# SAFER: WHEN IS ACQUIRING FORENSIC EVIDENCE WORTH THE COST IN AGENTIC AI-GENERATED VIDEO DETECTION?

**Anonymous authors**
Paper under double-blind review

## ABSTRACT

Agentic approaches have recently emerged for AI-generated video detection, allowing vision-language models (VLMs) to acquire and reason over specialized forensic evidence. However, forensic tool calls incur additional computational latency, while their marginal utility relative to this cost remains unquantified. To close this gap, we propose SAFER, a cost-aware agentic framework that treats forensic analysis as evidence acquisition under a computational budget. SAFER combines a VLM frontline with a dispatcher that decides whether additional spatial, spectral, or latent forensic evidence is worth acquiring, and an arbiter that integrates the acquired evidence. A same-reasoner no-tool prediction on every forward pass provides a controlled baseline for measuring the marginal value of forensic evidence without confounding tool utility with differences between models. Experiments on Chrono-66k and four external benchmarks show that the forensic tools contain discriminative signal in isolation, yet provide limited incremental value beyond the VLM frontline. In-domain, full tool acquisition fixes only 6 of 20 remaining errors while increasing latency by $1.71\times$; across 44 unseen generator and editing categories, its mean AUC contribution is $-0.007$. The cost-aware dispatcher consequently learns to bypass unnecessary tool calls, matching or exceeding full-tool baselines while operating at approximately $0.60\times$ their latency. Finally, we demonstrate that generalization failures under domain shift stem primarily from miscalibrated scoring rather than missing forensic evidence; on three of four external benchmarks, simple decision-threshold recalibration using a small target sample improves balanced accuracy by 4.3 to 12.9 points, outperforming adapted tool-routing strategies.

## 1 INTRODUCTION

Nowadays, generative video models can produce increasingly realistic visual content, while AI-based editing methods can selectively manipulate regions of authentic videos. Detectors built on vision-language models (VLMs) judge what a video shows, and a growing line of work adds forensic tools that measure how it was made: spectral statistics (Qian et al., 2020; Corvi et al., 2025), reconstruction error under a diffusion autoencoder (Wang et al., 2023; Liu et al., 2024), and local texture and motion cues. Agentic detectors go further and let a multimodal model call such tools and reason over their outputs (Liu et al., 2025a; Li et al., 2026; Wu et al., 2026).

Two questions are usually left open. The first is cost. Every tool call adds latency, and a detector that screens large volumes of video, pays that latency on every item, so a tool is worth calling only if it changes enough decisions. The second is measurement. The value of a tool is the difference between a reasoner with the tool and the same reasoner without it. Comparing a tool-using model against a different model, such as a smaller scanner, mixes the value of the tool with the difference between the two models. We therefore ask: *when is acquiring forensic evidence worth its computational cost for an agentic VLM-based detector?*

To answer this, we introduce SAFER, a cost-aware detection framework explicitly engineered to measure and optimize the marginal utility of forensic tools. SAFER enforces a strict separation of concerns. A fast VLM frontline (comprising a scanner and an arbiter) executes a zero-tool forward



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.

pass on every video. A latency-aware dispatcher then evaluates the frontline's state to determine if invoking forensic tools, which compute 23 distinct spatial, spectral, and latent measurements., is worth its computational cost. If invoked, the arbiter processes these measurements as an evidence graph. Because SAFER computes a zero-tool baseline on every single run, it intrinsically measures the precise value added by external forensics against the same reasoning engine, at no additional computational cost.

To rigorously evaluate this framework, we curated a comprehensive 66k-video meta-dataset called Chrono-66k and standardized evaluation across four external benchmarks, testing specifically for generalization under domain shift. Our empirical analysis reveals a counter-intuitive reality about modern deepfake detection. While forensic tools isolate valid signals (achieving $96.39\%$ macro-F1 in isolation), they are overwhelmingly redundant when paired with a highly capable VLM frontline. In our in-domain testing, running the full tool suite fixed only 6 out of 20 remaining errors while inflating latency by $1.71\times$. Moreover, on 44 unseen generative categories, the average value added by tools collapsed to zero.

Our findings indicate that when detectors fail on novel synthetic media, many of the remaining thresholded errors under domain shift are associated with miscalibrated scoring rather than a lack of recoverable forensic evidence. SAFER's dispatcher independently learns this reality, defaulting to a tool-skipping policy that matches or exceeds full-tool baselines at $0.60$ the latency.

We formalize our contributions as follows:

1. **The SAFER Architecture:**We propose a cost-aware detection framework that dynamically routes forensic tool execution and natively computes the marginal value of tool usage without auxiliary forward passes.(Section 3).
2. **Quantifying Forensic Redundancy:** We empirically demonstrate that while forensic tools provide strong isolated signals, they are largely redundant alongside a modern VLM frontline, yielding negligible performance gains (fixing 6 of 20 errors) at severe latency costs ($1.71\times$). (Section 4.1).
3. **Exposing Tool Failure Under Domain Shift:** We show that tool utility does not transfer to unseen generative distributions. Across 44 unseen categories, the delta in AUC when adding tools averages $-0.007$, a failure masked by uncontrolled comparisons. (Sections 4.2 and 4.3).
4. **Calibration over Computation:** We demonstrate that threshold recalibration using a minimal labeled sample from a target domain improves balanced accuracy by 4.3 to 12.9 points across three benchmarks, vastly outperforming complex tool-routing strategies (0.6 to 3.3 points) and indicating that calibration can be a more effective lever than tool routing under the evaluated distribution shifts.(Section 4.4).

## 2 Related Works

AI-generated video detection has evolved alongside increasingly diverse evaluation benchmarks. Early datasets primarily focused on face manipulation (Rössler et al., 2019; Jiang et al., 2020; Dolhansky et al., 2019), while recent benchmarks cover broader forms of video generation, including GenVideo (Chen et al., 2024b), GenVidBench (Ni et al., 2026), GenBuster (Wen et al., 2025), and FaceVid-Forensics-100K (Zou et al., 2026). FakePartsBench further considers partial manipulation of otherwise real videos (Liu et al., 2025b). However, differences in source quality, frame rate, encoding, and sampling between real and generated videos can introduce shortcuts that inflate detection performance (Fang et al., 2026; Michels et al., 2026). We therefore audit these effects and evaluate on format-normalized videos.

Beyond benchmark design, prior work has explored forensic cues that expose artifacts not readily captured by semantic appearance. Frequency-aware representations have shown robustness to compression (Qian et al., 2020); for generated video, SpecSem-Net combines spectral and semantic features (Wei et al., 2026), Corvi et al. (2025) identify discriminative mid-to-high frequency components after compression, and PwTF-DVD applies temporal Fourier analysis at the pixel level (Kim et al., 2025). Reconstruction-based approaches provide another source of forensic evidence: DIRE measures diffusion reconstruction error (Wang et al., 2023), while DIVID and ReConFuse extend

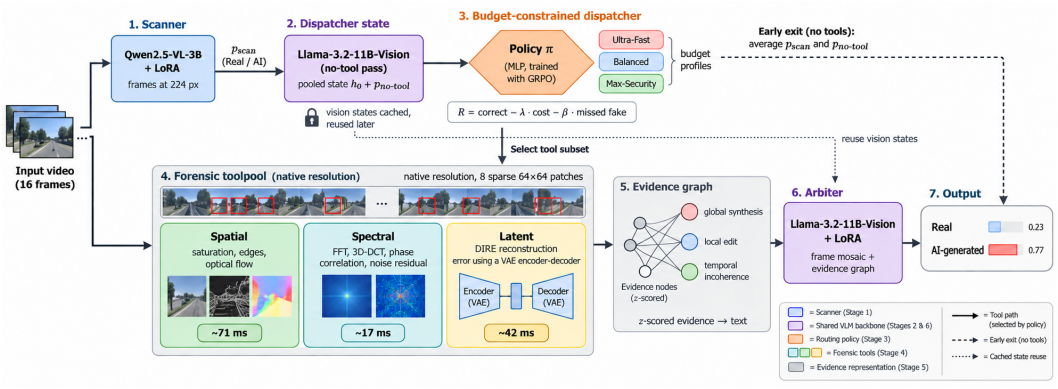



Figure 1: SAFER architecture. The frontline (scanner and no-tool arbiter pass) runs on every video and gives the dispatcher its state. The dispatcher either exits without tools (dashed arrow) or pays for a tool subset. Only the tools compute forensic cues; they run at native resolution on sparse patches, their outputs form an evidence graph, and the arbiter reads the graph with the video, reusing its cached vision states. The no-tool pass doubles as the same-reasoner baseline for measuring tool value.

reconstruction-based detection to video (Liu et al., 2024; Chen et al., 2026). These approaches demonstrate the utility of specialized forensic cues, but generally incorporate them as detector components rather than treating their acquisition as an explicit costed decision.

More recently, agentic approaches have made forensic evidence acquisition part of the detection process itself. LAVID allows a vision-language model to select and reason over forensic and knowledge tools (Liu et al., 2025a); multi-agent approaches distribute analysis across specialized texture, lighting, motion, and physics experts (Zou et al., 2026); Hermes retrieves forensic tools following perceptual analysis and organizes their outputs into an evidence reasoning graph (Li et al., 2026); and TUE-Detector trains a multimodal model to invoke thirteen forensic tools through supervised fine-tuning and GRPO (Wu et al., 2026). Collectively, these methods demonstrate how specialized evidence can be selected and integrated into multimodal reasoning. However, they leave a complementary question largely unexplored: whether the marginal predictive value obtained from forensic tool calls justifies the computational latency they incur. In particular, evaluating tool utility against a different detector can confound the value of the evidence with differences between the underlying reasoning models.

