SPLIT: Training-Free AI-Generated and Partially Edited Video Detection via Spatial Patch-Level Incoherence and Temporal Roughness

Jongyeop Hyun[1] and Hyounghun Kim[1,2⋆]

[1] Graduate School of Artificial Intelligence, POSTECH
[2] Department of Computer Science and Engineering, POSTECH
{mldljyh,h.kim}@postech.ac.kr

Abstract. Deploying AI-generated video detectors in real-world services demands an ultra-low false positive rate (FPR) on real videos to avoid falsely rejecting authentic content, a regime where standard metrics such as AUROC fail to reflect actual operating behavior. We introduce **S**patial **P**atch-**L**evel **I**ncoherence and **T**emporal Roughness (SPLIT), a training-free detector that operates on patch tokens from a frozen vision encoder to detect both fully generated and partially edited videos. SPLIT computes two complementary signals: Two-step Temporal Roughness (TTR), capturing non-smooth patch trajectories via one-step and two-step feature variation contrast, and Local Spatial Motion Incoherence (LSMI), measuring spatially inconsistent temporal changes through gradients of a feature-space motion field. The two are fused multiplicatively with gamma correction to sharpen real–fake separation at strict thresholds. We further propose a service-aligned evaluation protocol based on Fake Recall at fixed FPR with real-only threshold calibration and cross-real threshold transfer. Across three benchmarks—FakeParts, GenVideo, and ViF-Bench—SPLIT achieves the highest Fake Recall at FPR $= 0.1\%$, substantially outperforming supervised and training-free baselines while remaining robust to post-processing with negligible overhead. The code is publicly available at `https://github.com/mldljyh/SPLIT`.

Keywords: AI-generated video detection · Training-free detection · Partial video manipulation

1 Introduction

Recent progress in text-to-video and image-to-video generation has made high-quality synthetic video broadly accessible, while also enabling subtle *partial* edits such as localized inpainting/outpainting, temporal interpolation/extrapolation, faceswap, and style transfer [2, 6, 18, 30, 40]. This creates an urgent need for reliable AI-generated video detection in deployment settings with stringent operating constraints: false rejections of authentic user content must remain extremely

⋆ Corresponding author.

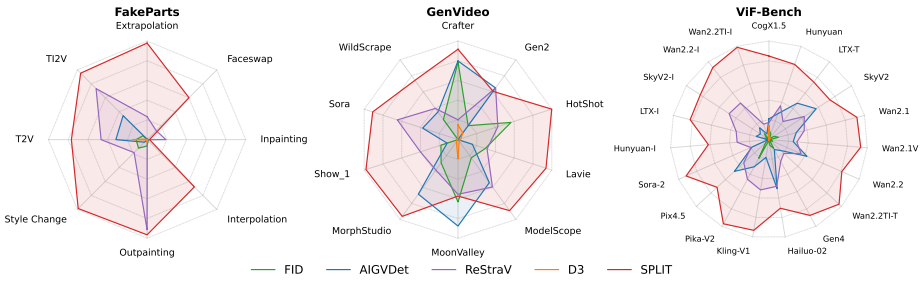

Fig. 1: Fake Recall (%) at FPR = 0.1% for FID, AIGVDet, ReStraV, D3, and SPLIT across FakeParts, GenVideo, and ViF-Bench. SPLIT (red) consistently achieves the highest and most uniform recall across all manipulation types and generators.

rare (*e.g.*, FPR on real videos $\leq 0.1\%$), yet recall on manipulated content must stay high [45]. In this regime, aggregate metrics such as AUROC can be misleading because they do not directly reflect behavior at ultra-low FPR [14, 45].

Prior work has largely focused on supervised detectors that exploit spatial artifacts, frequency cues, or spatiotemporal inconsistencies, often leveraging large pretrained video encoders [1, 7, 11, 12, 21, 28, 37, 47, 52]. While effective within their training distributions, these methods can degrade under distribution shift from new generators, post-processing, and especially partial edits where most of the content remains real [23]. Training-free approaches such as D3 [51] improve cross-generator generalization via clip-level second-order temporal volatility in frozen features, but pooling over the clip can dilute sparse, localized manipulation evidence—as highlighted by partial-edit benchmarks such as FakeParts [23].

This paper targets the intersection of three practical requirements: (i) detecting both fully generated and partially edited videos, (ii) generalizing across generators without task-specific training, and (iii) operating under service-aligned ultra-low-FPR constraints with thresholds calibrated using real data only. We introduce **S**patial **P**atch-**L**evel **I**ncoherence and **T**emporal Roughness (SPLIT), a training-free detector that operates on *patch tokens* from a frozen pretrained vision encoder. Rather than collapsing a frame into a single embedding, SPLIT preserves localized spatiotemporal evidence and aggregates it into a single manipulation score without any learned parameters.

SPLIT computes two complementary patch-token signals. **Two-step Temporal Roughness (TTR)** measures deviations from smooth temporal evolution by comparing accumulated one-step and two-step feature variations, capturing artifacts such as flicker and local temporal misalignment. **Local Spatial Motion Incoherence (LSMI)** measures whether temporal changes are spatially coordinated by constructing a feature-space motion field and computing local spatial gradients; manipulated regions tend to induce spatially inconsistent temporal changes. We fuse these signals multiplicatively with a gamma correction to amplify subtle but consistent roughness differences.

We also advocate deployment-constrained evaluation, reporting *Fake Recall at FPR* $= \alpha$ for $\alpha \in \{0.1\%, 1\%, 5\%\}$, with thresholds calibrated on real videos only, and introduce *cross-real threshold transfer* to quantify calibration stability when the real-data source shifts. We evaluate on three complementary benchmarks: FakeParts [23] (partial edits), GenVideo [7] (earlier generators; near-domain for supervised baselines), and ViF-Bench [20] (newer 2024–2025 generators; out-of-distribution stress test). As shown in Fig. 1 at FPR = 0.1%, SPLIT outperforms representative supervised baselines and the training-free D3 across manipulation types and generators. Our contributions are as follows:

- We introduce **SPLIT**, a training-free patch-token detector that measures localized spatiotemporal incoherence through **TTR** and **LSMI**, fused with gamma correction to sharpen separation under strict thresholds.
- We formalize service-aligned AI-generated video detection via Fake Recall at ultra-low FPR with real-only threshold calibration, and propose cross-real threshold transfer to measure calibration stability across heterogeneous real domains.
- We demonstrate strong generalization and robustness across partial edits and diverse generators on FakeParts, GenVideo, and ViF-Bench, with negligible overhead beyond the frozen encoder forward pass.

2 Related Work

AI-Generated Video Detection. Early deepfake detection targeted face-centric manipulations using spatial artifacts [12, 21], later incorporating temporal modeling for cross-frame consistency [47, 52]. With the rise of diffusion-based generators [2, 6], recent detectors target broader video-level artifacts beyond faces. These approaches span multiple strategies: frequency-aware methods like NPR [37] exploit upsampling artifacts that persist across generators; spatiotemporal methods such as STIL [11] learn joint appearance-dynamics inconsistencies, while FTCN [52] and MINTIME [8] model temporal coherence primarily for face forgery scenarios. Foundation-model-based approaches leverage pretrained encoders—XCLIP [28] extends CLIP [34] with temporal attention, and DeMamba [7] augments such encoders with a Mamba [10] module that captures local spatiotemporal inconsistencies through continuous scanning of partitioned regions. AIGVDet [1] combines spatial CNN features with optical-flow branches for video-level fusion.

A complementary frozen-feature line uses representation dynamics rather than training large video detectors. ReStraV [17], inspired by perceptual straightening, observes that natural videos trace straighter trajectories than AI-generated videos in DINOv2 feature space, summarizing frame-level step distances and curvature statistics to train a lightweight classifier. D3 [51] is fully training-free and measures clip-level second-order temporal volatility via central differences in frozen features. These methods show that frozen representation dynamics provide useful cross-generator cues, but they rely on global frame/clip summaries that can dilute sparse manipulation evidence. In contrast, SPLIT operates on

patch tokens and explicitly targets localized spatiotemporal incoherence: TTR compares accumulated 1-step and 2-step patch changes, while LSMI measures spatial disagreement of patch-wise temporal motion. This patch-level formulation is especially suited to partial edits, where manipulation evidence is region-specific.

