LEARNING RECONSTRUCTION ERRORS AS SPATIOTEMPORAL EVIDENCE FOR AI-GENERATED VIDEO DETECTION

Anonymous authors
Paper under double-blind review

ABSTRACT

The proliferation of AI-generated videos has raised growing concerns about misinformation, fraud, and potential societal risks, creating an urgent need for detection. Reconstruction-error-based methods have shown strong empirical performance for AI-generated image detection by exploiting discrepancies between an input and its reconstructed version. Their extension to video detection, however, faces two challenges: (i) a fixed reconstructor is not optimized for the detection objective and may yield weak discriminative signals; and (ii) an additional raw-video feature stream is often introduced to model temporal dependencies, increasing architectural overhead. To address these challenges, we develop a detector that learns and exploits reconstruction errors as a unified source of spatiotemporal evidence. First, we jointly optimize a reconstructor and the downstream classifier to produce reconstruction errors that are more informative for detecting AI-generated videos. Second, we characterize temporal information using relative second-order temporal differences of the reconstruction errors. We further show that a 3D convolution over three consecutive errors can represent this relative second-order feature without explicitly constructing it, while simultaneously integrating spatial evidence. Extensive experiments on over 30 existing AI-video generators demonstrate competitive performance against existing training-based detectors.

1 INTRODUCTION

Nowadays, state-of-the-art video generation models, such as the closed-source models Sora and Veo2 and the open-source models CogVideoX and LTX-Video (HaCohen et al., 2024; Yang et al., 2024), can produce highly realistic videos when users provide text descriptions or input images. On the one hand, this makes video generation more convenient and thus enriches entertainment in ordinary life and the video/cinema business. On the other hand, misuse of video generation models can yield videos intended to spread misinformation and facilitate fraud, causing various risks to society and businesses. To avoid these potential risks, there is an important and increasingly urgent need to identify videos generated by AI.

Early AI-video detection methods (Qian et al., 2020; Frank et al., 2020; Tan et al., 2024; Yan et al., 2024) focus on DeepFake-style facial forgeries. More recently, state-of-the-art video generators can produce videos with diverse scenarios (Liu et al., 2025; Ma et al., 2026; Ni et al., 2026), making detection more challenging. Various methods for detecting such diverse AI-generated videos have accordingly been developed (Chen et al., 2024b; Liu et al., 2024a; Chen et al., 2026b; Park et al., 2026). Reconstruction-based methods are among the earliest techniques used to detect AI-generated videos. Specifically, a reconstruction-based method (a) reconstructs the video to be detected using a generative model, (b) computes the difference between the original and reconstructed videos, and (c) classifies the video as real or fake based on that difference. Reconstruction-based methods have demonstrated strong empirical performance in detecting AI-generated images (Cao et al., 2022; Wang et al., 2023; Luo et al., 2024; Chu et al., 2025; Qi et al., 2026), and it is natural to extend this technique to AI-generated video detection.

However, reconstruction-based AI-video detection methods face two challenges. First, existing reconstruction-error-based video detectors often exploit the errors primarily as frame-wise spatial

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.

cues while underusing the temporal information in the errors (Liu et al., 2024a; Song et al., 2024; Chen et al., 2026b). As temporal information is important for detecting AI-generated videos (Chen et al., 2024b; Zheng et al., 2025), existing methods introduce additional features from the raw video to supplement the information provided by reconstruction errors. Second, when real and generated videos produce similar reconstruction-error patterns, the errors provide only a weak discriminative signal, limiting their predictive power. These observations raise a central question:

Can we produce more discriminative reconstruction errors and exploit them as a unified source of spatial and temporal evidence for AI-generated video detection?

To answer this question, we develop a detector that learns and exploits reconstruction errors within a single stream. First, we jointly optimize a reconstructor and the downstream classifier to produce reconstruction errors that are more informative for detecting AI-generated videos. Second, we characterize a temporal detection signal using second-order differences of the reconstruction errors. Furthermore, we show that a 3D convolution over three consecutive reconstruction errors can represent this second-order feature without explicitly constructing it, while simultaneously integrating spatial evidence. Empirically, we test the proposal on several benchmark datasets covering more than 30 existing AI-video generators. The evaluated generators include both text-to-video and image-to-video models. Across these experiments, even when using only reconstruction errors, we find that the proposal typically achieves the highest detection performance when compared with more than 10 baselines.

2 Related Work

AI Video Generation. Modern AI video generation is largely built upon three broad methodological paradigms: diffusion-based, flow-matching, and autoregressive models. Diffusion-based generative frameworks (Ho et al., 2022b; Blattmann et al., 2023b) are widely used for video generation. Building on these frameworks, subsequent studies have developed models for high-quality text-to-video and image-to-video generation (He et al., 2022; Ho et al., 2022a; Singer et al., 2022; Blattmann et al., 2023a; Guo et al., 2023; Chen et al., 2024a; Wang et al., 2025). Furthermore, diffusion transformers provide a scalable backbone for image generation (Peebles & Xie, 2023). Related Transformer-based architectures have also been developed for text-to-image generation (Esser et al., 2024). Subsequent studies have extended and refined Transformer-based video generation (Gupta et al., 2024; Yang et al., 2024; Zheng et al., 2024; Ma et al., 2025; Zhao et al., 2025; Chen et al., 2026a; Jia et al., 2026). More recently, flow matching (Lipman et al., 2023) has emerged as a new approach for video generation (Jin et al., 2025; Ifriqi et al., 2026; Kodaira et al., 2026; Pan et al., 2026) and has shown competitive performance. Autoregressive methods (Yan et al., 2021; Villegas et al., 2022; Kondratyuk et al., 2024) generate video tokens or segments conditioned on previously generated content. Recent autoregressive diffusion models further support long-form and streaming video generation (Yang et al., 2025; Yin et al., 2025; Huang et al., 2026; Liu et al., 2026). Such diversity in generative formulations and architectures may produce heterogeneous spatiotemporal artifacts, posing challenges for generalization to unseen video generators.

AI-generated Video Detection. Existing AI-generated video detectors are mainly training-based methods that leverage datasets containing real and AI-generated videos to train classifiers (Corvi et al., 2026; Li et al., 2026b). Training-based methods can be categorized into several subtypes. Inconsistency-based detection methods typically use pretrained CLIP-type models (Radford et al., 2021; Ni et al., 2022) to extract features, which are then used to train a binary classifier to identify discrepancies between AI-generated and real videos in the spatial domain, the temporal domain, or both (Song et al., 2024; Chen et al., 2024b; Ma et al., 2024; Xiang et al., 2026; Wei et al., 2026). Furthermore, methods in this category may also incorporate reconstruction errors, the differences between observed videos and their reconstructed counterparts, as additional features for training the classifier (Liu et al., 2024a; Chen et al., 2026b; Du et al., 2026). Reasoning-based methods (Wen et al., 2025; Li et al., 2026a; Park et al., 2026; Tan et al., 2026) exploit high-capacity pretrained multimodal large language models and use post-training techniques, such as supervised fine-tuning and reinforcement learning, to detect forgeries and provide explanations. Physics-driven methods (Internò et al., 2025; He et al., 2026; Zhang et al., 2025) identify AI-generated videos by detecting deviations from natural motion. Finally, although most methods are training-based, training-

free methods exploit second-order features, spatiotemporal likelihoods, or temporal over-coherence for detection (Zheng et al., 2025; Ben Hayun et al., 2026; Brokman et al., 2026).

Reconstruction-Based AI-Generated Image/Video Detection. Diffusion Reconstruction (DIRE)-based detection exploits discrepancies between an input and its AI-reconstructed counterpart as forensic evidence. Wang et al. (2023) found that real and synthetic images exhibit systematically different DIRE, and thus, DIRE-based detection performs well in AI-generated image detection. Motivated by the success of DIRE, Ricker et al. (2024); Luo et al. (2024) proposed fast variants of DIRE to make it more practical to use. Furthermore, new methods enrich the reconstruction signal to enhance the detection power through diffusion inversion (Cazenavette et al., 2024), semantically guided image reconstruction (Jin et al., 2026; Yu et al., 2024), frequency decomposition (Chu et al., 2025), structural edits in reconstruction error (Jiang et al., 2026b), the second-order changes in reconstruction error (Qi et al., 2026), and correction of instance-specific biases in reconstruction error (Choi et al., 2026). Reconstruction-based methods are currently also among the important baselines in video detection (Liu et al., 2024a; Song et al., 2024; Chen et al., 2026b; Du et al., 2026).

3 Preliminaries

In this section, we review two approaches to AI-generated video detection that motivate our method: reconstruction-based detection (Section 3.1) and the second-order difference method for detection (Section 3.2). Throughout the paper, we use $x \in \mathbb{R}^{T\times C\times H\times W}$ to denote a video where T is the number of frames, C is the number of channels, and H and W are the frame height and width, respectively. The t-th frame is denoted by $x_t \in \mathbb{R}^{C\times H\times W}$, for $t = 1, \ldots, T$.

3.1 Reconstruction-based method for AI-video Detection

In reconstruction-based detection, one critical component is the reconstructor $\mathcal{R}(\cdot)$ that maps an observed video sample x to a reconstructed video. In practice, $\mathcal{R}$ is typically implemented using an autoencoder (Song et al., 2024; Chen et al., 2026b), such as a vector-quantised variational autoencoder (VQ-VAE; Van Den Oord et al., 2017) or a Wavelet Flow VAE (WF-VAE; Li et al., 2025). Reconstruction-based detectors exploit the resulting error

$$x - \mathcal{R}(x) \in \mathbb{R}^{T\times C\times H\times W}$$

as forensic evidence for detecting AI-generated videos. The underlying empirical rationale is that real and generated videos can exhibit systematically different reconstruction errors (see Figure 1(a) for an illustration). The resulting error forms a temporal sequence of frame-wise spatial discrepancies. However, existing approaches often use it primarily as a spatial cue and introduce an additional raw-video branch to model temporal dependencies (Liu et al., 2024a; Song et al., 2024; Chen et al., 2026b). Specifically, the raw video and its reconstruction error are concatenated and fused via a dedicated fusion module, as shown in Figure 1(b). The fused information is then used to predict whether the video is real or fake.

