SCIENCE CHINA
Information Sciences

• RESEARCH PAPER •

June 2026, Vol. 69, Iss. 6, 162103:1–162103:15
https://doi.org/10.1007/s11432-024-4894-0

DeMamba: AI-generated video detection on million-scale GenVideo benchmark

Haoxing CHEN[1†], Yan HONG[1†], Zizheng HUANG[1,2], Zhuoer XU[1], Zhangxuan GU[1*], Yaohui LI[2], Jun LAN[1], Huijia ZHU[1], Jianfu ZHANG[3*], Weiqiang WANG[1] & Huaxiong LI[2]

[1]*Tiansuan Lab, Ant Group, Hangzhou 310023, China*
[2]*Department of Control Science and Intelligence Engineering, Nanjing University, Nanjing 210023, China*
[3]*Qingyuan Research Institute, Shanghai Jiao Tong University, Shanghai 200240, China*

Received 10 December 2024/Revised 6 June 2025/Accepted 28 July 2025/Published online 22 May 2026

Abstract Recently, video generation techniques have advanced rapidly. Given the popularity of video content on social media platforms, these models intensify concerns about the spread of fake information. Therefore, there is a growing demand for detectors capable of distinguishing between artificial intelligence (AI) generated videos and mitigating the potential harm caused by fake information. However, the lack of large-scale datasets from the most advanced video generators poses a barrier to the development of such detectors. To address this gap, we introduce the first AI-generated video detection dataset, GenVideo. It features the following characteristics: (1) a large volume of videos, including over one million AI-generated and real videos collected; (2) a rich diversity of generated content and methodologies, covering a broad spectrum of video categories and generation techniques. We conduct extensive studies of the dataset and propose two evaluation methods tailored for real-world scenarios to assess the detectors' performance: the cross-generator video classification task assesses the generalizability of trained detectors on generators; the degraded video classification task evaluates the robustness of detectors to handle videos that have degraded in quality during dissemination. Moreover, we introduced a plug-and-play module, named detail mamba (DeMamba), designed to enhance the detectors by identifying AI-generated videos through the analysis of inconsistencies in temporal and spatial dimensions. Our extensive experiments demonstrate DeMamba's superior generalizability and robustness on GenVideo compared to existing detectors. We believe that the GenVideo dataset and the DeMamba module will significantly advance the field of AI-generated video detection. Our code and dataset are available at https://github.com/chenhaoxing/DeMamba.

Keywords generative model, video detection, dataset, deepfake, vision mamba

Citation Chen H X, Hong Y, Huang Z Z, et al. DeMamba: AI-generated video detection on million-scale GenVideo benchmark. Sci China Inf Sci, 2026, 69(6): 162103, https://doi.org/10.1007/s11432-024-4894-0

1 Introduction

Advancements in generative models [1–3] have been impressive, enabling the creation of highly realistic images with less effort and expertise. As these models become capable of generating sufficiently realistic images, more researchers are exploring how to improve video creation [4–7]. Currently, certain generative algorithms, such as Sora [8] and Gen2 [9], are capable of producing high-quality videos through the use of straightforward inputs, including text and images. While these generative algorithms can reduce manual labor and enhance creativity, they also introduce risks [10]. For example, they could be utilized to misinform the public in critical domains such as politics or economics. A notable incident involved an artificial intelligence (AI) generated video of Taylor Swift that spread widely on Twitter, harming her reputation. This situation highlights the pressing need for technology that can detect these fake videos and avoid potential harm.

To assist in developing robust and highly generalizable detectors, we have created the first million-scale dataset of AI-generated videos, named GenVideo. GenVideo leverages state-of-the-art models to generate massive amounts of video, providing comprehensive training and validation for detectors of AI-generated videos. Unlike deepfake video datasets [11,12] which focus on human face videos, GenVideo encompasses a broad spectrum of scene contents and motion variations, closely simulating the real-world authentication challenges posed by video generation models in

* Corresponding author (email: guzhangxuan.gzx@antgroup.com, c.sis@sjtu.edu.cn)
† These authors contributed equally to this work.

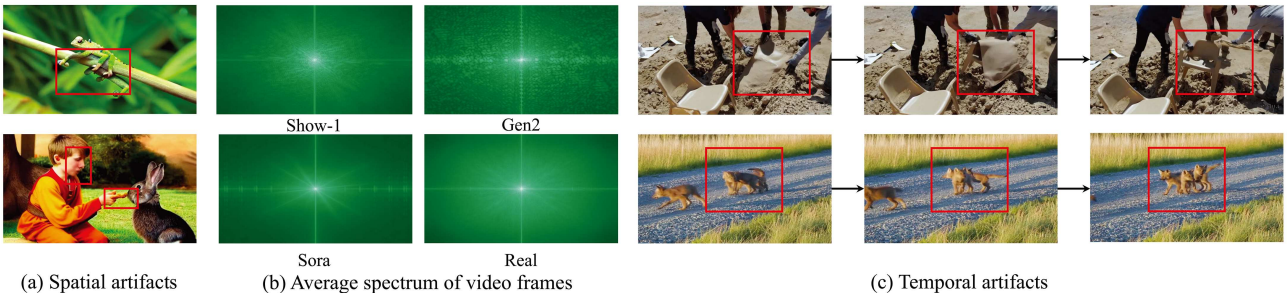

Figure 1 (Color online) We illustrate the spatial and temporal artifacts present in the generated videos. (a) Artifact errors in local appearance; (b) frequency inconsistency: average spectrum of video frames for real videos and fake videos generated; (c) temporal inconsistency.

Table 1 An overview of fake video detection datasets.

Dataset	Fake video scale	Generation method	Video source	Task	Venue
Faceforensics [16]	1004	2	Generated	Face	arXiv'18
FakeAVCeleb [17]	19500	4	Generated	Face	NeurIPS'21
GVD [18]	11618	11	Collection	General	PRCV'24
GVF [19]	964	9	Generated	General	arXiv'24
GenVideo	1081083	20	Collection&Generated	General	–

various practical settings. GenVideo includes 1081083 generated videos and 1213511 real videos. The fake videos consist of those generated in-house and those collected from the internet, while the real videos come from the Youku-mPLUG [13], Kinetics-400 [14], and MSR-VTT [15] datasets. It should be noted that all fake videos are generated by prompts or images rather than being edited based on real videos. Due to the scale of the data, we can prevent detectors from merely learning the content differences between real and fake videos, instead focusing on subtle signs that determine video authenticity. As shown in Table 1, GenVideo offers more generation methods, a larger quantity, and diverse sources compared to previous artificial intelligence generated content (AIGC) video detection datasets [16, 17], which can enhance the diversity of our dataset. We propose two tasks that align with real-world detection challenges: (1) cross-generator video classification, where a trained detector is tasked with identifying videos from unseen generators; and (2) degraded video classification, where the detector assesses videos that have been degraded, such as those with low resolution, compression artifacts, or Gaussian blur. GenVideo can significantly advance the development of detectors aimed at identifying AI-generated videos in society.

In this paper, we evaluat state-of-the-art detection models [20–24] on GenVideo. However, the generalization capabilities of these models are compromised due to the limitations of existing image detection methods, which cannot model temporal inconsistencies, and video detection methods, which struggle to efficiently model local spatial inconsistencies. As shown in Figure 1, generated videos often exhibit both spatial and temporal artifacts, and modeling only one aspect (either spatial or temporal) may not be sufficient to cover all types of artifacts. Building a detector with satisfactory generalization performance requires modeling the spatial-temporal local details. In this paper, we introduce a plug-and-play module called detail mamba (DeMamba), which leverages a structured state space model to capture spatial-temporal inconsistencies across different regions, thereby discerning the authenticity of videos. Extensive experiments on GenVideo demonstrate that DeMamba can be used as a plug-and-play addition to existing feature extractors, significantly enhancing the generalizability and robustness of models.

Our contributions are summarized as follows.

- We introduce the first million-scale dataset for AI-generated video detection, GenVideo, which includes fake videos from various scenes, contents, and models.

- We design two tasks to evaluate the performance of detectors: cross-generator video classification and degraded video classification.

- We propose a plug-and-play detector, DeMamba, capable of modeling spatial-temporal inconsistencies. Extensive experimental results validate the generalizability and robustness of our DeMamba in identifying AI-generated videos.

2 Related studies

2.1 Video generation methods

Video generation methods [25,26] have become powerful tools for producing high-quality video content from textual or image prompts. Currently, video generation primarily encompasses two major tasks: text-to-video (T2V) and image-to-video (I2V). T2V involves inputting a text prompt to the model to generate videos based on textual instructions, while I2V aims to generate videos based on an input image, describing videos content or specific frames. Based on the types of these video generation methods, they can be separated into three categories: diffusion-based methods with U-Nets, diffusion-based methods with Transformers, and other methods.

Diffusion with U-Nets. A first family directly extends 2D diffusion to 3D latent volumes or adds lightweight temporal adapters on top of frozen image checkpoints. Early attempts [27–36] validate the idea, after which works such as Text2Video-Zero [37], AnimateDiff [38], PIA [39], and LaVie [6] refine the temporal module for better consistency. I2VGen-XL [40], VideoCrafter [41, 42], DynamiCrafter [43], ModelScope-T2V [44], SVD [4], VideoComposer [45] and SEINE [46] further push resolution or controllability. This paradigm excels at per-frame fidelity but its convolutional receptive field grows slowly over long sequences.

Diffusion with Transformers. A second line replaces the U-Net denoiser with a DiT/ViT-style Transformer, giving longer context windows with fewer parameters. Pioneering studies [8, 47] inspire Latte [48], CogVideo [49] and OpenAI's Sora [8,47], all of which use hierarchical or multi-frame-rate training. The backbone is usually a DiT variant [50]. These models capture global motion well but need careful positional encoding and memory sharding.

