Beyond Deepfake Images: Detecting AI-Generated Videos

Danial Samadi Vahdati, Tai D. Nguyen, Aref Azizpour, Matthew C. Stamm
Drexel University
Philadelphia, PA, USA
{ds3729,tdn47,aa4639,mcs382}@drexel.edu

Abstract

Recent advances in generative AI have led to the development of techniques to generate visually realistic synthetic video. While a number of techniques have been developed to detect AI-generated synthetic images, in this paper we show that synthetic image detectors are unable to detect synthetic videos. We demonstrate that this is because synthetic video generators introduce substantially different traces than those left by image generators. Despite this, we show that synthetic video traces can be learned, and used to perform reliable synthetic video detection or generator source attribution even after H.264 re-compression. Furthermore, we demonstrate that while detecting videos from new generators through zero-shot transferability is challenging, accurate detection of videos from a new generator can be achieved through few-shot learning.

1. Introduction

In recent years, substantial progress in generative AI has produced numerous techniques for generating visually realistic synthetic images. These advances have also introduced significant misinformation and disinformation threats. Synthetic images can be easily produced and used as falsified visual evidence to deceive a target audience.

To combat this, researchers have developed a number of techniques to detect synthetic images. These techniques operate by searching for statistical traces left in synthetic images by their source generator. For example, prior work by Zhang et al. has shown that the upsampling operation used in many generator architectures to grow an image from a small latent representation to a full sized image leaves behind traces similar to those left by resampling [94]. A number of approaches have been successfully developed to accurately detect synthetic images made by a wide variety of generators [3, 15, 19, 22, 47, 53, 78, 95, 97] and attribute them to their source [23, 77, 97].

Very recently, AI-based synthetic video generators have begun to emerge. These range from text-prompted approaches such as Stable Video Diffusion, VideoCrafter, or OpenAI's recently released Sora, to others such as Luma AI's NeRF-basesd approach which allows synthetic videos to be generated and manipulated based on a set of input images. The emergence of synthetic video generators represents not only a major technological advancement, but also a significant escalation in the potential misinformation and disinformation threats caused by generative AI.

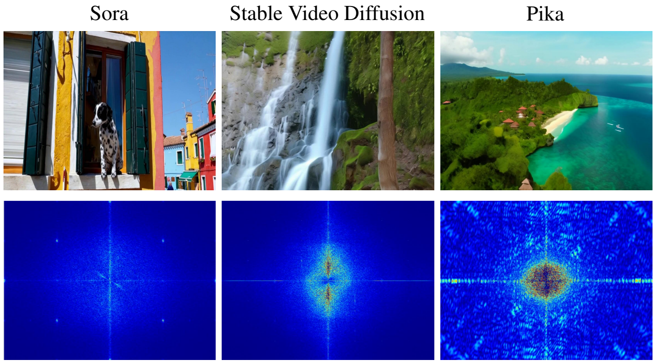

Figure 1. Top row: video frames taken from AI-generated videos. Bottom row: Fourier transforms of the residual forensic traces for each corresponding frame above. The process to produce visual results in the bottom row is described in Sec. 4.

One would reasonably assume that synthetic image detectors should accurately detect synthetic videos. In this paper, however, we demonstrate that synthetic image detectors do not accurately detect synthetic videos. Furthermore, we show that this is not due to performance degradation caused by H.264 compression. Instead, we demonstrate that synthetic video generators leave distinct traces that are not detected by image detectors. Encouragingly, we show that these traces can be learned and utilized to perform accurate synthetic video detection and generator source attribution. In addition, we investigate the transferability of synthetic video detectors and show that they can be adapted to detect videos from new generators that contain substantially different traces using very little data. The novel contributions of this paper are listed below:

1. We show that synthetic image detectors do not reliably

Figure 2. Sample frames from different video generators. The figure shows the synthetic video frames: from left to right Luma [1], VideoCrafter [13], CogVideo [34], Pika [45], Sora [9], and Stable Video Diffusion [8].

detect AI-generated videos, and empirically verify this is not due to the degradation effects of H.264 compression.

2. We demonstrate that synthetic video generators leave substantially different forensic traces than those left by synthetic image generators. This is the primary cause of synthetic image detectors' poor performance on video.
3. Furthermore, we show that synthetic video traces can be learned and used to perform reliable synthetic video detection or source attribution even in the presence of H.264 re-compression.
4. We demonstrate that while detecting videos from new generators through zero-shot transferability is challenging, accurate detection of videos from a new generator can be achieved through few-shot learning.
5. We create a new, publicly available dataset of synthetic videos from a number of state-of-the-art video generators that can be used to train and benchmark the performance of synthetic video detectors. [1]

2. Background

Synthetic Image Generation. The field of computer-generated media has seen significant advancements, beginning with the introduction of Generative Adversarial Networks (GANs) by Goodfellow et al. [27], a seminal work that has since spurred a multitude of subsequent innovations [17, 40–43, 56, 81, 96]. These innovations have significantly enhanced the capabilities of generative models in producing images that are diverse, realistic, and of high quality. Recent works have explored using Transformers for improving generated image consistency [11, 21, 24, 30, 67, 92]. However, a notable milestone was achieved with the advent of the diffusion model by Ho et al. [32], which has since fueled a vast array of research leading to cutting-edge generation methods like Stable Diffusion [74], DALL·E [72], Midjourney [61], and Cascade Diffusion [33], to name a few [4, 20, 29, 84, 98].