This question is closely related to adaptive computation, where additional computation is used only when its expected benefit justifies its cost. Early-exit networks terminate inference once a prediction is sufficiently confident (Teerapittayanon et al., 2016), while LLM cascades and routers direct queries toward the least costly model expected to solve them (Chen et al., 2023; Ong et al., 2024). Motivated by this principle, SAFER treats forensic analysis as cost-aware evidence acquisition: the agent decides whether additional evidence is worth collecting based on its measured latency, while a same-reasoner no-tool baseline allows its marginal value to be measured without changing the underlying reasoner.

## 3 Methodology

We first describe the task, dataset construction, and evaluation protocol, followed by the SAFER architecture and our methodology for measuring the marginal value of forensic evidence and the limits of tool routing.

### 3.1 Task, Data, and Metrics

We consider binary AI-generated video detection, where the goal is to determine whether a video is real or AI-generated. All models are trained on Chrono-66k and evaluated on its held-out test split. To assess generalization under domain shift, we further evaluate the trained models on four external

benchmarks without additional fine-tuning. For training and in-domain evaluation, we introduce Chrono-66k, a curated dataset of 66,646 videos, comprising 33,334 real clips from Kinetics-400 (Kay et al., 2017) and 33,312 AI-generated clips from VidProM using Pika and VideoCrafter2 (Wang & Yang, 2024; Chen et al., 2024a). Videos are assigned to training, validation, and test sets at the source-clip level using SHA-256 hashing and an 80/10/10 stratified split, yielding 53,334, 6,636, and 6,676 videos, respectively. The test split is isolated from model selection and dispatcher training.

Because differences in video format can introduce shortcuts unrelated to generative artifacts, we explicitly audit and normalize these properties. A gradient-boosted classifier trained only on file metadata—including duration, resolution, frame rate, codec, and bitrate—can distinguish the original real and generated sets, indicating the presence of format-related shortcuts. We therefore letterbox every video to $854\times480$, resample it to 24 fps, retain the first 10 seconds, remove audio, and re-encode it using H.264 (CRF 23, yuv420p).

To evaluate generalization beyond Chrono-66k, we use normalized versions of four external benchmarks: GenBuster (1,000 real and 2,150 generated videos), ViF-Bench (165 real and 2,995 generated videos across 19 architectures (JoeLeelyf, 2026)), FakePartsBench (2,000 real, 2,950 generated, and 2,700 partially edited videos), and GenVidBench (500 real and 1,000 generated videos, excluding Pika and VideoCrafter2). Together, these benchmarks contain 44 generator and editing categories unseen during training: 8 image/reference-conditioned, 9 local-editing, and 27 text-to-video categories (Appendix Table 16). We refer to evaluation on these unseen categories as *domain shift*. For the balanced in-domain test set, we report macro-F1, total errors, miss rate, and false-alarm rate. Under domain shift, we report ROC-AUC with 95% bootstrap confidence intervals to measure threshold-independent ranking performance, together with balanced accuracy at the default threshold of 0.5. Computational cost is measured as median end-to-end latency per video, and paired differences in classification errors are evaluated using McNemar's test.

### 3.2 The SAFER Architecture

SAFER processes 16 uniformly sampled frames per video through a chronologically routed, four-stage architecture (Figure 1).

**Stage 1: The Frontline Baseline.** A lightweight scanner, Qwen2.5-VL-3B-Instruct (Bai et al., 2025) with LoRA adapters (Hu et al., 2021), evaluates the frames at $224\times224$ resolution to yield a preliminary prediction $p_{\text{scan}}(y \mid V)$. Concurrently, the Arbiter a Llama-3.2-11B-Vision-Instruct model (Grattafiori et al., 2024) utilizing 4-bit LoRA reads a $2\times2$ frame mosaic at $560\times560$ resolution. This initial Arbiter pass executes *without* forensic tools, generating a baseline prediction $p_{\text{nt}}$ and a pooled state $h_0$, while caching its vision states to eliminate redundant computation later.

**Stage 2: Cost-Aware Dispatcher.** A two-layer MLP, $\pi_\phi(a \mid s)$, acts as the routing manager. It receives the concatenated Frontline state $s = [h_0, p_{\text{scan}}, p_{\text{nt}}]$ and selects one of six discrete actions: exit without tools, or execute a combination of tools (Spatial, Spectral, Latent, Spatial+Spectral, or All). Crucially, the Dispatcher does not see the forensic measurements; it predicts solely whether the potential accuracy gain justifies the latency cost. We optimize this router using GRPO (Shao et al., 2024), applying a non-differentiable reward function:

$$R(V, a) = \mathbb{I}[\hat{y}_a = y] - \lambda \frac{C(a)}{\max_{a'} C(a')} - \beta\, \mathbb{I}[y \neq \text{Real} \wedge \hat{y}_a = \text{Real}] \tag{1}$$

where $C(a)$ is the empirically measured latency of action $a$. We train three Dispatcher profiles: Ultra-Fast ($\lambda$=1, $\beta$=0, Latent disabled), Balanced ($\lambda$=0.2, $\beta$=0.5), and Max-Security ($\lambda$=0, $\beta$=2).

**Stage 3: Forensic Toolpool.** If the Dispatcher authorizes tools, they are executed on native-resolution frames (capped at 1024 pixels) to preserve high-frequency artifacts. The pool comprises 23 features across three families. The **Spatial** group (9 features, 71 ms median) assesses pixel-level inconsistencies (e.g., optical flow, edge sharpness). The **Spectral** group (9 features, 17 ms median) evaluates frequency artifacts (e.g., 3D-DCT energy, phase correlation). The **Latent** group (5 features, 42 ms median) computes reconstruction errors using a Stable Video Diffusion VAE (Blattmann et al., 2023), following the DIRE methodology (Wang et al., 2023).

Table 1: Held-out Chrono-66k test split, Real vs. AI-Generated ($n$=6,676). Miss and FA (false alarm) in %. Latency is median end-to-end time per video (batch size 4, resident models). Static runs the arbiter once with all tools and no dispatcher-state pass. The default is the Ultra-Fast and Balanced dispatcher, which exits without tools on every video.

| Model | Macro-F1 | Errors | Miss | FA | Tool cost (ms) | p50 (ms) |
|---|---|---|---|---|---|---|
| Toolpool-GBDT (tools only, no VLM) | 96.39 | 241 | 3.57 | 3.65 | 125.9 | 235.9 |
| SAFER-Scanner | 99.63 | 25 | 0.39 | 0.36 | – | 142.2 |
| Arbiter, no tools (part of the default) | 99.36 | 43 | 0.60 | 0.69 | – | – |
| SAFER-Static (arbiter with all tools) | 99.54 | 31 | 0.39 | 0.54 | 125.9 | 348.5 |
| SAFER low-latency mode ($\tau$=0.5) | 99.66 | 23 | 0.39 | 0.30 | 0.1 | **143.0** |
| SAFER default (learned, no-tool exit) | 99.70 | 20 | 0.27 | 0.33 | 0.0 | 239.1 |
| SAFER evidence mode, spectral | 99.75 | 17 | 0.24 | 0.27 | 17.4 | 282.2 |
| SAFER evidence mode, all tools | **99.79** | **14** | **0.15** | **0.27** | 125.9 | 409.4 |
| Oracle (cheapest correct action) | 99.81 | 13 | – | – | – | 239.1 |

**Stage 4: Evidence Graph & Final Arbiter Pass.** Tool outputs are $z$-scored against the training distribution and structured into an evidence graph mapping each feature to its mechanistic hypothesis (e.g., temporal incoherence). Tools skipped by the Dispatcher are explicitly marked as unexecuted. The Arbiter ingests this serialized graph alongside the video. Because the Arbiter's vision states were cached during Stage 1, this second forward pass is highly efficient. During training, tool groups are randomly dropped with $p = 0.3$ to ensure the Arbiter remains robust to partial or empty evidence graphs.

### 3.3 Quantifying Tool Value and Routing Limits

The fundamental advantage of SAFER is its capacity to quantify the exact marginal utility of forensic tools without confounding variables.

**Same-Reasoner Counterfactual.** We define *tool value* for a given subset $T$ as the Arbiter's AUC with $T$ minus the Arbiter's AUC with an empty graph. Because both predictions emerge from the exact same network weights, viewing the exact same frames, relying on the exact same cached vision states, this delta cleanly isolates the signal provided by the forensic evidence.

**The Oracle Upper Bound.** To establish the theoretical ceiling of any routing strategy, we construct an *Oracle*. By precomputing the outcomes and latencies of all six possible Dispatcher actions on every test video, the Oracle retrospectively selects the lowest-latency action that yields a correct prediction (at the 0.5 threshold). If an optimal, omniscient Oracle provides negligible improvement over the baseline, it mathematically demonstrates that no predictive router could meaningfully enhance performance via tool usage on that dataset.

## 4 Experiments

Our empirical evaluation is structured to answer four fundamental questions regarding the utility of forensic tools in VLM-based detection: (1) Do tools improve performance on familiar (in-domain) data, and at what computational cost? (2) Does tool utility transfer to unseen generative architectures? (3) Can a dynamic router predict when tools will actually help? (4) If tools fail to resolve domain shift, what mechanism actually fixes the remaining errors?

### 4.1 In-Domain Efficacy versus Computational Cost

We first establish whether forensic features contain valid signals. As shown in Table 1, they do: a simple GBDT trained exclusively on the 23 tool features achieves 96.39% macro-F1 without any multimodal VLM assistance. When integrated into the Arbiter, all tools reduce baseline errors from 43 to 31 (McNemar $p = 0.03$).