Non-diffusion generators. Finally, generative adversarial networks (GANs) [51, 52] and autoregressive token transformers [7, 53] remain relevant. Exploratory works [54, 55] analyze transformer fundamentals; FlashVideo [55] accelerates inference, while VideoPoet [56] and MagViT [57] quantize clips into vector quantization (VQ) tokens for ultra-long generation. Temporal GAN variants [51, 52, 57] yield crisp frames with low latency, yet struggle to maintain coherence.

2.2 AI-generated content detection

AI-generated visual content can amplify the spread of misinformation, so researchers have devoted considerable effort to designing forgery-detection models and constructing benchmark datasets. Recently, much of this work has focused on identifying generated images [58–61], drawing on AI-generated image datasets [62–64], and in particular on evaluating how well detectors handle images produced by previously unseen generative models. To date, the studies in [65, 66] have examined deepfake video detection, yet research on detecting generated videos in wider scenarios beyond human faces remains noticeably sparse. GenVidDet (GVD) [18] and GenerativeDeepfake (GVF) [19] represent early explorations of generic generated-video detection, but their scales are limited (GVD contains only 11k clips, GVF merely 964 videos), which restricts generalization and thorough evaluation. In contrast, our GenVideo dataset comprises more than one million generated videos. We hope that this paper will provide pioneering and insightful contributions to the field of AIGC video detection.

3 GenVideo

3.1 Overview of GenVideo

In response to the critical need for evaluating the generalizability of datasets and detectors (i.e., the capacity of training detectors to accurately recognize unseen videos from the open world) and the robustness of these detectors (i.e., their ability to maintain high performance against various corruptions to fake videos), we have developed the GenVideo dataset. This dataset is characterized by two main features.

- Large scale. The GenVideo dataset is organized hierarchically, encompassing cross-generators such as diffusion-based generators and transformer-based generators, and cross architectures within the same type of generator, like different motion modules combined with the same T2I base model [34, 35]. This structure facilitates covering a broader range of generated content and producing fake videos on a larger scale. The training (resp., testing) set in GenVideo contains a total of 2294594 (resp., 18588) video clips, comprising 1213511 (resp., 10000) real videos and 1081083 (resp., 8588) fake videos.
- Diverse content. GenVideo includes a wide array of high-quality fake videos sourced from open-source websites, along with videos produced using both user-trained and officially provided pre-trained video generation models, including T2V and I2V models. The generated video content encompasses a diverse range of scenes, including

Table 2 Statistics of real and generated videos in the GenVideo dataset.

Video source	Type	Task	Time	Resolution	FPS	Length (s)	Training set	Test set	Total count
Kinetics-400 [14]	Real	–	17.05	224–340	–	5–10	260232	–	1213511
Youku-mPLUG [13]		–	23.07	–	–	10–120	953279	–	
MSR-VTT [15]	Real	–	16.05	–	–	10–30	–	10000	10000
ZeroScope [68]	Fake	T2V	23.07	1024×576	8	3	133169	–	1081083
I2VGen-XL [40]		I2V	23.12	1280×720	8	2	61975	–	
SVD [4]		I2V	23.12	1024×576	8	4	149026	–	
VideoCrafter [42]		T2V	24.01	1024×576	8	2	39485	–	
Pika [69]		T2V&I2V	24.02	1088×640	24	3	98377	–	
DynamicCrafter [43]		I2V	24.03	1024×576	8	3	46205	–	
SD [39]		T2V&I2V	23.12	512–1024	8	2–6	200720	–	
SEINE [46]		I2V	24.04	1024×576	8	2–4	24737	–	
Latte [48]		T2V	24.03	512×512	8	2	149979	–	
OpenSora [47]		T2V	24.03	512×512	8	2	177410	–	
ModelScope [44]	Fake	T2V	23.03	256×256	8	4	–	700	8588
MorphStudio [70]		T2V	23.08	1280×720	8	2	–	700	
MoonValley [71]		T2V	24.01	1024×576	16	3	–	626	
HotShot [74]		T2V	23.10	672×384	8	1	–	700	
Show_1 [72]		T2V	23.10	576×320	8	4	–	700	
Gen2 [73]		I2V&T2V	23.09	896×512	24	4	–	1380	
Crafter [41]		T2V	23.04	256×256	8	4	–	1400	
LaVie [6]		T2V	23.09	1280×2048	8	2	–	1400	
Sora [8]		T2V	24.02	–	–	27–62	–	56	
WildScape		T2V&I2V	24	512–1024	8–16	2–6	–	926	
Total count	–	–	–	–	–	–	2294594	18588	2313182

landscapes, people, buildings, objects, and more. The duration of the videos is primarily between 2 to 6 s, and the aspect ratios of the video resolutions vary widely. This diverse collection ensures a comprehensive set of fake videos, significantly enriching the understanding of AI-generated video detection across numerous real-world contexts, and enhancing the generalizability and robustness of detectors.

Evaluation objectives. To avoid trivial detection caused by the same distribution from the same generator, as observed in previous AI-generated image detection datasets [59, 60, 67], we conduct two tasks to verify the performance of detection models: cross-generator generalization and degraded video classification. Cross-generator generalization refers to the model being trained on data generated by some generators and validated on unseen data generated by other generators, which is meant to test the model's generalization ability. Degraded video classification, on the other hand, is used to validate the model's robustness by testing its ability to recognize videos of different types of degradation.

3.2 Organization of GenVideo

The GenVideo dataset primarily consists of real videos and fake videos shown in Table 2. The real videos are mainly sourced from existing datasets related to video action datasets [14] and video description datasets [13,15]. The fake videos are obtained through external web scraping, internal generation pipelines based on open-source projects, and a number of existing video evaluation datasets [5].

Considering the emergence of video generation models, which primarily focus on diffusion-based methods [4, 39–43, 68] and methods based on autoregressive models [47, 48], the training set of the GenVideo dataset predominantly comprises videos generated by these two popular types of algorithms, as shown in Table 2. Additionally, following [5], we generate 98377 videos using the service provided by the Pika website [69]. To balance the quantity ratio between real videos and fake videos, we sampled 260232 and 953279 video clips from the existing video datasets Kinetics-400 [14] and Youku-mPLUG [13], respectively, to form the white sample of the training set.

For the test set, the real videos are sourced from the MSR-VTT dataset [15], which is a large video description dataset. The fake videos are mainly sourced from two parts: the first part comes from the Evalcrafter benchmark [5, 70–73], which is used to assess the temporal smoothness, quality, and other metrics of different generation models. The second part of the data comes from external web scraping, covering generated videos from existing popular video generation methods [27, 29, 31–36, 45, 53, 54, 56, 74–82]. This data encompasses most of the currently available video

generation methods and advanced derivative methods of mainstream video generation techniques. The scraped data are denoted as WildScape in Table 2.

3.3 Video collection details of GenVideo

We synthesize fake videos and gather real videos to construct the GenVideo dataset utilizing the hierarchical structure and the corresponding generators. It is crucial to underline that the primary objective of an AI-generated video detection dataset is to achieve robust and generalizable detection capabilities, rather than solely focusing on video quality for assessment purposes. A diverse and large-scale collection ensures that the dataset encompasses a wide range of video categories, facilitating detailed evaluations of AI-generated video detection algorithms and their effectiveness across various contexts.

Real video collection. Considering that fake videos from generators are limited to specific domains determined by training datasets such as Kinetics-400 [14] and Youku-mPLUG [13], we sample parts of videos from those datasets as the real part of the GenVideo dataset. Specifically, we randomly sample 953279 videos from Youku-mPLUG [13] and randomly slice 10-s segments from each video to form real samples.

Fake video collection. The guiding principle for collecting fake videos is to ensure maximal diversity in content and generators. We prioritize generating additional fake videos using the most recent generators due to their superior quality. To collect diverse fake videos from different resources as training samples, we have established a video generation pipeline for text-to-video generation and image-to-video generation. This pipeline facilitates the production of videos using popular generative mechanisms, including diffusion-based models [4, 39–43, 46, 68], transformer-based methods [47,48], and the service-based method Pika website [69]. For image-to-video generation, we employed various text-to-image models to produce diverse images, including different versions of stable diffusion (SD [83], SDXL [84]). In order to produce videos with rich semantics through different generative approaches, including semantically diverse text and image prompts across persons, objects, and diverse scenes, we first construct a rich prompt dictionary. In detail, we selected 100 common categories such as humans, animals, and plants as foreground keywords, and typical 20 scenes like "in the park" or "on the lawn" as background keywords. Leveraging a large language model [85], these foreground and background keywords are expanded into about 4000 comprehensive textual prompts. Besides, we also randomly sample 1600 textual prompts from VBench [86] with consideration of semantic diversity and style diversity. Next, each prompt from the constructed prompt dictionary is fed into T2V (resp., T2I) generators to produce diverse videos (resp., images). I2V generators leverage images generated from T2I generators to produce videos.

To assemble a representative test set, we investigated current video generators based on different model architectures [4,8,57] and scraped example videos from their projects, as illustrated by WildScape in Table 2. This includes prominent video generation models such as VideoPoet [53], Emu [32], and Sora [8]. Additionally, we collected videos generated by various condition-guided models [30, 33, 34] that focus on social contexts and characters. We also included some non-mainstream generation algorithms, such as those based on latent flow diffusion models [23], masked generative video transformer [56], or autoregressive models [87]. This approach ensures coverage of both popular algorithms and those generating high-quality content, particularly around character-centric videos. We integrated an existing video quality evaluation dataset [5] that includes typical generation methods and demonstrates relatively high generation quality.