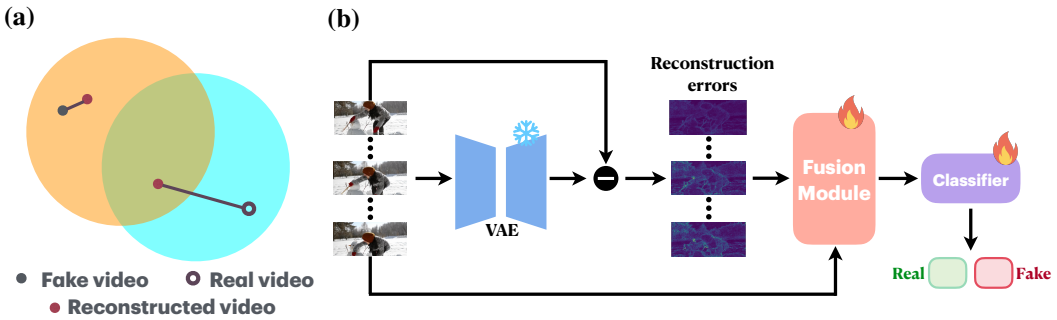

Figure 1: Panel (a). Graphical illustration of the rationale behind existing reconstruction-error-based methods. Real and generated videos can exhibit systematically different reconstruction errors. Panel (b). A high-level workflow of existing reconstruction-error-based methods. The VAE, fusion module, and classifier vary across methods. Some methods may incorporate additional information from large multimodal models.

3.2 SECOND-ORDER DIFFERENCE METHOD

As noted above, AI-generated and natural videos can exhibit different temporal behaviors, and existing studies have shown that such differences are informative for AI-generated video detection. Existing methods therefore seek to incorporate temporal information into a reconstruction-based detector. For example, Zheng et al. (2025) propose that second-order temporal features can distinguish real from AI-generated videos. Specifically, for a sequence of video frames $x_1, \ldots, x_T$, they define the approximation of the acceleration at time t as the second-order difference between raw video frames:

$$\Delta_t^2 x_t = (x_t - x_{t-1}) - (x_{t-1} - x_{t-2}) = x_t - 2x_{t-1} + x_{t-2}.$$

Based on the second-order difference, Zheng et al. (2025) design a training-free detection method that leverages the detection signal

$$\Delta_t^2 x_t^r - \Delta_t^2 x_t^f$$

where x^r and x^f denote a real video and an AI-generated video, respectively. Figure 2 provides an illustration of this signal for discriminating real and fake videos. For AI-generated videos, the second-order differences concentrate on the moving subject, while for real videos, the differences appear in both the moving subject and the background. This forms the practical principle for second-order difference-based detection.



Figure 2: Comparison of the second-order differences in real videos (left panel) and fake videos generated by Sora (middle panel) and AnimateDiff-v2 (right panel).

4 METHODOLOGY

4.1 MOTIVATION

This paper asks whether we can adaptively learn and exploit reconstruction errors as a unified source of spatial and temporal evidence for AI-generated video detection. We begin by discussing two limitations of existing implementations.

- *Discriminativeness of reconstruction errors*. Existing reconstruction-based detectors typically treat $\mathcal{R}$ as fixed (Wang et al., 2023; Ricker et al., 2024; Chen et al., 2026b; Du et al., 2026). However, as the fixed reconstructor is not optimized for the detection objective, the reconstruction errors may not be sufficiently distinguishable, particularly when real videos closely resemble AI-generated videos and vice versa (see Figure 4(a)). In such cases, the reconstruction errors provide only a weak discriminative signal to the subsequent classifier, limiting its predictive power.
- *Neglect of temporal signals in reconstruction errors*. Reconstruction errors form a temporal sequence across video frames, but existing methods often use them primarily as frame-wise spatial cues rather than directly modeling their temporal structure. To compensate for this underuse, these methods introduce an additional raw-video feature-extraction branch and a fusion module to incorporate temporal information, thereby increasing architectural and inference overhead.

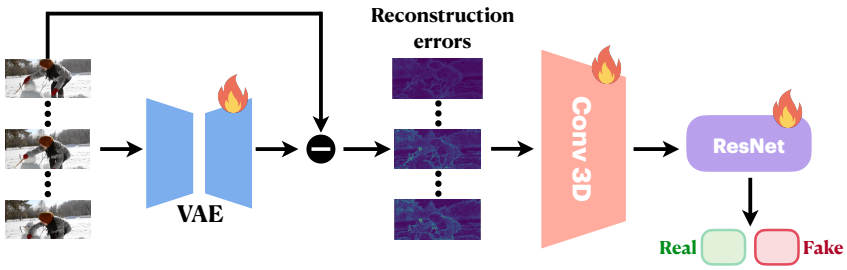

Figure 3: Overview of the proposed detection framework. An adaptive variational autoencoder produces reconstruction errors for detection. A 3D convolution integrates spatial evidence from these errors and can represent second-order temporal features without explicitly constructing them.

Together, these limitations concern how reconstruction errors are produced and used. Our method addresses both through (i) an adaptive reconstructor that produces more discriminative reconstruction errors and (ii) a 3D convolution that integrates spatial information and can represent second-order temporal features without explicit construction. Figure 3 gives an overview of the proposed method, with the details of (i) and (ii) presented in Sections 4.2 and 4.3, respectively.

4.2 Adaptive Reconstruction Errors

We propose learning adaptive reconstruction errors that provide a more discriminative signal for detecting AI-generated videos. Specifically, suppose that we have access to a dataset of real videos $\mathcal{D}_r$ and a dataset of AI-generated videos $\mathcal{D}_f$, both of which are readily available in practice. For instance, $\mathcal{D}_r$ can be a pre-collected video dataset, while $\mathcal{D}_f$ can be constructed by prompting a text-to-video model to generate videos. We parameterize the reconstructor as $\mathcal{R}_\psi(\cdot)$, where ψ comprises all learnable parameters in the reconstructor. For an input video x, the adaptive reconstructor produces the learned reconstruction error $x - \mathcal{R}_\psi(x) \in \mathbb{R}^{T\times C\times H\times W}$. We then feed the error into a classifier $\mathcal{C}_\phi$, where ϕ comprises all parameters in the classifier. We interpret $\mathcal{C}_\phi(x - \mathcal{R}_\psi(x)) \in (0, 1)$ as the predicted probability that x is AI-generated. We jointly learn ϕ and ψ by minimizing the binary cross-entropy loss

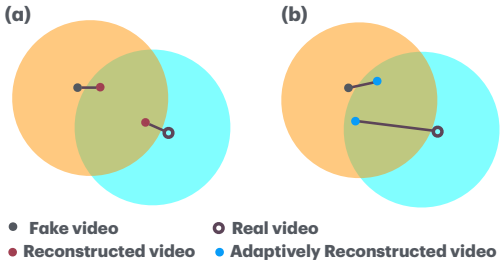

Figure 4: Panel (a): the reconstruction errors obtained with a fixed reconstructor. Panel (b): the reconstruction errors obtained with an adaptive reconstructor. Adaptation can make reconstruction-error patterns more distinguishable.

$$\mathcal{L}(\phi, \psi) = -\sum_{x\in\mathcal{D}_f} \log \mathcal{C}_\phi(x - \mathcal{R}_\psi(x)) - \sum_{x\in\mathcal{D}_r} \log\left[1 - \mathcal{C}_\phi(x - \mathcal{R}_\psi(x))\right].$$

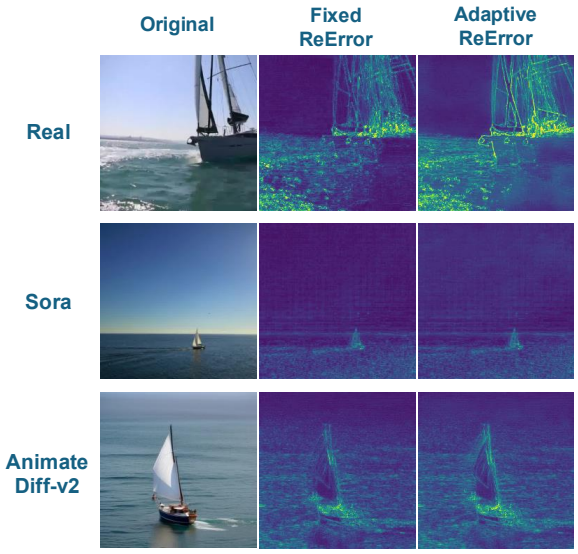

Figure 5: Graphical comparison of the reconstruction errors returned by a fixed reconstructor (middle column) and an adaptive reconstructor (right column) for the original videos (left column). The first row shows a real video, while the second and third rows show videos from Sora and AnimateDiff-v2, respectively.

To instantiate the learnable reconstructor in practice, we initialize $\mathcal{R}_\psi$ from a pretrained VAE and adapt it via low-rank adaptation (LoRA; Hu et al., 2022), freezing the pretrained weights and optimizing only the introduced LoRA parameters. The pretrained initialization provides a reconstruction-oriented starting point, while LoRA keeps the overall number of trainable parameters small. Since $\mathcal{C}_\phi(x - \mathcal{R}_\psi(x))$ maps an input video directly to a classification probability and both ϕ and ψ are optimized using the detection loss, the proposed detector can be viewed as an end-to-end error-learning model. Algebraically, $x - \mathcal{R}_\psi(x) = x + [-\mathcal{R}_\psi(x)]$ has an identity-plus-learned-branch form analogous to that used in residual learning (He et al., 2016). Such additive skip connections are also used in Transformer and YOLOv3 architectures (Vaswani et al., 2017; Redmon & Farhadi, 2018).

Figure 5 provides an illustrative example of the resulting change in reconstruction errors. For the frames shown, adaptive reconstruction changes the error pattern more visibly for the real video than for the two generated videos. In this example, the resulting error patterns are therefore easier to distinguish. The

figure provides qualitative evidence for this intended mechanism. The detection loss encourages the adaptive reconstructor to produce reconstruction-error patterns that better distinguish real and generated videos.