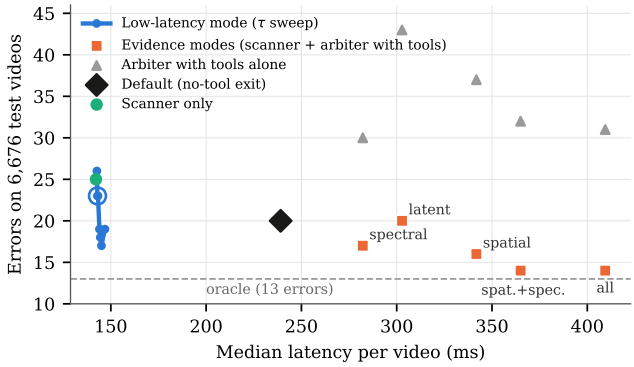



Figure 2: Errors against median latency on the in-domain test split. The arbiter with tools alone (grey) makes more errors than the frontline without tools (black); averaged with the scanner (orange), tools remove up to 6 more errors at 1.2 to 1.7× the default latency. The low-latency curve sweeps $\tau$; the circled point is the $\tau$ chosen on validation.

Table 2: Zero-shot AUC (95% bootstrap CI) on normalized external benchmarks. SAFER rows come from the per-action run. †Excludes Pika and VideoCrafter2 (training generators); the metadata and zero-shot rows were not run on this subset. With VideoCrafter2 included, the normalized metadata classifier reaches 0.98 on GenVidBench (Appendix D).

| Detector | GenBuster | ViF-Bench | FakeParts | GenVidBench† |
|---|---|---|---|---|
| Metadata-GBDT, normalized | 0.18 | 0.48 | 0.50 | – |
| Qwen2.5-VL-3B, zero-shot | 0.69 (.67–.71) | 0.56 (.52–.61) | 0.64 (.63–.66) | – |
| SAFER-Scanner | **0.867** (.85–.88) | 0.638 (.60–.68) | 0.714 (.70–.73) | 0.830 (.81–.85) |
| Arbiter, no tools | 0.808 (.79–.82) | 0.621 (.58–.66) | 0.692 (.68–.70) | 0.802 (.78–.82) |
| SAFER-Static (arbiter with all tools) | 0.735 (.72–.75) | 0.616 (.57–.66) | 0.690 (.68–.70) | 0.788 (.76–.81) |
| SAFER evidence mode, all tools | 0.800 (.79–.82) | 0.645 (.60–.69) | **0.725** (.71–.74) | 0.821 (.80–.84) |
| SAFER default | 0.853 (.84–.87) | **0.646** (.61–.69) | **0.725** (.71–.74) | **0.831** (.81–.85) |

However, this signal is overwhelmingly redundant when paired with a highly capable VLM frontline. The zero-tool SAFER default achieves 99.70% macro-F1 (20 errors) at 239 ms. Forcing the system to execute all tools (evidence mode) pushes macro-F1 to 99.79% by fixing just 6 remaining errors out of 6,676 videos, but inflates end-to-end latency by 1.71×. This equates to approximately 190 seconds of raw compute spent per single error avoided. Crucially, these results are robust against shortcut learning: while container metadata alone perfectly separates classes on raw files (Fang et al., 2026), our format normalization degrades metadata-only AUC to 0.80, while SAFER retains an AUC of 0.999 (see Appendix Table 15).

**Takeaway:** In-domain, tools offer marginal accuracy gains at severe latency premiums. Their deployment is only justified in settings where a fraction of a percent in accuracy supersedes throughput constraints.

### 4.2 Generalization to Unseen Domains

We next evaluate whether tools provide critical evidence when the VLM encounters 44 entirely unseen generator and editing categories (JoeLeelyf, 2026). Table 2 details the zero-shot AUC under this domain shift. Crucially, the normalized metadata classifier performs near chance on ViF-Bench and FakePartsBench, and exhibits an inverted AUC (0.18) on GenBuster. This confirms that file-level differences point opposite to the training data and cannot artificially inflate SAFER's scores.

Our Chrono-66k training transfers effectively to new domains: the scanner gains 0.07 to 0.18 AUC over the identical backbone used in a zero-shot capacity. Consequently, the SAFER default (which skips tools) achieves the highest AUC on both GenVidBench and ViF-Bench the latter being the most challenging dataset, featuring 2025-era generators like Sora-2, Kling, and Wan2.2 while tying

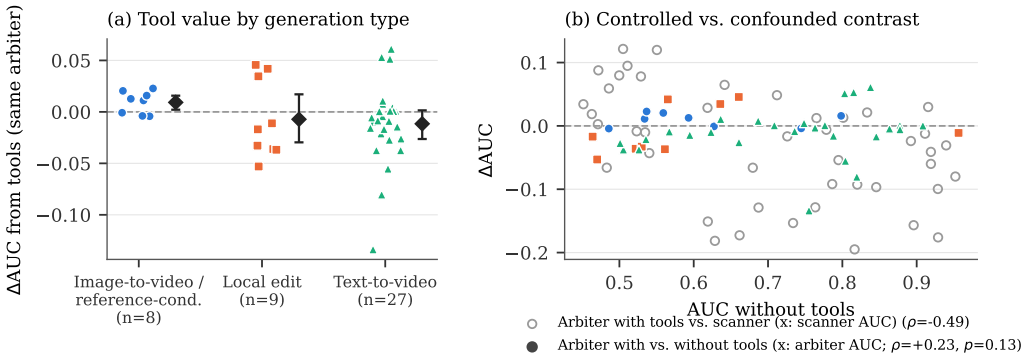



Figure 3: Tool value on 44 unseen categories. (a) AUC of the arbiter with all tools minus the same arbiter without tools, by generation process, with the mean and 95% bootstrap interval. (b) The controlled contrast (color) shows no detectable relation to how well the arbiter already does. The uncontrolled contrast (grey: arbiter with tools minus the scanner, against scanner AUC) produces a strong negative correlation that comes from comparing two different models.

Table 3: Dispatchers and routers on the in-domain test split ($n$=6,676). Tool groups is the mean number of tool groups run per video. The supervised router's $\lambda$ is selected on the validation outcome table; the confidence gate uses $\lambda$=0.05, the smallest weight used for the dispatchers.

| Router | Macro-F1 | Errors | p50 (ms) | Tool groups |
|---|---|---|---|---|
| GRPO dispatcher, $\lambda \in [0.05, 1]$ (SAFER default) | 99.70 | 20 | 239.1 | 0.00 |
| Supervised cost-aware router ($\lambda$=0.2) | 99.70 | 20 | 239.5 | 0.01 |
| Confidence gate, matched actions ($\lambda$=0.05) | 99.73 | 18 | 239.4 | 0.01 |
| GRPO without cost term ($\lambda$=0, $\beta$=0.5) | 99.63 | 25 | 254.4 | 0.54 |
| GRPO, Max-Security profile ($\lambda$=0, $\beta$=2) | 99.61 | 26 | 259.2 | 0.48 |

the all-tools evidence mode on FakePartsBench. Most notably, a direct comparison reveals that adding tools to the Arbiter strictly *lowers* its AUC across every single benchmark (comparing the "Arbiter, no tools" row against "Static").

Figure 3 dissects this performance drop by category. Across all 44 unseen categories, adding the full tool suite to the Arbiter changes the AUC by an average of $-0.007$, with a 95% bootstrap interval of $[-0.017, 0.003]$ that firmly contains zero. At the benchmark level, the tool-induced AUC shifts are universally negative: $-0.005$ on ViF-Bench, $-0.002$ on FakePartsBench, $-0.014$ on GenVidBench, and $-0.072$ on GenBuster.

The utility of forensic tools under shift is highly fragmented. Bootstrap intervals reveal a clear accuracy gain in only 6 categories, a clear loss in 14 (including all three GenBuster categories), and no statistically detectable change in the remaining 24. The few isolated gains stem from two inpainting methods and the AnyV2V style editor on FakePartsBench (gaining 0.035 to 0.046 AUC; ProPainter gaining 0.034), alongside three text-to-video sources (gaining 0.051 to 0.061). Image-to-video and reference-conditioned generators demonstrate the most consistent positive signal (5 of 8 positive, mean $+0.009$, interval $[0.002, 0.016]$). However, even an Oracle that perfectly identifies each video's generator and selects the optimal matched action raises the mean per-category AUC by at most 0.010. (Appendix E provides comprehensive equivalence tests and further validation).

On unseen generative architectures, we conclude that while forensic tools do alter individual classification decisions, they do not yield a net increase in detector accuracy. Even prior knowledge of the exact generator via Oracle routing cannot meaningfully change this reality.

Table 4: Change in balanced accuracy (points, relative to the SAFER default at the 0.5 threshold) on normalized external benchmarks. Zero-shot rows use routers fitted on Chrono-66k. Adaptation rows are fitted on labeled videos of one generator-disjoint half of each benchmark and tested on the other (mean ± std over 10 random splits). The routing oracle picks the cheapest correct action at the 0.5 threshold. Changes are computed from unrounded values, so they can differ by 0.1 from differences of the rounded absolute values in Appendix Table 12, which also gives AUC. †Pika and VideoCrafter2 excluded.

| | ViF-Bench | FakeParts | GenBuster | GenVidBench† |
|---|---|---|---|---|
| SAFER default (balanced accuracy, %) | 53.7 | 62.2 | 68.1 | 73.8 |
| Static (arbiter with all tools) | −0.4 | −0.5 | +0.7 | −2.6 |
| Supervised router, zero-shot | −0.1 | −0.3 | 0.0 | −0.3 |
| Routing oracle (cheapest correct action) | +1.6 | +3.8 | +7.2 | +5.1 |
| Router refit on target labels | +0.9 ±0.1 | +0.6 ±0.1 | +3.3 ±0.8 | −0.6 ±0.4 |
| SAFER adaptation mode (recalibration) | **+7.4** ±0.8 | **+4.3** ±0.9 | **+12.9** ±0.5 | −0.6 ±0.5 |

### 4.3 The Dispatcher's Learned Policy and the Routing Pitfall

If tools only help sporadically, can a routing mechanism learn to trigger them selectively? Our experiments reveal a fundamental flaw in this assumption: a lack of confidence in the frontline does *not* correlate with tool utility.