4 DeMamba

4.1 Preliminaries

Structured state space sequence models (S4) [88–90] are grounded in continuous systems, facilitating the mapping of a one-dimensional function or sequence, denoted as $x(t) \in \mathbb{R}^L$ to $y(t) \in \mathbb{R}^L$, via an intermediary hidden state $h(t) \in \mathbb{R}^N$. In a formal context, S4 leverage the subsequent ordinary differential equation to represent the input data:

$$h'(t) = \mathbf{A}h(t) + \mathbf{B}x(t), \quad y(t) = \mathbf{C}h(t), \tag{1}$$

where $\mathbf{A} \in \mathbb{R}^{N\times N}$ embodies the system's evolutionary matrix, with $\mathbf{B} \in \mathbb{R}^{N\times L}$ and $\mathbf{C} \in \mathbb{R}^{L\times N}$ serving as the projection matrices. To navigate the transition from continuous to discrete modeling in contemporary S4, the Mamba framework utilizes a timescale parameter $\triangle$, facilitating the conversion of $\mathbf{A}$ and $\mathbf{B}$ into their discrete equivalents $\bar{\mathbf{A}}$ and $\bar{\mathbf{B}}$ through the zero-order hold methodology [88], expressed as

$$\bar{\mathbf{A}} = \exp(\triangle\mathbf{A}), \quad \bar{\mathbf{B}} = (\triangle\mathbf{A})^{-1}(\exp(\triangle\mathbf{A}) - \mathbf{I}) \cdot \triangle\mathbf{B}, \quad h_t = \bar{\mathbf{A}}h_{t-1} + \bar{\mathbf{B}}x_t, \quad y_t = \mathbf{C}h_t. \tag{2}$$

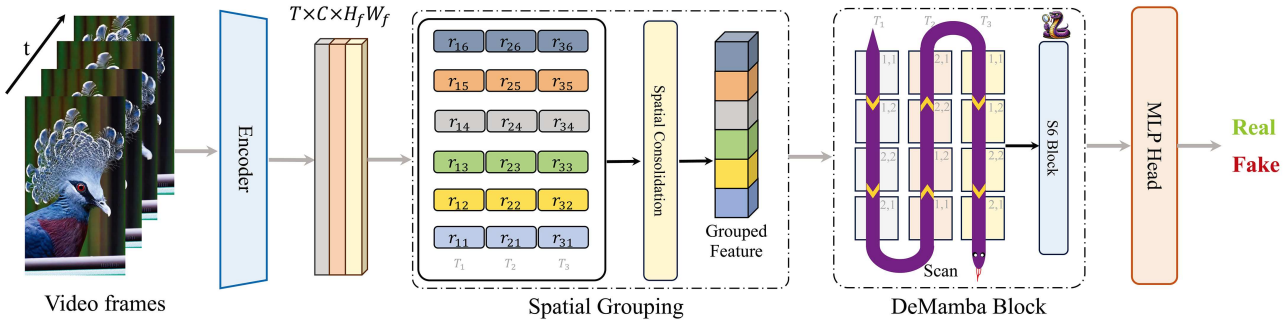

Figure 2 (Color online) The overall framework of our DeMamba.

Contrary to traditional models that primarily rely on linear time-invariant S4, Mamba [91] distinguishes itself by implementing a selection mechanism computed with scan for S4 (S6). Within the S6 framework, parameters $\mathbf{B} \in \mathbb{R}^{B\times L\times N}$, $\mathbf{C} \in \mathbb{R}^{B\times L\times N}$, and $\triangle \in \mathbb{R}^{B\times L\times D}$ are inherently derived from the input $\mathbf{x} \in \mathbb{R}^{B\times L\times D}$, formulating an intrinsic structure for contextual perceptiveness and adaptive modulation of weights.

4.2 AI-generated video detection with DeMamba module

Overview. As illustrated in Figure 2, our proposed method comprises a feature encoder, a DeMamba block, and a multi-layer perceptron (MLP) classification head. Specifically, we employ state-of-the-art vision encoders (e.g., CLIP [92] and XCLIP [23]) to encode the input video frames $\mathbf{X}^v \in \mathbb{R}^{3\times T\times H\times W}$ into a sequence of features, denoted by $\mathbf{F} \in \mathbb{R}^{T\times C\times H_f\times W_f}$, where C symbolizes the channel dimensionality, and H_f, W_f represent the spatial dimensions, i.e., height and width of the feature maps, respectively. Following this, the extracted features are spatially grouped, and the DeMamba module is applied to model the intra-group feature consistency. Finally, we aggregate the features from different groups to determine whether the input video is generated by AI.

DeMamba block. We first apply spatial consolidation: given the feature $\mathbf{F}$, we split it into s^2 zones along both the height and width dimension where each zone of $\mathbf{F}$ is denoted as $\mathbf{F}_{jk} \in \mathbb{R}^{T\times C\times (H_f/s)\times (W_f/s)}$, where $j,k = \{1,\ldots,s\}$. In Figure 2, we adapt the 1D Mamba layer for handling spatial-temporal input by expanding its capability to a 3D scan. In the previous Mamba approaches [91, 93, 94], a sweep-scan mechanism was utilized, which might not effectively capture the inherent contextual relationships between adjacent tokens. To address this limitation, we propose a continuous scan strategy for each segmented region, aimed at maintaining spatial continuity throughout the entire scanning phase. Suppose a zone consists of four spatial positions: (1,1), (1,2), (2,1), and (2,2), corresponding to the top-left, top-right, bottom-left, and bottom-right corners, respectively. The sweep-scan order is (1,1) → (1,2)→ (2,1) → (2,2), whereas in the continuous scan, the order is (1,1) → (1,2) → (2,2) → (2,1). This method organizes spatial tokens based on their proximity and subsequently aligns them sequentially across successive frames.

It facilitates the coherent integration of spatial and temporal dynamics, enhancing the capability of the model to capture complex spatial-temporal relationships. After modeling the spatial-temporal inconsistency of each partitioned region using DeMamba, we can obtain the feature $\mathbf{F}'_{jk} \in \mathbb{R}^{T\times C\times (H_f W_f/s^2)}$, where $j,k = \{1,\ldots,s\}$.

Classification head. To leverage more comprehensive features for classification, we aggregate both global and local features. Specifically, we temporally and spatially average the input features $\mathbf{F}$ before the DeMamba block to obtain the global feature $\mathbf{F}^{\text{global}} \in \mathbb{R}^C$, and average pool the temporal and spatial features $\mathbf{F}'_{jk}$ after the DeMamba processing into pooled features $\mathbf{F}^{\text{pool}}_{jk} \in \mathbb{R}^C$. Then we concatenate the local features with the global features and apply a simple MLP for classification:

$$y_{\text{pred}} = \text{Sigmoid}(\text{MLP}([\mathbf{F}^{\text{global}}; \mathbf{F}^{\text{pool}}_{11}; \ldots; \mathbf{F}^{\text{pool}}_{ss}])). \tag{3}$$

Finally, we use binary cross-entropy loss to train our model to classify real/fake videos.

5 Experiments

5.1 Implementation details

Datasets. To comprehensively analyze the performance of various detectors, we divided the dataset into two distinct parts: the basic training set D_{train} and the out-of-domain test set $D_{\text{v-ood}}$. D_{train} and $D_{\text{v-ood}}$ contain fake

Table 3 Training parameter settings.

Model	LR	Frames	Epochs	Scheduler
F3Net	1e−5	8	30	[20, 25], lr×0.1
NPR	1e−5	8	30	[20, 25], lr×0.1
STIL	1e−5	8	30	[20, 25], lr×0.1
VideoMAE-B	1e−5	16	30	[20, 25], lr×0.1
VideoMamba-M	1e−5	16	30	[20, 25], lr×0.1
CLIP-B-FT	1e−6	8	10	–
MINTIME-CLIP-B	1e−6	8	10	–
FTCN-CLIP-B	1e−6	8	10	–
TALL	1e−6	8	10	–
XCLIP-FT	1e−6	8	10	–
DeMamba-CLIP-FT	1e−6	8	10	–
DeMamba-XCLIP-FT	1e−6	8	10	–

videos produced by different generative methods and real videos from different sources. D_{train} includes 1213511 real videos and 1081083 generated videos produced using 10 baseline generative methods. $D_{\text{v-ood}}$ contains 10000 real videos and 8588 generated videos created with 10 generative methods.

Evaluation metrics. Consistent with the methodologies employed in prior studies, the effectiveness of the detectors is measured by evaluating the recall on generated samples, overall average precision (AP), and F1 score. The recall on generated samples reflects the extent to which generated samples are retrieved, while AP and F1 indicate the rate of false positives on real samples. The classification threshold is set to 0.5. For image-based detection methods, frame-level predictions are aggregated into video-level predictions to ensure consistent analysis across different media formats. It is particularly noteworthy that when evaluating a dataset generated by a specific synthesis method, the recall for that synthesis method is calculated based on the dataset itself. Additionally, in the process of calculating AP and F1, all real videos are taken into account to achieve a more comprehensive assessment.

Baselines. Three types of image-level methods and seven video-level methods are treated as baselines for comparison: F3Net [16], NPR [17], CLIP [88], STIL [20], XCLIP [21], VideoMAE [95], VideoMamba [24], MINTIME [96], FTCN [97], TALL [18].

Training Parameters. We adopted the experimental settings reported in the original papers of all methods and performed hyperparameter tuning accordingly. All methods were optimized using a batch size of 128, the AdamW optimizer, and binary cross-entropy loss. The specific training hyperparameters for each method are detailed in Table 3. It is important to note that the hyperparameters for both many-to-many and one-to-many generalization tasks are identical for each model. All of our experiments were conducted on a system equipped with 8 Tesla A-100 80G GPUs and an Intel(R) Xeon(R) Platinum 8369B CPU @ 2.90 GHz.

5.2 Task1: cross generator generalization

Due to the rapid iteration of generation methods, we propose a cross-dataset generalization task to test the generalization performance of detectors. Specifically, it consists of two types of generalization tasks: (1) the one-to-many generalization task, and (2) the many-to-many generalization task.

One-to-many generalization task. Following AI-generated image detection setting [59,61], we also perform a one-to-many generalization task. Unlike the many-to-many generalization task, the one-to-many generalization task involves training on one baseline category and then testing on each subset and the average detection performance on $D_{\text{v-ood}}$. As shown in Table 4, our DeMamba-XCLIP-FT achieves better generalization performance in three one-to-many generalization tasks due to the learning of spatial-temporal inconsistency in DeMamba.