Synthetic Video Generation. Another modality of synthetic media synthesis is synthetic video generation. Recently, lots of research attention has been devoted to developing synthetic video generation methods. These methods ranges from diffusion models [8, 13, 14], to Transformers [25, 34, 39, 52, 71, 76, 86, 87, 91]. Moreover, there exists generation techniques that use a combination of methods such as SORA by OpenAI [9], and commercially available products that do not disclose the exact method used for content generation [45, 62].

Synthetic Image Detection. As synthetic image generators proliferate, researchers aim to devise detection methods. Wang et al. [83] were among the first to tackle this by training a CNN on a single generator, enabling the detection and classification of numerous synthetic images. Subsequently, as generators grew more complex, researchers developed sophisticated detectors, including methods proposed by Marra et al. [54, 55], Zhang et al. [94], and others [10, 23, 47, 60, 64, 65, 69, 82, 85, 93]. Recently, detection methods have extended to newer image generation techniques like diffusion models [2, 19, 51, 70, 77, 97].

3. Using Synthetic Image Detectors On Video

Given that a video can be seen as a sequence of images, it is reasonable to expect that synthetic image detectors should be effective at detecting AI-generated synthetic videos. Surprisingly, however, we have found that synthetic image detectors do not successfully identify synthetic videos. Furthermore, we have found that this issue is not primarily caused by the degradation of forensic traces due to H.264 video compression.

To demonstrate these findings, we conducted a series of experiments in which we evaluated synthetic image detector's ability to detect synthetic videos. The details of these experiments, as well as their outcomes, are presented below.

3.1. Experimental Setup

Detectors. We evaluated the performance of a broad set of detection algorithms. These include both publicly available pretrained detectors made specifically for synthetic image detection, and other architectures that are widely used to perform image forensic tasks. The complete list of these detectors and their referred-to names is provided in Table 1.

Image Training Dataset. To train detectors that weren't pretrained, we used a dataset of 100,000 images equally divided between real and synthetic. For real images, we utilized a subset of the COCO dataset [49] and the LSUN dataset [90], as was done in [19]. For synthetic images, we

[1] Link to our dataset: https://huggingface.co/datasets/ductai199x/synth-vid-detect

	Detection Algorithms (P=Pretrained, R=Retrained, T=Trained-by-us)		
Training	Refer to as	Architecture	Used by
P	Corvi et al.[19]	ResNet-50[31]	[3, 19, 48, 78, 95]
P	Sinitsa et al.[77]	DIF[77]	[15, 77, 97]
P	Zhu et al.[97]	Swin-Transformer[50]	[97]
R	ResNet-50[31]	ResNet-50[31]	[3, 19, 48, 78, 95]
R	DIF[77]	DIF[77]	[77]
R	Swin-T[50]	Swin-Transformer[50]	[15, 97]
T	ResNet-34[31]	ResNet-34[31]	[3, 22, 53, 94, 95]
T	VGG-16[75]	VGG-16[75]	[3, 26, 47, 68, 89]
T	Xception[18]	Xception[18]	[3, 12, 16, 38, 79]
T	DenseNet[37]	DenseNet[37]	[46, 54, 59, 88]
T	MISLnet[23]	MISLnet[5, 7]	[23, 28]

Table 1. List of detection algorithms used in this paper and the names as they are referred to in our paper.

Method	Images	Videos				
	Baseline	Luma	CogVideo	VC-v1	SVD	Average
Corvi et al. [19]	0.974	0.583	0.704	0.590	0.682	0.640
Sinitsa et al. [77]	0.992	0.500	0.500	0.500	0.500	0.500
Zhu et al. [97]	0.891	0.652	0.694	0.728	0.719	0.698
ResNet-50 [31]	0.946	0.572	0.736	0.604	0.710	0.656
DIF [77]	0.991	0.581	0.603	0.617	0.573	0.594
Swin-T [50]	0.911	0.638	0.685	0.698	0.692	0.678
ResNet-34 [31]	0.983	0.576	0.623	0.616	0.647	0.615
VGG-16 [75]	0.990	0.635	0.652	0.684	0.669	0.660
Xception [18]	0.996	0.592	0.638	0.670	0.664	0.641
DenseNet [37]	0.975	0.559	0.584	0.647	0.678	0.624
MISLnet [23]	0.983	0.626	0.718	0.710	0.707	0.690

Table 2. Detection performance of existing synthetic image detectors, that were trained or pretrained on synthetic images, on different synthetic video generation methods. Performance numbers are measured using AUC.