When we force tool usage on the least-confident third of predictions under domain shift, the tools break significantly more correct decisions than they fix (e.g., breaking 116 while fixing 69 on FakePartsBench). This exposes the "different-model pitfall." Previous literature often claims tools are helpful by comparing a weak model against a larger, tool-augmented model. However, when controlling for model capacity, we find that a weak frontline merely indicates that a *second model* might help, not that *forensic tools* will help.

Consequently, SAFER's GRPO-trained Dispatcher organically converges to the mathematically optimal policy: it exits without tools on nearly every video (Table 3). A supervised router reaches the exact same conclusion. Tool invocation is simply not worth the latency risk when the probability of introducing an error outpaces the probability of fixing one.

### 4.4 What fixes the errors that remain under shift?

An analysis of per-action outcomes decomposes the remaining errors into two distinct categories, exposing the structural limitations of forensic tools. The vast majority of errors are ubiquitous across every possible action: 2,659 out of 3,160 errors on ViF-Bench and 3,744 out of 7,650 errors on FakePartsBench occur regardless of tool usage at the default threshold. Nearly all of these represent generated videos misclassified as real. Because no combination of forensic evidence can resolve these ubiquitous errors, they explain the severe gap between high ROC-AUC and trailing balanced accuracy. Conversely, the second category comprises errors that specific tool actions could theoretically resolve. However, these are exceedingly rare: just 7 out of 6,676 in-domain videos, and between 95 and 389 videos per external benchmark. This scarcity mathematically bounds the maximum theoretical gain of any routing policy at the 0.5 threshold to a mere 1.6 to 7.2 points (Table 4), while practical routers fitted in-domain fail to exceed a 0.1-point gain.

The prevalence of the first error type indicates a grading-curve collapse rather than an evidence failure: the VLM still successfully ranks videos (preserving AUC), but the rigid 0.5 decision threshold, optimized for the training distribution no longer separates the classes under domain shift. To test this, we evaluate an adaptation mode that shifts the decision threshold using a disjoint labeled sample from the target domain. Specifically, we compute a new threshold using one half of the target generators and evaluate on the remaining half. This strict protocol ensures the test generators remain completely unseen (real videos are split uniformly at random, and test-half videos are strictly isolated during fitting).

This simple recalibration mechanism yields massive gains in balanced accuracy: +7.4 points on ViF-Bench, +4.3 points on FakePartsBench, and +12.9 points on GenBuster, decisively outperforming

the omniscient routing oracle on all three benchmarks. By contrast, utilizing the exact same labeled subset to refit the routing policy yields trivial gains of only 0.9, 0.6, and 3.3 points, respectively. (Neither approach proves effective on GenVidBench, where the split isolates only a single generator per half, though leveraging real target videos alone recovers partial gains; see Appendix D).

Ultimately, these dynamics reveal a core truth about generalization in deepfake detection. Under domain shift, what successfully transfers across generative architectures is the continuous scale of the model's scores, not the discrete rules for when to trigger forensic tools. When confronted with novel generators, allocating a minimal labeled sample to threshold calibration is profoundly more effective than utilizing that same sample to optimize tool routing.

### 4.5 Scope and limitations

Our empirical findings are bounded by specific architectural choices. All conclusions are derived from a single scanner/arbiter pair (Qwen2.5-VL-3B and Llama-3.2-11B-Vision) and a curated pool of 23 hand-crafted forensic features standardized into an evidence graph. Alternative fusion mechanisms or different VLM backbones may leverage tool evidence differently. Also, while format normalization isolates content differences from container shortcuts, the external benchmarks strictly test domain shift involving both novel generators and re-encoding. Future evaluation against external agentic architectures (e.g., D3 (Zheng et al., 2025), SPLIT (Hyun & Kim, 2026), and DeMamba (Chen et al., 2024b)) under this rigorous protocol would further contextualize SAFER's absolute capabilities.

## 5 Conclusion

In this work, we investigated a central question in agentic AI-generated video detection: when is acquiring forensic evidence worth its computational cost? To answer this question, we introduced SAFER, a cost-aware framework that measures the marginal value of forensic evidence against a controlled, same-reasoner no-tool baseline. Our experiments show that although forensic tools provide strong discriminative signals in isolation, they offer limited incremental value beyond the VLM frontline. Across 44 unseen generator and editing categories, the marginal value of full tool acquisition is effectively zero on average ($\Delta\text{AUC} = -0.007$), while frontline confidence does not reliably identify the cases in which additional evidence is beneficial.

Under domain shift, our results further indicate that the observed decision failures stem primarily from miscalibrated scoring rather than missing forensic evidence. On three external benchmarks, decision-threshold recalibration using a small labeled target sample improves balanced accuracy by 4.3–12.9 points, substantially outperforming adapted tool-routing strategies. Consistent with these findings, SAFER's cost-aware dispatcher learns to bypass forensic tools by default, matching or exceeding full-tool baselines while operating at approximately $0.60\times$ their latency.

These findings also motivate a methodological recommendation for tool-augmented detection. The value of forensic evidence should be measured against the same reasoning model operating without that evidence. Comparisons between a tool-equipped detector and a different baseline can confound the contribution of forensic evidence with differences in model capacity or reasoning, potentially overstating the value attributable to the tools themselves. Measuring both marginal evidence value and acquisition cost provides a more controlled basis for determining when forensic tool use is justified.

## AI Use Statement

We used generative AI assistants for literature search, for drafting and editing the text of this paper, and for writing parts of the analysis, plotting, and evaluation code. All experiments were run by the authors, all reported numbers come from our own runs, and we checked AI-assisted code and text before use. We take responsibility for the full content of this paper.

## Ethics Statement

Detectors of AI-generated video can be used to flag misinformation and can also be used to screen content at scale, where false alarms fall on real footage and real people. We report false-alarm rates and threshold behavior for this reason, and we show that accuracy drops on unseen generators, so the detector should not be used as sole evidence. Chrono-66k is built from public datasets under their licenses and is distributed with gated access.

## Reproducibility Statement

Section 3.1 and Appendix A describe the data, splits, and normalization; Section 3.2 and Appendix B give the architecture and all training settings; Appendix D gives the external protocol and hardware. The tool-value and routing analyses in Sections 4.2 to 4.4 use only the saved per-action outcome tables. Code and evaluation scripts are available at `https://anonymous.4open.science/r/safer-5C41`.

## Acknowledgement

## References

Shuai Bai, Ke qin Chen, Xue-Jing Liu, Jia-Lin Wang, Wenbin Ge, Sibo Song, Kai Dang, Peng Wang, Shijie Wang, Jun Tang, Humen Zhong, Yuanzhi Zhu, Mingkun Yang, Zhaohai Li, Jian-Qiang Wan, Pengfei Wang, Wei Ding, Zheren Fu, Yiheng Xu, Jiabo Ye, Xi Zhang, Tianbao Xie, Zesen Cheng, Hang Zhang, Zhibo Yang, Haiyang Xu, and Junyang Lin. Qwen2.5-vl technical report. *ArXiv*, abs/2502.13923, 2025. URL `https://api.semanticscholar.org/CorpusID:276449796`.

Andreas Blattmann, Tim Dockhorn, Sumith Kulal, Daniel Mendelevitch, Maciej Kilian, Dominik Lorenz, Yam Levi, Zion English, Vikram Voleti, Adam Letts, et al. Stable video diffusion: Scaling latent video diffusion models to large datasets. *arXiv preprint arXiv:2311.15127*, 2023.

Haoxin Chen, Yong Zhang, Xiaodong Cun, Menghan Xia, Xintao Wang, Chao Weng, and Ying Shan. Videocrafter2: Overcoming data limitations for high-quality video diffusion models, 2024a.

Haoxing Chen, Yan Hong, Zizheng Huang, Zhuoer Xu, Zhangxuan Gu, Yaohui Li, Jun Lan, Huijia Zhu, Jianfu Zhang, Weiqiang Wang, and Huaxiong Li. Demamba: Ai-generated video detection on million-scale genvideo benchmark, 2024b. URL `https://arxiv.org/abs/2405.19707`.

Lingjiao Chen, Matei Zaharia, and James Zou. FrugalGPT: How to use large language models while reducing cost and improving performance. *arXiv preprint arXiv:2305.05176*, 2023.

Xiaojing Chen, Xinyu Lu, Changtao Miao, and Yunfeng Diao. ReConFuse: Reconstruction-error guided semantic fusion for AI-generated video detection. *arXiv preprint arXiv:2606.04706*, 2026.

Riccardo Corvi, Davide Cozzolino, Ekta Prashnani, Shalini De Mello, Koki Nagano, and Luisa Verdoliva. Seeing what matters: Generalizable AI-generated video detection with forensic-oriented augmentation. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2025.

Brian Dolhansky, Russ Howes, Ben Pflaum, Nicole Baram, and Cristian Canton Ferrer. The deepfake detection challenge (dfdc) preview dataset, 2019. URL `https://arxiv.org/abs/1910.08854`.

Ziyuan Fang, Tianyi Wei, Guanjie Wang, Weiming Zhang, Nenghai Yu, and Wenbo Zhou. Real data lies: Unveiling and closing the quality shortcut in generalizable AI-generated video detection. In *International Conference on Machine Learning (ICML)*, 2026.

Aaron Grattafiori et al. The Llama 3 herd of models. *arXiv preprint arXiv:2407.21783*, 2024.

Edward J Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. Lora: Low-rank adaptation of large language models. *arXiv preprint arXiv:2106.09685*, 2021.

Jongyeop Hyun and Hyounghun Kim. SPLIT: Training-free AI-generated and partially edited video detection via spatial patch-level incoherence and temporal roughness. *arXiv preprint arXiv:2607.02886*, 2026.