Benchmarks and Partial Edits. Several benchmarks have been developed for AI-generated video detection [7, 20, 22, 29, 42]. GenVideo [7] provides a million-scale dataset covering diverse generators with cross-generator and degraded-video evaluation tasks. ViF-Bench [20] extends coverage to the latest generators including Sora-2 [30], Wan2.2 [40], and Kling [18], with fine-grained artifact annotations. These benchmarks primarily target fully AI-generated videos (T2V, TI2V), driving progress on global detectors. FakeParts [23] addresses a gap by introducing localized spatial (inpainting, outpainting, faceswap), temporal (interpolation, extrapolation), and style manipulations applied to otherwise real videos, with pixel- and frame-level annotations. Their evaluation reveals performance drops of up to 50% for many detectors on partial edits, confirming that preserved real context defeats global approaches—a finding that highlights the value of our patch-level formulation [23]. While some recent deepfake benchmarks report strict operating points [45], AI-generated video and partial-edit benchmarks still predominantly emphasize aggregate metrics such as AUROC, which can obscure behavior at ultra-low false-positive rates. To our knowledge, they do not evaluate real-only threshold calibration at FPR targets as low as 0.1% or the stability of such thresholds under shifts in the real-data domain; we address this via Fake Recall at FPR $\in \{0.1\%, 1\%, 5\%\}$ with real-only calibration and cross-real threshold transfer.

3 Method

3.1 Overview

We propose Spatial Patch-Level Incoherence and Temporal Roughness (SPLIT), a training-free detector for AI-generated and partially edited videos under service-aligned operating constraints where the false positive rate (FPR) on real videos must remain extremely low (*e.g.*, $\leq 0.1\%$). SPLIT analyzes a clip using patch-level representations from a frozen pretrained vision encoder and computes two complementary signals: (1) Two-step Temporal Roughness (TTR), which measures how strongly patch features deviate from smooth temporal evolution by comparing 1-step and 2-step temporal variations; and (2) Local Spatial Motion Incoherence (LSMI), which measures spatial disagreement of patch-wise temporal changes by computing local spatial gradients of a feature-space motion field. These signals are fused multiplicatively with a gamma exponent to obtain a final score, where larger values indicate a higher likelihood of manipulation. An overview of SPLIT is shown in Fig. 2.

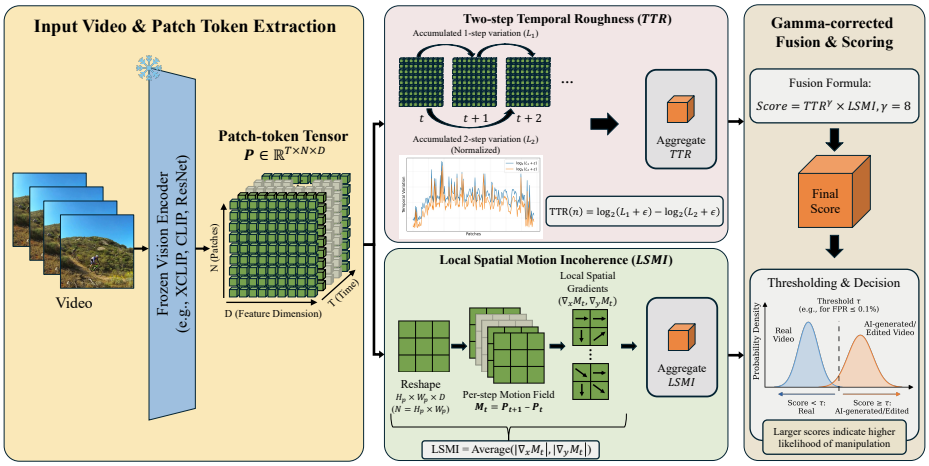

Fig. 2: Overview of SPLIT. A frozen vision encoder produces patch tokens $P \in \mathbb{R}^{T\times N\times D}$, from which two signals are computed: *Two-step Temporal Roughness (TTR)*, comparing 1-step and 2-step feature variations, and *Local Spatial Motion Incoherence (LSMI)*, measuring spatial gradients of a feature-space motion field. These are combined via gamma-corrected multiplicative fusion and thresholded using real-only calibration to classify videos as real or AI-generated/edited.

3.2 Patch Token Extraction

Given an input video clip $x \in \mathbb{R}^{T\times 3\times H\times W}$, we extract per-frame patch tokens using a frozen pretrained vision encoder. Let $Z_t \in \mathbb{R}^{N\times D}$ denote the N patch tokens of dimension D at time t.

For transformer encoders (*e.g.*, CLIP [34], XCLIP [28], DINO [31]), we use the final-layer patch tokens and exclude the class token (`[CLS]`), yielding $Z_t \in \mathbb{R}^{N\times D}$. For CNN encoders (*e.g.*, ResNet [15], VGG [36], EfficientNet [38], MobileNet [16]), we extract the last convolutional feature map $F_t \in \mathbb{R}^{C\times H'\times W'}$ and treat each spatial cell as a token by reshaping to $Z_t \in \mathbb{R}^{(H'W')\times C}$, so that $N = H'W'$ and $D = C$. Stacking across time gives a patch-token tensor

$$P = \{p_{t,n}\} \in \mathbb{R}^{T\times N\times D}, \tag{1}$$

where $p_{t,n} \in \mathbb{R}^D$ is the feature token of patch n at time t. To support spatial neighborhood operations, we also view tokens on a 2D grid of size $H_p \times W_p$ such that $N = H_pW_p$. For ViT-style patchification, (H_p, W_p) corresponds to the patch grid; for CNN feature maps, $(H_p, W_p) = (H', W')$.

3.3 Two-step Temporal Roughness (TTR)

SPLIT targets localized temporal artifacts that frequently arise from generation or partial edits, such as patch-wise flicker, inconsistent texture refresh, or local

temporal misalignment. Rather than relying on a single notion of global volatility, TTR measures two-step consistency of temporal evolution in representation space by comparing accumulated 1-step and 2-step changes. For each patch n, we define the accumulated 1-step variation as

$$L_1(n) = \sum_{t=1}^{T-1} \|p_{t+1,n} - p_{t,n}\|_2, \tag{2}$$

and the accumulated 2-step variation $\hat{L}_2(n)$ together with its normalized form $L_2(n)$, which accounts for step length and short sequences, as

$$\hat{L}_2(n) = \sum_{t=1}^{T-2} \|p_{t+2,n} - p_{t,n}\|_2, \qquad L_2(n) = \left(\frac{\hat{L}_2(n)}{2}\right) \cdot \frac{T-1}{T-2}, \tag{3}$$

which requires $T \geq 3$. TTR is then defined as a log-ratio:

$$\mathrm{TTR}(n) = \log_2(L_1(n) + \epsilon) - \log_2(L_2(n) + \epsilon), \tag{4}$$

where ϵ is a small constant for numerical stability. $L_1(n)$ is the consecutive-step path length of patch n's feature trajectory, while $L_2(n)$ is a normalized two-step chord-length proxy. For constant-velocity evolution, the normalization gives $L_2(n) = L_1(n)$, and the two remain close for smooth motion. Irregular temporal evolution, such as flicker, texture refresh, or local temporal misalignment, makes successive patch-token displacements less aligned, increasing $L_1(n)$ relative to $L_2(n)$ and therefore increasing $\mathrm{TTR}(n)$. Thus, TTR is best interpreted as a statistical patch-level cue, not a necessary property of every fake clip: generated or edited videos are more likely to contain locally rough token trajectories, while real videos may also exhibit such behavior under cuts, shake, or abrupt motion. Our real-only threshold calibration accounts for this natural tail of the real-video score distribution. The log-ratio reduces sensitivity to absolute motion magnitude. We convert patch-wise roughness into a clip-level statistic by averaging in log-space:

$$\mathbf{TTR} = \frac{1}{N} \sum_{n=1}^{N} \mathrm{TTR}(n). \tag{5}$$

This choice has two motivations. First, since $\mathrm{TTR}(n) = \log_2\left(\frac{L_1(n)+\epsilon}{L_2(n)+\epsilon}\right)$, the average corresponds to the log of the geometric mean of per-patch roughness ratios, $\mathbf{TTR} = \log_2\left(\prod_n \frac{L_1(n)+\epsilon}{L_2(n)+\epsilon}\right)^{1/N}$, which aggregates multiplicative evidence while reducing sensitivity to extreme outliers. Second, mean pooling estimates the expected local temporal roughness across the frame and yields a low-variance score, which is important for our real-only threshold calibration at ultra-low FPR. This patch-level design preserves sensitivity to localized edits while remaining robust to content and object motion through aggregation.