Method	Images	Videos					
	Baseline	Luma	CogVideo	VC-v1	SVD	Avg.	Vs. no H.264
ResNet-50 [31]	0.963	0.604	0.770	0.646	0.738	0.689	+0.033
DIF [77]	0.994	0.617	0.634	0.655	0.624	0.632	+0.038
Swin-T [50]	0.948	0.679	0.730	0.758	0.742	0.727	+0.049
ResNet-34 [31]	0.989	0.663	0.687	0.700	0.727	0.694	+0.079
VGG-16 [75]	0.993	0.719	0.743	0.754	0.729	0.736	+0.076
Xception [18]	0.979	0.642	0.692	0.734	0.708	0.694	+0.053
DenseNet [37]	0.980	0.604	0.628	0.691	0.703	0.656	+0.032
MISLnet [23]	0.995	0.674	0.759	0.784	0.760	0.744	+0.054

Table 3. Detection performance of existing synthetic image detectors, that were retrained on H.264-compressed synthetic images, on different synthetic video generation methods. Performance numbers are measured using AUC.

used synthetic images from the datasets used in [23] consisting of images made by CycleGAN [96], StarGAN [17], StyleGAN3 [43], ProGAN [40], and Stable Diffusion [74].

Image Testing Dataset. To benchmark the performance of each detector, we created a testing set of 20,000 images equally divided between real and synthetic. This set was made by utilizing disjoint subsets of the datasets used to create the real and synthetic image training data.

Video Testing Dataset. To measure the performance of each detector, we created a testing dataset of both real and synthetically generated videos. Real videos were taken equally from the Moments in Time (MiT) [63] and Video-ACID [36] datasets. Synthetic videos were generated using four different publicly available video generators: Luma [1], VideoCrafter-v1 [13], CogVideo [34], and Stable Video Diffusion [8]. These synthetic videos were created using a common set of diversified content and motion text prompts, with the exception of videos from Luma, which were gathered from a similarly diverse set of publicly shared videos. The qualitative samples of these videos are shown in Fig. 2. Further details of this test set are provided in Table 4 and in Sec. 5.1 below.

Metrics. The detection performance of each detector was measured using the area under its ROC curve (AUC).

3.2. Synthetic Image Detector Performance

We first established the baseline performance of each synthetic image detector on our image testing dataset. These results are presented in the second column of Table 2. The majority of detectors achieved an AUC of 0.94 or greater, except for the pre-trained version of Swin-T with an AUC of 0.891. These baseline results verify that when assessed on images, each detector can achieve strong performance.

Next we evaluated each synthetic image detector using our video testing dataset. These results are also shown in Table 2. These results show that all detectors experience significant performance drops when evaluating synthetic videos. The highest average AUC achieved was 0.698, with most detectors scoring an AUC of 0.65 or lower. This drop in performance cannot be attributed to a single challenging generator, as AUCs for each detector on a single generator are consistently less than 0.74.

These results demonstrate that synthetic image detectors face significant challenges in detecting synthetic videos. This difficulty persists across various detector architectures and whether detectors are pre-trained by others or retrained using our dataset.

3.3. Effect of H.264 Robust Training

It is well known that compression alters forensic traces and degrade a detector's performance. Hence, a plausible explanation for synthetic image detector's the poor performance on synthetic video could be that H.264 video compression is degrading synthetic video traces[66].

To test this hypothesis, we conducted an additional set of experiments in which each synthetic image detector was retrained to be robust against H.264 compression. Ro-

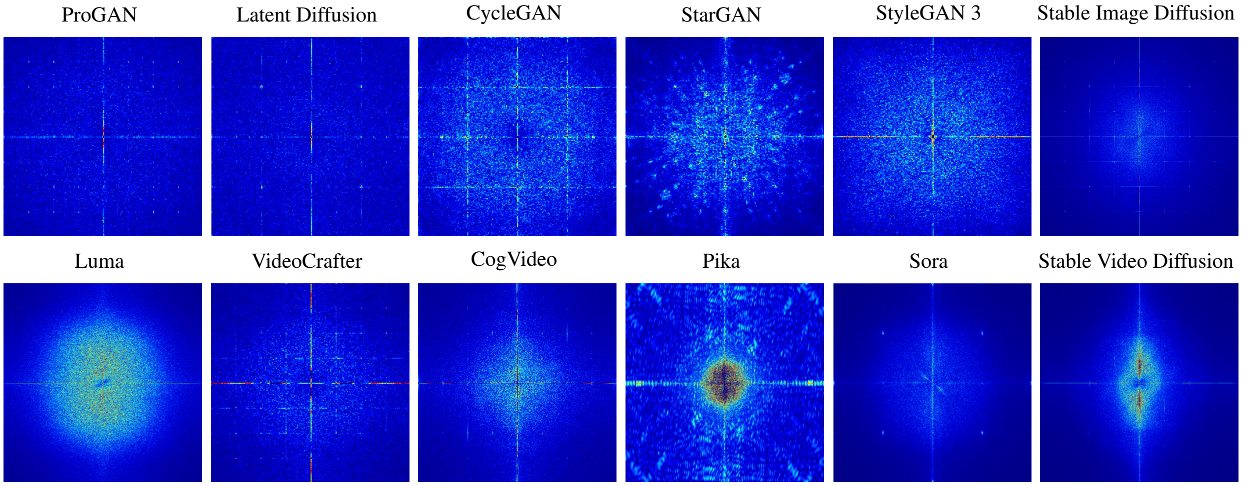

Figure 3. Fourier transform analysis of the forensic traces extracted from different synthetic image and video generators.