4.3 Implicit Reconstruction-Error-Based Temporal Signal and Its Fusion

We next describe how temporal information can be extracted from reconstruction errors. Recall that Section 3.2 detects AI videos use the fact that the second-order temporal difference $\Delta_t^2 x_t^r$ differs from $\Delta_t^2 x_t^f$. Motivated by the second-order temporal cue, we expect the reconstruction process to preserve the temporal behavior of AI-generated videos more closely than that of real videos. We thereby develop the concept of the *relative second-order temporal difference* (RSTD):

$$\Delta_t^2 x_t - \Delta_t^2 \widetilde{x}_t,$$

where $\widetilde{x}$ is the reconstructed video corresponding to x. Accordingly, for real videos, the second-order temporal difference $\Delta_t^2 x_t^r$ is expected to differ substantially from that of the reconstructed version, $\Delta_t^2 \widetilde{x}_t^r$. By contrast, for AI-generated videos, the second-order temporal difference $\Delta_t^2 x_t^f$ may be more similar to that of the reconstructed version, $\Delta_t^2 \widetilde{x}_t^f$. These contrasting relationships may lead to distinct RSTD patterns that provide a signal for detection. This hypothesis is supported by the evidence presented in Figure 6. Specifically, for a real video, the RSTD is distributed throughout the frame, whereas for a fake video, the RSTD appears locally around the moving object.

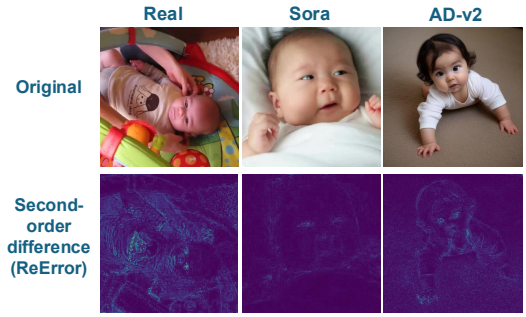

Figure 6: Comparison of the relative second-order temporal differences in real videos (left column) and fake videos generated by Sora (middle column) and AnimateDiff-v2 (right column).

More importantly, RSTD can be expressed as

$$\begin{aligned}\Delta_t^2 x_t - \Delta_t^2 \widetilde{x}_t &= \big((x_t - x_{t-1}) - (x_{t-1} - x_{t-2})\big) - \big((\widetilde{x}_t - \widetilde{x}_{t-1}) - (\widetilde{x}_{t-1} - \widetilde{x}_{t-2})\big), \\ &= (x_t - \widetilde{x}_t) - 2(x_{t-1} - \widetilde{x}_{t-1}) + (x_{t-2} - \widetilde{x}_{t-2}),\end{aligned} \tag{1}$$

where the reconstruction error at the t-th frame is denoted by $r_t = x_t - \widetilde{x}_t$. A notable observation from Equation (1) is that RSTD can be computed using only the reconstruction errors. Thus, it does not require the raw video as an additional input, thereby avoiding a separate raw-video branch in the classifier.

Next, we seek to fuse the temporal feature $r_t - 2r_{t-1} + r_{t-2}$ with the reconstruction errors r_t, r_{t-1}, and r_{t-2}. A direct fusion strategy is to concatenate them as $[r_t - 2r_{t-1} + r_{t-2}, r_t, r_{t-1}, r_{t-2}]$ and feed the resulting tensor into a 3D convolution (Ji et al., 2013) to extract joint spatiotemporal information. This strategy requires the model to process RSTD as an additional input slice, thereby increasing the computational cost. Instead, we show that its contribution can be absorbed into the parameters of a 3D convolution operating directly on the three original errors. The following proposition formalizes this.

Proposition 1. *For any 3D convolution* $\mathrm{Conv3D}_H$ *with four temporal kernel slices* H_0, H_1, H_2, H_3 *applied to* $[r_t - 2r_{t-1} + r_{t-2}, r_t, r_{t-1}, r_{t-2}]$*, there exists another 3D convolution* $\mathrm{Conv3D}_G$ *with three temporal kernel slices* G_0, G_1, G_2 *such that*

$$\mathrm{Conv3D}_H([r_t - 2r_{t-1} + r_{t-2}, r_t, r_{t-1}, r_{t-2}]) = \mathrm{Conv3D}_G([r_t, r_{t-1}, r_{t-2}]), \tag{2}$$

for all r_t, r_{t-1}, r_{t-2}*. Conversely, every 3D convolution of* $[r_t, r_{t-1}, r_{t-2}]$ *can be represented by a 3D convolution of* $[r_t - 2r_{t-1} + r_{t-2}, r_t, r_{t-1}, r_{t-2}]$.

Proposition 1 shows that explicitly constructing RSTD does not enlarge the available function class in this local convolutional setting: its contribution can be absorbed into the temporal kernel weights. In other words, directly feeding $[r_t, r_{t-1}, r_{t-2}]$ into a 3D convolution with three temporal kernel slices allows the model to exploit and fuse the spatiotemporal information in the reconstruction errors. Learning the three temporal kernel slices independently allows the detector to recover RSTD as a special case while adapting to other local temporal evidence under the detection objective. Therefore, we view RSTD as an interpretable reference for the learned temporal kernels rather than as a fixed stencil that must be explicitly imposed.

5 EXPERIMENTS

5.1 EXPERIMENTAL SETUP

Baselines. We compare our method with 11 representative AI-generated video detectors, most of which are training-based detectors, including ReConFuse (Chen et al., 2026b), two DIVID variants (Liu et al., 2024a) (DIVID-CNN and DIVID-LSTM), G2VD-CLIP (Du et al., 2026), DeMamba (Chen et al., 2024b), Ivy-xDetector (Jiang et al., 2026a), DeepTraceReward (Fu et al., 2025), and MM-Det (Song et al., 2024). We also compare it with three training-free detectors: D3 (Zheng et al., 2025), STALL (Ben Hayun et al., 2026), and Over-Coherence (Brokman et al., 2026).

Training setup. We train the proposed method on 139,002 class-balanced videos: 69,501 real videos from Kinetics-400 (Kay et al., 2017) and 69,501 synthetic videos generated in the VBench dataset (Huang et al., 2024; 2025). The checkpoint is selected using a hold-out validation set that includes fake videos from the MovieGen dataset (Polyak et al., 2024) and real videos from the Panda-70M dataset (Chen et al., 2024c). Baselines that do not provide checkpoints are retrained on the same dataset to ensure fairness.

Evaluation. We evaluate all the baselines on datasets that were not used during training or validation, including (i) a dataset mixed with fake videos from the EvalCrafter dataset (Liu et al., 2024b) and real videos from the MSR-VTT dataset (Xu et al., 2016), (ii) the GenVideo dataset that has both real and fake videos (Chen et al., 2024b), (iii) a dataset mixed with fake videos from the FakeParts dataset (Liu et al., 2025) and real videos from the TenKReal and ROVI datasets (Wu et al., 2024), and (iv) a dataset mixed with fake videos from VideoPhy (Bansal et al., 2025) and real videos from MSR-VTT. These datasets collectively over 30 synthetic-video subsets spanning text-to-video and image-to-video generation. We report video-level ROC-AUC and average precision (AP), treating generated videos as the positive class.

5.2 COMPARISON TO EXISTING METHODS

First, from the results on the EvalCrafter dataset (see Table 1), we can see that our method has the highest AUC for 10 of the 14 generators and still ranks second for 3 generators and third for ModelScope. This leads to the best overall AUC on the EvalCrafter dataset, which exceeds 0.99 on average. Notably, it uniformly outperforms the other reconstruction-based baselines, such as ReConFuse, MM-Det, and two DIVID variants. These results demonstrate the effectiveness of our proposal on leveraging the reconstruction errors. In terms of the AP results presented in Table A1, our method still delivers the leading performance. For 12 of the 14 generators, our method has the highest AP, and for the remaining generators, it has the second-best AP. The empirical superiority of the proposal extends to the GenVideo dataset. It typically has the best performance across video generators in the dataset (as presented in Table 2), demonstrating the empirical advantages of the proposal. Finally, the comparison on the more challenging FakeParts dataset, in which the AI-generated videos may come from fine-grained manipulations such as style changes and inpainting, is reported in Table 3. On this dataset, our method has the highest AUC and AP for three of the four generators; for the remaining generator, the proposal ranks second. Its overall competitive performance gives the proposal the highest average AUC and AP.

5.3 ABLATION STUDY

In this part, we conduct ablation studies to investigate the advantages of the proposed components: the adaptive reconstructor and reconstruction-error-based temporal modeling. We compare our method with three variants: (i) a vanilla method that differs from our method only in its use of a fixed VAE reconstructor, denoted as FixReError; (ii) a method that uses a 3D convolutional kernel with a temporal size of 1 and therefore reduces temporal aggregation, denoted as AdaReErrorOnly; and (iii) a more complicated method that explicitly uses an additional branch to construct the temporal information RSTD, and explicit fuse it with reconstruction errors via 3D convolutional kernel, denoted as ExplicitTemporal. All methods use the same training configuration (see Appendix B).