3.4 Local Spatial Motion Incoherence (LSMI)

To capture whether temporal changes are spatially coordinated, we define a feature-space motion field and measure its local spatial disagreement. We first reshape tokens into a grid $P_t \in \mathbb{R}^{H_p \times W_p \times D}$ and compute the per-step motion field:

$$M_t = P_{t+1} - P_t \in \mathbb{R}^{H_p \times W_p \times D}, \quad t = 1, \ldots, T-1. \tag{6}$$

We then compute local spatial gradients of motion using forward differences:

$$\nabla_x M_t(i,j) = M_t(i,j) - M_t(i,j+1), \quad j = 1, \ldots, W_p - 1, \tag{7}$$

$$\nabla_y M_t(i,j) = M_t(i,j) - M_t(i+1,j), \quad i = 1, \ldots, H_p - 1. \tag{8}$$

LSMI is the mean magnitude of these gradients, averaged across time and spatial locations:

$$\textbf{LSMI} = \frac{1}{2}\left(\mathbb{E}_{t,i,j}\left[\|\nabla_x M_t(i,j)\|_2\right] + \mathbb{E}_{t,i,j}\left[\|\nabla_y M_t(i,j)\|_2\right]\right). \tag{9}$$

Intuitively, real motion tends to be spatially coherent, with neighboring patches changing similarly, whereas localized generation or editing artifacts often yield spatially inconsistent temporal changes, thereby increasing LSMI.

3.5 Gamma-corrected Fusion and Scoring

We fuse the two signals with a multiplicative rule:

$$s(x) = \textbf{TTR}^{\gamma} \cdot \textbf{LSMI}, \tag{10}$$

where larger scores indicate higher likelihood of manipulation. Here γ is a fixed, non-learned sharpening constant that amplifies subtle TTR differences. We selected $\gamma = 8$ once on a small held-out validation split and then kept it fixed for all benchmarks and FPR targets.

To meet service-aligned constraints, we choose a threshold τ using real-only threshold calibration such that the resulting FPR on real calibration data matches a target (*e.g.*, 0.1%, 1%, or 5%). At test time, we predict manipulated if $s(x) \geq \tau$ and real otherwise.

4 Experiments

4.1 Experimental Setup

Training Data. SPLIT is training-free and therefore requires no training dataset. For comparison with learning-based detectors, we train all supervised baselines following the experimental settings of DeMamba [7] and D3 [51] using GenVideo-100K [7] as the sole training set.

Test Benchmarks. We evaluate on three benchmarks. FakeParts [23] targets partially edited videos with eight manipulation categories including T2V, TI2V, interpolation, extrapolation, inpainting, outpainting, faceswap, and style change. GenVideo [7] focuses on T2V content from ten earlier generators. ViF-Bench [20] targets newer 2024–2025 systems spanning fourteen T2V and five I2V generators (*e.g.*, Sora-2, Wan2.2, Kling), providing an out-of-distribution test of generalization. Complete lists of generators for each benchmark are provided in the supplementary material.

Baselines. We compare SPLIT against the training-free D3 [51] method as well as ten supervised detectors: three image-level methods (FID [50], NPR [37], STIL [11]) and seven video-level methods (FTCN [52], MINTIME [8], TALL [47], XCLIP [28], AIGVDet [1], DeMamba [7], ReStraV [17]). NPR, STIL, FTCN, MINTIME, TALL, XCLIP, and DeMamba are reproduced using the official DeMamba repository, while FID, AIGVDet, and ReStraV are trained and evaluated using their respective official implementations. D3 is evaluated using its official repository.

Implementation Details. We adopt a pretrained XCLIP-B/16 [28] vision encoder for patch token extraction. Following D3 [51], we extract segments of up to 2 seconds, sample frames at 8 fps with uniform intervals, crop 10% from the longer side, and resize all frames to 224×224 pixels. All experiments are conducted on an AMD EPYC 9554 64-Core Processor with an NVIDIA RTX A6000 GPU.

Service-Aligned Evaluation Metric. Standard metrics such as AUROC do not reflect deployment settings where falsely rejecting authentic user content must be extremely rare. We therefore report Fake Recall at fixed False Positive Rate (FPR) on real videos α for $\alpha \in \{0.1\%, 1\%, 5\%\}$. Let $s(x)$ be the detector score (larger means more likely manipulated), and classify a video as fake if $s(x) \geq \tau$. We choose a threshold τ_α using a real-only calibration set $\mathcal{R}$ such that the fraction of real videos flagged as fake is at most α:

$$\mathrm{FPR}(\tau) \triangleq \Pr_{x \sim \mathcal{R}}\big[s(x) \geq \tau\big], \qquad \tau_\alpha \triangleq \inf\{\tau : \mathrm{FPR}(\tau) \leq \alpha\}. \tag{11}$$

We then evaluate detection performance on manipulated videos $\mathcal{F}$ at this operating point:

$$\text{Fake Recall@}\alpha \triangleq \Pr_{x \sim \mathcal{F}}\big[s(x) \geq \tau_\alpha\big]. \tag{12}$$

Cross-Real Threshold Transfer. To measure robustness to the choice of real calibration source, we calibrate thresholds on two disjoint real sets used in prior work: ROVI [44] (the real subset of FakeParts) and MSR-VTT [46] (the real subset of GenVideo). For each FPR target α, we compute τ_{ROVI} and $\tau_{\text{MSR-VTT}}$ separately, apply each threshold to the test benchmarks, and report the harmonic mean of the resulting Fake Recall values. This prevents over-crediting methods that are well-calibrated on only one real domain. Per-real-set results and conventional AUROC metrics are provided in the supplementary material.

Table 1: Fake Recall (%) on FakeParts under cross-real threshold transfer at FPR $\in \{0.1\%, 1\%, 5\%\}$. Methods are grouped by horizontal lines. Overall denotes the simple average across all categories. Best results are in **bold**.

Method	Extrapolation			Faceswap			Inpainting			Interpolation			Outpainting		
	0.1%	1%	5%	0.1%	1%	5%	0.1%	1%	5%	0.1%	1%	5%	0.1%	1%	5%
NPR	0.00	1.03	8.43	0.00	0.00	0.53	0.12	2.75	12.45	0.00	0.02	1.28	0.00	0.00	0.16
STIL	0.09	0.18	2.79	0.08	0.16	2.40	0.69	1.36	7.56	0.17	0.48	2.58	0.18	0.34	1.96
FID	0.00	0.70	5.58	0.16	8.70	35.39	0.33	6.48	28.19	1.63	10.84	34.24	6.44	28.19	52.72
MINTIME	0.04	0.33	2.63	0.00	0.03	0.78	0.06	0.82	3.53	2.17	11.67	29.55	0.07	0.50	2.60
FTCN	0.23	1.35	3.69	0.00	0.24	1.49	0.05	0.68	3.54	1.25	6.10	15.03	0.00	0.64	2.05
TALL	0.05	1.27	5.22	0.00	0.13	0.88	0.26	2.04	8.32	0.24	1.59	7.29	0.19	1.09	5.08
XCLIP	0.00	0.03	0.70	0.00	0.12	1.36	0.08	0.78	4.92	0.18	1.03	4.72	0.09	0.45	2.18
AIGVDet	0.42	1.77	4.87	0.89	6.47	14.63	0.98	7.10	19.79	0.43	3.18	7.62	0.56	1.80	3.50
DeMamba	0.00	0.05	0.45	0.03	0.21	1.92	0.07	1.06	5.01	0.00	1.35	8.40	0.14	1.32	6.06
ReStraV	23.27	61.48	81.58	12.98	50.72	77.63	**19.09**	**30.70**	**34.53**	0.18	11.72	50.69	91.45	98.92	99.79
D3	0.00	16.77	58.66	0.04	40.86	86.89	0.00	1.20	7.05	0.00	11.60	53.61	2.72	66.03	94.13
SPLIT	**97.86**	**99.37**	**99.86**	**60.09**	**84.21**	**96.35**	2.48	14.63	33.56	**67.81**	**87.57**	**96.58**	**96.52**	**99.23**	**99.85**

Method	Style Change			T2V			TI2V			Overall		
	0.1%	1%	5%	0.1%	1%	5%	0.1%	1%	5%	0.1%	1%	5%
NPR	0.04	1.36	4.78	0.38	4.25	18.45	0.45	4.67	21.43	0.12	1.76	8.44
STIL	2.19	4.20	15.63	6.72	11.30	31.79	3.52	6.85	22.96	1.70	3.11	10.96
FID	12.26	38.59	64.28	11.15	37.61	62.68	5.67	30.49	63.15	4.71	20.20	43.28
MINTIME	3.54	20.74	41.95	2.19	12.99	30.92	4.24	25.27	56.10	1.54	9.04	21.01
FTCN	13.98	33.91	55.65	8.57	24.13	42.81	19.41	50.10	72.75	5.44	14.65	24.63
TALL	0.30	1.65	6.53	0.32	1.89	6.84	0.05	0.71	3.39	0.18	1.30	5.44
XCLIP	1.56	8.65	26.55	3.91	14.88	34.28	9.53	29.74	58.89	1.92	6.96	16.70
AIGVDet	3.69	8.54	12.26	31.65	47.88	56.74	34.39	61.26	76.42	9.13	17.25	24.48
DeMamba	1.31	8.76	25.52	1.59	9.47	24.13	3.37	17.61	43.48	0.81	4.98	14.37
ReStraV	18.60	69.91	90.06	46.65	79.45	90.37	73.13	92.26	97.40	35.67	61.89	77.75
D3	0.34	40.14	79.63	14.46	68.30	**93.22**	0.66	54.49	91.56	2.28	37.42	70.59
SPLIT	**98.80**	**99.71**	**99.92**	**76.74**	**85.34**	91.64	**95.27**	**98.61**	**99.86**	**74.45**	**83.58**	**89.70**

Table 2: Fake Recall (%) on GenVideo under cross-real threshold transfer at FPR $\in \{0.1\%, 1\%, 5\%\}$.

