of that table and sharpen where the boundary falls. RLVS (Song et al., 2025) selects key frames with a reinforcement-learning policy and estimates the gradient sign only on them, and CubeAgent (Shi et al., 2025) selects spatiotemporal "video cubes" with a deep reinforcement-learning policy. Both choose *which* coordinates enter the search and adapt that choice from victim feedback during the attack, so they are query-adaptive support selection; neither grades how strongly a retained coordinate is explored, and neither derives its structure before the first query. We position them here rather than comparing numbers, for the same reason as the other support-selection methods: the quantity they control is a different one.

Table 27: Query-based black-box attacks by the aspect of the zeroth-order search their structure governs. *Prior from*: *fixed* (chosen a priori), *dataset* (natural-image statistics), *input* (the clip under attack), *victim* (queried), *history* (the attack's past queries), *fitness* (queried objective values), *surrogate* (a separate model); —: none. *Space*: restricts reachable directions. *Support*: selects where perturbation may be placed. *Reuse*: reuses transformed copies of one pattern across frames. *Scale*: grades search magnitude per coordinate. [a]Coordinate sampling probability proportional to the perturbation found so far. [b]Per-coordinate step from running moment estimates. [c]General-purpose optimizers (with separable CMA-ES), not attacks; the diagonal scale is adapted online from queried fitness rather than read from the input. †Hard-label attack built on OPT-ATTACK, listed for its structure only; it does not share the score-based access of the other rows.

Category	Method	Prior from	Space	Support	Reuse	Scale
No structure	V-BAD (Jiang et al., 2019)	—				
	NES (Ilyas et al., 2018)	—				
Search-space parameterization	SimBA (Guo et al., 2019)	fixed	✓			
	Square (Andriushchenko et al., 2020)	fixed	✓			
	Bandits-TD (Ilyas et al., 2019)	dataset + history	✓			
	P-RGF (Cheng et al., 2019)	surrogate	✓			
	TenAd (Haghjooei & Rezghi, 2025)†	input	✓			
Support selection	AstFocus (Wei et al., 2023)	victim		✓		
	QEBB (Haghjooei & Rezghi, 2024)	input		✓		
	FDP (Jin et al., 2025)	victim		✓		
Perturbation reuse	Geo-TRAP (Li et al., 2021)	fixed			✓	
	ME-Sampler (Zhang et al., 2020)	input			✓	
Coordinate-wise scale	SNES (Wierstra et al., 2014)	fitness[c]				✓
	ZOO (Chen et al., 2017)	history				✓[a]
	ZO-AdaMM (Chen et al., 2019)	history				✓[b]
	CSA (Ours)	**input**				✓