We conduct the ablation study on the challenging FakeParts dataset. Table A3 shows that, compared with FixReError, our method improves the AUC on all generators, with relative improvement ranges from 19.1%–89.5%. The average relative improvement reaches 68.5%. Furthermore, the

Table 1: AUC scores across generators in the EvalCrafter datasets; the best method is highlighted in cyan, the second-best method in orange. Each column represents one video generator and each row is one method.

	floor33	Gen-2	Gen-2 Dec.	hotshot	LaVIE base	LaVIE interp.	ModelScope
D3	65.00	86.49	88.89	85.43	73.02	73.97	61.74
STALL	79.43	86.70	89.55	73.33	84.66	81.88	73.35
Over-Coherence	47.55	32.07	49.01	70.92	45.44	33.90	42.55
ReConFuse	98.61	99.05	99.26	97.08	98.09	97.16	96.78
DIVID-CNN	95.94	97.44	98.52	83.43	96.86	93.80	92.83
DIVID-LSTM	95.07	96.70	97.72	70.26	94.48	91.14	87.28
G2VD-CLIP	99.57	99.45	99.47	93.41	98.78	96.29	97.57
Ivy-xDetector	94.17	95.05	98.16	93.30	90.91	81.67	90.72
DeepTraceReward	99.03	99.82	99.92	82.65	84.77	93.69	96.43
MM-Det	55.09	58.21	65.23	59.82	50.95	56.74	55.11
DeMamba	99.36	99.61	99.36	96.88	98.53	98.28	97.79
Ours	99.93	99.69	99.89	100.00	100.00	99.59	97.08
	Moonvalley	Pika	Pika Dec.	show_1	VideoCrafter2	VideoCrafter1	zeroscope
D3	85.10	82.46	85.32	76.24	86.49	80.91	56.10
STALL	78.63	83.48	87.60	78.85	85.40	82.77	66.04
Over-Coherence	43.24	32.51	38.83	28.59	55.06	57.32	42.22
ReConFuse	99.65	98.85	99.00	97.31	99.33	98.93	92.87
DIVID-CNN	99.24	97.32	97.24	88.50	99.14	97.45	79.19
DIVID-LSTM	99.17	96.72	96.94	90.77	98.47	96.38	71.48
G2VD-CLIP	99.85	99.41	99.40	95.52	99.86	99.24	93.63
Ivy-xDetector	98.70	94.73	94.18	85.38	98.67	96.45	87.20
DeepTraceReward	99.54	99.86	99.85	86.91	99.99	99.25	91.82
MM-Det	62.98	53.75	51.45	51.94	50.87	52.67	50.22
DeMamba	99.86	99.78	98.72	98.83	99.69	99.12	97.15
Ours	100.00	99.87	99.96	99.98	99.96	100.00	99.92

Table 2: AUC (upper part) and AP (lower part) on the GenVideo dataset. Avg. denotes the mean across the generators.

	Crafter	Gen-2	LaVIE	ModelScope	Moonvalley	MorphStudio	Show-1	Sora	Wildscrape	Avg.
D3	98.24	99.13	97.24	95.23	99.00	97.58	98.36	97.12	94.66	97.40
STALL	85.98	90.25	79.70	72.40	81.14	79.22	77.97	76.00	68.44	79.01
Over-Coherence	27.12	10.73	29.60	35.81	17.60	25.62	20.17	23.44	22.83	23.66
ReConFuse	98.64	99.00	96.71	96.65	99.40	98.03	97.55	94.41	89.95	96.70
DIVID-CNN	95.17	94.96	87.76	86.47	98.39	93.48	78.90	89.35	70.08	88.28
DIVID-LSTM	91.31	88.90	82.45	75.61	95.81	88.42	77.35	74.04	61.30	81.69
G2VD-CLIP	99.19	99.26	93.81	96.43	99.82	99.45	89.49	96.41	82.90	95.20
Ivy-xDetector	97.75	96.32	86.67	90.76	98.76	94.64	85.59	92.84	91.69	92.78
DeepTraceReward	99.55	99.97	88.19	96.82	99.49	98.80	87.56	99.17	89.28	95.43
MM-Det	51.73	62.41	53.80	54.29	63.35	55.89	52.17	70.62	58.17	58.05
DeMamba	99.38	99.55	98.17	97.84	99.84	99.29	98.57	96.40	85.84	97.21
Ours	99.96	100.00	100.00	100.00	100.00	99.89	100.00	99.99	93.69	99.28
D3	84.87	89.21	77.27	70.47	88.31	77.19	81.92	52.08	66.50	76.42
STALL	47.77	58.16	36.83	23.19	33.70	35.68	31.06	11.17	26.29	33.76
Over-Coherence	6.34	5.39	6.44	7.48	5.80	6.25	5.77	1.70	5.86	5.67
ReConFuse	93.19	94.76	84.82	74.38	92.79	84.44	78.36	36.96	63.94	78.18
DIVID-CNN	85.74	80.09	59.18	55.47	93.80	79.48	38.08	43.94	42.99	64.31
DIVID-LSTM	73.85	52.20	40.20	29.18	80.41	58.55	27.05	14.22	29.67	45.04
G2VD-CLIP	94.81	92.90	76.70	81.54	97.96	94.87	60.33	61.40	56.90	79.71
Ivy-xDetector	82.57	77.70	40.16	48.25	88.90	64.45	31.43	37.77	67.23	59.83
DeepTraceReward	97.75	99.73	58.15	88.39	96.89	95.58	54.11	91.27	74.11	84.00
MM-Det	10.57	23.95	11.71	12.89	26.23	14.08	10.49	30.19	17.31	17.49
DeMamba	95.94	96.38	87.73	87.70	99.05	96.36	90.40	59.58	65.55	86.52
Ours	99.73	100.00	100.00	99.97	100.00	99.76	100.00	99.00	88.23	98.52

comparison between AdaReErrorOnly and the proposal shows that using temporal information across the reconstruction errors indeed improves performance. The relative improvement over AdaReErrorOnly is 39.5% on average, and exceeds 55% in detecting video generated by LTX-Video. Finally, the comparison between ExplicitTemporal and our method shows that, overall, explicitly constructing RSTD does not lead to a substantial improvement. Their AUC values are very close, and the average AUC of ExplicitTemporal is even slightly worse than that of our method. This demonstrates the advantage of the simplicity of using Conv3D, as described in Section 4.3.

Table 3: AUC and AP on the FakeParts dataset.

Model	CogVideoX		LTX-Video		AllegroAI		Veo2		Avg.	
	AUC	AP	AUC	AP	AUC	AP	AUC	AP	AUC	AP
D3	93.60	93.86	91.13	90.58	83.08	79.26	79.41	72.27	86.81	83.99
STALL	83.68	83.80	67.77	69.09	68.99	70.67	60.01	58.54	70.11	70.53
Over-Coherence	41.14	43.83	45.45	43.04	37.13	40.02	35.94	39.63	39.92	41.63
ReConFuse	95.13	92.76	85.30	86.53	96.64	96.69	89.73	89.65	91.70	91.41
DIVID-CNN	58.75	65.11	89.72	89.60	98.21	98.20	90.15	90.38	84.21	85.82
DIVID-LSTM	59.63	57.25	82.66	86.71	98.82	98.93	88.96	88.77	82.52	82.92
G2VD-CLIP	97.13	96.57	91.01	90.72	99.65	99.64	96.61	95.29	96.10	95.56
Ivy-xDetector	96.62	95.95	94.52	94.54	93.66	93.00	90.05	88.47	93.71	92.99
DeepTraceReward	74.11	70.49	87.03	84.20	92.98	93.21	91.23	91.59	86.34	84.87
MM-Det	53.18	51.70	55.53	55.10	60.26	56.10	54.99	53.04	55.99	53.99
DeMamba	96.70	94.57	86.96	87.62	98.62	98.47	92.75	91.19	93.76	92.96
Ours	98.76	98.80	96.12	95.24	99.39	99.33	96.90	96.72	97.79	97.52

5.4 Analysis of Training Configurations

We perform an analysis of the proposed detector with respect to two important training configurations: (i) the number of epochs and (ii) the LoRA rank. The analysis is conducted on the FakeParts dataset. When studying the number of epochs, the LoRA rank is fixed; the same applies when studying the LoRA rank. For each configuration, we consider 3 possible values.

Figure 7 shows that performance is relatively stable across the number of epochs, with more training epochs generally providing better overall performance, although the gains diminish at later epochs. Empirically, the final number of epochs is selected based on detection performance on a standalone validation set, and we find that this choice already yields desirable performance. As for the LoRA rank, we compare three LoRA ranks $r \in \{16, 32, 64\}$. We can see that, overall, the value of r has only a marginal impact on detection performance—the variation in average AUC/AP is less than ± 1.0. This suggests that the method is not sensitive to r.

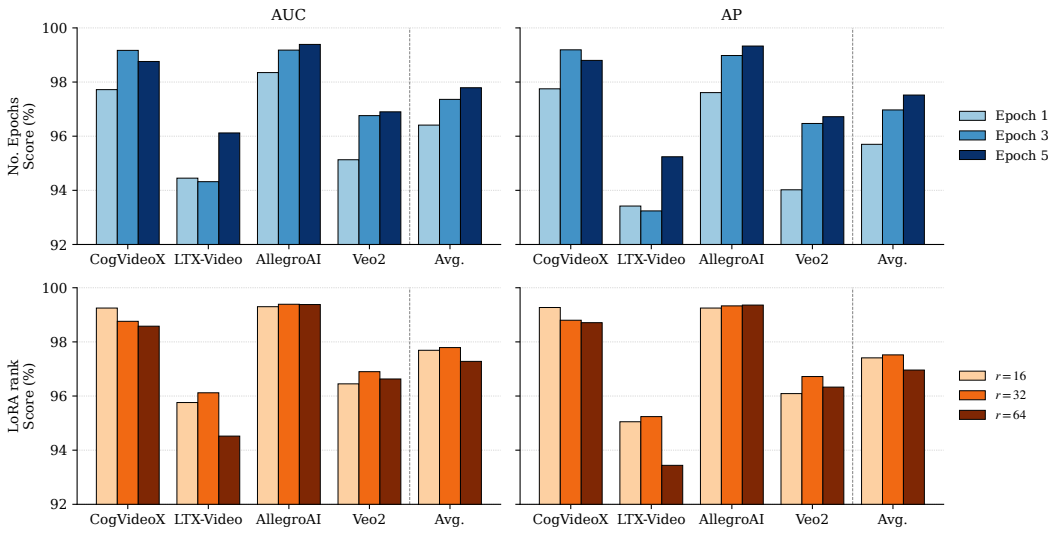

Figure 7: Sensitivity to the number of epochs (upper panels) and LoRA rank (lower panels). Left panels are AUC, while the right one are AP. The default training epochs is selected on the validation set and the LoRA rank is $r = 32$.