Method	Crafter [5]			Gen2 [9]			HotShot [27]			Lavie [43]			ModelScope [41]			MoonValley [25]		
	0.1%	1%	5%	0.1%	1%	5%	0.1%	1%	5%	0.1%	1%	5%	0.1%	1%	5%	0.1%	1%	5%
NPR	0.42	7.61	30.60	0.28	3.27	15.14	0.00	1.68	20.25	0.00	0.84	9.56	1.64	8.88	23.97	0.63	4.53	25.30
STIL	22.75	31.19	60.45	12.83	20.49	53.42	12.71	21.59	46.23	10.50	16.46	38.78	10.14	17.10	43.23	20.61	31.22	64.96
FID	78.71	93.76	98.23	17.47	54.61	80.12	56.46	90.16	97.91	29.09	69.23	89.30	22.88	64.27	87.77	63.23	91.98	98.22
MINTIME	15.06	59.57	90.95	9.10	51.27	85.51	1.49	10.46	30.85	13.23	56.53	85.93	6.81	29.50	58.57	9.42	43.20	79.71
FTCN	61.09	88.03	96.60	43.44	77.09	90.16	15.56	42.77	66.90	33.32	68.78	86.64	21.76	48.92	69.52	30.80	71.41	91.26
TALL	0.46	2.26	6.70	0.22	2.49	9.00	0.84	3.27	10.13	0.00	0.77	3.54	0.41	3.30	8.94	0.00	1.00	4.88
XCLIP	12.00	42.72	74.03	5.98	28.32	68.55	5.53	18.99	48.72	5.78	18.73	45.42	5.98	23.47	58.17	7.78	30.57	70.79
AIGVDet	79.71	93.15	97.26	64.78	81.91	90.58	0.00	0.00	0.00	15.39	31.14	44.37	53.99	77.86	89.40	**87.62**	**96.50**	**98.47**
DeMamba	21.94	54.77	79.07	22.96	59.21	83.98	4.08	21.31	45.52	3.74	22.57	50.95	3.89	20.40	46.27	2.86	16.72	40.36
ReStraV	20.12	67.68	90.78	**64.86**	84.82	93.07	42.47	93.37	98.84	31.00	74.97	88.58	59.37	86.48	94.19	55.44	80.02	87.01
D3	15.05	75.72	94.00	8.39	**87.91**	**98.73**	3.05	61.21	89.38	2.24	65.21	88.34	1.96	53.09	84.63	19.41	85.35	97.18
SPLIT	**91.65**	**97.49**	**99.71**	60.49	81.73	94.96	**99.93**	**100.00**	**100.00**	**93.80**	**96.96**	**98.82**	**89.01**	**94.56**	**98.14**	57.22	81.43	95.77

Method	MorphStudio [26]			Show_1 [49]			Sora [3]			WildScrape [7]			Overall		
	0.1%	1%	5%	0.1%	1%	5%	0.1%	1%	5%	0.1%	1%	5%	0.1%	1%	5%
NPR	0.28	2.74	17.94	0.00	0.00	0.82	0.00	0.00	16.48	0.22	3.25	15.57	0.35	3.28	17.56
STIL	15.29	21.80	49.91	8.14	14.37	40.08	7.14	12.24	33.01	7.22	11.56	30.99	12.73	19.80	46.11
FID	29.10	71.67	91.75	18.19	56.37	86.94	0.00	8.83	25.25	25.17	46.08	66.00	34.03	64.69	82.15
MINTIME	14.03	55.51	88.93	6.99	36.54	73.00	2.38	14.06	28.57	13.00	43.49	67.29	9.15	40.01	68.93
FTCN	55.20	82.67	94.62	20.81	53.68	75.76	10.42	18.99	33.01	35.55	55.63	67.50	32.80	60.80	77.20
TALL	0.78	2.95	9.39	0.00	2.82	9.52	0.00	0.00	2.68	0.17	1.03	5.90	0.29	1.99	7.07
XCLIP	19.04	43.27	71.59	6.73	28.76	54.24	5.10	17.86	43.30	8.34	24.11	48.90	8.23	27.68	58.37
AIGVDet	68.36	86.08	94.58	11.58	24.31	34.16	37.72	67.81	79.37	34.64	46.88	55.78	45.38	60.56	68.40
DeMamba	18.61	47.15	73.28	10.58	38.06	67.43	2.86	12.61	27.42	8.50	26.35	45.90	10.00	31.91	56.02
ReStraV	38.15	81.76	93.72	40.09	87.82	96.83	64.67	89.11	96.30	39.07	72.82	83.88	45.52	81.89	92.32
D3	2.78	59.61	89.99	1.40	72.99	95.73	0.00	64.56	94.61	0.44	39.31	72.63	5.47	66.50	90.52
SPLIT	**96.19**	**98.86**	**99.79**	**98.13**	**99.93**	**100.00**	**90.93**	**99.10**	**100.00**	**74.30**	**80.89**	**86.27**	**85.16**	**93.09**	**97.35**

Table 3: Fake Recall (%) on ViF-Bench under cross-real threshold transfer at FPR = 0.1%. Results for FPR ∈ {1%, 5%} are provided in the supplementary material.

Method	T2V Models															I2V Models					Overall
	CogX1.5 [48]	Hunyuan [19]	LTX-T [13]	SkyV2 [4]	Wan2.1 [40]	Wan2.1V [40]	Wan2.2 [40]	Wan2.2TI-T [40]	Gen4 [35]	Hailuo-02 [24]	Kling-V1 [18]	Pika-V2 [32]	Pix4.5 [33]	Sora-2 [30]	Hunyuan-I [19]	LTX-I [13]	SkyV2-I [4]	Wan2.2-I [40]	Wan2.2TI-I [40]	
NPR	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
STIL	2.02	3.05	5.58	3.54	3.03	0.51	6.57	1.52	1.32	3.51	1.71	1.08	2.21	1.11	1.01	1.52	0.51	1.01	0.00	2.15
FID	10.59	5.89	0.96	6.09	9.84	0.97	3.91	0.91	10.21	5.88	1.07	21.54	4.63	0.93	0.00	5.10	2.39	7.41	0.93	5.22
MINTIME	11.10	10.71	7.44	10.66	15.38	2.95	10.88	2.96	9.25	19.58	9.39	17.86	14.29	2.71	0.00	3.65	2.13	0.81	0.85	8.03
FTCN	18.61	24.57	14.07	23.43	16.39	6.17	16.72	1.68	16.81	42.84	22.26	28.46	39.23	3.45	0.00	7.84	4.02	4.71	2.50	15.46
TALL	0.00	0.51	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.51	0.05
XCLIP	1.21	2.71	0.00	3.37	0.00	0.51	2.42	0.00	0.00	2.63	0.57	0.00	0.55	0.00	0.00	1.01	0.00	0.00	0.00	0.79
AIGVDet	21.04	27.76	46.82	57.24	28.51	24.69	42.14	16.32	11.47	50.35	18.29	31.32	47.51	18.66	0.92	13.30	10.39	15.20	6.13	25.69
DeMamba	1.73	7.41	0.76	3.52	5.37	1.44	5.31	2.69	4.85	9.93	2.85	1.62	5.87	1.67	0.68	1.62	1.53	2.34	1.44	3.30
ReStraV	17.35	36.31	23.22	42.81	36.97	28.47	35.59	20.63	25.08	48.00	51.70	45.72	34.63	19.95	32.32	34.25	48.16	46.94	16.58	33.93
D3	14.32	0.00	2.01	0.00	0.00	0.00	0.00	0.00	0.00	0.00	1.13	0.00	0.00	0.00	1.01	0.00	0.00	0.00	0.00	0.97
SPLIT	**84.77**	**80.60**	**73.98**	**73.79**	**92.05**	**93.37**	**80.71**	**96.66**	**87.38**	**70.53**	**93.23**	**97.01**	**71.42**	**91.70**	**61.58**	**82.38**	**83.09**	**92.64**	**98.98**	**84.52**

4.2 Results and Analysis

Fake Recall under cross-real threshold transfer is reported in Tabs. 1 to 3. Across all three benchmarks, SPLIT consistently achieves the highest Fake Recall despite being entirely training-free.