Many-to-many generalization task. This task involves training on 10 baseline categories and then testing on each subset and the average detection performance on $D_{\text{v-ood}}$. As shown in Table 5, video models achieve better recognition accuracy compared to image models because video models can model temporal sequences. Moreover, our DeMamba model can be effectively integrated into existing models, achieving significant improvements. For example, integrating the DeMamba module into XCLIP results in DeMamba-XCLIP-FT achieving an average recall/F1/AP of 0.9392/0.9020/0.9710, respectively, which marks an improvement of 11.26%/17.72%/12.02% in recall/F1/AP over the original XCLIP.

Table 4 Comparisons to the SOTAs on the one-to-many generalization task. The best performance is highlighted in bold.

Model	Type	Pika			OpenSora			SEINE		
		R	F1	AP	R	F1	AP	R	F1	AP
NPR	Image	0.5144	0.5306	0.6497	0.5929	0.5232	0.5763	0.4618	0.5385	0.6111
STIL	Video	0.7383	0.5165	0.6304	0.4336	0.4892	0.5256	0.7239	0.5057	0.6083
MINTIME-CLIP-B	Video	0.6135	0.6538	0.7443	0.2166	0.3263	0.5738	0.6889	0.7269	0.8286
FTCN-CLIP-B	Video	0.6287	0.6423	0.7194	0.1863	0.3005	0.5689	0.6279	0.6991	0.8036
TALL	Video	0.7141	0.5572	0.6225	0.4916	0.5315	0.5710	0.6568	0.6085	0.6805
Video-Mamba	Video	0.6985	0.6374	0.7225	0.5183	0.5235	0.6389	0.7432	0.7175	0.8135
XCLIP-B-FT	Video	0.6096	0.6579	0.7831	0.6615	0.6497	0.7154	0.7201	**0.7898**	0.8880
DeMamba-XCLIP-B-FT	Video	**0.7572** (+0.1476)	**0.7263** (+0.0684)	**0.8166** (+0.0335)	**0.7382** (+0.0767)	**0.6713** (+0.0216)	**0.7382** (+0.0228)	**0.8098** (+0.0897)	0.7873 (−0.0025)	**0.8943** (+0.0063)

5.3 Task2: degraded video classification

In practical detection scenarios, the robustness of the detector to perturbations is also of paramount importance. In this regard, we investigated the impact of perturbations on the detector on 8 different types: H.264 compression, JPEG compression, FLIP, Crop, text watermark, image watermark, Gaussian blur, and color transform. Here, we provide the specific implementation details for each task of degraded video classification.

(1) H.264 compression. H.264, also known as advanced video coding (AVC), is a widely used video compression standard. In this paper, we set the CRF to 28 to compress the video.

(2) JPEG compression. JPEG compression is a widely used image compression standard designed for efficient compression of digital images. The JPEG algorithm is based on the characteristics of the human visual system, taking advantage of the insensitivity of human eyes to the loss of image details, thus achieving lossy compression of data. In this paper, we set the quality to 35 for the degradation experiment.

(3) FLIP. We randomly select either horizontal flip or vertical flip with equal probability for the degradation experiment.

(4) Crop. We randomly crop the video from the original video with a scale of 71% to 93%.

(5) Text watermark. We randomly add textual watermarks at random positions in the video.

(6) Image watermark. We randomly add visual watermarks at random positions in the video.

(7) Gaussian blur. We add Gaussian blur to the video with a setting of $\sigma = 7$.

(8) Color transform. We randomly select one color transformation from brightness, contrast, saturation, hue, and set the parameter to 0.5.

Table 6 shows the performance of the models trained in the many-to-many task under the influence of these perturbations. We can observe that, under data degradation conditions, DeMamba-XCLIP-FT still achieves the best performance in tasks other than JPEG compression, indicating that our model demonstrates good robustness when facing degraded data. DeMamba exhibits poorer generalization when confronted with stronger JPEG compression, which may be because compression hinders effective modeling of inconsistencies, thereby confusing real and generated samples. However, compared to XCLIP, DeMamba still effectively enhances robustness.

5.4 Ablation study

Ablation testing. We conduct ablation experiments to validate the effectiveness of DeMamba. As shown in Table 7, DeMamba effectively enhances the generalization performance of the model. Additionally, when using fused features, the model achieves its best performance.

Influence of scanning orders. As shown in Table 8, the continuous scan proposed in this paper effectively enhances performance compared to the traditional scanning method.

Influence of different zone sizes. We investigate the impact of zone size in dividing zones in DeMamba on modeling temporal inconsistency. As shown in Table 9, the best performance is observed when the zone size is 2. Smaller zones enable the model to concentrate more on local details, leading to superior modeling performance. However, excessively small zones may result in the loss of spatial contextual information. Therefore, selecting an appropriate zone size is crucial.

Necessity of large volume of GenVideo for generalizability. To clarify the impact of data volume on detection performance, we conduct experiments using DeMamba on various scaled subsets of the GenVideo dataset. We design two experiments to explore: (1) variation in the number of training samples within each generator: for the data generated by each generator, we randomly selected between 100 and 20000 video samples to train the

Table 5 Comparisons to the SOTAs in F1 score (F1) and average precision (AP) on the many-to-many generalization task. The best performance is highlighted in bold.

Model	Detection level	Metric	Sora	Morph studio	Gen2	HotShot	LaVie	Show-1	Moon valley	Crafter	Model scope	Wild scape	Avg.
F3Net	Image	R	0.8393	0.9971	0.9862	0.7757	0.5700	0.3657	0.9952	0.9971	0.8943	0.7678	0.8188
		F1	0.5000	0.9406	0.9628	0.8169	0.6988	0.4904	0.9332	0.9688	0.8873	0.8251	0.8024
		AP	0.6827	0.9989	0.9967	0.8935	0.8524	0.6317	0.9958	0.9989	0.9380	0.8841	0.8873
NPR	Image	R	0.9107	0.9957	0.9949	0.2429	0.8964	0.5771	0.9712	0.9986	0.9429	0.8780	0.8408
		F1	0.2786	0.8441	0.9131	0.3028	0.8627	0.5944	0.8170	0.9164	0.8184	0.8163	0.7164
		AP	0.6717	0.9914	0.9920	0.2276	0.9391	0.6176	0.9633	0.9972	0.9415	0.9040	0.8245
CLIP-B-FT	Image	R	0.9464	0.9986	0.9138	0.7729	0.8814	0.8600	0.9968	0.9979	0.8429	0.8467	0.9057
		F1	0.2818	0.8422	0.8691	0.7204	0.8521	0.7698	0.8252	0.9137	0.7608	0.7955	0.7631
		AP	0.8067	0.9967	0.9524	0.8220	0.9348	0.8862	0.9955	0.9979	0.8693	0.8908	0.9152
STIL	Video	R	0.7857	0.9814	0.9804	0.7600	0.6179	0.5329	0.9936	0.9736	0.9457	0.6501	0.8222
		F1	0.3805	0.9068	0.9458	0.7824	0.7232	0.6217	0.9039	0.9433	0.8884	0.7267	0.7823
		AP	0.5721	0.9908	0.9932	0.8619	0.8224	0.7043	0.9925	0.9896	0.9718	0.8132	0.8712
VideoMAE-B	Video	R	0.6786	0.9600	0.9841	0.9614	0.7714	0.8043	0.9744	0.9693	0.9629	0.6836	0.8750
		F1	0.6230	0.9593	0.9819	0.9600	0.8608	0.8722	0.9644	0.9745	0.9615	0.7977	0.8955
		AP	0.6649	0.9885	0.9977	0.9927	0.9655	0.9531	0.9949	0.9969	0.9927	0.9074	0.9454
VideoMamba-M	Video	R	0.6932	0.9855	0.9832	0.9536	0.7865	0.8233	0.9765	0.9687	0.9635	0.7124	0.8846
		F1	0.6252	0.9637	0.9775	0.9452	0.8727	0.8985	0.9662	0.9712	0.9619	0.8125	0.8994
		AP	0.6757	0.9905	0.9987	0.9927	0.9783	0.9512	0.9953	0.9973	0.9945	0.9052	0.9480
MINTIME-CLIP-B	Video	R	0.8929	1.0000	0.9899	0.2643	0.9679	0.9814	0.9984	1.0000	0.8429	0.8238	0.8762
		F1	0.4902	0.9340	0.9606	0.3760	0.9499	0.9246	0.9266	0.9659	0.8495	0.8539	0.8231
		AP	0.8321	0.9999	0.9967	0.5084	0.9920	0.9927	0.9976	0.9999	0.9183	0.9177	0.9155
FTCN-CLIP-B	Video	R	0.8750	1.0000	0.9891	0.1771	0.9771	0.9186	1.0000	1.0000	0.8529	0.8283	0.8618
		F1	0.7840	0.9859	0.9873	0.2922	0.9813	0.9442	0.9843	0.9929	0.9073	0.8960	0.8755
		AP	0.9179	0.9999	0.9979	0.4594	0.9976	0.9780	0.9999	0.9999	0.9469	0.9232	0.9221
TALL	Video	R	0.9107	0.9828	0.9783	0.8300	0.7657	0.7957	0.9952	0.9893	0.9414	0.6631	0.8852
		F1	0.2615	0.8240	0.8964	0.7430	0.7782	0.7234	0.8133	0.9032	0.8027	0.6736	0.7419
		AP	0.7115	0.9689	0.9851	0.7938	0.8459	0.7938	0.9879	0.9902	0.9270	0.7647	0.8791
DeMamba-CLIP-B-FT	Video	R	0.9571	1.0000	0.9870	0.6914	0.9243	0.9329	1.0000	1.0000	0.8357	0.8294	0.9158 (+0.0101)
		F1	0.6463	0.9615	0.9739	0.7803	0.9414	0.9276	0.9572	0.9804	0.8723	0.8782	0.8919 (+0.1288)
		AP	0.8550	1.0000	0.9959	0.7615	0.9678	0.9699	0.9997	1.0000	0.8980	0.8972	0.9345 (+0.1930)
XCLIP-B-FT	Video	R	0.8214	0.9957	0.9362	0.6129	0.7936	0.6971	0.9792	0.9979	0.7714	0.8359	0.8441
		F1	0.3147	0.8805	0.9041	0.6530	0.8242	0.7099	0.8602	0.9370	0.7570	0.8212	0.7662
		AP	0.6442	0.9973	0.9678	0.7098	0.9035	0.7728	0.9734	0.9984	0.8201	0.8897	0.8677
DeMamba-XCLIP-FT	Video	R	0.9821	1.0000	0.9986	0.6543	0.9486	0.9886	1.0000	1.0000	0.9286	0.8909	**0.9392** (+0.0951)
		F1	0.6467	0.9602	0.9790	0.7539	0.9537	0.9551	0.9557	0.9797	0.9240	0.9120	**0.9020** (+0.1358)
		AP	0.9332	1.0000	0.9997	0.8555	0.9897	0.9960	0.9998	1.0000	0.9777	0.9575	**0.9710** (+0.1043)