bust training involves augmenting the training dataset by re-compressing all data points using different compression strengths, i.e. different quality factor for JPEG or different compression rate factors (CRFs) for H.264. It is well known that robust training can significantly mitigate the negative effects of compression [6, 19, 44, 80]. As a result, if H.264 compression was truly the cause of a detector's low performance on synthetic video, we would expect the detector's performance to increase to a level much closer to its baseline performance after robust training.

Table 3 shows the performance of each synthetic image detector after it was robustly trained by augmenting the image training dataset with H.264 compressed images using CRFs between 0 and 20. The baseline detector performances on images show that robust training improves each detector's already strong performance on images. Despite this, robust training did not substantially improve each detector's performance when evaluated on our video testing set. The detector with the strongest performance on video after robust training was VGG-16, which obtained an AUC of 0.736 on video as opposed to its baseline performance of 0.993 on images. Most detectors achieved an average AUC of below 0.74. And on average, robust training only improved the AUC of each detector by 0.052. The largest AUC gain was seen by ResNet-34 [31, 94], increasing its average AUC by 0.079 to 0.694.

These results indicate that H.264 is not the primary cause of synthetic image detectors' poor performance when detecting synthetic videos. Instead, the poor performance of synthetic image detectors after H.264 robust training suggests that a different factor is causing this phenomenon.

4. Synthetic Video Forensic Traces

Here, we present evidence that forensic traces in synthetic video are substantially different than those in synthetic images. We qualitatively demonstrate this by visualizing the low-level forensic traces left by a number of different image and video generators using the approach proposed in [94].

To do this, we collected a set of 1000 images and video frames created using several different image and video generators. These image generators included ProGAN [40], CycleGAN [96], StarGAN [17], StyleGAN 3 [43], Latent Diffusion [73], and Stable Diffusion [74], while video generators included Luma [1], VideoCrafter v1 [13], CogVideo [34], Pika [45], Sora [9], and Stable Video Diffusion [8]. We then created a noise residual for each image and video frame x_k by de-noising it using a de-noising algorithm ϕ, then subtracting the denoised image or frame from the original. All noise residuals from a single generator were averaged to produce an aggregate noise residual $y = \frac{1}{N}\sum_{k=1}^{N} x_k - \phi(x_k)$. Frequency domain representations of these aggregate noise residuals were then created by taking their Fourier transforms, then their magnitudes were plotted to produce trace visualizations.

Image and Video Generator's Trace Comparison. The resulting low-level forensic traces for each image and video generator are shown in Fig. 3. By examining these traces, we can clearly see that synthetic images contain substantially different traces than synthetic videos. For example, traces left by image generators (e.g. ProGAN, CycleGAN, Latent Diffusion) typically include periodic spectral peaks or a grid-like structure that Zhang et al. [94] showed are caused by up-sampling operations used in a generator's architecture. By contrast, some video generation techniques such as Luma employ neural radiance fields (NeRFs) or other architectures that do not utilize up-sampling. As a result, the distinct patterns seen in synthetic image traces will not be present in traces from these video generators.

Additionally, industry generators may use undisclosed techniques to protect trade secrets. For instance, full details

	Dataset	Training		Validation		Testing	
		# Videos	# Frames	# Videos	# Frames	# Videos	# Frames
Real	MIT [63]	3,991	80,000	377	8,000	945	20,000
	Video-ACID [36]	3,663	80,000	407	8,000	716	20,000
Total		7,654	160,000	784	16,000	1,661	40,000
Synthetic	Luma [1]	312	40,000	32	4,000	78	10,000
	VC-v1 [13]	1,428	40,000	143	4,000	280	10,000
	CogVideo [34]	1,600	40,000	163	4,000	357	10,000
	SVD [8]	2,857	40,000	286	4,000	714	10,000
Total		6,197	160,000	624	16,000	1,429	40,000

Table 4. Dataset statistics for training and evaluating detection systems on synthetic video data. VC stands for VideoCrafter.

about Pika's generation method are not currently public, but the significant difference between its traces and others' suggests Pika uses a noticeably different technique.

Due to the stark contrasts between forensic traces left by image and video generators, it is highly likely that this is the major reason why synthetic image detectors exhibit substantially lower performance on video. Even when robustly trained, synthetic image detectors learn features to capture forensic traces similar to what they have seen before. Since video traces can be substantially different in nature, synthetic image detectors are not suited to capture these traces.