FakeParts. SPLIT attains 74.45% Fake Recall at FPR = 0.1% overall (95% category-stratified bootstrap CI [72.42%, 76.14%]; see supplementary material), compared to 35.67% for the best supervised baseline (ReStraV) and 2.28% for D3 (CI [0.53%, 5.43%]). Several categories are near-saturated even at this strictest threshold: Extrapolation (97.86%), Outpainting (96.52%), Style Change (98.80%), and TI2V (95.27%). Faceswap (60.09%) and Interpolation (67.81%) also show substantial improvements over all baselines. Inpainting remains the most difficult category (2.48% at FPR = 0.1%). Nevertheless, SPLIT ranks second among the compared methods, reflecting the challenge of detecting edits confined to a small spatial region per frame.

GenVideo. SPLIT achieves 85.16% Fake Recall at FPR = 0.1%, rising to 93.09% at 1% and 97.35% at 5%. The strongest supervised competitor under the strictest constraint is ReStraV at 45.52%, while D3 reaches only 5.47%. SPLIT exceeds 90% at FPR = 0.1% on six of the ten generators, with performance improves substantially as FPR relaxes, consistent with a score distribution that provides stable separation between real and generated content.

ViF-Bench. At FPR = 0.1%, SPLIT achieves 84.52% overall Fake Recall and ranks first on every individual generator. Supervised detectors degrade sharply: the best baseline (ReStraV) reaches 33.93%, while D3 drops to 0.97%. This confirms that patch-level TTR and LSMI cues capture generation artifacts that persist across architectural generations of video synthesizers, without requiring retraining.

Table 4: Cross-real threshold transfer stability. Actual FPR (%) on a held-out real domain when the threshold is calibrated to a target FPR on a disjoint real domain. Values closer to the nominal target (0.1, 1, or 5%) indicate more stable cross-domain calibration.

	MSR-VTT → ROVI			ROVI → MSR-VTT		
Method	@0.1	@1	@5	@0.1	@1	@5
FID	0.62	8.25	38.64	0.01	0.16	0.67
FTCN	0.48	2.28	7.40	0.01	0.28	2.68
AIGVDet	13.22	25.12	31.69	0.00	0.00	0.02
ReStraV	0.00	0.00	0.00	15.30	39.86	66.10
D3	0.00	0.07	1.52	1.18	4.06	9.04
SPLIT	0.03	0.29	4.22	0.56	2.18	5.49

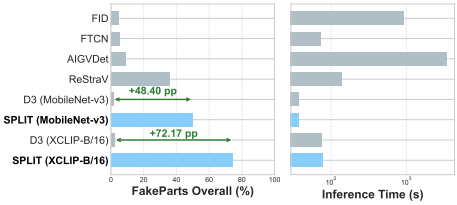

Fig. 3: Comparison of FakeParts Overall Fake Recall at FPR = 0.1% (left) and end-to-end inference time on 1,000 FakeParts T2V samples and batch size of 1 (right, log scale). Inference time for AIGVDet includes RAFT [39]-based optical flow preprocessing.

Analysis. The performance gap is most pronounced at FPR = 0.1%, precisely the regime relevant to deployment. This supports our central hypothesis: localized spatiotemporal incoherence—measured at the patch level on frozen encoder tokens—provides a discriminative and broadly generalizable signal. By avoiding clip-level aggregation, SPLIT preserves sensitivity to partial manipulations while remaining robust across diverse and newly emerging video generators, without any generator-specific training or adaptation, under real-only threshold calibration with cross-real transfer.

4.3 Cross-Real Threshold Transfer

A practical deployment concern is whether a threshold calibrated on one real-only source remains reliable when the target real distribution differs. As shown in Tab. 4, we report the actual rejection rate on a held-out real domain when the threshold is set to achieve a target FPR on a different real domain. We compare against the five strongest competing methods selected by FakeParts Overall Fake Recall at FPR = 0.1%.

Our method maintains the transferred FPR close to the intended target in both directions, indicating stable calibration across real domains rather than over-specialization to a particular real set. Several baselines, by contrast, exhibit pronounced domain-dependent drift: a threshold tuned on one real source can become substantially more permissive or more restrictive on the other, undermining operation at ultra-low FPR. This stability, combined with the strong Fake Recall reported in §4.2, confirms that SPLIT's patch-level signals generalize not only across generators but also across real-data distributions—a prerequisite for practical threshold setting without access to target-domain real samples.

Table 5: Fake Recall (%) on FakeParts Overall at FPR $\in \{0.1\%, 1\%, 5\%\}$ for Gaussian blur ($\sigma = 1, 2$), JPEG compression ($q = 90, 80$), and Axis Flips (x, y) post-processing, compared against the unperturbed baseline.

Method	Baseline			Gaussian Blur ($\sigma = 1$)			Gaussian Blur ($\sigma = 2$)			JPEG ($q = 90$)			JPEG ($q = 80$)			Flip (x-axis)			Flip (y-axis)		
	0.1%	1%	5%	0.1%	1%	5%	0.1%	1%	5%	0.1%	1%	5%	0.1%	1%	5%	0.1%	1%	5%	0.1%	1%	5%
NPR	0.12	1.76	8.44	0.25	2.10	9.49	0.24	1.96	6.67	0.11	1.66	8.24	0.13	1.70	8.29	0.12	1.77	9.03	0.14	1.79	8.78
STIL	1.70	3.11	10.96	1.41	3.35	11.58	1.08	2.67	9.70	1.68	2.97	10.52	1.69	3.15	10.24	1.21	3.36	11.33	1.67	3.05	10.01
FID	4.71	20.20	43.28	3.83	11.63	24.72	1.48	6.50	18.68	4.78	20.22	43.68	4.98	21.82	44.11	4.20	21.25	44.78	0.22	1.18	5.58
MINTIME	1.54	9.04	21.01	1.18	6.36	16.60	0.43	3.84	11.75	1.22	7.31	17.72	0.94	5.98	15.80	1.95	10.02	22.61	1.20	6.64	16.56
FTCN	5.44	14.65	24.63	4.74	10.68	19.97	1.50	6.21	14.19	5.48	13.70	23.33	5.74	12.62	22.10	6.13	15.95	26.53	5.18	12.81	22.93
TALL	0.18	1.30	5.44	0.18	1.15	5.28	0.15	1.16	5.00	0.18	1.28	5.45	0.18	1.29	5.52	0.19	1.25	5.34	0.19	1.29	5.63
XCLIP	1.92	6.96	16.70	2.12	8.05	17.49	1.67	6.15	14.29	1.31	5.92	14.78	1.19	5.41	13.98	3.06	8.11	17.00	1.49	6.11	15.06
AIGVDet	9.13	17.25	24.48	9.35	18.21	25.10	9.96	18.65	26.96	9.99	16.43	24.12	9.49	16.76	24.01	9.67	17.15	24.03	8.27	16.86	23.54
DeMamba	0.81	4.98	14.37	1.29	4.80	13.05	0.82	3.21	9.86	1.07	4.56	13.55	1.03	4.81	13.28	1.65	6.09	15.83	0.99	5.04	15.05
ReStraV	35.67	61.89	77.75	44.46	72.60	82.99	54.73	**75.13**	83.33	32.66	55.99	75.08	24.42	47.32	71.83	30.25	54.85	75.48	38.50	60.50	77.55
D3	2.28	37.42	70.59	1.53	28.60	66.87	1.89	23.57	63.79	2.16	38.27	71.03	2.78	39.84	71.73	2.50	40.98	74.70	2.09	37.65	70.32
SPLIT	**74.45**	**83.58**	**89.70**	**62.54**	**77.49**	**88.71**	**55.42**	73.50	**88.47**	**70.45**	**80.89**	**88.83**	**64.40**	**77.11**	**87.12**	**75.73**	**82.79**	**88.84**	**66.92**	**80.42**	**88.52**

4.4 Inference Efficiency

Inference time on 1,000 FakeParts T2V samples is reported in Fig. 3. Because feature encoding dominates the computational cost, our lightweight patch-level TTR and LSMI computations add negligible overhead to the frozen encoder forward pass. When paired with the same backbone, SPLIT matches D3's runtime while boosting the FakeParts Overall Fake Recall at FPR $= 0.1\%$ from 2.28% to 74.45% with XCLIP-B/16. SPLIT delivers substantially stronger detection under service-aligned thresholds without sacrificing practical throughput.