AI USE STATEMENT

In preparing this paper, LLMs were used only to aid and polish the writing and to retrieve and discover related work. All cited references were manually checked by the authors to confirm that the cited works exist. This use did not affect the core methodology.

ETHICS STATEMENT

This work uses only synthetic and existing public benchmark datasets and involves no participant recruitment or new collection of personal data.

REPRODUCIBILITY STATEMENT

Section 4 in the main text describes the proposed method and the training and evaluation protocols. Appendix B documents the model architecture, preprocessing, optimization settings, baseline implementation details, data sources, and hardware and software configuration.

REFERENCES

Hritik Bansal, Zongyu Lin, Tianyi Xie, Zeshun Zong, Michal Yarom, Yonatan Bitton, Chenfanfu Jiang, Yizhou Sun, Kai-Wei Chang, and Aditya Grover. Videophy: Evaluating physical commonsense for video generation. In *The Thirteenth International Conference on Learning Representations*, 2025. URL https://openreview.net/forum?id=9D2QvO1uWj.

Omer Ben Hayun, Roy Betser, Meir Yossef Levi, Levi Kassel, and Guy Gilboa. Training-free detection of generated videos via spatial-temporal likelihoods. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 16299–16310, 2026.

Andreas Blattmann, Tim Dockhorn, Sumith Kulal, Daniel Mendelevitch, Maciej Kilian, Dominik Lorenz, Yam Levi, Zion English, Vikram Voleti, Adam Letts, Varun Jampani, and Robin Rombach. Stable video diffusion: Scaling latent video diffusion models to large datasets. 2023a.

Andreas Blattmann, Robin Rombach, Huan Ling, Tim Dockhorn, Seung Wook Kim, Sanja Fidler, and Karsten Kreis. Align your latents: High-resolution video synthesis with latent diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 22563–22575, 2023b.

Jonathan Brokman, Oren Rachmil, Omer Hofman, Roy Betser, Amit Giloni, Roman Vainshtein, and Hisashi Kojima. Training-free detection of text-to-video generations via over-coherence. In *2026 IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*, pp. 3993–4003. IEEE, 2026.

Junyi Cao, Chao Ma, Taiping Yao, Shen Chen, Shouhong Ding, and Xiaokang Yang. End-to-end reconstruction-classification learning for face forgery detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 4113–4122, 2022. doi: 10.1109/CVPR52688.2022.00408.

George Cazenavette, Avneesh Sud, Thomas Leung, and Ben Usman. Fakeinversion: Learning to detect images from unseen text-to-image models by inverting stable diffusion. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 10759–10769, 2024. doi: 10.1109/CVPR52733.2024.01023.

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. In *2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 7310–7320. IEEE, 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.

Junsong Chen, Yuyang Zhao, Jincheng YU, Ruihang Chu, Junyu Chen, Shuai Yang, Xianbang Wang, Yicheng Pan, Zhou Daquan, Huan Ling, Haozhe Liu, Hongwei Yi, Hao Zhang, Muyang Li, Yukang Chen, Han Cai, Sanja Fidler, Ping Luo, Song Han, and Enze Xie. Sana-video: Efficient video generation with block linear diffusion transformer. In C. Vondrick, B. Hariharan, C. Raffel, L. Pinto, D. Yang, and A. Faust (eds.), *International Conference on Learning Representations*, volume 2026, pp. 38973–38997, 2026a. URL https://proceedings.iclr.cc/paper_files/paper/2026/file/41b93c59da0d0f835907fd661d419db2-Paper-Conference.pdf.

Tsai-Shien Chen, Aliaksandr Siarohin, Willi Menapace, Ekaterina Deyneka, Hsiang-wei Chao, Byung Eun Jeon, Yuwei Fang, Hsin-Ying Lee, Jian Ren, Ming-Hsuan Yang, and Sergey Tulyakov. Panda-70m: Captioning 70m videos with multiple cross-modality teachers. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2024c.

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*, 2026b.

Sungik Choi, Hankook Lee, Jaehoon Lee, Robin Kim, Stanley Jungkyu Choi, and Moontae Lee. A debiased reconstruction-based framework for training-free detection of ai-generated images. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2026.

Beilin Chu, Xuan Xu, Xin Wang, Yufei Zhang, Weike You, and Linna Zhou. FIRE: Robust detection of diffusion-generated images via frequency-guided reconstruction error. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 12830–12839, 2025. doi: 10.1109/CVPR52734.2025.01197.

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. *Advances in neural information processing systems*, 38:26418–26446, 2026.

Meng Du, Hongchang Chen, Ran Li, Junjie Zhang, Qi Ouyang, and Shuxin Liu. G2VD: Generalizable ai-generated video detection via counterfactual intervention and causal disentanglement. *arXiv preprint arXiv:2607.04607*, 2026.

Patrick Esser, Sumith Kulal, Andreas Blattmann, Rahim Entezari, Jonas Müller, Harry Saini, Yam Levi, Dominik Lorenz, Axel Sauer, Frederic Boesel, et al. Scaling rectified flow transformers for high-resolution image synthesis. In *Forty-first international conference on machine learning*, 2024.

Joel Frank, Thorsten Eisenhofer, Lea Schönherr, Asja Fischer, Dorothea Kolossa, and Thorsten Holz. Leveraging frequency analysis for deep fake image recognition. In *International conference on machine learning*, pp. 3247–3258. PMLR, 2020.

Xingyu Fu, Siyi Liu, Yinuo Xu, Pan Lu, Yejin Choi, James Zou, Dan Roth, and Chris Callison-Burch. Learning human-perceived fakeness in AI-generated videos via multimodal LLMs. In *Generative and Protective AI for Content Creation*, 2025. URL https://openreview.net/forum?id=L1jeJCkXZi.

Yuwei Guo, Ceyuan Yang, Anyi Rao, Zhengyang Liang, Yaohui Wang, Yu Qiao, Maneesh Agrawala, Dahua Lin, and Bo Dai. Animatediff: Animate your personalized text-to-image diffusion models without specific tuning. *arXiv preprint arXiv:2307.04725*, 2023.

Agrim Gupta, Lijun Yu, Kihyuk Sohn, Xiuye Gu, Meera Hahn, Fei-Fei Li, Irfan Essa, Lu Jiang, and José Lezama. Photorealistic video generation with diffusion models. In *European Conference on Computer Vision*, pp. 393–411. Springer, 2024.

Yoav HaCohen, Nisan Chiprut, Benny Brazowski, Daniel Shalem, Dudu Moshe, Eitan Richardson, Eran Levin, Guy Shiran, Nir Zabari, Ori Gordon, et al. Ltx-video: Realtime video latent diffusion. *arXiv preprint arXiv:2501.00103*, 2024.

Haojin He, Hao Tan, Zichang Tan, Ajian Liu, and Jun Wan. MotionPhys: Detecting ai-generated videos via physical consistency of optical-flow trajectories. *arXiv preprint arXiv:2608.20770*, 2026.

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2016.

Yingqing He, Tianyu Yang, Yong Zhang, Ying Shan, and Qifeng Chen. Latent video diffusion models for high-fidelity long video generation. *arXiv preprint arXiv:2211.13221*, 2022.

Jonathan Ho, William Chan, Chitwan Saharia, Jay Whang, Ruiqi Gao, Alexey Gritsenko, Diederik P Kingma, Ben Poole, Mohammad Norouzi, David J Fleet, et al. Imagen video: High definition video generation with diffusion models. *arXiv preprint arXiv:2210.02303*, 2022a.

Jonathan Ho, Tim Salimans, Alexey Gritsenko, William Chan, Mohammad Norouzi, and David J. Fleet. Video diffusion models. In *Advances in Neural Information Processing Systems*, volume 35, pp. 8633–8646, 2022b.

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. In *International Conference on Learning Representations*, 2022. URL https://openreview.net/forum?id=nZeVKeeFYf9.

Xun Huang, Zhengqi Li, Guande He, Mingyuan Zhou, and Eli Shechtman. Self forcing: Bridging the train-test gap in autoregressive video diffusion. *Advances in Neural Information Processing Systems*, 38:167283–167308, 2026.

Ziqi Huang, Yinan He, Jiashuo Yu, Fan Zhang, Chenyang Si, Yuming Jiang, Yuanhan Zhang, Tianxing Wu, Qingyang Jin, Nattapol Chanpaisit, Yaohui Wang, Xinyuan Chen, Limin Wang, Dahua Lin, Yu Qiao, and Ziwei Liu. VBench: Comprehensive benchmark suite for video generative models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2024.

Ziqi Huang, Fan Zhang, Xiaojie Xu, Yinan He, Jiashuo Yu, Ziyue Dong, Qianli Ma, Nattapol Chanpaisit, Chenyang Si, Yuming Jiang, Yaohui Wang, Xinyuan Chen, Ying-Cong Chen, Limin Wang, Dahua Lin, Yu Qiao, and Ziwei Liu. VBench++: Comprehensive and versatile benchmark suite for video generative models. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2025. doi: 10.1109/TPAMI.2025.3633890.

Tariq Berrada Ifriqi, John Nguyen, Karteek Alahari, Jakob Verbeek, and Ricky TQ Chen. Flowception: Temporally expansive flow matching for video generation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 16185–16195, 2026.

Christian Internò, Robert Geirhos, Markus Olhofer, Sunny Liu, Barbara Hammer, and David Klindt. Ai-generated video detection via perceptual straightening. In *Advances in Neural Information Processing Systems*, volume 38, pp. 23468–23501, 2025. doi: 10.52202/085713-0697.

Shuiwang Ji, Wei Xu, Ming Yang, and Kai Yu. 3d convolutional neural networks for human action recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 35(1):221–231, 2013. doi: 10.1109/TPAMI.2012.59.

Weinan Jia, Yuning Lu, Mengqi Huang, Hualiang Wang, Binyuan Huang, Nan Chen, Jidong Jiang, Zhendong Mao, et al. Moga: Mixture-of-groups attention for end-to-end long video generation. In *International Conference on Learning Representations*, volume 2026, pp. 115944–115964, 2026.