We note that our findings align with prior research. Specifically, Corvi et al. [19] found that Stable and Latent diffusion models produce different forensic traces than image generators such as ADM and DALL-E 2. They also showed that even robustly trained synthetic image detectors "still cannot reliably detect images that present artifacts significantly different from those seen during training." [19]

5. Learning Synthetic Video Forensic Traces

Results presented in the previous two sections show that traces left by synthetic video generators are different than those left by image generators, and that synthetic image detectors do not reliably detect these traces.

In this section, however, we show that synthetic video traces can be learned. Through a series of experiments, we show that CNNs can be trained to accurately perform synthetic video detection and source attribution. Furthermore, we demonstrate that robust training can improve these detectors even after H.264 re-compression. Additionally, we show how video-level detection can be performed to boost performance over frame-level detection.

5.1. Experimental Setup

The following experiments all used the same experimental setup detailed here.

Video Training Data. To train synthetic video detectors in the following experiments, we collected a diverse set of real and synthetic videos. For real videos, we gathered videos from the Moments in Time (MIT) [63] dataset and the Video Authentication and Camera Identification Database (Video-ACID) [36]. For the set of synthetic videos, we used 4 publicly available video generators to generate a large dataset for both training and testing purposes. These generation methods are: Luma [1], VideoCrafter-v1 [13], CogVideo [34], and Stable Video Diffusion [8]. To create the synthetic videos in our dataset, we utilized a common set of text prompts chosen to represent a diversified set of scenes and activities. Videos created using Luma were gathered from publicly shared videos, and similarly chosen to represent a diverse set of contents and motion. Additionally, all videos are by default compressed using H.264 at constant rate factor 23. More detailed are provided in Table 4.

Video Testing Data. To create our video testing data, we utilized the same collection and generation methodology as for our training data. However, we kept an exclusive set of testing prompts to only be used to create testing data. Regarding videos from Luma, we also gathered a disjoint set of videos, completely separate from those in the training set. We note that this testing set is the same set used for experiments in Sec. 3. More detailed are provided in Table 4.

Out-of-distribution Test-only Synthetic Videos. In addition to our in-distribution video testing set, we also evaluated the performance of different detection algorithms on an out-of-distribution, test-only set of videos. The synthetic videos in this set is collected from three recently emerging generation methods: Sora [9], Pika [45], and VideoCrafter-v2 [14]. More details on this dataset are provided in Table 7

Metrics. In the following experiments, the performance of each detector was measured using the area under its ROC curve (AUC). In addition, to highlight performance differences, we provided the Relative Error Reduction (RER) with respect to the second best performing method, reported as a percentage. This metric is calculated as follows:

$$RER = 100 \times \frac{\text{AUC}_N - \text{AUC}_R}{1 - \text{AUC}_R}, \quad (1)$$

where AUC_R is the AUC of the referencing method, and AUC_N is the AUC of the method being compared against.

Detectors. To demonstrate the performance of different detection methods on detecting synthetic videos, we conducted our experiments with the diverse set of detectors listed in Table 1 and Sec. 3.1.

5.2. Synthetic Video Detection

First, we conducted an experiment in which we trained each candidate detector network to perform synthetic video detection using the training dataset described above. We then evaluated their ability to detect each of the four synthetic video generators in the test set. These experiments were carried out at a patch-level, i.e. all detection decisions were obtained using one patch taken from a single video frame.

The results of this experiment are presented in Table 5.

Method	**Videos**				
	Luma	**CogVideo**	**VC-v1**	**SVD**	**Average**
ResNet-50 [31]	0.921	**0.935**	0.940	0.939	0.934
DIF [77]	0.938	0.957	0.969	0.973	0.959
Swin-T [50]	0.960	0.964	0.986	0.991	0.975
ResNet-34 [31]	0.916	0.930	0.942	0.951	0.935
VGG-16 [75]	0.937	0.944	0.965	0.960	0.951
Xception [18]	0.928	0.951	0.973	0.969	0.955
DenseNet [37]	0.918	0.924	0.964	0.968	0.943
MISLnet [23]	**0.975**	**0.980**	**0.991**	**0.987**	**0.983**

Table 5. Detection performance of methods trained and tested on different synthetic video generation methods. Performance numbers are measured using AUC.

Method	**Videos**				
	Luma	**CogVideo**	**VC-v1**	**SVD**	**Overall**
Resnet-50 [19]	0.937	0.958	0.970	0.968	0.962
DIF [77]	0.894	0.909	0.926	0.924	0.917
Swin-T [50]	**0.976**	0.973	0.991	0.988	0.986
ResNet-34 [31]	0.925	0.947	0.963	0.932	0.948
VGG-16 [75]	0.901	0.936	0.945	0.959	0.935
Xception [18]	0.925	0.971	0.960	0.952	0.950
DenseNet [37]	0.954	0.959	0.976	0.970	0.966
MISLnet [23]	0.975	**0.984**	**0.998**	**0.992**	**0.991**

Table 6. Synthetic video source attribution performance of each detection systems on individual generation method. Performance numbers are measured using AUC.