4.5 Robustness to Post-Processing

We evaluate robustness on FakeParts Overall under common post-processing operations—Gaussian blur ($\sigma = 1, 2$), JPEG compression ($q = 90, 80$), and geometric axis flips (horizontal x, vertical y)—and report Fake Recall at service-aligned FPR targets in Tab. 5.

Across perturbations and operating points, SPLIT remains the best method by a wide margin. Blur is the most disruptive: at FPR $= 0.1\%$, Fake Recall drops from 74.45% to 55.42% under $\sigma = 2$, yet still exceeds the best supervised baseline (ReStraV, 54.73%). JPEG compression is substantially less harmful (SPLIT retains 70.45% at $q = 90$ and 64.40% at $q = 80$), suggesting our patch-level spatiotemporal cues do not depend on fragile high-frequency pixel artifacts. Geometric flips have minimal impact: horizontal flipping leaves performance essentially unchanged, while vertical flipping causes only a modest decrease (66.92% at FPR $= 0.1\%$), consistent with the fact that our scoring aggregates patch-wise evidence without relying on absolute spatial orientation.

At the looser FPR $= 5\%$ operating point, performance remains near-saturated and stable across all perturbations (87.12–89.70%), indicating that SPLIT maintains robust detection under realistic video transformations while preserving a substantial advantage in the deployment-critical ultra-low-FPR regime.

Table 6: Component ablation on FakeParts Overall. Fake Recall (%) is reported under cross-real threshold transfer at FPR $\in \{0.1\%, 1\%, 5\%\}$. Checkmarks denote active components.

Patch-level TTR		LSMI	FakeParts		
			0.1%	1%	5%
	✓		54.21	76.72	88.77
✓	✓		68.29	82.09	89.37
✓		✓	8.04	14.63	25.07
✓	✓	✓	**74.45**	**83.58**	**89.70**

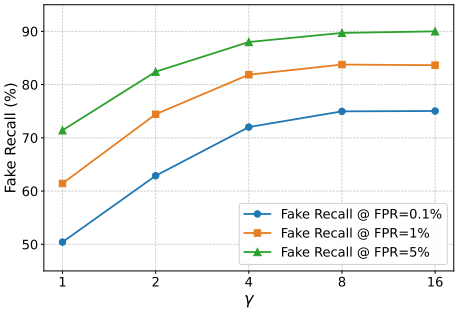

Fig. 4: Effect of the gamma exponent γ on FakeParts Overall Fake Recall (%).

5 Ablation Studies

5.1 Contribution of Each Component

We isolate the contribution of each SPLIT component on FakeParts Overall under cross-real threshold transfer, as summarized in Tab. 6. Replacing global `[CLS]` embeddings with patch-level tokens for TTR yields the single largest improvement, boosting Fake Recall by +14.08 points at FPR = 0.1% (from 54.21% to 68.29%), confirming that patch-level granularity is essential for detecting partial manipulations. LSMI alone is a weaker standalone signal, yet it captures complementary spatial-coherence evidence: fusing Patch-level TTR with LSMI achieves the best performance across all operating points, with the largest margin at the strictest threshold (+6.16 over Patch-level TTR alone). These results confirm that patch-level temporal roughness is the primary discriminative cue, while LSMI provides additional spatial incoherence information that further strengthens separation under strict real-only calibration. We include an extended analysis of TTR step size and qualitative diagnostics in the supplementary material.

5.2 Effect of Gamma Exponent γ

As shown in Fig. 4, we examine the influence of the gamma exponent γ in the fusion rule $s(x) = \mathbf{TTR}^{\gamma} \cdot \mathbf{LSMI}$ and report FakeParts Overall Fake Recall under cross-real threshold transfer at FPR $\in \{0.1\%, 1\%, 5\%\}$. Increasing γ from 1 to 8 yields consistent and substantial gains across all operating points, confirming that gamma correction effectively amplifies subtle TTR differences to sharpen the separation between real and manipulated videos. Beyond $\gamma = 8$, performance saturates: $\gamma = 16$ offers negligible further improvement and can even produce marginal regression at certain FPR levels. We therefore adopt $\gamma = 8$ throughout. Additional sensitivity analyses on GenVideo and ViF-Bench show the same trend and are provided in the supplementary material.

Table 7: Effect of the vision encoder on Fake Recall (%).

Vision Encoder	Method	FakeParts			GenVideo			ViF-Bench		
		0.1%	1%	5%	0.1%	1%	5%	0.1%	1%	5%
CLIP-B/16	D3	1.66	39.79	69.94	2.37	58.19	86.13	0.74	48.62	88.78
	SPLIT	**71.47**	**81.56**	**89.39**	**85.87**	**93.14**	**97.81**	**85.29**	**94.86**	**98.38**
CLIP-B/32	D3	0.64	31.17	67.19	0.14	47.89	85.39	0.50	45.53	89.50
	SPLIT	**47.37**	**72.01**	**88.56**	**63.55**	**82.33**	**95.54**	**54.76**	**87.22**	**98.12**
DINOv2-B	D3	4.00	36.41	61.49	16.32	64.01	84.08	3.97	54.90	83.40
	SPLIT	**47.77**	**70.73**	**87.63**	**68.03**	**84.18**	**95.50**	**56.64**	**86.52**	**97.98**
DINOv2-L	D3	4.12	28.24	54.54	15.99	55.26	78.43	6.21	44.00	75.86
	SPLIT	**49.06**	**68.25**	**86.45**	**68.71**	**83.32**	**95.09**	**55.95**	**82.64**	**97.47**
XCLIP-B/16	D3	2.28	37.42	70.59	5.47	66.50	90.52	0.97	46.77	87.77
	SPLIT	**74.45**	**83.58**	**89.70**	**85.16**	**93.09**	**97.35**	**84.52**	**95.08**	**98.46**
XCLIP-B/32	D3	0.67	27.63	65.75	0.94	54.34	87.43	0.53	34.43	81.72
	SPLIT	**60.23**	**77.34**	**89.07**	**70.19**	**85.17**	**95.25**	**68.25**	**91.36**	**98.34**
ResNet18	D3	1.93	34.70	68.55	1.68	53.89	85.89	0.94	43.73	87.99
	SPLIT	**61.23**	**77.87**	**88.01**	**76.10**	**89.38**	**96.57**	**73.53**	**89.35**	**96.65**
VGG16	D3	4.32	51.01	74.85	2.48	68.33	87.57	1.67	65.61	92.85
	SPLIT	**58.75**	**73.70**	**85.18**	**78.62**	**88.77**	**95.63**	**70.59**	**85.53**	**94.63**
EfficientNet-b4	D3	1.41	21.06	51.74	3.64	40.52	72.99	0.97	26.81	66.49
	SPLIT	**49.03**	**73.38**	**87.46**	**60.25**	**81.24**	**93.02**	**56.74**	**85.39**	**95.99**
MobileNet-v3	D3	1.77	31.60	62.67	1.60	50.15	80.48	1.02	41.68	82.13
	SPLIT	**49.81**	**71.03**	**86.02**	**67.15**	**83.76**	**94.42**	**61.07**	**84.57**	**95.52**

5.3 Effect of the Vision Encoder

We ablate the frozen vision encoder used for patch token extraction, as summarized in Tab. 7, comparing SPLIT against D3 under cross-real threshold transfer across all three benchmarks. SPLIT consistently and substantially outperforms D3 with every tested backbone, spanning transformer (CLIP, XCLIP, DINOv2) and CNN (ResNet18, VGG16, EfficientNet-b4, MobileNet-v3) architectures, with margins peaking at the deployment-critical FPR = 0.1%. This universal improvement confirms that the gains stem from patch-level spatiotemporal scoring rather than from any particular pretrained representation.

Encoder choice modulates performance but does not alter the conclusion. Transformer backbones with finer spatial granularity yield the strongest results, B/16 variants consistently outperform their B/32 counterparts, consistent with our hypothesis that localized artifacts are better captured at higher patch resolution. Lightweight CNN encoders still deliver large improvements over D3, demonstrating that the detection signal is encoder-agnostic. These results jointly show that replacing D3's global clip-level aggregation with localized TTR and LSMI measurement is the dominant driver of performance, while stronger vision tokenizers further amplify this advantage under strict real-only calibration.