Liming Jiang, Ren Li, Wayne Wu, Chen Qian, and Chen Change Loy. Deeperforensics-1.0: A large-scale dataset for real-world face forgery detection, 2020. URL https://arxiv.org/abs/2001.03024.

JoeLeelyf. ViF-Bench. Hugging Face dataset https://huggingface.co/datasets/JoeLeelyf/ViF-Bench, revision 67e11331, 2026.

Will Kay, Joao Carreira, Karen Simonyan, Brian Zhang, Chloe Hillier, Sudheendra Vijayanarasimhan, Fabio Viola, Tim Green, Trevor Back, Paul Natsev, Mustafa Suleyman, and Andrew Zisserman. The kinetics human action video dataset. *CoRR*, abs/1705.06950, 2017. URL http://arxiv.org/abs/1705.06950.

Kim et al. Beyond spatial frequency: Pixel-wise temporal frequency-based deepfake video detection. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, 2025.

Shuaibo Li, Pengfei HAO, Hongtao Wu, Jianfeng Dong, Ping Li, Xiaohong Liu, and Lei Zhu. Hermes: An evidence-driven agentic framework for trustworthy and explainable AI-generated video detection. In *Forty-third International Conference on Machine Learning*, 2026. URL https://openreview.net/forum?id=mgtSv2NUik.

Qingyuan Liu, Pengyuan Shi, Yun-Yun Tsai, Chengzhi Mao, and Junfeng Yang. Turns out I'm not real: Towards robust detection of AI-generated videos. *arXiv preprint arXiv:2406.09601*, 2024.

Qingyuan Liu, Yun-Yun Tsai, Ruijian Zha, Victoria Li, Pengyuan Shi, Chengzhi Mao, and Junfeng Yang. Lavid: An agentic lvlm framework for diffusion-generated video detection, 2025a. URL https://arxiv.org/abs/2502.14994.

Ziyi Liu, Firas Gabetni, Awais Hussain Sani, Xi Wang, Soobash Daiboo, Gaëtan Brison, Gianni Franchi, and Vicky Kalogeiton. FakeParts: a new family of AI-generated DeepFakes. *arXiv preprint arXiv:2508.21052*, 2025b.

Joren Michels, Lode Jorissen, and Nick Michiels. Dataset biases and shortcut learning in motion-based AI-generated video detection. *arXiv preprint arXiv:2607.00948*, 2026.

Zhenliang Ni, Qiangyu Yan, Mouxiao Huang, Tianning Yuan, Yehui Tang, Hailin Hu, Xinghao Chen, and Yunhe Wang. GenVidBench: A 6-million benchmark for AI-generated video detection. In *Proceedings of the AAAI Conference on Artificial Intelligence*, 2026.

Isaac Ong, Amjad Almahairi, Vincent Wu, Wei-Lin Chiang, Tianhao Wu, Joseph E. Gonzalez, M. Waleed Kadous, and Ion Stoica. RouteLLM: Learning to route LLMs with preference data. *arXiv preprint arXiv:2406.18665*, 2024.

Yuyang Qian, Guojun Yin, Lu Sheng, Zixuan Chen, and Jing Shao. Thinking in frequency: Face forgery detection by mining frequency-aware clues, 2020. URL https://arxiv.org/abs/2007.09355.

Andreas Rössler, Davide Cozzolino, Luisa Verdoliva, Christian Riess, Justus Thies, and Matthias Nießner. Faceforensics++: Learning to detect manipulated facial images, 2019. URL https://arxiv.org/abs/1901.08971.

Zhihong Shao, Peiyi Wang, Qihao Zhu, Runxin Xu, Junxiao Song, Xiao Bi, Haowei Zhang, Mingchuan Zhang, YK Li, Yang Wu, et al. Deepseekmath: Pushing the limits of mathematical reasoning in open language models. *arXiv preprint arXiv:2402.03300*, 2024.

Surat Teerapittayanon, Bradley McDanel, and H. T. Kung. BranchyNet: Fast inference via early exiting from deep neural networks. In *International Conference on Pattern Recognition (ICPR)*, 2016.

Wenhao Wang and Yi Yang. Vidprom: A million-scale real prompt-gallery dataset for text-to-video diffusion models. *NuerIPS*, 2024. URL https://openreview.net/forum?id=pYNl76onJL.

Zhendong Wang, Jianmin Bao, Wengang Zhou, Weilun Wang, Hezhen Hu, Hong Chen, and Houqiang Li. Dire for diffusion-generated image detection. In *2023 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 22388–22398. IEEE, 2023.

Zixi Wei, Huixuan Zhang, and Xiaojun Wan. SpecSem-Net: Integrating spectral and semantic features for robust AI-generated video detection. *arXiv preprint arXiv:2605.17311*, 2026.

Haiquan Wen et al. BusterX: MLLM-powered AI-generated video forgery detection and explanation. *arXiv preprint arXiv:2505.12620*, 2025.

Yichen Wu, Haoxuan Qu, Yongxing Dai, Yan Bai, Yihang Lou, Yuqi Lin, Hossein Rahmani, and Jun Liu. TUE-Detector: A tool-using expert MLLM-based detector for AI-generated videos. *arXiv preprint arXiv:2608.30704*, 2026.

Chende Zheng, Ruiqi Suo, Chenhao Lin, Zhengyu Zhao, Le Yang, Shuai Liu, Minghui Yang, Cong Wang, and Chao Shen. D3: Training-free AI-generated video detection using second-order features. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, 2025.

Xuechao Zou, Shun Zhang, Kai Li, Yi Zhou, Xinyu Sun, Yuhui Chen, Zhe Wu, Congyan Lang, and Junliang Xing. Multi-agent forensic reasoning for generalizable deepfake video detection, 2026. URL https://arxiv.org/abs/2608.06865.

# APPENDIX

## A DATASET DETAILS

Table 5: Composition of Chrono-66k. Splits are grouped by source clip (SHA-256) and stratified by class at 80/10/10: 53,334 training, 6,636 validation, and 6,676 test videos.

| Class | Source | Videos | Test videos |
|---|---|---|---|
| Real | Kinetics-400 | 33,334 | 3,345 |
| AI-Generated | VidProM (Pika, VideoCrafter2) | 33,312 | 3,331 |
| Total | | 66,646 | 6,676 |

## B IMPLEMENTATION DETAILS

**Dispatcher.** Each dispatcher is a two-layer MLP with 256 hidden units, trained on the 6,000-video validation outcome table for 1,500 GRPO iterations with 256 videos per iteration, 16 sampled actions per video, two inner epochs, learning rate $3\times10^{-4}$, PPO clip range 0.2, KL coefficient 0.02 to the initial policy, and entropy bonus 0.01. The checkpoint with the best greedy reward on a held-out selection split is kept. All runs use seed 42. Training used four NVIDIA H200 NVL GPUs (141 GB each).

## C ADDITIONAL IN-DOMAIN RESULTS

The Llama pass that produces the dispatcher state is the largest single cost (131.8 ms). Computing the routing state from the scanner alone would remove most of the gap between no-tool exit (244.6 ms) and the scanner (about 113 ms for decoding and scanning).

Table 6: Training configuration of the two VLM components.

| | Scanner (Qwen2.5-VL-3B) | Arbiter (Llama-3.2-11B-Vision) |
|---|---|---|
| Precision | bf16 | 4-bit, 8-bit AdamW |
| Input | 16 frames, $224^2$ | 4-frame mosaic, $560^2$, evidence text |
| Trainable parameters | 59.9M | 104.9M |
| Epochs / optimizer steps | 2 / 1,666 | 1 / 833 |
| Effective batch size | 64 | 64 |
| LoRA rank / $\alpha$ / dropout | 32 / 64 / 0.05 | 32 / 64 / 0.05 |
| Learning rate (adapter / head) | $10^{-4}$ / $5\times10^{-4}$ | $10^{-4}$ / $5\times10^{-4}$ |
| Label smoothing | 0.05 | 0.05 |
| Tool-group dropout | – | 0.3 |
| Best validation macro-F1 | 99.64 | 99.65 |
| Training time | 2.1 h | 1.6 h |
| Peak memory | 9.8 GiB | 29.3 GiB |

Table 7: Per-class precision and recall (%) on the in-domain test split (3,345 Real, 3,331 AI-Generated).

| Model | Real P | Real R | Gen. P | Gen. R |
|---|---|---|---|---|
| SAFER-Scanner | 99.61 | 99.64 | 99.64 | 99.61 |
| SAFER-Static | 99.61 | 99.46 | 99.46 | 99.61 |
| SAFER default | 99.73 | 99.67 | 99.67 | 99.73 |
| SAFER, Max-Security profile | 99.58 | 99.64 | 99.64 | 99.58 |

The learned dispatchers route as follows on the in-domain test split: Ultra-Fast and Balanced send 100% of videos to no-tool exit; Max-Security sends 52.2% to no-tool exit, 28.9% to latent, 17.4% to spectral, 1.4% to spatial, and 0.1% to spatial+spectral.

## D External Benchmark Details

No tool group raises benchmark-level AUC over the no-tool arbiter by more than 0.004. Tool actions return the arbiter alone; the matched comparison with the scanner average is the evidence mode in Table 2.

**Adaptation-mode stability.** Over 10 random generator-disjoint splits, the share of the routing oracle's balanced-accuracy gain recovered by routing adaptation is $57 \pm 9\%$ on ViF-Bench (range 43–67%), $46 \pm 12\%$ on GenBuster (24–55%), $16 \pm 2\%$ on FakePartsBench (13–18%), and $-11 \pm 7\%$ on GenVidBench, where each half contains a single generator.