These results clearly show that synthetic videos can be reliably detected using each of these detectors. All detectors evaluated achieved an average AUC of at least 0.93. MISLnet achieved the highest average AUC of 0.983 and maintained consistently strong detection performance for each video generator. We note that each detector trained on synthetic video experienced an improvement of at least 0.23 in average AUC when compared to performance of the same detector robustly trained on synthetic images. This further reinforces that synthetic video traces can be learned by existing architectures used for synthetic image detection.

5.3. Synthetic Video Source Attribution

Next, we conducted an experiment evaluating each network's ability in source attribution. The forensic network identifies a video's source generator or determines its authenticity. To adapt each network for this multi-class classification, we replaced its final layer with one containing neurons corresponding to each generator and one for real. The trained network's AUC was assessed using the one-vs-the-rest strategy, where any incorrect source attribution was counted as a miss regardless of the class.

The results of this experiment are shown in Table 6. From these results, we can see that all networks achieved an AUC of at least 0.91, with most achieving an AUC of 0.95 or higher. Again, the best performing network was MISLnet, which acheived an AUC of 0.991. These results indicate that existing networks can be trained to perform accurate synthetic video source attribution. We note that this result makes sense in light of the synthetic video traces visualized in Fig. 3. As discussed in Sec. 4, videos are generated using a wide variety of generation strategies and generator architectures. Since each technique imparts significant different traces, this makes it much easier for networks to accurately discriminate between each generator.

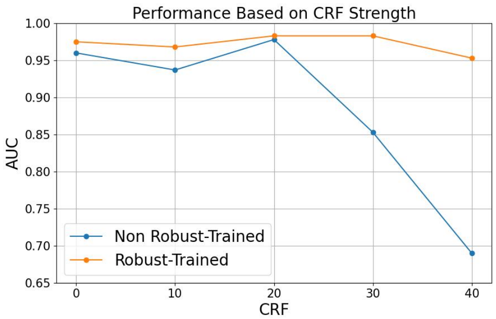

Figure 4. Detection performance of MISLnet[23] before and after robust-training on videos with constant rate factors from 0 to 40.

5.4. Effect of H.264 Re-compression

As we discussed in Sec. 3, it is well known that re-compression can significantly reduce the performance of a forensic system. This is particularly important since re-compression is often utilized by social media platforms. In light of this, we conducted a set of experiments to understand the effect of H.264 compression on detection performance. Additionally, we conducted experiments to assess the ability of robust training to mitigate these effects. In light of space constraints, results are reported for the MISLnet detector, which achieved the highest detection and attribution performance in the previous experiments.

To carry out these experiments, we re-compressed each video in the testing set with constant rate factors (CRFs) ranging from 0 (weak) to 40 (strong). We note that videos are all initially H.264 compressed at CRF 23 either by the camera or the generator. We then re-compressed each video in the training set using the same CRF levels, and used them to robustly train MISLnet to perform synthetic video detection. After this, we evaluated the performance of both the non-robustly and robustly trained version of this detector.

The results of this experiment are displayed in Fig. 4, which shows the AUC achieved by the detector at each CRF both with and without robust training. From these results, we can see that without robust training, the detector's performance decreases as the CRF increases. The notable exception to this is at CRF 20, which is close to the typical default CRF of 23. Performance increases here because the CRF used during re-compression is close to the default CRF

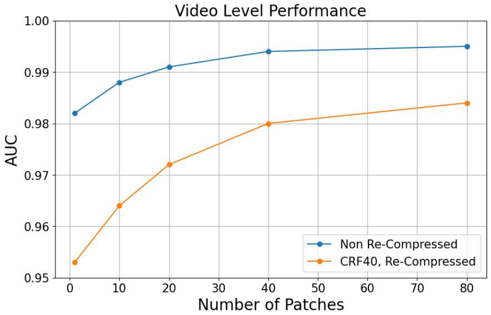

Figure 5. Video-level performance of MISLnet over different number of patches used for obtaining video-level detection score.

that has already been seen during training.

The results presented in Fig. 4 also show that when robust training is utilized, the detector's performance remains consistently strong across all CRFs. Specifically, the detector is able to achieve an AUC of 0.95 or higher for all CRFs. Furthermore, when the CRF is 30 or higher, the detector achieves an AUC of 0.97 or higher. These results demonstrate that robust training enables accurate synthetic video detection even after re-compression.

5.5. Video-Level Detection Performance

Unlike synthetic images, synthetic videos consist of a sequence of AI-generated frames. Because of this, generator traces are distributed temporally throughout a video. This information can be exploited to perform detection at a video-level with greater accuracy.

To perform video-level detection, we first form a patch-level embedder $\psi(\cdot)$ by discarding the final layer of a pre-trained patch-level synthetic video detector. The pre-softmax activations produced by this network correspond to patch-level embeddings that capture video generator traces. Next, a sequence of N temporally distributed patches x_k are gathered throughout a video and added together to form a single video-level embedding. This is then passed through a soft-max layer to produce the final output detection score

$$\delta = \sigma\Big(\sum_{k=1}^{N} \psi(x_k)\Big), \qquad (2)$$

where $\sigma(\cdot)$ is the soft-max function[35, 58].