6 Conclusion

We presented SPLIT, a training-free detector for AI-generated and partially edited videos that measures localized manipulation evidence on patch tokens from a frozen vision encoder through two complementary signals—Two-step Temporal Roughness (TTR) and Local Spatial Motion Incoherence (LSMI)—fused via gamma correction without any learned parameters. Together with a service-aligned evaluation protocol based on Fake Recall at fixed FPR with real-only threshold calibration and cross-real threshold transfer, we evaluated SPLIT across three benchmarks spanning partial edits (FakeParts), near-domain generators (GenVideo), and out-of-distribution 2024–2025 generators (ViF-Bench). SPLIT consistently achieved the highest Fake Recall at FPR targets as strict as 0.1%, substantially outperforming both supervised and training-free baselines while remaining robust to post-processing with negligible overhead. These results demonstrate that localized spatiotemporal incoherence on frozen patch tokens provides a strong and generalizable signal for reliable video manipulation detection under practical deployment constraints. We advocate reporting service-aligned operating points, such as Fake Recall at $\alpha = 0.1\%$ with real-only threshold calibration and cross-real threshold transfer, as a key metric for real-world AI-generated video detection.

Acknowledgements

This work was supported by Institute of Information & Communications Technology Planning & Evaluation (IITP) grant funded by the Korea government (MSIT) (No. RS-2019-II191906, Artificial Intelligence Graduate School Program (POSTECH)).

References

1. Bai, J., Lin, M., Cao, G., Lou, Z.: AI-Generated Video Detection via Spatial-Temporal Anomaly Learning. In: Lin, Z., Cheng, M.M., He, R., Ubul, K., Silamu, W., Zha, H., Zhou, J., Liu, C.L. (eds.) Pattern Recognition and Computer Vision. pp. 460–470. Springer Nature (2025). https://doi.org/10.1007/978-981-97-8792-0_32
2. Blattmann, A., Dockhorn, T., Kulal, S., Mendelevitch, D., Kilian, M., Lorenz, D., Levi, Y., English, Z., Voleti, V., Letts, A., Jampani, V., Rombach, R.: Stable Video Diffusion: Scaling Latent Video Diffusion Models to Large Datasets (2023). https://doi.org/10.48550/arXiv.2311.15127, http://arxiv.org/abs/2311.15127
3. Brooks, T., Peebles, B., Holmes, C., DePue, W., Guo, Y., Jing, L., Schnurr, D., Taylor, J., Luhman, T., Luhman, E., et al.: Video generation models as world simulators. OpenAI Blog **1**(8), 1 (2024)
4. Chen, G., Lin, D., Yang, J., Lin, C., Zhu, J., Fan, M., Zhang, H., Chen, S., Chen, Z., Ma, C., Xiong, W., Wang, W., Pang, N., Kang, K., Xu, Z., Jin, Y., Liang, Y., Song, Y., Zhao, P., Xu, B., Qiu, D., Li, D., Fei, Z., Li, Y., Zhou, Y.: Skyreels-v2: Infinite-length film generative model (2025), https://arxiv.org/abs/2504.13074

5. Chen, H., Xia, M., He, Y., Zhang, Y., Cun, X., Yang, S., Xing, J., Liu, Y., Chen, Q., Wang, X., Weng, C., Shan, Y.: Videocrafter1: Open diffusion models for high-quality video generation (2023), https://arxiv.org/abs/2310.19512
6. Chen, H., Zhang, Y., Cun, X., Xia, M., Wang, X., Weng, C., Shan, Y.: VideoCrafter2: Overcoming Data Limitations for High-Quality Video Diffusion Models. In: 2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). pp. 7310–7320 (2024). https://doi.org/10.1109/CVPR52733.2024.00698
7. Chen, H., Hong, Y., Huang, Z., Xu, Z., Gu, Z., Li, Y., Lan, J., Zhu, H., Zhang, J., Wang, W., Li, H.: DeMamba: AI-Generated Video Detection on Million-Scale GenVideo Benchmark (2024). https://doi.org/10.48550/arXiv.2405.19707
8. Coccomini, D.A., Zilos, G.K., Amato, G., Caldelli, R., Falchi, F., Papadopoulos, S., Gennaro, C.: MINTIME: Multi-Identity Size-Invariant Video Deepfake Detection. IEEE Transactions on Information Forensics and Security **19**, 6084–6096 (2024). https://doi.org/10.1109/TIFS.2024.3409054
9. Esser, P., Chiu, J., Atighehchian, P., Granskog, J., Germanidis, A.: Structure and content-guided video synthesis with diffusion models. In: Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV). pp. 7346–7356 (October 2023)
10. Gu, A., Dao, T.: Mamba: Linear-Time Sequence Modeling with Selective State Spaces. In: First Conference on Language Modeling (2024)
11. Gu, Z., Chen, Y., Yao, T., Ding, S., Li, J., Huang, F., Ma, L.: Spatiotemporal Inconsistency Learning for DeepFake Video Detection. In: Proceedings of the 29th ACM International Conference on Multimedia. pp. 3473–3481. ACM (2021). https://doi.org/10.1145/3474085.3475508
12. Güera, D., Delp, E.J.: Deepfake Video Detection Using Recurrent Neural Networks. In: 2018 15th IEEE International Conference on Advanced Video and Signal Based Surveillance (AVSS). pp. 1–6 (2018). https://doi.org/10.1109/AVSS.2018.8639163
13. HaCohen, Y., Chiprut, N., Brazowski, B., Shalem, D., Moshe, D., Richardson, E., Levin, E., Shiran, G., Zabari, N., Gordon, O., Panet, P., Weissbuch, S., Kulikov, V., Bitterman, Y., Melumian, Z., Bibi, O.: Ltx-video: Realtime video latent diffusion (2024), https://arxiv.org/abs/2501.00103
14. Hand, D.J.: Measuring classifier performance: A coherent alternative to the area under the ROC curve. Machine Learning **77**(1), 103–123 (Oct 2009). https://doi.org/10.1007/s10994-009-5119-5
15. He, K., Zhang, X., Ren, S., Sun, J.: Deep Residual Learning for Image Recognition. In: CVPR. pp. 770–778 (2016)
16. Howard, A., Sandler, M., Chu, G., Chen, L.C., Chen, B., Tan, M., Wang, W., Zhu, Y., Pang, R., Vasudevan, V., Le, Q.V., Adam, H.: Searching for MobileNetV3. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 1314–1324 (2019)
17. Internò, C., Geirhos, R., Olhofer, M., Liu, S., Hammer, B., Klindt, D.: Ai-generated video detection via perceptual straightening. In: Belgrave, D., Zhang, C., Lin, H., Pascanu, R., Koniusz, P., Ghassemi, M., Chen, N. (eds.) Advances in Neural Information Processing Systems. vol. 38, pp. 20672–20705. Curran Associates, Inc. (2025), https://proceedings.neurips.cc/paper_files/paper/2025/file/1d9a43752c2819e03967c5c1b708169c-Paper-Conference.pdf
18. Kling: Kling AI: Next-Generation AI Creative Studio (2025), https://klingai.com/global/