model; (2) variation in the number of training samples across different generators: we randomly selected subsets to simulate scenarios with 1 to 10 generators in the dataset. In each scenario, we collected 20000 video samples for the videos generated by each generator. Note that, for real videos, we randomly selected an equal number of samples to the total number of training generated video data to participate in the training. The test sets used in these experiments are the same as those mentioned in the main paper; they consist of the full test set videos, and the model selected for the experiment is DeMamba-XCLIP-FT. The comparative results of these experiments are detailed in Figure 3. The results clearly indicate a significant association between the scale of the dataset and the improvement in detection performance, further affirming the critical role of massive data in enhancing detection capabilities.

Training and inference efficiency. As detailed in Table 10, we report the training and inference times on 8 Tesla A100-80G GPUs, with QPS (queries per second) measured on a single GPU at batch size 1. Incorporating DeMamba into XCLIP does not significantly impact inference time compared to CLIP and XCLIP. Slower training is due to Mamba's training characteristics, but it offers efficient inference post-training. Our method maintains

Table 6 Robustness evaluation of different detectors on many-to-many generalization task. The best performance is highlighted in bold.

Model	Detection level	Metric	Original	Compression		Transformation		Watermarks		Gaussian blur	Color transform
				CRF = 28	JPEG	Flip	Crop	Text	Image		
F3Net	Image	R	0.8188	0.7721 (−0.0467)	0.8108 (−0.0080)	0.8041 (−0.0147)	0.6782 (−0.1406)	0.7513 (−0.0675)	0.7668 (−0.0520)	0.8025 (−0.0163)	0.8102 (−0.0086)
		F1	0.8024	0.7967 (−0.0057)	0.6984 (−0.1040)	0.7945 (−0.0079)	0.5988 (−0.2036)	0.7858 (−0.0166)	0.7944 (−0.0080)	0.7910 (−0.0114)	0.7912 (−0.0112)
		AP	0.8873	0.8739 (−0.0134)	0.8102 (−0.0771)	0.8770 (−0.0173)	0.6705 (−0.2168)	0.8701 (−0.0172)	0.8793 (−0.0070)	0.8717 (−0.0156)	0.8717 (−0.0156)
NPR	Image	R	0.8408	0.8012 (−0.0396)	0.7375 (−0.1033)	0.8365 (−0.0043)	0.7144 (−0.1264)	0.8285 (−0.0123)	0.8200 (−0.0208)	0.8392 (−0.0016)	0.8378 (−0.0030)
		F1	0.7164	0.6382 (−0.0782)	0.4588 (−0.2576)	0.6896 (−0.0268)	0.1655 (−0.5509)	0.5580 (−0.1584)	0.6583 (−0.0581)	0.4791 (−0.2373)	0.6564 (−0.0600)
		AP	0.8245	0.7612 (−0.0633)	0.5372 (−0.2873)	0.7905 (−0.0340)	0.3636 (−0.4609)	0.6716 (−0.1528)	0.7702 (−0.0543)	0.5755 (−0.2490)	0.7681 (−0.0564)
CLIP-B-FT	Image	R	0.9057	0.8253 (−0.0804)	0.7632 (−0.1425)	0.8878 (−0.0179)	0.6782 (−0.2275)	0.7877 (−0.1180)	0.7832 (−0.1225)	0.8812 (−0.0245)	0.8809 (−0.0248)
		F1	0.7631	0.7476 (−0.0155)	0.6982 (−0.0649)	0.7533 (−0.0098)	0.6273 (−0.1358)	0.7125 (−0.0506)	0.7537 (−0.0094)	0.6743 (−0.0888)	0.7542 (−0.0089)
		AP	0.9152	0.9042 (−0.0110)	0.8533 (−0.0619)	0.9105 (−0.0047)	0.7635 (−0.1517)	0.8894 (−0.0258)	0.9105 (−0.0047)	0.8455 (−0.0697)	0.9105 (−0.0047)
STIL	Video	R	0.8222	0.7712 (−0.0510)	0.7215 (−0.1007)	0.8076 (−0.0146)	0.7678 (−0.0544)	0.7694 (−0.0528)	0.7401 (−0.0821)	0.8205 (−0.0017)	0.8195 (−0.0027)
		F1	0.7823	0.7412 (−0.0411)	0.5124 (−0.2699)	0.7435 (−0.0388)	0.6589 (−0.1234)	0.7276 (−0.0547)	0.7301 (−0.0522)	0.6472 (−0.1351)	0.7566 (−0.0257)
		AP	0.8712	0.8504 (−0.0208)	0.5938 (−0.2774)	0.8521 (−0.0191)	0.7638 (−0.1074)	0.8259 (−0.0453)	0.8309 (−0.0403)	0.7454 (−0.1258)	0.8665 (−0.0047)
VideoMamba-M	Video	R	0.8846	0.8425 (−0.0421)	0.8775 (−0.0071)	0.8753 (−0.0093)	0.5753 (−0.3093)	0.8259 (−0.0587)	0.8043 (−0.0803)	0.8812 (−0.0034)	0.8835 (−0.0011)
		F1	0.8994	0.8705 (−0.0289)	**0.8477** (−0.0517)	0.8562 (−0.0432)	0.6352 (−0.2642)	0.8715 (−0.0279)	0.8843 (−0.0151)	0.7537 (−0.1457)	0.8687 (−0.0307)
		AP	0.9480	0.9207 (−0.0273)	**0.9253** (−0.0227)	0.9275 (−0.0205)	0.6442 (−0.3038)	0.9320 (−0.0160)	0.9376 (−0.0104)	0.8439 (−0.1040)	0.9465 (−0.0015)
MINTIME-CLIP-B	Video	R	0.8724	0.8215 (−0.0509)	0.8199 (−0.0525)	0.8215 (−0.0508)	0.5534 (−0.3190)	0.8084 (−0.0640)	0.7972 (−0.0752)	0.8628 (−0.0096)	0.8715 (−0.0009)
		F1	0.8700	0.8486 (−0.0214)	0.8417 (−0.0283)	0.8486 (−0.0214)	0.5485 (−0.3215)	0.8468 (−0.0232)	0.8379 (−0.0321)	0.4734 (−0.3966)	0.8658 (−0.0042)
		AP	0.9369	0.9207 (−0.0158)	0.9143 (−0.0226)	0.9207 (−0.0158)	0.6199 (−0.3170)	0.9071 (−0.0298)	0.9227 (−0.0142)	0.8146 (−0.1223)	0.9364 (−0.0005)
FTCN-CLIP-B	Video	R	0.8935	0.8519 (−0.0416)	0.8127 (−0.0808)	0.8858 (−0.0077)	0.5546 (−0.3389)	0.8284 (−0.0651)	0.8547 (−0.0388)	0.8874 (−0.0061)	0.8904 (−0.0031)
		F1	0.8739	0.8730 (−0.0009)	0.5495 (−0.3244)	0.8484 (−0.0451)	0.5573 (−0.3166)	0.8689 (−0.0050)	0.8648 (−0.0091)	0.6002 (−0.2735)	0.8638 (−0.0101)
		AP	0.9418	0.9294 (−0.0124)	0.6884 (−0.2534)	0.9316 (−0.0102)	0.6015 (−0.3403)	0.9256 (−0.0162)	0.9386 (−0.0032)	0.8503 (−0.0915)	0.9389 (−0.0029)
TALL	Video	R	0.8852	0.8413 (−0.0439)	0.4272 (−0.4580)	0.8785 (−0.0067)	0.7233 (−0.1619)	0.8061 (−0.0791)	0.8045 (−0.0807)	0.8834 (−0.0018)	0.8816 (−0.0036)
		F1	0.7419	0.7411 (−0.0008)	0.5309 (−0.2110)	0.7248 (−0.0171)	0.5844 (−0.1575)	0.7329 (−0.0090)	0.7360 (−0.0059)	0.5996 (−0.1423)	0.6719 (−0.0700)
		AP	0.8791	0.8673 (−0.0118)	0.6439 (−0.2352)	0.8774 (−0.0017)	0.6744 (−0.2047)	0.8608 (−0.0183)	0.8767 (−0.0024)	0.7880 (−0.0911)	0.8439 (−0.0352)
DeMamba-CLIP-FT	Video	R	0.9158	0.8572 (−0.0586)	0.8479 (−0.0679)	0.9047 (−0.0111)	0.6172 (−0.2986)	0.8352 (−0.0806)	0.8664 (−0.0494)	0.9037 (−0.0121)	0.9105 (−0.0053)
		F1	0.8919	0.8750 (−0.0169)	0.6325 (−0.2594)	0.8912 (−0.0007)	0.5233 (−0.3686)	0.8455 (−0.0464)	0.8895 (−0.0024)	0.7672 (−0.1247)	0.8910 (−0.0009)
		AP	0.9345	0.9093 (−0.0252)	0.7403 (−0.1942)	0.9344 (−0.0001)	0.6112 (−0.3233)	0.9008 (−0.0337)	0.9335 (−0.0010)	0.8299 (−0.1046)	0.9339 (−0.0006)
XCLIP-B-FT	Video	R	0.8441	0.8387 (−0.0054)	0.5377 (−0.3064)	0.8378 (−0.0063)	0.5597 (−0.2844)	0.8388 (−0.0023)	0.8254 (−0.0187)	0.8195 (−0.0216)	0.8421 (−0.0020)
		F1	0.7662	0.7329 (−0.0333)	0.4188 (−0.3474)	0.7592 (−0.0070)	0.3062 (−0.4600)	0.7595 (−0.0067)	0.7593 (−0.0069)	0.6322 (−0.1340)	0.7488 (−0.0174)
		AP	0.8677	0.8653 (−0.0024)	0.5411 (−0.3266)	0.8676 (−0.0001)	0.4610 (−0.4067)	0.8671 (−0.0006)	0.8674 (−0.0003)	0.6805 (−0.1489)	0.8601 (−0.0076)
DeMamba-XCLIP-FT	Video	R	**0.9392**	**0.9052** (−0.0340)	**0.9326** (−0.0116)	**0.9376** (−0.0016)	**0.7298** (−0.2244)	**0.8915** (−0.0477)	**0.9094** (−0.0288)	**0.9382** (−0.0010)	**0.9342** (−0.0050)
		F1	**0.9020**	**0.8770** (−0.0250)	0.6823 (−0.2197)	**0.8820** (−0.0200)	**0.6595** (−0.2425)	**0.8755** (−0.0265)	**0.8990** (−0.0030)	**0.7962** (−0.1058)	**0.9005** (−0.0005)
		AP	**0.9710**	**0.9600** (−0.0110)	0.7629 (−0.2081)	**0.9676** (−0.0034)	**0.6718** (−0.2992)	**0.9430** (−0.0280)	**0.9707** (−0.0003)	**0.8505** (−0.1205)	**0.9705** (−0.0005)