**Recalibration from real videos only.** Setting the threshold at 95% specificity on the labeled real videos of one half, without any labeled fakes, gains $+2.7 \pm 0.2$ points on ViF-Bench, $+0.6 \pm 0.0$ on FakePartsBench, $+5.3 \pm 0.3$ on GenBuster, and $-4.7 \pm 0.4$ on GenVidBench (10 splits). Labeled fakes from other target generators supply the rest of the gain in Table 4.

**Per-action outcome run.** The SAFER rows of Table 2, Figure 3, and Tables 4, 10, and 16 come from a second pass that records every action's prediction and latency per video, with the same models and normalized files. No-tool-exit and static scores agree with the first run (share of videos differing by more than 0.02: at most 0.9% and 0.03%), and all AUCs agree to the third decimal. Scanner scores differ by more than 0.02 on 1.2–3.9% of videos, mostly where the scanner is unsure. Running the same 30 ViF-Bench videos twice in fresh processes changes scanner scores by up to 0.011, and the fresh runs agree better with the per-action run than with the first run, so we attribute the differences to non-deterministic bf16 kernels under the shared-GPU load of the first run. Static and all-tools cached predictions are identical in every benchmark, so vision-state reuse is exact. Table 14 gives the AUC of every action on each ViF-Bench generator.

Table 8: Mean latency components (ms) per video in the in-domain harness, 1,000 held-out videos, resident models. Totals include patch proposal (1–2 ms).

| Route | Decode | Scanner | Dispatcher state | Tools | Arbiter | Total |
|---|---|---|---|---|---|---|
| No-tool exit | 48.8 | 64.0 | 131.8 | 0.0 | 0.0 | 244.6 |
| Spectral | 48.8 | 64.0 | 131.8 | 15.0 | 48.4 | 309.2 |
| Latent | 48.8 | 64.0 | 131.8 | 41.1 | 44.3 | 331.1 |
| Spatial | 48.8 | 64.0 | 131.8 | 59.0 | 47.7 | 353.0 |
| All tools | 48.8 | 64.0 | 131.8 | 112.1 | 54.6 | 412.5 |

Table 9: How often the routing decision can change correctness on the in-domain outcome tables. "Tools fix exit": no-tool exit is wrong and some tool route is right; "tools break exit": no-tool exit is right and some tool route is wrong.

| Outcome table | Videos | All correct | All wrong | Tools fix exit | Tools break exit |
|---|---|---|---|---|---|
| Validation (GRPO) | 6,000 | 5,962 | 3 | 8 | 27 |
| Test | 6,676 | 6,625 | 13 | 7 | 31 |

**Container audit per benchmark.** On raw files the metadata classifier reaches AUC 1.00 on GenVidBench, 0.85 on FakePartsBench, 0.54 on ViF-Bench, and 0.22 on GenBuster. After normalization these become 0.98, 0.50, 0.48, and 0.18. An AUC well below 0.5 still means the classes differ in container properties, only in the opposite direction from our training data, so GenBuster also keeps a residual difference after normalization.

**Overlapping generators.** On the 1,000 GenVidBench videos from Pika and VideoCrafter2, both training generators, the scanner flags 96.7% as fake and no-tool exit 98.3%. These are excluded from Tables 2 and 4.

**Latency in the external runs.** The first external run, which produced the metadata and zero-shot rows of Table 2 and the operating-point table, used one NVIDIA H200 NVL shared by six worker processes at batch size 1, with the GPU fully loaded. There, median latency was 0.66 s for the scanner, 1.12 s for static tool use, and 1.37 s for no-tool exit, so no-tool exit appeared slower than static. The per-action run used four H200 NVL GPUs with one worker each and nothing else on them (batch size 1, 17,460 videos in about 75 minutes). There the ordering matches the in-domain one: scanner 99–104 ms, no-tool exit 232–238 ms, static 282–288 ms. We therefore attribute the reversal in the first run to contention on the shared GPU.

Table 16: Per-category AUC on unseen generators and editing methods (against all real videos of the benchmark), from the per-action run, grouped by generation process and sorted by tool value. No-tool is the arbiter with an empty evidence graph; $\Delta$ = arbiter with all tools minus no-tool.

| Benchmark | Category | Type | $n$ | Scanner | No-tool | All tools | $\Delta$ |
|---|---|---|---|---|---|---|---|
| FakeParts | TI2V/WAN | I2V/ref. | 93 | 0.801 | 0.798 | 0.814 | +0.016 |
| FakeParts | TI2V/Open-Sora-768px | I2V/ref. | 300 | 0.795 | 0.745 | 0.741 | -0.004 |
| ViF-Bench | Wan2.2-TI2V-5B-I | I2V/ref. | 165 | 0.472 | 0.537 | 0.559 | +0.023 |
| ViF-Bench | SkyReels-V2-I2V-14B-540P | I2V/ref. | 164 | 0.500 | 0.559 | 0.579 | +0.020 |
| ViF-Bench | LTX-Video-13B-I | I2V/ref. | 165 | 0.511 | 0.593 | 0.606 | +0.013 |
| ViF-Bench | HunyuanVideo-I2V | I2V/ref. | 165 | 0.486 | 0.534 | 0.545 | +0.011 |
| ViF-Bench | Wan2.1-VACE-1.3B-T | I2V/ref. | 165 | 0.505 | 0.627 | 0.626 | -0.001 |
| ViF-Bench | Wan2.2-I2V-14B | I2V/ref. | 165 | 0.463 | 0.486 | 0.482 | -0.004 |
| FakeParts | Change_of_style/AnyV2V | Edit | 300 | 0.642 | 0.661 | 0.706 | +0.046 |
| FakeParts | Inpainting/DiffuEraser | Edit | 300 | 0.529 | 0.565 | 0.607 | +0.042 |
| FakeParts | Inpainting/Propainter | Edit | 300 | 0.550 | 0.636 | 0.670 | +0.035 |
| FakeParts | Change_of_style/RAVE | Edit | 300 | 0.915 | 0.956 | 0.945 | -0.011 |
| FakeParts | Faceswap/Insightface | Edit | 300 | 0.629 | 0.464 | 0.447 | -0.017 |
| FakeParts | Outpainting/akira2 | Edit | 300 | 0.540 | 0.530 | 0.497 | -0.033 |
| FakeParts | Inpainting/ROVI/Inpainting_MP4 | Edit | 300 | 0.452 | 0.522 | 0.486 | -0.036 |

| Benchmark | Category | Type | $n$ | Scanner | No-tool | All tools | $\Delta$ |
|---|---|---|---|---|---|---|---|
| FakeParts | Outpainting/akira | Edit | 300 | 0.535 | 0.561 | 0.524 | -0.037 |
| FakeParts | Interpolation/Framer | Edit | 300 | 0.483 | 0.470 | 0.417 | -0.053 |
| FakeParts | T2V/sora-720p | T2V | 72 | 0.911 | 0.838 | 0.899 | +0.061 |
| FakeParts | T2V/Wan_21 | T2V | 129 | 0.892 | 0.815 | 0.868 | +0.053 |
| FakeParts | T2V/sora-1080p | T2V | 128 | 0.833 | 0.803 | 0.855 | +0.051 |
| FakeParts | T2V/HunyuanVideo | T2V | 300 | 0.929 | 0.748 | 0.753 | +0.004 |
| FakeParts | T2V/Mochi1 | T2V | 300 | 0.765 | 0.776 | 0.777 | +0.001 |
| FakeParts | T2V/LTXVideo | T2V | 300 | 0.919 | 0.877 | 0.878 | +0.001 |
| FakeParts | T2V/Open-Sora | T2V | 300 | 0.940 | 0.908 | 0.909 | +0.001 |
| FakeParts | T2V/CogVideoX | T2V | 300 | 0.919 | 0.863 | 0.859 | -0.004 |
| FakeParts | T2V/allegroai | T2V | 128 | 0.952 | 0.878 | 0.872 | -0.005 |
| FakeParts | T2V/veo2-720p | T2V | 300 | 0.820 | 0.736 | 0.727 | -0.008 |
| FakeParts | Extrapolation/Cosmos-Predict2 | T2V | 300 | 0.471 | 0.501 | 0.474 | -0.027 |
| GenBuster | id/fake | T2V | 1000 | 0.846 | 0.804 | 0.749 | -0.055 |
| GenBuster | ood/fake | T2V | 1000 | 0.896 | 0.819 | 0.739 | -0.080 |
| GenBuster | wild/fake | T2V | 150 | 0.816 | 0.755 | 0.621 | -0.134 |
| ViF-Bench | Wan2.2-TI2V-5B-T | T2V | 165 | 0.619 | 0.636 | 0.647 | +0.011 |
| ViF-Bench | Wan2.2-T2V-14B | T2V | 165 | 0.786 | 0.687 | 0.695 | +0.008 |
| ViF-Bench | HunyuanVideo | T2V | 164 | 0.726 | 0.708 | 0.709 | +0.001 |
| ViF-Bench | SkyReels-V2 | T2V | 165 | 0.712 | 0.763 | 0.760 | -0.003 |
| ViF-Bench | sora-2 | T2V | 150 | 0.687 | 0.567 | 0.558 | -0.009 |
| ViF-Bench | CogVideoX1.5-5B-T | T2V | 165 | 0.680 | 0.623 | 0.613 | -0.010 |
| ViF-Bench | kling-v1 | T2V | 141 | 0.734 | 0.595 | 0.580 | -0.014 |
| ViF-Bench | LTX-Video-13B-T | T2V | 165 | 0.778 | 0.788 | 0.772 | -0.016 |
| ViF-Bench | pixverse-v4-5 | T2V | 152 | 0.929 | 0.846 | 0.829 | -0.017 |
| ViF-Bench | gen4-turbo | T2V | 121 | 0.523 | 0.536 | 0.515 | -0.021 |
| ViF-Bench | hailuo | T2V | 137 | 0.764 | 0.661 | 0.635 | -0.026 |
| ViF-Bench | Wan2.1-T2V-1.3B | T2V | 165 | 0.662 | 0.526 | 0.489 | -0.037 |
| ViF-Bench | pika-v2 | T2V | 151 | 0.619 | 0.505 | 0.468 | -0.037 |

## E Additional Statistical Analyses

**Equivalence of tool value to zero.** A two one-sided test places the mean category-level tool value ($-0.007$) within $\pm 0.02$ AUC of zero ($p = 0.009$). The margin was chosen after the analysis; the smallest margin that passes at $\alpha = 0.05$ is $\pm 0.016$. The 44 categories share each benchmark's real videos, so they are not fully independent, and we treat the bootstrap interval as the primary result. With 44 per-category intervals, about two would exclude zero by chance.