Changjiang Jiang, Wenhui Dong, Zhonghao Zhang, Fengchang Yu, Wei Peng, Xinbin Yuan, Yifei Bi, Ming Zhao, Zian Zhou, Chenyang Si, and Caifeng Shan. Ivy-fake: A unified explainable framework and benchmark for image and video aigc detection. In *Proceedings of the 2026 International Conference on Multimedia Retrieval*, ICMR '26, pp. 2438–2447, New York, NY, USA, 2026a. Association for Computing Machinery. ISBN 9798400726170. doi: 10.1145/3805622.3810615.

Wan Jiang, Jing Yan, Xiaojing Chen, Ling Shen, Chenhao Lin, Yunfeng Diao, and Richang Hong. Eires:training-free ai-generated image detection via edit-induced reconstruction error shift. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) Findings*, pp. 6486–6495, June 2026b.

Fanli Jin, Feng Lin, Gaojian Wang, Tong Wu, and Zhisheng Yan. DySy-Det: A synergistic framework with dynamic reconstruction-path consistency for ai-generated image detection. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 40, pp. 35571–35579, 2026. doi: 10.1609/aaai.v40i42.40868.

Yang Jin, Zhicheng Sun, Ningyuan Li, Kun Xu, Hao Jiang, Nan Zhuang, Quzhe Huang, Yang Song, Yadong Mu, and Zhouchen Lin. Pyramidal flow matching for efficient video generative modeling. In *International Conference on Learning Representations*, volume 2025, pp. 23378–23402, 2025.

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

Akio Kodaira, Tingbo Hou, Ji Hou, Markos Georgopoulos, Felix Juefei-Xu, Masayoshi Tomizuka, and Yue Zhao. Streamdit: Real-time streaming text-to-video generation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 29200–29210, 2026.

Dan Kondratyuk, Lijun Yu, Xiuye Gu, Jose Lezama, Jonathan Huang, Grant Schindler, Rachel Hornung, Vighnesh Birodkar, Jimmy Yan, Ming-Chang Chiu, Krishna Somandepalli, Hassan Akbari, Yair Alon, Yong Cheng, Joshua V. Dillon, Agrim Gupta, Meera Hahn, Anja Hauth, David Hendon, Alonso Martinez, David Minnen, Mikhail Sirotenko, Kihyuk Sohn, Xuan Yang, Hartwig Adam, Ming-Hsuan Yang, Irfan Essa, Huisheng Wang, David A Ross, Bryan Seybold, and Lu Jiang. VideoPoet: A large language model for zero-shot video generation. In Ruslan Salakhutdinov, Zico Kolter, Katherine Heller, Adrian Weller, Nuria Oliver, Jonathan Scarlett, and Felix Berkenkamp (eds.), *Proceedings of the 41st International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*, pp. 25105–25124. PMLR, 21–27 Jul 2024.

Yifei Li, Wenzhao Zheng, Yanran Zhang, Runze Sun, Yu Zheng, Lei Chen, Jie Zhou, and Jiwen Lu. Skyra: Ai-generated video detection via grounded artifact reasoning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 4482–4493, 2026a.

Zhengcen Li, Chenyang Jiang, Hang Zhao, Shiyang Zhou, Yunyang Mo, Feng Gao, Fan Yang, Qiben Shan, Shaocong Wu, and Jingyong Su. Preserving forgery artifacts: Ai-generated video detection at native scale. In *The Fourteenth International Conference on Learning Representations*, 2026b.

Zongjian Li, Bin Lin, Yang Ye, Liuhan Chen, Xinhua Cheng, Shenghai Yuan, and Li Yuan. Wf-vae: Enhancing video vae by wavelet-driven energy flow for latent video diffusion model. In *2025 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 17778–17788. IEEE, 2025.

Yaron Lipman, Ricky T. Q. Chen, Heli Ben-Hamu, Maximilian Nickel, and Matthew Le. Flow matching for generative modeling. In *International Conference on Learning Representations*, 2023.

Kunhao Liu, Wenbo Hu, Jiale Xu, Ying Shan, and Shijian Lu. Rolling forcing: Autoregressive long video diffusion in real time. In C. Vondrick, B. Hariharan, C. Raffel, L. Pinto, D. Yang, and A. Faust (eds.), *International Conference on Learning Representations*, volume 2026, pp. 91177–91196, 2026. URL https://proceedings.iclr.cc/paper_files/paper/2026/file/935151cc6cb5d8b6816133b75233775a-Paper-Conference.pdf.

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*, 2024a.

Yaofang Liu, Xiaodong Cun, Xuebo Liu, Xintao Wang, Yong Zhang, Haoxin Chen, Yang Liu, Tieyong Zeng, Raymond Chan, and Ying Shan. Evalcrafter: Benchmarking and evaluating large video generation models. In *2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 22139–22149. IEEE, 2024b.

Ziyi Liu, Firas Gabetni, Awais Hussain Sani, Xi Wang, Soobash Daiboo, Gaetan Brison, Gianni Franchi, and Vicky Kalogeiton. Fakeparts: a new family of ai-generated deepfakes. *arXiv preprint arXiv:2508.21052*, 2025.

Yunpeng Luo, Junlong Du, Ke Yan, and Shouhong Ding. LaRE^2: Latent reconstruction error based method for diffusion-generated image detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 17006–17015, 2024.

Long Ma, Zhiyuan Yan, Qinglang Guo, Yong Liao, Haiyang Yu, and Pengyuan Zhou. Detecting ai-generated video via frame consistency. 2024.

Long Ma, Zihao Xue, Yan Wang, Zhiyuan Yan, Jin Xu, Xiaorui Jiang, Haiyang Yu, Yong Liao, and Zhen Bi. Your one-stop solution for ai-generated video detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 4458–4470, 2026.

Xin Ma, Yaohui Wang, Xinyuan Chen, Gengyun Jia, Ziwei Liu, Yuan-Fang Li, Cunjian Chen, and Yu Qiao. Latte: Latent diffusion transformer for video generation. *Transactions on Machine Learning Research*, 2025.

Bolin Ni, Houwen Peng, Minghao Chen, Songyang Zhang, Gaofeng Meng, Jianlong Fu, Shiming Xiang, and Haibin Ling. Expanding language-image pretrained models for general video recognition. In *European conference on computer vision*, pp. 1–18. Springer, 2022.

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*, volume 40, pp. 15582–15590, 2026. doi: 10.1609/aaai.v40i18.38587.

Zirui Pan, Xin Wang, Yipeng Zhang, Yuwei Zhou, and Wenwu Zhu. Temporal-aware flow matching for video generation with temporally coherent motion. In *Forty-third International Conference on Machine Learning*, 2026.

Kyoungjun Park, Yifan Yang, Juheon Yi, Muhammad Muaz, Shicheng Zheng, Yifei Shen, Dongqi Han, Caihua Shan, and Lili Qiu. VidGuard-R1: Ai-generated video detection and explanation via reasoning MLLMs and RL. In *The Fourteenth International Conference on Learning Representations*, 2026.

William Peebles and Saining Xie. Scalable diffusion models with transformers. In *2023 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 4172–4182. IEEE, 2023.

Adam Polyak, Amit Zohar, Andrew Brown, Andros Tjandra, Animesh Sinha, Ann Lee, Apoorv Vyas, Bowen Shi, Chih-Yao Ma, Ching-Yao Chuang, et al. Movie gen: A cast of media foundation models. *arXiv preprint arXiv:2410.13720*, 2024.

Xinyi Qi, Kai Ye, Chengchun Shi, Ying Yang, Jin Zhu, and Hongyi Zhou. A difference-in-difference approach to detecting ai-generated images. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 42964–42975, 2026.

Yuyang Qian, Guojun Yin, Lu Sheng, Zixuan Chen, and Jing Shao. Thinking in frequency: Face forgery detection by mining frequency-aware clues. In *European conference on computer vision*, pp. 86–103. Springer, 2020.

Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In *International conference on machine learning*, pp. 8748–8763. PmLR, 2021.

Joseph Redmon and Ali Farhadi. Yolov3: An incremental improvement. *arXiv preprint arXiv:1804.02767*, 2018.

Jonas Ricker, Denis Lukovnikov, and Asja Fischer. AEROBLADE: Training-free detection of latent diffusion images using autoencoder reconstruction error. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 9130–9140, 2024. doi: 10.1109/CVPR52733.2024.00872.

Uriel Singer, Adam Polyak, Thomas Hayes, Xi Yin, Jie An, Songyang Zhang, Qiyuan Hu, Harry Yang, Oron Ashual, Oran Gafni, et al. Make-a-video: Text-to-video generation without text-video data. *arXiv preprint arXiv:2209.14792*, 2022.

Xiufeng Song, Xiao Guo, Jiache Zhang, Qirui Li, Lei Bai, Xiaoming Liu, Guangtao Zhai, and Xiaohong Liu. On learning multi-modal forgery representation for diffusion generated video detection. *Advances in Neural Information Processing Systems*, 37:122054–122077, 2024.

Chuangchuang Tan, Yao Zhao, Shikui Wei, Guanghua Gu, Ping Liu, and Yunchao Wei. Rethinking the up-sampling operations in cnn-based generative network for generalizable deepfake detection. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 28130–28139, 2024.

Hao Tan, Jun Lan, Senyuan Shi, Zichang Tan, Zijian Yu, Huijia Zhu, Weiqiang Wang, Jun Wan, and Zhen Lei. Videoveritas: Ai-generated video detection via perception pretext reinforcement learning. *arXiv preprint arXiv:2602.08828*, 2026.

Du Tran, Heng Wang, Lorenzo Torresani, Jamie Ray, Yann LeCun, and Manohar Paluri. A closer look at spatiotemporal convolutions for action recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 6450–6459, 2018.

Aaron Van Den Oord, Oriol Vinyals, et al. Neural discrete representation learning. *Advances in neural information processing systems*, 30, 2017.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. *Advances in neural information processing systems*, 30, 2017.