We conducted a series of experiments where we measured the performance of this video-level detection strategy using different numbers of patches. In these experiments, we used the MISLnet detector architecture since it achieved the best performance for patch-level detection and attribution. Performance in terms of AUC was measured on both our non-recompressed testing set, as well as on a version of the testing set that was re-compressed with a CRF of 40 for different numbers of patches N ranging from 1 to 80. Additionally, the percentage of relative error reduction (RER) over a patch-level detector was also calculated for each N.

The results of these experiments are displayed in Fig. 5

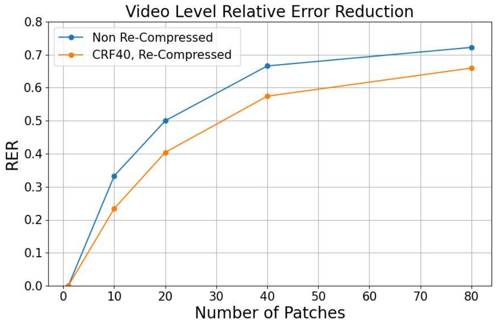

Figure 6. Relative Error Reduction in video-level performance versus frame-level performance of MISLnet over different number of patches used for obtaining video-level detection score.

which shows video-level detection AUC vs number of patches used for detection, and in Fig. 6 which shows the RER vs. number of patches used for detection. From these figures, we can see that performance increases as more patches are used to perform video-level detection. This is particularly strong for re-compressed video, where the AUC grows from 0.953 to 0.984 as 80 patches are used, corresponding to a RER of 66% over the patch-level detector. These results show that video-level detection can achieve important performance gains by leveraging traces throughout an entire video.

We also note that while the increase in AUC for non-recompressed videos is not large in absolute terms, the RER achieved is substantial. Specifically, when 80 patches are used, the video-level detector achieves an RER of 72.2% over the patch-level detector. This is particularly important when performing synthetic video detection at scale, such as if a social media company were to examine all videos uploaded to its service. Because of the large number of videos examined, small gains in performance can correspond to a large reduction in the number of false alarms.

6. Detection Transferability to New Generators

New synthetic video generator architectures and generation approaches are emerging at a rapidly. Hence, it is important to understand the transferability of synthetic video detectors to new generators as we know this could be achieved as demonstrated by[57]. In this section, we conduct a series of experiments to examine detector transferability in both zero-shot and few-shot transfer scenarios.

6.1. Zero-Shot Transferability

In our first set of experiment, we examined synthetic video detectors' zero-shot transferability performance. This corresponds to a detector's ability to detect videos from a new generator without any re-training.

We began by training the best performing detector (MISLnet) using data from three of the generators in our training set. We benchmarked the detector's performance on the

Out-of-distribution Generation Method	Test-only	
	# Videos	# Frames
Sora [9]	38	5,000
Pika [45]	163	10,000
VC-v2 [14]	200	10,000
Total	401	25,000

Table 7. Statistics of the out-of-distribtion, test-only synthetic video dataset used in Sec. 6.

Generation Method	Seen Sources	Unseen Source
VideoCrafter v1[13]	0.993	0.773
Cogvideo[34]	0.990	0.671
Luma[1]	0.991	0.702
SVD[8]	0.985	0.760

Table 8. Zero-shot detection performance of MISLnet [23], which was trained on 3 out of 4 synthetic video generation sources and test on the remaining one. Performance numbers are in AUC.

portion of the test dataset corresponding to generators seen during training. Then, we measured the detector's zero-shot transferability by using it to detect videos generated by a generator not included in training, i.e. unseen sources.

Results from this experiment are presented in Table 8. These results show that while the detector achieves strong performance on videos from generators seen during training, performance drops significantly when evaluating on new generators. Specifically, the AUC drops from an average of 0.990 for "seen" generators to an average of 0.727 for new, "unseen" generators.

We conducted a similar experiment, in which we used the detector trained on all four generators in our training set to detect synthetic videos from the generators in our Out-of-Distribution testing set described in Section 5.1. These videos were generated using three different generators: Sora, Pika, and VideoCrafter v2.

The results of this experiment are presented in the second colum of Table 9, labeled "Zero-Shot". These experiments yielded similar results, in which the detector had significant difficulty detecting videos from these new, unseen generators. The only notable exception to this was VideoCrafter v2, with a generator closely similar to VideoCrafter v1.

The results of these experiments are somewhat unsurprising. As we can see from the generator trace visualizations in Fig. 3, traces left by different generators can vary substantially. Furthermore, we know that both video generator architectures and generation approaches (i.e. NeRF, diffusion, transformer, etc.) vary significantly from generator to generator. As a result, it is difficult for a detector to capture traces from new generation that leave different traces than those seen in training. Plus, this aligns with similar findings obtained for synthetic image detection [19].