**Groups and correlations.** Image-to-video and reference-conditioned categories gain more from tools than text-to-video categories (Mann–Whitney $p = 0.015$); the groups were formed after inspection, so we read this as description. In a post-hoc check without the three GenBuster categories, the same-reasoner tool value is positively related to the arbiter's AUC ($\rho = +0.40$, $p = 0.01$), so tools, if anything, help where the arbiter is already good. No single tool group raises the arbiter's benchmark-level AUC by more than 0.004 (Table 10).

**Matched comparison under shift.** The default beats the all-tools evidence mode by 0.053 AUC on GenBuster (95% interval $[0.046, 0.060]$) and 0.010 on GenVidBench ($[0.006, 0.015]$), and ties it on ViF-Bench ($+0.001$, $[-0.009, 0.010]$) and FakePartsBench ($0.000$, $[-0.003, 0.003]$), at a median latency of 232 to 238 ms against 391 to 398 ms. The scanner alone runs at $0.43\times$ the default's latency, has higher AUC on GenBuster, and is within 0.011 elsewhere. Static, which is not a router, gains 0.7 points of balanced accuracy on GenBuster while losing 0.118 AUC there.

**Recalibrated thresholds.** With each score at its own in-sample Youden threshold, tools break more decisions than they fix on FakePartsBench, GenBuster, and GenVidBench, and fix more than they break on ViF-Bench (264 against 49). That count reflects ViF-Bench's 95% share of fakes: at these thresholds balanced accuracy is 62.2% for the default and 62.4% for the evidence mode (Table 13).

Table 10: AUC of the scanner, the arbiter without tools, and each dispatcher action on the normalized external benchmarks (per-action run). The no-tool exit is the mean of the scanner and the no-tool arbiter. †Pika and VideoCrafter2 excluded.

| Benchmark | Scanner | No-tool | No-tool exit | Spatial | Spectral | Latent | Spat.+Spec. | All |
|---|---|---|---|---|---|---|---|---|
| GenBuster | 0.867 | 0.808 | 0.853 | 0.771 | 0.741 | 0.786 | 0.732 | 0.735 |
| ViF-Bench | 0.638 | 0.621 | 0.646 | 0.624 | 0.611 | 0.620 | 0.612 | 0.616 |
| FakePartsBench | 0.714 | 0.692 | 0.725 | 0.690 | 0.686 | 0.685 | 0.691 | 0.690 |
| GenVidBench† | 0.830 | 0.802 | 0.831 | 0.796 | 0.794 | 0.804 | 0.787 | 0.788 |

Table 11: Operating-point metrics of the SAFER default and SAFER-Scanner on normalized external benchmarks (%): TPR at 5% FPR, specificity on real videos, and fake recall at the 0.5 threshold. Rows other than GenVidBench come from the first external run; all AUC and routing conclusions in the main text use the per-action run.

| | SAFER default | | | SAFER-Scanner | | |
|---|---|---|---|---|---|---|
| Benchmark | TPR@5% | Spec. | Recall | TPR@5% | Spec. | Recall |
| GenBuster | 52.0 | 97.1 | 39.1 | 51.3 | 97.4 | 33.4 |
| ViF-Bench | 17.4 | 99.4 | 8.1 | 18.9 | 98.8 | 9.3 |
| FakePartsBench | 30.7 | 96.2 | 28.2 | 34.2 | 96.6 | 28.7 |
| GenVidBench† | 44.8 | 88.8 | 58.9 | 44.2 | 90.2 | 56.8 |

**Cost break-even and routing details.** The dispatcher's tool actions return the arbiter alone, while the no-tool exit averages it with the scanner; part of the loss of the $\lambda$=0 dispatchers (25 and 26 errors against 20) comes from dropping that average. With matched actions, the all-tools evidence mode changes the test decision on 8 of 6,676 videos (7 fixed, 1 broken), a net accuracy gain of 0.09%. Under Equation 1 the no-tool exit has cost 0 and the all-tools action the maximum, so their normalized cost gap is 1, and skipping beats always-on tools for any $\lambda$ above about 0.001 (0.0005 with the validation counts); the smallest weight we train with is 0.05. Under shift the net count is negative at the default threshold, so no latency weight makes the tools pay. The confidence gate (a supervised router over matched actions) uses $\lambda$=0.05, the smallest dispatcher weight; a test-set sweep over $\lambda$ from 0.002 to 0.01 reaches 16 in-domain errors at a mean latency of 242.5 ms.

## F Cue Statistics

Figure 4 shows, for two real and two generated videos, the noise-residual energy, the Fourier spectrum of the residual, the per-pixel temporal high-frequency energy, and the VAE reconstruction error, each on a shared color scale. Over 19 real and 20 generated native videos, the spectral cues separate the classes best: generated videos have a flatter spatial spectrum (slope $-3.09$ vs. $-3.83$, AUC 0.78) and a higher frame high-frequency ratio (AUC 0.78). Residual strength (AUC 0.76) and residual flicker (AUC 0.66) follow, and the latent cue is weaker (AUC 0.65). Several residual maps also contain $8\times8$ block patterns from compression, and the two classes come at different resolutions, which is why all detector results use normalized video.

Table 17 lists the per-video cue statistics behind Figure 4 (native format, 16 consecutive frames, $256\times256$ center crop). AUC is computed with AI-generated as the positive class; values below 0.5 mean the cue is lower for generated videos. Figure 5 shows the class-averaged spectra and Figure 6 the per-video distributions. The generated videos in this sample come at $512\times320$ (12 videos) and $1088\times640$ (8), while real videos range from $320\times240$ to $1280\times720$, so these statistics mix generation traces with encoding differences.

## G Forensic Features

**Spatial (9):** saturation mean, standard deviation, and entropy; standard deviation and maximum of inter-frame luminance jumps; global Laplacian variance and its patch-to-global ratio; optical-flow inconsistency; patch-to-context color gap. **Spectral (9):** global FFT high-frequency ratio;

Table 12: Routing on normalized external benchmarks, absolute values: balanced accuracy (%) at the 0.5 threshold / AUC. Adaptation rows are means over 10 generator-disjoint splits (recalibration leaves AUC unchanged); "GRPO, target-domain" is GRPO trained on the same labeled halves ($\lambda$=0.2, one split). Median latency: scanner 99–104 ms, no-tool exit 232–238 ms, static 282–288 ms, evidence mode 391–398 ms. †Pika and VideoCrafter2 excluded.

| Router | ViF-Bench | FakeParts | GenBuster | GenVidBench† |
|---|---|---|---|---|
| Scanner only | 54.1 / .638 | 62.7 / .714 | 65.4 / .867 | 73.5 / .830 |
| SAFER default (no-tool exit) | 53.7 / .646 | 62.2 / .725 | 68.1 / .853 | 73.8 / .831 |
| Static (all tools) | 53.3 / .616 | 61.7 / .690 | 68.8 / .735 | 71.2 / .788 |
| GRPO, Max-Security profile | 53.8 / .627 | 62.2 / .705 | 68.0 / .806 | 72.5 / .814 |
| Supervised router, zero-shot | 53.6 / .643 | 62.0 / .721 | 68.1 / .848 | 73.6 / .825 |
| SAFER low-latency mode, in-domain $\tau$ | 53.6 / .622 | 61.7 / .696 | 65.9 / .851 | 72.7 / .824 |
| GRPO, target-domain | 53.7 / .646 | 62.2 / .725 | 68.1 / .853 | – |
| SAFER evidence mode, all tools | 53.1 / .645 | 61.9 / .725 | 67.0 / .800 | 73.4 / .821 |
| SAFER adaptation mode: routing | 54.6 / .642 | 62.8 / .722 | 71.4 / .849 | 73.3 / .825 |
| SAFER adaptation mode: recalibration | 61.1 / .646 | 66.5 / .725 | 81.0 / .853 | 73.2 / .831 |
| Oracle (cheapest correct action) | 55.3 / – | 66.1 / – | 75.3 / – | 79.0 / – |

Table 13: Decisions changed by tools, relative to the SAFER default: decisions fixed / broken. "0.5": all-tools evidence mode at the default threshold (all changes fall in the least-confident third). "Recal.": each score at its own in-sample Youden threshold on the benchmark (a sensitivity check); balanced accuracy of the default / evidence mode at those thresholds is 62.2 / 62.4 (ViF-Bench), 67.4 / 67.5 (FakePartsBench), 81.2 / 77.0 (GenBuster), and 75.3 / 74.0 (GenVidBench). "Gate": supervised router over matched actions, fitted on the Chrono-66k validation table ($\lambda$=0.05), with the share of videos sent to tools.

| Test set | 0.5 | Recal. | Gate (tool share) |
|---|---|---|---|
| Chrono-66k test | 7 / 1 | – | 2 / 0 (0.8%) |
| ViF-Bench | 0 / 37 | 264 / 49 | 0 / 14 (6.5%) |
| FakePartsBench | 69 / 116 | 105 / 145 | 18 / 34 (10.6%) |
| GenBuster | 25 / 66 | 24 / 133 | 9 / 28 (17.0%) |
| GenVidBench† | 28 / 37 | 26 / 67 | 3 / 13 (18.7%) |

radial spectrum slope; patch FFT high-frequency mean and standard deviation; global and patch 3D-DCT high-frequency energy; global phase correlation; phase-shift spike; noise-residual coefficient of variation across patches. **Latent (5):** global DIRE; patch DIRE mean, minimum, and standard deviation; patch-to-global DIRE ratio.