19. Kong, W., Tian, Q., Zhang, Z., Min, R., Dai, Z., Zhou, J., Xiong, J., Li, X., Wu, B., Zhang, J., Wu, K., Lin, Q., Yuan, J., Long, Y., Wang, A., Wang, A., Li, C., Huang, D., Yang, F., Tan, H., Wang, H., Song, J., Bai, J., Wu, J., Xue, J., Wang, J., Wang, K., Liu, M., Li, P., Li, S., Wang, W., Yu, W., Deng, X., Li, Y., Chen, Y., Cui, Y., Peng, Y., Yu, Z., He, Z., Xu, Z., Zhou, Z., Xu, Z., Tao, Y., Lu, Q., Liu, S., Zhou, D., Wang, H., Yang, Y., Wang, D., Liu, Y., Jiang, J., Zhong, C.: Hunyuanvideo: A systematic framework for large video generative models (2025), https://arxiv.org/abs/2412.03603
20. Li, Y., Zheng, W., Zhang, Y., Sun, R., Zheng, Y., Chen, L., Zhou, J., Lu, J.: Skyra: AI-Generated Video Detection via Grounded Artifact Reasoning (2025). https://doi.org/10.48550/arXiv.2512.15693
21. Li, Y., Lyu, S.: Exposing DeepFake Videos By Detecting Face Warping Artifacts. In: CVPRW. pp. 46–52 (2019)
22. Liu, Y., Cun, X., Liu, X., Wang, X., Zhang, Y., Chen, H., Liu, Y., Zeng, T., Chan, R., Shan, Y.: EvalCrafter: Benchmarking and Evaluating Large Video Generation Models. In: CVPR. pp. 22139–22149 (2024)
23. Liu, Z., Gabetni, F., Sani, A.H., Wang, X., Daiboo, S., Brison, G., Franchi, G., Kalogeiton, V.: FakeParts: A New Family of AI-Generated DeepFakes (2025). https://doi.org/10.48550/arXiv.2508.21052
24. MiniMax: Hailuo 02 - #2 Global AI Video Generation Model. https://hailuo-02.com/ (2025)
25. Moonvalley: Moonvalley - the imagination research company. https://www.moonvalley.com/ (2022)
26. Morph Studio: Morph Studio - AI Image & Video Generator: Free Art Creator & Editor (2023), https://www.morphstudio.com/
27. Mullan, J., Crawbuck, D., Sastry, A.: Hotshot-XL (Oct 2023), https://github.com/hotshotco/hotshot-xl
28. Ni, B., Peng, H., Chen, M., Zhang, S., Meng, G., Fu, J., Xiang, S., Ling, H.: Expanding Language-Image Pretrained Models for General Video Recognition. In: Avidan, S., Brostow, G., Cissé, M., Farinella, G.M., Hassner, T. (eds.) Computer Vision – ECCV 2022. pp. 1–18. Springer Nature Switzerland (2022). https://doi.org/10.1007/978-3-031-19772-7_1
29. Ni, Z., Yan, Q., Huang, M., Yuan, T., Tang, Y., Hu, H., Chen, X., Wang, Y.: GenVidBench: A 6-Million Benchmark for AI-Generated Video Detection (2025). https://doi.org/10.48550/arXiv.2501.11340
30. OpenAI: Sora 2 is here (2025), https://openai.com/index/sora-2/
31. Oquab, M., Darcet, T., Moutakanni, T., Vo, H.V., Szafraniec, M., Khalidov, V., Fernandez, P., Haziza, D., Massa, F., El-Nouby, A., Assran, M., Ballas, N., Galuba, W., Howes, R., Huang, P.Y., Li, S.W., Misra, I., Rabbat, M., Sharma, V., Synnaeve, G., Xu, H., Jegou, H., Mairal, J., Labatut, P., Joulin, A., Bojanowski, P.: DINOv2: Learning Robust Visual Features without Supervision. TMLR (2023), https://openreview.net/forum?id=a68SUt6zFt
32. Pika Art: Pika art. https://pika.art/ (2025)
33. PixVerse AI: PixVerse | Create Amazing AI Videos from Text & Photos with AI Video Generator. https://app.pixverse.ai (2025)
34. Radford, A., Kim, J.W., Hallacy, C., Ramesh, A., Goh, G., Agarwal, S., Sastry, G., Askell, A., Mishkin, P., Clark, J., Krueger, G., Sutskever, I.: Learning Transferable Visual Models From Natural Language Supervision. In: Proceedings of the 38th International Conference on Machine Learning. pp. 8748–8763. PMLR (2021)
35. Runway: Runway Research | Introducing Runway Gen-4. https://runwayml.com/research/introducing-runway-gen-4 (2025)

36. Simonyan, K., Zisserman, A.: Very Deep Convolutional Networks for Large-Scale Image Recognition (2015). https://doi.org/10.48550/arXiv.1409.1556, http://arxiv.org/abs/1409.1556
37. Tan, C., Liu, H., Zhao, Y., Wei, S., Gu, G., Liu, P., Wei, Y.: Rethinking the Up-Sampling Operations in CNN-Based Generative Network for Generalizable Deepfake Detection. In: 2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). pp. 28130–28139 (2024). https://doi.org/10.1109/CVPR52733.2024.02657
38. Tan, M., Le, Q.: EfficientNet: Rethinking Model Scaling for Convolutional Neural Networks. In: Proceedings of the 36th International Conference on Machine Learning. pp. 6105–6114. PMLR (2019)
39. Teed, Z., Deng, J.: RAFT: Recurrent All-Pairs Field Transforms for Optical Flow. In: Vedaldi, A., Bischof, H., Brox, T., Frahm, J.M. (eds.) Computer Vision – ECCV 2020. pp. 402–419. Springer International Publishing, Cham (2020). https://doi.org/10.1007/978-3-030-58536-5_24
40. Wan, T., Wang, A., Ai, B., Wen, B., Mao, C., Xie, C.W., Chen, D., Yu, F., Zhao, H., Yang, J., Zeng, J., Wang, J., Zhang, J., Zhou, J., Wang, J., Chen, J., Zhu, K., Zhao, K., Yan, K., Huang, L., Feng, M., Zhang, N., Li, P., Wu, P., Chu, R., Feng, R., Zhang, S., Sun, S., Fang, T., Wang, T., Gui, T., Weng, T., Shen, T., Lin, W., Wang, W., Wang, W., Zhou, W., Wang, W., Shen, W., Yu, W., Shi, X., Huang, X., Xu, X., Kou, Y., Lv, Y., Li, Y., Liu, Y., Wang, Y., Zhang, Y., Huang, Y., Li, Y., Wu, Y., Liu, Y., Pan, Y., Zheng, Y., Hong, Y., Shi, Y., Feng, Y., Jiang, Z., Han, Z., Wu, Z.F., Liu, Z.: Wan: Open and Advanced Large-Scale Video Generative Models (2025). https://doi.org/10.48550/arXiv.2503.20314
41. Wang, J., Yuan, H., Chen, D., Zhang, Y., Wang, X., Zhang, S.: Modelscope text-to-video technical report (2023), https://arxiv.org/abs/2308.06571
42. Wang, W., Yang, Y.: VidProM: A Million-scale Real Prompt-Gallery Dataset for Text-to-Video Diffusion Models. NeurIPS **37**, 65618–65642 (2024). https://doi.org/10.52202/079017-2096
43. Wang, Y., Chen, X., Ma, X., Zhou, S., Huang, Z., Wang, Y., Yang, C., He, Y., Yu, J., Yang, P., Guo, Y., Wu, T., Si, C., Jiang, Y., Chen, C., Loy, C.C., Dai, B., Lin, D., Qiao, Y., Liu, Z.: LaVie: High-Quality Video Generation with Cascaded Latent Diffusion Models. International Journal of Computer Vision **133**(5), 3059–3078 (May 2025). https://doi.org/10.1007/s11263-024-02295-1
44. Wu, J., Li, X., Si, C., Zhou, S., Yang, J., Zhang, J., Li, Y., Chen, K., Tong, Y., Liu, Z., Loy, C.C.: Towards Language-Driven Video Inpainting via Multimodal Large Language Models. In: CVPR. pp. 12501–12511 (2024)
45. Xiong, X., Patel, P., Fan, Q., Wadhwa, A., Selvam, S., Guo, X., Qi, L., Liu, X., Sengupta, R.: Talkingheadbench: A multi-modal benchmark & analysis of talking-head deepfake detection. In: Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision (WACV). pp. 4139–4149 (March 2026)
46. Xu, J., Mei, T., Yao, T., Rui, Y.: MSR-VTT: A Large Video Description Dataset for Bridging Video and Language. In: CVPR. pp. 5288–5296 (2016)
47. Xu, Y., Liang, J., Jia, G., Yang, Z., Zhang, Y., He, R.: TALL: Thumbnail Layout for Deepfake Video Detection. In: 2023 IEEE/CVF International Conference on Computer Vision (ICCV). pp. 22601–22611. IEEE (2023). https://doi.org/10.1109/ICCV51070.2023.02071
48. Yang, Z., Teng, J., Zheng, W., Ding, M., Huang, S., Xu, J., Yang, Y., Hong, W., Zhang, X., Feng, G., Yin, D., Yuxuan.Zhang, Wang, W., Cheng, Y., Xu, B., Gu, X., Dong, Y., Tang, J.: Cogvideox: Text-to-video diffusion models with an expert

transformer. In: The Thirteenth International Conference on Learning Representations (2025), https://openreview.net/forum?id=LQzN6TRFg9
49. Zhang, D.J., Wu, J.Z., Liu, J.W., Zhao, R., Ran, L., Gu, Y., Gao, D., Shou, M.Z.: Show-1: Marrying Pixel and Latent Diffusion Models for Text-to-Video Generation. International Journal of Computer Vision **133**(4), 1879–1893 (Apr 2025). https://doi.org/10.1007/s11263-024-02271-9
50. Zheng, C., Lin, C., Zhao, Z., Wang, H., Guo, X., Liu, S., Shen, C.: Breaking Semantic Artifacts for Generalized AI-generated Image Detection. NeurIPS **37**, 59570–59596 (2024). https://doi.org/10.52202/079017-1903
51. Zheng, C., Suo, R., Lin, C., Zhao, Z., Yang, L., Liu, S., Yang, M., Wang, C., Shen, C.: D3: Training-Free AI-Generated Video Detection Using Second-Order Features. In: ICCV. pp. 12852–12862 (2025)
52. Zheng, Y., Bao, J., Chen, D., Zeng, M., Wen, F.: Exploring Temporal Coherence for More General Video Face Forgery Detection. In: 2021 IEEE/CVF International Conference on Computer Vision (ICCV). pp. 15024–15034 (2021). https://doi.org/10.1109/ICCV48922.2021.01477