Table 7 Ablation study of DeMamba. The best performance is highlighted in bold.

Model	Metric	Sora	Morph studio	Gen2	HotShot	LaVie	Show-1	Moon valley	Crafter	Model scope	Wild scape	Avg.
w/o DeMamba	R	0.8214	0.9957	0.9362	0.6129	0.7936	0.6971	0.9792	0.9979	0.7714	0.8359	0.8411 (−0.0971)
	F1	0.3147	0.8805	0.9041	0.6530	0.8242	0.7099	0.8602	0.9370	0.7570	0.8212	0.7662 (−0.1358)
	AP	0.6442	0.9973	0.9678	0.7098	0.9035	0.7728	0.9734	0.9984	0.8201	0.8897	0.8677 (−0.1043)
w/o global	R	0.9733	0.9985	0.9945	0.6766	0.9355	0.9788	1.0000	1.0000	0.9134	0.8654	0.9336 (−0.0046)
	F1	0.4252	0.9053	0.9475	0.7183	0.9176	0.8960	0.8962	0.9508	0.8618	0.8562	0.8375 (−0.0645)
	AP	0.8732	0.9935	0.9922	0.8244	0.9799	0.9942	0.9942	0.9985	0.9732	0.9498	0.9573 (−0.0137)
DeMamba-XCLIP-FT	R	0.9821	1.0000	0.9986	0.6543	0.9486	0.9886	1.0000	1.0000	0.9286	0.8909	**0.9382**
	F1	0.6467	0.9602	0.9790	0.7539	0.9537	0.9551	0.9557	0.9797	0.9240	0.9120	**0.9020**
	AP	0.9332	1.0000	0.9997	0.8555	0.9897	0.9960	0.9998	1.0000	0.9777	0.9575	**0.9710**

Table 8 Influence of scanning order in DeMamba. The best performance is highlighted in bold.

Model	Scan order	Metric	Sora	Morph studio	Gen2	HotShot	LaVie	Show-1	Moon valley	Crafter	Model scope	Wild scape	Avg.
DeMamba -XCLIP-FT	Sweep	R	0.9521	1.0000	0.9255	0.6177	0.9269	0.9633	0.9987	0.9932	0.8153	0.7311	0.8924 (−0.0468)
		F1	0.4649	0.9211	0.9200	0.6906	0.9212	0.9029	0.9124	0.9557	0.8207	0.7856	0.8295 (−0.0725)
		AP	0.9332	0.9995	0.9732	0.8555	0.9588	0.9754	0.9966	0.9754	0.8713	0.8082	0.9347 (−0.0363)
	Continuous	R	0.9821	1.0000	0.9986	0.6543	0.9486	0.9886	1.0000	1.0000	0.9286	0.8909	0.9392
		F1	0.6467	0.9602	0.9790	0.7539	0.9537	0.9551	0.9557	0.9797	0.9240	0.9120	**0.9020**
		AP	0.9332	1.0000	0.9997	0.8555	0.9897	0.9960	0.9998	1.0000	0.9777	0.9575	**0.9710**

Table 9 Influence of different zone sizes in DeMamba. The best performance is highlighted in bold.

Model	Zone size	Metric	Sora	Morph studio	Gen2	HotShot	LaVie	Show-1	Moon valley	Crafter	Model scope	Wild scape	Avg.
DeMamba-CLIP-FT	1×1	R	1.0000	0.9986	0.8478	0.9557	0.8143	0.8229	1.0000	0.9979	0.7829	0.8110	0.9031
		F1	0.3218	0.8556	0.8398	0.8334	0.8216	0.7624	0.8414	0.9215	0.7390	0.7849	0.7722
		AP	0.9380	0.9964	0.9178	0.9603	0.9004	0.8613	0.9992	0.9981	0.8212	0.8696	0.9262
	2×2	R	0.9571	1.0000	0.9870	0.6914	0.9243	0.9329	1.0000	1.0000	0.8357	0.8294	0.9158
		F1	0.6463	0.9615	0.9739	0.7803	0.9414	0.9276	0.9572	0.9804	0.8723	0.8782	**0.8919**
		AP	0.8550	1.0000	0.9959	0.7615	0.9678	0.9699	0.9997	1.0000	0.8980	0.8972	**0.9345**
	7×7	R	0.9821	1.0000	0.9841	0.6500	0.9400	0.9543	1.0000	1.0000	0.8086	0.8434	0.9163
		F1	0.3724	0.8855	0.9308	0.6811	0.9085	0.8630	0.8737	0.9393	0.7828	0.8271	0.8064
		AP	0.8728	0.9999	0.9906	0.7351	0.9747	0.9700	0.9976	0.9997	0.8575	0.8995	0.9297
	14×14	R	1.0000	0.9986	0.9341	0.8857	0.8171	0.7014	1.0000	0.9993	0.7643	0.8002	0.8901
		F1	0.4628	0.9149	0.9214	0.8550	0.8555	0.7430	0.9059	0.9556	0.7845	0.8245	0.8223
		AP	0.9494	0.9942	0.9703	0.9356	0.9402	0.8456	0.9912	0.9922	0.7760	0.8946	0.9289
DeMamba-XCLIP-FT	1×1	R	0.9107	1.0000	0.9935	0.7371	0.8950	0.9629	1.0000	1.0000	0.9471	0.8121	0.9258
		F1	0.5952	0.9569	0.9748	0.8066	0.9227	0.9387	0.9521	0.9780	0.9298	0.8644	0.8919
		AP	0.8665	0.9999	0.9983	0.9064	0.9785	0.9881	0.9991	0.9999	0.9843	0.9218	0.9643
	2×2	R	0.9821	1.0000	0.9986	0.6543	0.9486	0.9886	1.0000	1.0000	0.9286	0.8909	0.9392
		F1	0.6467	0.9602	0.9790	0.7539	0.9537	0.9551	0.9557	0.9797	0.9240	0.9120	**0.9020**
		AP	0.9332	1.0000	0.9997	0.8555	0.9897	0.9960	0.9998	1.0000	0.9777	0.9575	**0.9710**
	7×7	R	0.9643	1.0000	0.9957	0.5600	0.8871	0.9600	1.0000	1.0000	0.9443	0.7948	0.9106
		F1	0.7013	0.9689	0.9821	0.6895	0.9244	0.9485	0.9653	0.9842	0.9409	0.8622	0.8967
		AP	0.9602	1.0000	0.9997	0.8285	0.9969	0.9952	0.9998	1.0000	0.8651	0.9745	0.9620
	14×14	R	0.7679	1.0000	0.9906	0.2286	0.8471	0.9514	0.9952	1.0000	0.8771	0.7225	0.8380
		F1	0.7748	0.9908	0.9909	0.3670	0.9126	0.9659	0.9873	0.9954	0.9253	0.8326	0.8742
		AP	0.8043	1.0000	0.9976	0.6569	0.9849	0.9931	0.9979	0.9991	0.9687	0.8831	0.9286

competitive computational times relative to other video models, outperforming F3Net, STIL, and TALL.

Impact of different generators on model performance. To investigate how data from different generators contribute to generalization, we conducted ablation experiments as shown in Table 11. Specifically, we removed the

Table 10 Training/inference time and QPS (queries per second) for different models.