Ruben Villegas, Mohammad Babaeizadeh, Pieter-Jan Kindermans, Hernan Moraldo, Han Zhang, Mohammad Taghi Saffar, Santiago Castro, Julius Kunze, and Dumitru Erhan. Phenaki: Variable length video generation from open domain textual description. *arXiv preprint arXiv:2210.02399*, 2022.

Yaohui Wang, Xinyuan Chen, Xin Ma, Shangchen Zhou, Ziqi Huang, Yi Wang, Ceyuan Yang, Yinan He, Jiashuo Yu, Peiqing Yang, et al. Lavie: High-quality video generation with cascaded latent diffusion models. *International Journal of Computer Vision*, 133(5):3059–3078, 2025.

Zhendong Wang, Jianmin Bao, Wengang Zhou, Weilun Wang, Hezhen Hu, Hong Chen, and Houqiang Li. DIRE for diffusion-generated image detection. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 22445–22455, 2023. doi: 10.1109/ICCV51070.2023.02051.

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, Yiwei He, Zhenglin Huang, Tianxiao Li, Zihan Yu, Xingru Huang, Lu Qi, Baoyuan Wu, Xiangtai Li, and Guangliang Cheng. Busterx: Mllm-powered ai-generated video forgery detection and explanation. *arXiv preprint arXiv:2505.12620*, 2025.

Jianzong Wu, Xiangtai Li, Chenyang Si, Shangchen Zhou, Jingkang Yang, Jiangning Zhang, Yining Li, Kai Chen, Yunhai Tong, Ziwei Liu, et al. Towards language-driven video inpainting via multimodal large language models. In *2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 12501–12511. IEEE, 2024.

Qianyu Xiang, Yunfeng Guo, Siqi Gu, Lizhe Xie, and Yining Hu. Detecting ai-generated videos via global semantic awareness and inter-frame semantic consistency. *Neurocomputing*, 684:133530, 2026. doi: 10.1016/j.neucom.2026.133530.

Jun Xu, Tao Mei, Ting Yao, and Yong Rui. Msr-vtt: A large video description dataset for bridging video and language. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 5288–5296, 2016.

Wilson Yan, Yunzhi Zhang, Pieter Abbeel, and Aravind Srinivas. Videogpt: Video generation using vq-vae and transformers. *arXiv preprint arXiv:2104.10157*, 2021.

Zhiyuan Yan, Taiping Yao, Shen Chen, Yandan Zhao, Xinghe Fu, Junwei Zhu, Donghao Luo, Chengjie Wang, Shouhong Ding, Yunsheng Wu, and Li Yuan. Df40: Toward next-generation deepfake detection. In *Advances in Neural Information Processing Systems*, volume 37, pp. 29387–29434. Curran Associates, Inc., 2024. doi: 10.52202/079017-0925.

Shuai Yang, Wei Huang, Ruihang Chu, Yicheng Xiao, Yuyang Zhao, Xianbang Wang, Muyang Li, Enze Xie, Yingcong Chen, Yao Lu, et al. Longlive: Real-time interactive long video generation. *arXiv preprint arXiv:2509.22622*, 2025.

Zhuoyi Yang, Jiayan Teng, Wendi Zheng, Ming Ding, Shiyu Huang, Jiazheng Xu, Yuan Yang, Wenyi Hong, Xiaohan Zhang, Guanyu Feng, Da Yin, Yuxuan Zhang, Weihan Wang, Ye Cheng, Bin Xu, Xiaotao Gu, Yuxiao Dong, and Jie Tang. Cogvideox: Text-to-video diffusion models with an expert transformer. 2024.

Tianwei Yin, Qiang Zhang, Richard Zhang, William T Freeman, Fredo Durand, Eli Shechtman, and Xun Huang. From slow bidirectional to fast autoregressive video diffusion models. In *2025 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 22963–22974. IEEE, 2025.

Xiao Yu, Kejiang Chen, Kai Zeng, Han Fang, Zijin Yang, Xiuwei Shang, Yuang Qi, Weiming Zhang, and Nenghai Yu. Semgir: Semantic-guided image regeneration based method for ai-generated image detection and attribution. In *Proceedings of the 32nd ACM International Conference on Multimedia*, MM '24, pp. 8480–8488, New York, NY, USA, 2024. Association for Computing Machinery. ISBN 9798400706868. doi: 10.1145/3664647.3680776. URL https://doi.org/10.1145/3664647.3680776.

Shuhai Zhang, Zihao Lian, Jiahao Yang, Daiyuan Li, Guoxuan Pang, Feng Liu, Bo Han, Shutao Li, and Mingkui Tan. Physics-driven spatiotemporal modeling for ai-generated video detection. In *Advances in Neural Information Processing Systems*, volume 38, pp. 193477–193524, 2025. doi: 10.52202/085713-5810.

Xuanlei Zhao, Xiaolong Jin, Kai Wang, and Yang You. Real-time video generation with pyramid attention broadcast. In *International Conference on Learning Representations*, volume 2025, pp. 3296–3319, 2025.

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*, pp. 12852–12862, 2025.

Zangwei Zheng, Xiangyu Peng, Tianji Yang, Chenhui Shen, Shenggui Li, Hongxin Liu, Yukun Zhou, Tianyi Li, and Yang You. Open-sora: Democratizing efficient video production for all. *arXiv preprint arXiv:2412.20404*, 2024.

A PROOF

For notational simplicity, we consider a bias-free 3D convolution with one output channel[1]. For a temporal kernel of size three, corresponding to the temporal-offset set $\mathcal{T} = \{0, 1, 2\}$, spatial support $\mathcal{S}$, and kernel G, 3D convolution is defined on $[r_t, r_{t-1}, r_{t-2}]$ as

$$\mathrm{Conv3D}_G([r_t, r_{t-1}, r_{t-2}])(p) = \sum_{j\in\mathcal{T}} \sum_{c=1}^{C} \sum_{u\in\mathcal{S}} G_{c,j,u} \times r_{t-j,c,p-u}. \tag{3}$$

Here, c indexes the input channel, $j \in \mathcal{T}$ indexes the temporal offset, p denotes a spatial location, and $u \in \mathcal{S}$ denotes a spatial offset. More specifically, $G_{c,j,u}$ is the learnable scalar kernel weight applied to $r_{t-j,c,p-u}$, the reconstruction-error value from input channel c at time $t - j$ and spatial location $p - u$. Their product is the contribution of this input value to the convolved value at location p, and the sums aggregate such contributions over temporality, channels, and spatiality.

A.1 PROOF OF PROPOSITION 1

Proof. We first prove the forward direction, i.e., Equation (2). Let $d_t = r_t - 2r_{t-1} + r_{t-2}$. At spatial location p, according to the definition of 3D convolution in Equation (3), the convolution over the four-slice augmented input $[d_t, r_t, r_{t-1}, r_{t-2}]$ can be written as

$$\begin{aligned}\mathrm{Conv3D}_H([d_t, r_t, r_{t-1}, r_{t-2}])(p) = &\sum_{c=1}^{C} \sum_{u\in\mathcal{S}} H_{c,0,u} \times d_{t,c,p-u} \\ &+ \sum_{j=0}^{2} \sum_{c=1}^{C} \sum_{u\in\mathcal{S}} H_{c,j+1,u} \times r_{t-j,c,p-u}.\end{aligned}$$

Substituting $d_t = r_t - 2r_{t-1} + r_{t-2}$ and collecting the coefficients of the three reconstruction errors r_t, r_{t-1}, r_{t-2} yields

$$\begin{aligned}&\sum_{c=1}^{C} \sum_{u\in\mathcal{S}} \big(H_{c,0,u} + H_{c,1,u}\big) \times r_{t,c,p-u} \\ &+ \sum_{c=1}^{C} \sum_{u\in\mathcal{S}} \big(-2H_{c,0,u} + H_{c,2,u}\big) \times r_{t-1,c,p-u} \\ &+ \sum_{c=1}^{C} \sum_{u\in\mathcal{S}} \big(H_{c,0,u} + H_{c,3,u}\big) \times r_{t-2,c,p-u}.\end{aligned}$$

Define the three temporal slices of G by

$$\begin{aligned}G_{c,0,u} &= H_{c,0,u} + H_{c,1,u}, \\ G_{c,1,u} &= -2H_{c,0,u} + H_{c,2,u}, \\ G_{c,2,u} &= H_{c,0,u} + H_{c,3,u},\end{aligned}$$

By Equation (3), the resulting expression is exactly $\mathrm{Conv3D}_G([r_t, r_{t-1}, r_{t-2}])(p)$. This proves the first direction.

For the converse, given any three-slice kernel G, set $H_{c,0,u} = 0$, $H_{c,1,u} = G_{c,0,u}$, $H_{c,2,u} = G_{c,1,u}$, and $H_{c,3,u} = G_{c,2,u}$ for every c and u. The corresponding convolution over the four-slice augmented input then ignores d_t and is identical to $\mathrm{Conv3D}_G([r_t, r_{t-1}, r_{t-2}])$. This proves the reverse direction. □

Additional Remark on Proposition 1. In general, a 3D convolutional block applies a linear Conv3D operator before normalization and activation, as in residual 3D CNNs (Tran et al., 2018) and the

[1]Including a bias does not change the result because the same bias can be used on both sides. The proof can be easily extended to the case of multiple outputs.

official Torchvision VideoResNet implementation[2]. Since the two input representations produce identical Conv3D feature maps, applying the same subsequent normalization and activation preserves their equality. Thus, the proposition applies to commonly used 3D convolutional blocks.

B Implementations and Experiments: Details

Video preprocessing. We sample video frames at 2 frames per second, using 9 frames when available and 5 otherwise, with last-frame repetition for shorter clips. Random temporal windows are used during training and the central window during evaluation. Frames are resized while preserving their aspect ratio, with the spatial resolution constrained to $[224^2, 448^2]$ pixels and both height and width adjusted to multiples of eight. No spatial cropping or padding is applied.