Table 14: AUC of each action per ViF-Bench generator (fake videos of the generator against all 165 real videos), sorted by no-tool-exit AUC. Bold marks the best action.

| Generator | $n$ | No-tool exit | Spatial | Spectral | Latent | Spat.+Spec. | All |
|---|---|---|---|---|---|---|---|
| Wan2.2-I2V-14B | 165 | 0.463 | **0.500** | 0.478 | 0.495 | 0.485 | 0.482 |
| Wan2.2-TI2V-5B-I | 165 | 0.499 | 0.552 | 0.556 | 0.547 | **0.559** | 0.559 |
| HunyuanVideo-I2V | 165 | 0.506 | **0.550** | 0.545 | 0.542 | 0.547 | 0.545 |
| pika-v2 | 151 | **0.535** | 0.485 | 0.505 | 0.499 | 0.471 | 0.468 |
| SkyReels-V2-I2V-14B-540P | 164 | 0.537 | 0.572 | 0.568 | 0.567 | 0.574 | **0.579** |
| gen4-turbo | 121 | **0.540** | 0.530 | 0.514 | 0.535 | 0.512 | 0.515 |
| LTX-Video-13B-I | 165 | 0.566 | 0.594 | 0.598 | 0.594 | 0.601 | **0.606** |
| Wan2.1-VACE-1.3B-T | 165 | 0.599 | **0.646** | 0.630 | 0.630 | 0.626 | 0.626 |
| Wan2.1-T2V-1.3B | 165 | **0.608** | 0.508 | 0.490 | 0.505 | 0.479 | 0.489 |
| sora-2 | 150 | **0.634** | 0.585 | 0.535 | 0.560 | 0.553 | 0.558 |
| Wan2.2-TI2V-5B-T | 165 | 0.647 | **0.657** | 0.623 | 0.636 | 0.646 | 0.647 |
| CogVideoX1.5-5B-T | 165 | **0.669** | 0.631 | 0.616 | 0.621 | 0.613 | 0.613 |
| kling-v1 | 141 | **0.686** | 0.590 | 0.573 | 0.592 | 0.574 | 0.580 |
| HunyuanVideo | 164 | **0.747** | 0.712 | 0.703 | 0.709 | 0.701 | 0.709 |
| Wan2.2-T2V-14B | 165 | **0.765** | 0.695 | 0.678 | 0.686 | 0.685 | 0.695 |
| hailuo | 137 | **0.768** | 0.638 | 0.626 | 0.643 | 0.629 | 0.635 |
| SkyReels-V2 | 165 | **0.778** | 0.767 | 0.752 | 0.760 | 0.755 | 0.760 |
| LTX-Video-13B-T | 165 | **0.802** | 0.790 | 0.767 | 0.797 | 0.766 | 0.772 |
| pixverse-v4-5 | 152 | **0.925** | 0.834 | 0.824 | 0.835 | 0.823 | 0.829 |

Table 15: Container audit on a balanced 1,000-video subset of the Chrono-66k test split. "Raw" uses the original files; the other rows use normalized copies (Section 3.1).

| Detector | AUC (95% CI) | Balanced acc. (%) |
|---|---|---|
| Metadata-GBDT, raw files | 1.000 (1.000–1.000) | 100.0 |
| Metadata-GBDT, normalized | 0.798 (0.767–0.826) | 62.4 |
| Qwen2.5-VL-3B, zero-shot | 0.915 (0.897–0.932) | 84.0 |
| SAFER-Scanner | 0.999 (0.999–1.000) | 98.5 |
| SAFER-Static | 0.998 (0.994–1.000) | 97.7 |
| SAFER default | 0.999 (0.999–1.000) | 98.7 |

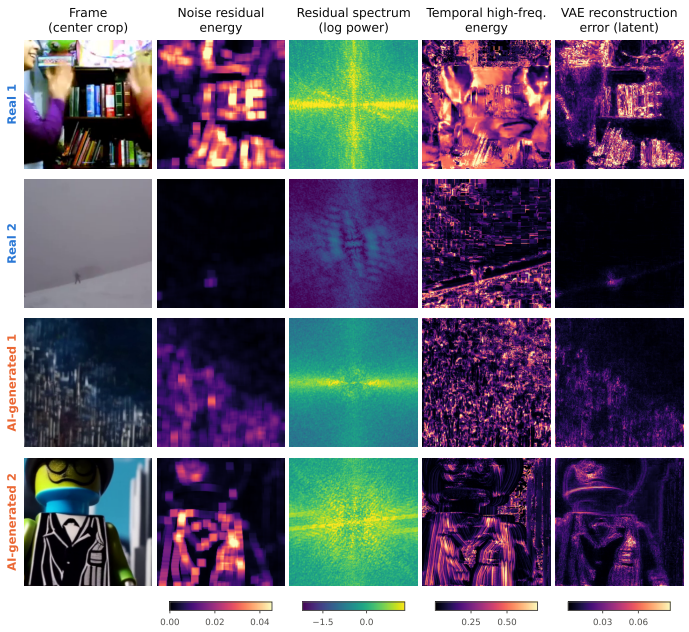



Figure 4: Forensic cues for two real and two AI-generated videos from Chrono-66k (native format, 256×256 center crop, shared color scale per column).

Table 17: Cue statistics over 19 real and 20 AI-generated videos (mean ± std).

| Cue | Real | AI-generated | AUC |
|---|---|---|---|
| Spectrum slope | $-3.83 \pm 0.34$ | $-3.09 \pm 0.97$ | 0.78 |
| Frame HF energy ratio | $0.0002 \pm 0.0002$ | $0.0103 \pm 0.0348$ | 0.78 |
| Noise residual std | $0.0078 \pm 0.0055$ | $0.0225 \pm 0.0215$ | 0.76 |
| Residual flicker | $0.0040 \pm 0.0039$ | $0.0114 \pm 0.0131$ | 0.66 |
| VAE reconstruction error | $0.0110 \pm 0.0083$ | $0.0138 \pm 0.0083$ | 0.65 |
| Residual HF energy ratio | $0.364 \pm 0.072$ | $0.423 \pm 0.159$ | 0.58 |
| Temporal HF ratio | $0.115 \pm 0.068$ | $0.163 \pm 0.133$ | 0.56 |
| Recon. error inconsistency | $0.699 \pm 0.243$ | $0.644 \pm 0.226$ | 0.43 |
| Noise inconsistency | $0.733 \pm 0.272$ | $0.629 \pm 0.242$ | 0.39 |

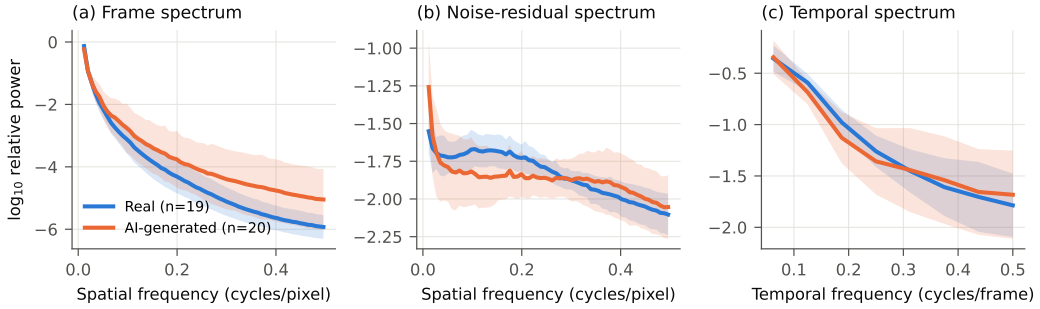



Figure 5: Class-averaged power spectra over 19 real and 20 AI-generated videos (mean ± one standard deviation): (a) frame spectrum, (b) noise-residual spectrum, (c) temporal spectrum. Generated videos keep more power at high spatial frequencies.

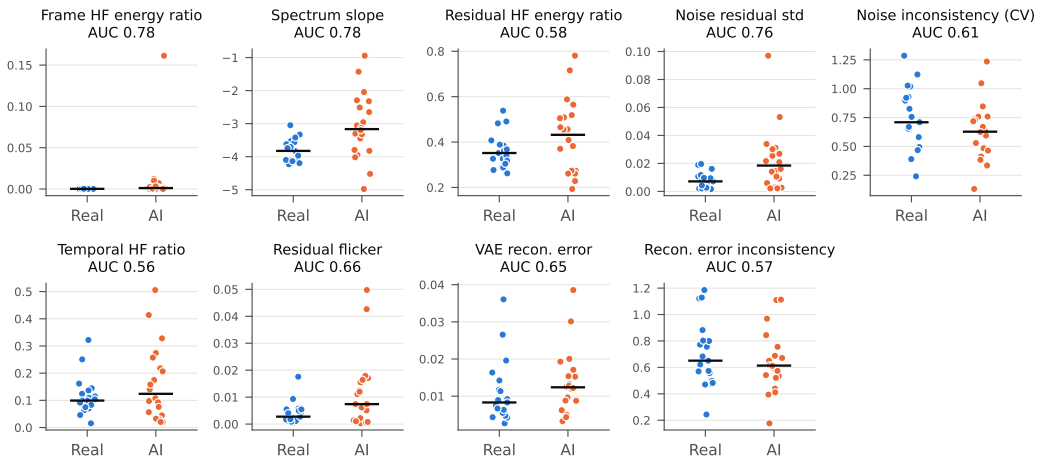



Figure 6: Per-video cue values by class, with the median of each class and the ROC-AUC of each cue.