Model	Training time (h)	Inference time (s)	QPS
F3Net	40	188	2.47
NPR	25.1	149	2.66
STIL	11.7	174	2.42
VideoMAE-B	52.5	283	1.32
VideoMamba-M	54.5	146	2.55
MINTIME-CLIP-B	18.1	148	2.61
FTCN-CLIP-B	11.8	148	2.68
TALL	6	150	2.71
CLIP-B-FT	10.1	148	2.56
DeMamba-CLIP-B-FT	14	150	2.54
XCLIP-FT	11.5	149	2.55
DeMamba-XCLIP-FT	15.1	153	2.53

Table 11 Impact of different generators on model performance ("w/o" denotes the exclusion of data from the specified generator). The best performance is highlighted in bold.

w/o	Zero scope	I2VGen -XL	SVD	Video crafter	Pika	Dynamic crafter	SD	SEINE	Latte	OpenSora	–
R	0.6754	0.8908	0.8708	0.9131	0.8291	0.9018	0.8483	**0.9349**	0.8820	0.8955	0.8795
F1	0.7030	0.8107	0.7975	0.6824	0.7883	0.8382	0.8504	0.8469	0.8323	0.8580	**0.8734**
AP	0.8244	0.9237	0.9105	0.8826	0.8915	0.9430	0.9422	0.9477	0.9322	0.9560	**0.9593**

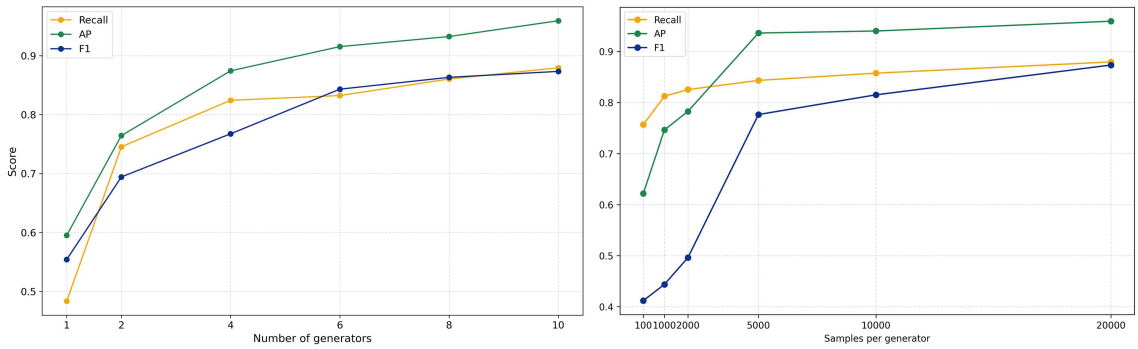

Figure 3 (Color online) Performance of training on scaled-up datasets on the test set.

generator under investigation from the training set, sampled 20000 video samples from each of the remaining nine generators, and combined these with real videos in equal proportions for training. As indicated by the results, the absence of ZeroScope led to the lowest AP, while excluding VideoCrafter resulted in the lowest F1 score, highlighting the significant contribution of these two datasets in improving generalization. Conversely, removing OpenSora had minimal impact on the results. Nevertheless, using all generators simultaneously produced the best performance, confirming our dataset's effectiveness in enhancing generalization.

6 Broader impacts

Our research focuses on utilizing machine learning to detect generated videos. We have introduced the first million-scale AI-generated video detection dataset and developed the DeMamba model. These efforts are crucial for protecting digital content and preventing the spread of misinformation. However, there is a potential for these tools to be misused, leading to competition between video generation and detection technologies. We aim to advocate for the ethical use of technology and promote creative research into tools that verify media authenticity. We believe this will help protect the public from the harm of misinformation, enhance the clarity and authenticity of information dissemination, and ensure the protection of personal privacy.

7 Conclusion and limitation

This paper introduces GenVideo, a dataset specifically designed for detecting fake videos generated by generative models. GenVideo is characterized by its large-scale nature, as well as the rich diversity of generated content and methods. We propose two tasks that mimic real-world scenarios, namely the cross-generator video classification task and the degraded video classification task, to evaluate the detection performance of existing detectors on GenVideo. Additionally, we introduce a plug-and-play effective detection model called DeMamba, which distinguishes AI-generated videos by analyzing inconsistencies in the spatial-temporal dimensions. This model has demonstrated its strong generalization and robustness across multiple tasks. We hope that this research will inspire the creation and improvement of other detection technologies, providing new avenues for the development of authentic and reliable AI-generated content applications.

The main limitation of this article lies in the suboptimal training efficiency of the proposed DeMamba, a common issue with the Mamba model. Consequently, we encourage the community to design more lightweight and generalized detection models to facilitate the regulation of AI-generated content.

Acknowledgements This work was supported by National Natural Science Foundation of China (Grant No. 62302295), Foundation of Key Laboratory of Artificial Intelligence, Ministry of Education, China, and Pioneer R&D Program of Zhejiang Province (Grant No. 2024C01024).

References

1 Zhang L M, Rao A Y, Agrawala M. Adding conditional control to text-to-image diffusion models. In: Proceedings of IEEE/CVF International Conference on Computer Vision, 2023. 3813–3824
2 Chen H X, Xu Z E, Gu Z X, et al. Diffute: universal text editing diffusion model. In: Proceedings of Advances in Neural Information Processing Systems, 2024
3 Li Y Y, Wang H, Jin Q, et al. Snapfusion: text-to-image diffusion model on mobile devices within two seconds. In: Proceedings of Advances in Neural Information Processing Systems, 2024
4 Blattmann A, Dockhorn T, Kulal S, et al. Stable video diffusion: scaling latent video diffusion models to large datasets. 2023. ArXiv:2311.15127
5 Liu Y F, Cun X D, Liu X B, et al. Evalcrafter: benchmarking and evaluating large video generation models. 2023. ArXiv:2310.11440
6 Wang Y H, Chen X Y, Ma X, et al. LaVie: high-quality video generation with cascaded latent diffusion models. 2023. ArXiv:2309.15103
7 GoogleAI. Veo. https://deepmind.google/technologies/veo/, 2024.
8 Brooks T, Peebles B, Holmes C, et al. Video generation models as world simulators. https://openai.com/index/sora/, 2024
9 Runway Research. Text driven video generation. https://research.runwayml.com/gen2, 2023
10 Barrett C, Boyd B, Bursztein E, et al. Identifying and mitigating the security risks of generative AI. Found Trends Privacy Security, 2023, 6: 1–52
11 Cai Z X, Ghosh S, Adatia A P, et al. AV-Deepfake1M: a large-scale llm-driven audio-visual deepfake dataset. In: Proceedings of International Conference on Multimedia, 2024. 7414–7423
12 Li Y Z, Yang X, Sun P, et al. Celeb-DF: a large-scale challenging dataset for deepfake forensics. In: Proceedings of IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2020. 3204–3213
13 Xu H Y, Ye Q H, Wu X, et al. Youku-mplug: a 10 million large-scale Chinese video-language dataset for pre-training and benchmarks. 2023. ArXiv:2306.04362
14 Kay W, Carreira J, Simonyan K, et al. The kinetics human action video dataset. 2017. ArXiv:1705.06950
15 Xu J, Mei T, Yao T, et al. MSR-VTT: a large video description dataset for bridging video and language. In: Proceedings of IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2016. 5288–5296
16 Rössler A, Cozzolino D, Verdoliva L, et al. Faceforensics: a large-scale video dataset for forgery detection in human faces. 2018. ArXiv:1803.09179
17 Ma L, Zhang J, Deng H, et al. DeCoF: generated video detection via frame consistency. 2024. ArXiv:2402.02085
18 Xu Y T, Liang J, Jia G Y, et al. TALL: thumbnail layout for deepfake video detection. In: Proceedings of IEEE/CVF International Conference on Computer Vision, 2023. 22601–22611
19 Gu Z H, Yao T P, Chen Y, et al. Hierarchical contrastive inconsistency learning for deepfake video detection. In: Proceedings of European conference on computer vision, 2022. 596–613
20 Qian Y Y, Yin G J, Sheng L, et al. Thinking in frequency: face forgery detection by mining frequency-aware clues. In: Proceedings of European Conference on Computer Vision, 2020. 86–103
21 Tan C C, Zhao Y, Wei S K, et al. Rethinking the up-sampling operations in CNN-based generative network for generalizable deepfake detection. In: Proceedings of IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2024. 28130–28139
22 Gu Z H, Chen Y, Yao T P, Ding S H, et al. Spatiotemporal inconsistency learning for deepfake video detection. In: Proceedings of International Conference on Multimedia, 2021. 3473–3481
23 Ni B L, Peng H W, Chen M H, et al. Expanding language-image pretrained models for general video recognition. In: Proceedings of European Conference on Computer Vision, 2022. 13664: 1–18
24 Li K C, Li X H, Wang Y, et al. VideoMamba: state space model for efficient video understanding. In: Proceedings of European Conference on Computer Vision, 2024. 237–255
25 Henschel R, Khachatryan L, Hayrapetyan D, et al. Streamingt2v: consistent, dynamic, and extendable long video generation from text. 2024. ArXiv:2403.14773
26 Zhou Y P, Daquan Zhou D Q, Cheng M M, et al. Storydiffusion: consistent self-attention for long-range image and video generation. 2024. ArXiv:2405.01434
27 Ma Z, Zhou D Q, Yeh C H, et al. Magic-me: identity-specific video customized diffusion. 2024. ArXiv:2402.09368
28 Zhang J H, Li D X, Le H, et al. Moonshot: towards controllable video generation and editing with multimodal conditions. 2024. ArXiv:2401.01827
29 Bar-Tal O, Chefer H, Tov O, et al. Lumiere: a space-time diffusion model for video generation. 2024. ArXiv:2401.12945
30 Wei Y J, Zhang S W, Qing Z W, et al. Dreamvideo: composing your dream videos with customized subject and motion. 2023. ArXiv:2312.04433
31 Ho J, Chan W, Saharia C, et al. Imagen video: high definition video generation with diffusion models. 2022. ArXiv:2210.02303
32 Girdhar R, Singh M, Brown A, et al. Emu video: factorizing text-to-video generation by explicit image conditioning. 2023. ArXiv:2311.10709
33 Feng M Y, Liu J L, Yu K, et al. Dreamoving: a human video generation framework based on diffusion models. 2023. ArXiv:2312.05107