Reconstructor and classifier. We use the pretrained Wan2.1 VAE[3] as the reconstruction backbone. The reconstruction error is processed by a randomly initialized 3D ResNet-18-style classifier with stage depths $(2, 2, 2, 2)$ and channel widths $(32, 64, 128, 256)$. We configure the 3D convolution with a temporal kernel size of 3 and a kernel height and width of 7. All downsampling is spatial, with no temporal downsampling throughout the network. Global average pooling of the final-stage feature map produces a 256-dimensional descriptor, followed by a $256 \to 512 \to 256 \to 1$ classification head.

LoRA and optimization. The pretrained Wan-VAE parameters are frozen, and rank-32 LoRA adapters are inserted into eight 3D convolutional layers in the encoder and decoder middle blocks of the VAE. The LoRA adapters and detector are jointly optimized for five epochs using AdamW with weight decay 10^{-4}. The initial learning rates are 5×10^{-6} for LoRA and 5×10^{-5} for the detector, with cosine decay to 10^{-7}. Training uses a global batch size of eight. The LoRA adapters are optimized solely through the video-level real/fake classification objective. Training and evaluation use BF16 automatic mixed precision.

B.1 Details of Experiments

Implementation baselines. For the training-based methods, we directly reuse their checkpoints whenever they are available. These methods include Ivy-xDetector[4], DeepTraceReward[5], and MM-Det[6]. When checkpoints are unavailable, we implement these methods ourselves or reuse their training code. These methods include ReConFuse, two variants of DIVID, G2VD, and DeMamba[7]. For these methods, we use the same training dataset as for our method. For the training-free methods, we use their official implementations, including D3[8], STALL[9], and Over-Coherence[10].

Data sources. Kinetics-400 videos[11] and released VBench generations[12] are obtained from their public repositories. MovieGen and Panda-70M videos are obtained from the MovieGenBench[13] and Panda-70M[14] releases, respectively. For GenVideo, we use the released `GenVideo-Val` split, following the download and preparation instructions in the DeMamba repository.[15] EvalCrafter[16] and

[2] https://docs.pytorch.org/vision/stable/_modules/torchvision/models/video/resnet.html
[3] https://huggingface.co/Wan-AI/Wan2.1-T2V-1.3B-Diffusers
[4] https://huggingface.co/AI-Safeguard/Ivy-Fake
[5] https://huggingface.co/DeeptraceReward/RewardModel
[6] https://huggingface.co/sparklexfantasy/llava-7b-1.5-rfrd
[7] https://github.com/chenhaoxing/DeMamba
[8] https://github.com/Zig-HS/D3
[9] https://github.com/OmerBenHayun/STALL
[10] https://github.com/FujitsuResearch/training-free-detection-of-text-to-video-generations-vi
[11] https://github.com/cvdfoundation/kinetics-dataset
[12] https://github.com/Vchitect/VBench
[13] https://github.com/facebookresearch/MovieGenBench
[14] https://github.com/snap-research/Panda-70M
[15] https://github.com/chenhaoxing/DeMamba
[16] https://github.com/EvalCrafter/EvalCrafter

VideoPhy[17] videos are obtained from their respective public releases. FakeParts videos and metadata are obtained from its Hugging Face release.[18]

Hardware setup. All experiments are conducted on NVIDIA GH200 Grace Hopper GPUs, each with 96 GB of GPU memory. Our method is trained on two GPUs using distributed data parallelism. The version of Python is 3.12 with `PyTorch 2.8.0a0`, and CUDA 12.9.

C Additional Results

Table A1: AP scores across generators in the EvalCrafter datasets.

Model	floor33	Gen-2	Gen-2 Dec.	hotshot	LaVIE base	LaVIE interp.	ModelScope
D3	36.89	55.37	62.85	54.81	46.07	45.56	31.14
STALL	58.78	66.77	77.06	49.37	68.89	64.03	48.71
Over-Coherence	23.18	17.89	23.55	38.18	23.16	19.09	22.36
ReConFuse	96.81	97.79	98.29	93.02	95.61	93.49	92.19
DIVID-CNN	93.81	95.83	97.52	74.03	94.27	88.87	86.30
DIVID-LSTM	91.74	92.68	95.42	51.77	89.50	82.29	75.21
G2VD-CLIP	98.59	98.24	98.26	85.11	97.00	91.70	94.04
Ivy-xDetector	82.29	83.11	94.97	81.48	74.71	57.59	73.29
DeepTraceReward	97.94	99.58	99.79	63.67	68.28	86.56	93.84
MM-Det	30.66	34.38	45.47	37.31	26.35	32.96	31.35
DeMamba	98.39	98.92	98.19	91.64	95.93	95.18	94.47
Ours	99.88	99.34	99.80	100.00	100.00	99.09	95.39
Model	Moonvalley	Pika	Pika Dec.	show_1	VideoCrafter2	VideoCrafter1	zeroscope
D3	54.47	55.74	60.71	48.55	60.14	57.84	24.22
STALL	52.58	66.89	71.73	57.22	66.58	66.34	33.05
Over-Coherence	19.65	18.32	19.93	17.74	26.26	28.11	21.07
ReConFuse	98.95	96.99	97.48	93.62	98.68	97.30	81.65
DIVID-CNN	98.77	95.33	95.75	80.77	98.60	95.72	58.56
DIVID-LSTM	98.20	93.30	94.76	82.01	97.44	94.22	49.08
G2VD-CLIP	99.37	98.09	98.09	88.90	99.51	97.99	84.33
Ivy-xDetector	95.02	82.70	79.08	58.48	95.59	89.48	58.39
DeepTraceReward	98.82	99.61	99.66	73.65	99.99	98.21	84.31
MM-Det	40.81	29.09	26.88	27.01	26.28	28.06	23.10
DeMamba	99.59	99.35	96.65	96.53	99.26	97.58	89.91
Ours	100.00	99.68	99.90	99.95	99.89	100.00	99.77

[17] https://github.com/Hritikbansal/videophy

[18] https://huggingface.co/datasets/hi-paris/FakeParts

Table A2: AUC and AP on classifying the fake videos in the VideoPhy dataset and real videos in the MSR-VTT dataset. AUC is demonstrated on the upper part while AP is on the lower part.

Model	LaVIE	cogvideox	cogvideox-5B	dream_machine	Gen-2	pika_video	svd	vc2	zeroscope	Avg.
D3	52.11	60.96	60.40	70.80	88.20	81.99	74.11	69.03	69.31	69.66
STALL	69.85	60.39	59.21	68.55	84.29	76.78	71.24	81.63	69.16	71.23
Over-Coherence	21.70	39.08	35.63	62.70	52.51	29.19	35.59	44.59	38.12	39.90
ReConFuse	96.30	92.85	94.27	98.23	99.65	99.32	97.94	99.41	95.52	97.05
DIVID-CNN	96.45	79.53	88.62	97.43	99.00	99.39	97.30	99.47	81.14	93.15
DIVID-LSTM	92.46	73.77	84.42	95.33	98.21	99.10	94.58	98.69	67.67	89.36
G2VD-CLIP	99.04	90.37	92.68	99.47	99.83	99.70	99.20	99.81	96.84	97.44
Ivy-xDetector	83.95	86.36	83.71	94.26	98.73	96.03	94.80	89.28	84.01	90.13
DeepTraceReward	70.98	88.08	77.03	95.45	99.32	98.59	94.68	78.32	77.67	86.68
MM-Det	50.03	52.60	57.87	52.34	55.69	50.59	56.52	51.56	51.42	53.18
DeMamba	98.68	97.00	97.96	99.38	99.76	99.89	99.49	99.91	98.01	98.90
Ours	99.86	100.00	99.99	100.00	100.00	100.00	99.55	100.00	99.90	99.92
D3	15.78	26.12	18.43	22.66	41.03	33.92	25.57	23.07	22.01	25.40
STALL	25.81	18.91	19.87	22.86	46.09	33.56	29.09	40.88	23.00	28.90
Over-Coherence	8.92	13.72	10.75	17.92	13.34	8.68	10.66	12.66	15.25	12.43
ReConFuse	84.40	69.38	77.73	91.93	97.80	95.76	91.12	97.49	78.49	87.12
DIVID-CNN	86.55	54.67	70.72	91.70	96.62	97.27	91.96	97.97	50.44	81.99
DIVID-LSTM	75.52	38.34	53.24	82.36	93.29	95.37	83.00	94.94	30.15	71.80
G2VD-CLIP	95.32	71.10	76.06	96.27	98.57	97.34	96.42	98.90	86.39	90.71
Ivy-xDetector	46.20	47.50	41.35	70.66	91.26	72.63	73.48	53.10	37.54	59.30
DeepTraceReward	35.60	68.42	47.38	91.78	98.23	94.69	89.16	39.78	40.51	67.28
MM-Det	14.65	16.29	23.16	15.98	18.46	13.33	21.02	15.27	15.29	17.05
DeMamba	91.66	81.41	89.06	96.56	98.36	99.45	97.76	99.46	87.08	93.42
Ours	99.70	100.00	99.97	100.00	100.00	100.00	98.93	100.00	99.45	99.78

Table A3: Ablation studies on the AUC on the FakeParts dataset. The best method among FixReError, AdaReErrorOnly and Ours in each column is highlighted in cyan. The fourth and fifth rows report the relative improvement of Ours over FixReError and AdaReErrorOnly, computed by $(\text{Ours} - \text{baseline})/(100 - \text{baseline})$.

	CogVideoX	LTX-Video	Allegro	Veo2	Avg.
FixReError	93.28	95.20	94.17	89.25	92.98
AdaReErrorOnly	98.68	90.81	99.36	96.64	96.37
Ours	98.76	96.12	99.39	96.90	97.79
Relative ↑ (FixReError)	81.55	19.17	89.54	71.16	68.58
Relative ↑ (AdaReErrorOnly)	6.06	57.78	4.69	7.74	39.15
ExplicitTemporal	98.99	95.00	99.02	95.97	97.25

D Conclusion and Discussion

In this paper, we develop an algorithm that produces discriminative reconstruction errors and exploits the temporal signal to detect AI-generated videos. The overall workflow is simple and easy to implement. Across the evaluated benchmarks, the method achieves competitive performance against the considered training-based and training-free detectors. As more videos combine natural and fake content, it is important for the proposed method to handle such cases.