34 Xu Z C, Zhang J F, Liew J H, et al. Magicanimate: temporally consistent human image animation using diffusion model. 2023. ArXiv:2311.16498
35 Hu L, Gao X, Zhang P, et al. Animate anyone: consistent and controllable image-to-video synthesis for character animation. 2023. ArXiv:2311.17117
36 Ni H M, Shi C H, Li K, et al. Conditional image-to-video generation with latent flow diffusion models. In: Proceedings of IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2023. 18444–18455
37 Khachatryan L, Movsisyan A, Tadevosyan V, et al. Text2video-zero: text-to-image diffusion models are zero-shot video generators. In: Proceedings of IEEE/CVF International Conference on Computer Vision, 2023. 15954–15964
38 Guo Y W, Ceyuan Yang C Y, Rao A Y, et al. Animatediff: animate your personalized text-to-image diffusion models without specific tuning. 2023. ArXiv:2307.04725
39 Zhang Y M, Xing Z N, Zeng Y H, et al. Pia: your personalized image animator via plug-and-play modules in text-to-image models. 2023. ArXiv:2312.13964
40 Wang X, Yuan H J, Zhang S W, et al. I2vgen-xl. https://modelscope.cn/models/damo/ Image-to-Video/summary, 2023
41 Chen H X, Xia M H, He Y Q, et al. Videocrafter1: open diffusion models for high-quality video generation. 2023. ArXiv:2310.19512
42 Chen H X, Zhang Y, Cun X D, et al. Videocrafter2: overcoming data limitations for high-quality video diffusion models. 2024. ArXiv:2401.09047
43 Xing J B, Xia M H, Zhang Y, et al. DynamiCrafter: animating open-domain images with video diffusion priors. 2023. ArXiv:2310.12190
44 Wang J N, Yuan H J, Chen D Y, et al. Modelscope text-to-video technical report. 2023. ArXiv:2308.06571
45 Wang X, Yuan H J, Zhang S W, et al. Videocomposer: compositional video synthesis with motion controllability. In: Proceedings of Advances in Neural Information Processing Systems, 2024
46 Chen X Y, Wang Y H, Zhang L J, et al. Seine: short-to-long video diffusion model for generative transition and prediction. In: Proceedings of International Conference on Learning Representations, 2023
47 Opensora team. Open-sora: democratizing efficient video production for all. https://github.com/ hpcaitech/Open-Sora, 2024
48 Ma X, Wang Y H, Jia G Y, et al. Latte: latent diffusion transformer for video generation. 2024. ArXiv:2401.03048
49 Hong W Y, Ding M, Zheng W D, et al. Cogvideo: large-scale pretraining for text-to-video generation via transformers. 2022. ArXiv:2205.15868
50 Peebles W, Xie S N. Scalable diffusion models with transformers. In: Proceedings of IEEE/CVF International Conference on Computer Vision, 2023. 4172–4182
51 Shen X Q, Li X, Elhoseiny M. Mostgan-v: video generation with temporal motion styles. In: Proceedings of IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2023. 5652–5661
52 Wang Y H, Jiang L M, Loy C C. Styleinv: a temporal style modulated inversion network for unconditional video generation. In: Proceedings of IEEE/CVF International Conference on Computer Vision, 2023. 22851–22861
53 Kondratyuk D, Yu L J, Gu X Y, et al. Videopoet: a large language model for zero-shot video generation. 2023. ArXiv:2312.14125
54 Yoo J, Kim S, Lee D, et al. Towards end-to-end generative modeling of long videos with memory-efficient bidirectional transformers. In: Proceedings of IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2023. 22888–22897
55 Lei B, Chen L,Ding C W. Flashvideo: a framework for swift inference in text-to-video generation. 2023. ArXiv:2401.00869
56 Yu L J, Cheng Y, Sohn K, et al. Magvit: masked generative video transformer. In: Proceedings of IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2023. 10459 C10469
57 Ghosh P, Sanyal S, Schmid C, et al. Raven: rethinking adversarial video generation with efficient tri-plane networks. 2024. ArXiv:2401.06035
58 Guo X, Liu X H, Ren Z Y, et al. Hierarchical fine-grained image forgery detection and localization. In: Proceedings of IEEE/CVF International Conference on Computer Vision, 2023. 3155–3165
59 Lorenz P, Durall R L, Keuper J. Detecting images generated by deep diffusion models using their local intrinsic dimensionality. In: Proceedings of IEEE/CVF International Conference on Computer Vision, 2023. 448–459
60 Wu H W, Zhou J T, Zhang S L. Generalizable synthetic image detection via language-guided contrastive learning. 2023. ArXiv:2305.13800
61 Wang Z D, Bao J M, Zhou W G, et al. DIRE for diffusion-generated image detection. In: Proceedings of IEEE/CVF International Conference on Computer Vision, 2023. 22388–22398
62 Wang Z J, Montoya E, Munechika D, et al. Diffusiondb: a large-scale prompt gallery dataset for text-to-image generative models. 2022. ArXiv:2210.14896
63 Zhu M J, Chen H T, Yan Q Y, et al. Genimage: a million-scale benchmark for detecting ai-generated image. In: Proceedings of Advances in Neural Information Processing Systems, 2024
64 Hong Y, Feng J M, Chen H X, et al. WildFake: a large-scale challenging dataset for AI-generated images detection. In: Proceedings of AAAI Conference on Artificial Intelligence, 2025
65 Khalid H, Tariq S, Kim M, et al. FakeAVCeleb: a novel audio-video multimodal deepfake dataset. In: Proceedings of Advances in Neural Information Processing Systems, 2021
66 Bai J, Lin M, Cao G, et al. AI-generated video detection via spatial-temporal anomaly learning. In: Proceedings of Chinese Conference on Pattern Recognition and Computer Vision, 2024. 460–470
67 Wang S Y, Wang O, Zhang R, et al. CNN-generated images are surprisingly easy to spot... for now. In: Proceedings of IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2020. 8692–8701
68 Zeroscope team. Zeroscope-v2-xl. https://huggingface.co/cerspense/zeroscope_v2_XL, 2024
69 Pika team. Pika art. https://pika.art/, 2022
70 Morph team. Morph studio. https://www.morphstudio.com/, 2023
71 Moonvalley team. moonvalley.ai. https://moonvalley.ai/, 2022
72 Zhang J H, Wu Z J, Liu J W, et al. Show-1: marrying pixel and latent diffusion models for text-to-video generation. 2023. ArXiv:2309.15818
73 Esser P, Chiu J, Atighehchian P, et al. Structure and content-guided video synthesis with diffusion models. In: Proceedings of IEEE/CVF International Conference on Computer Vision, 2023. 7346–7356
74 Hotshot team. Hotshot. https://huggingface.co/hotshotco/Hotshot-XL, 2023
75 Yang S Y, Hou L, Huang H B, et al. Direct-a-video: customized video generation with user-directed camera movement and object motion. 2024. ArXiv:2402.03162
76 Ren W M, Yang H, Zhang G, et al. Consisti2v: enhancing visual consistency for image-to-video generation. 2024. ArXiv:2402.04324
77 Wang X, Zhang S W, Yuan H J, et al. A recipe for scaling up text-to-video generation with text-free videos. 2023. ArXiv:2312.15770
78 Z W, Zhang S W, Wang J Y, et al. Hierarchical spatio-temporal decoupling for text-to-video generation. 2023. ArXiv:2312.04483
79 Ge S W, Nah S, Liu G L, et al. Preserve your own correlation: a noise prior for video diffusion models. In: Proceedings of IEEE/CVF International Conference on Computer Vision, 2023. 22930–22941
80 Guo X, Zheng M W, Hou L, et al. I2v-adapter: a general image-to-video adapter for video diffusion models. 2023. ArXiv:2312.16693
81 Tian L R, Wang Q, Zhang B, et al. Emo: emote portrait alive-generating expressive portrait videos with audio2video diffusion model under weak conditions. 2024. ArXiv:2402.17485
82 Zeng Y, Wei G Q, Zheng J N, et al. Make pixels dance: high-dynamic video generation. 2023. ArXiv:2311.10982
83 Rombach R, Blattmann A, Lorenz D, et al. High-resolution image synthesis with latent diffusion models. In: Proceedings of IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2022. 10684–10695

84 Podell D, English Z, Lacey K, et al. Sdxl: improving latent diffusion models for high-resolution image synthesis. 2023. ArXiv:2307.01952
85 Scao T L, Fan A, Akiki C, et al. Bloom: a 176b- parameter open-access multilingual language model. 2022. ArXiv:2211.05100
86 Huang Z Q, He Y N, Yu J S, et al. Vbench: comprehensive benchmark suite for video generative models. In: Proceedings of IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2023. 21807–21818
87 Gupta A, Yu L J, Sohn K, et al. Photorealistic video generation with diffusion models. 2023. ArXiv:2312.06662
88 Gu A, Goel K, Ré C. Efficiently modeling long sequences with structured state spaces. In: Proceedings of International Conference on Learning Representations, 2022
89 Gu A, Johnson I, Goel K, et al. Combining recurrent, convolutional, and continuous-time models with linear state space layers. In: Proceedings of Advances in Neural Information Processing Systems, 2021. 572–585
90 Smith J T H S, Warrington A, Linderman S W. Simplified state space layers for sequence modeling. In: Proceedings of International Conference on Learning Representations, 2023
91 Gu A, Dao T. Mamba: linear-time sequence modeling with selective state spaces. In: Proceedings of Conference on Language Modeling, 2024
92 Radford A, Kim J W, Hallacy C, et al. Learning transferable visual models from natural language supervision. In: Proceedings of International Conference on Machine Learning, 2021. 8748–8763
93 Liu Y, Tian Y J, Zhao Y Z, et al. Vmamba: visual state space model. 2024. ArXiv:2401.10166
94 Zhu L H, Liao B C, Zhang Q, et al. Vision Mamba: efficient visual representation learning with bidirectional state space model. In: Proceedings of International Conference on Learning Representations, 2024
95 Tong Z, Song Y B, Wang J, et al. VideoMAE: masked autoencoders are data-efficient learners for self-supervised video pre-training. In: Proceedings of Advances in Neural Information Processing Systems, 2022
96 Coccomini D A, Kordopatis-Zilos G, Amato G, et al. MINTIME: multi-identity size-invariant video deepfake detection. IEEE Trans Inf Forensics Secur, 2024, 19: 6084–6096
97 Zheng Y L, Bao J M, Chen D, et al. Exploring temporal coherence for more general video face forgery detection. In: Proceedings of IEEE International Conference on Computer Vision, 2021. 15024–15034