Generation Method	Zero-Shot	Few-Shot	RER
Sora [9]	0.530	0.982	96.2%
Pika [45]	0.620	0.989	97.1%
VideoCrafter v2 [14]	0.939	0.996	93.4%

Table 9. Zero-Shot and Few-Shot detection performance of MISLnet [23], which was trained on all training generators, and tested on new generation sources. Performance numbers are measured using AUC and RER.

6.2. Transferabilty Through Few-Shot Learning

Next, we examined the ability of a synthetic video detector to detect a new generator through few-shot learning.

We began the experiments with a pre-trained detector that identified all four generators in the training set, utilizing MISLnet. We then fine-tuned the detector to recognize each generator in the Out-of-Distribution testing set, using less than one minute of video from each generator. Notably, the fine-tuning videos were not part of the testing set. Subsequently, we evaluated the updated detector's performance on each new generator in the Out-of-Distribution testing set.

Results of this experiment are shown in the second column of Table 9 from the right, titled Few-Shot. These results show that the detector can very accurately transfer to detect new generators through few-shot learning. In each case, the detector achieves an AUC of 0.98 or higher. This is a substantial increase over the AUCs achieved by zero-shot transferability.

Notably, for Sora we increase the AUC from 0.530 to 0.982 through few-shot learning. This corresponds to an AUC boost of 0.452 and an RER of 96%. These results show that synthetic video detectors can be transferred to reliably detect new generators through few-shot learning. This is particularly important given the rapid pace with which new generators such as Sora are emerging.

7. Conclusion

Our paper highlights the challenges of detecting synthetically generated videos, showing that forensic traces in synthetic images and videos differ significantly. Leading to the poor performance of existing synthetic image detectors on AI-generated videos. However, we also showed that it is possible to learn synthetic video traces through the process of training. Resulting in strong and robust detection and attribution using existing synthetic image detectors' architectures. Additionally, we showed that while these detectors have difficulties directly transferring to an unseen generator, strong performance is attainable using very little data.

Acknowledgments. This material is based on research sponsored by DARPA and the Air Force Research Laboratory (AFRL) under agreement number HR0011-20-C-0126 and by the National Science Foundation under Award No. 2320600.

References

[1] Luma AI. https://lumalabs.ai/. 2, 3, 4, 5, 8

[2] Quentin Bammey. Synthbuster: Towards detection of diffusion model generated images. *IEEE Open Journal of Signal Processing*, 5:1–9, 2024. 2

[3] Samah S Baraheem and Tam V Nguyen. Ai vs. ai: Can ai detect ai-generated images? *Journal of Imaging*, 9(10):199, 2023. 1, 3

[4] Georgios Batzolis, Jan Stanczuk, Carola-Bibiane Schönlieb, and Christian Etmann. Conditional image generation with score-based diffusion models, 2021. 2

[5] Belhassen Bayar and Matthew C Stamm. A deep learning approach to universal image manipulation detection using a new convolutional layer. In *Proceedings of the 4th ACM workshop on information hiding and multimedia security*, pages 5–10, 2016. 3

[6] Belhassen Bayar and Matthew C Stamm. Augmented convolutional feature maps for robust cnn-based camera model identification. In *2017 IEEE International Conference on Image Processing (ICIP)*, pages 4098–4102. IEEE, 2017. 4

[7] Belhassen Bayar and Matthew C Stamm. Constrained convolutional neural networks: A new approach towards general purpose image manipulation detection. *IEEE Transactions on Information Forensics and Security*, 13(11):2691–2706, 2018. 3

[8] Andreas Blattmann, Tim Dockhorn, Sumith Kulal, Daniel Mendelevitch, Maciej Kilian, Dominik Lorenz, Yam Levi, Zion English, Vikram Voleti, Adam Letts, et al. Stable video diffusion: Scaling latent video diffusion models to large datasets. *arXiv preprint arXiv:2311.15127*, 2023. 2, 3, 4, 5, 8

[9] Tim Brooks, Bill Peebles, Connor Holmes, Will DePue, Yufei Guo, Li Jing, David Schnurr, Joe Taylor, Troy Luhman, Eric Luhman, Clarence Ng, Ricky Wang, and Aditya Ramesh. Video generation models as world simulators. 2024. 2, 4, 5, 8

[10] Tiago Carvalho, Edmar R. S. de Rezende, Matheus T. P. Alves, Fernanda K. C. Balieiro, and Ricardo B. Sovat. Exposing computer generated images by eye's region classification via transfer learning of vgg19 cnn. In *2017 16th IEEE International Conference on Machine Learning and Applications (ICMLA)*, pages 866–870, 2017. 2

[11] Huiwen Chang, Han Zhang, Jarred Barber, AJ Maschinot, Jose Lezama, Lu Jiang, Ming-Hsuan Yang, Kevin Murphy, William T Freeman, Michael Rubinstein, et al. Muse: Text-to-image generation via masked generative transformers. *arXiv preprint arXiv:2301.00704*, 2023. 2

[12] Beijing Chen, Xingwang Ju, Bin Xiao, Weiping Ding, Yuhui Zheng, and Victor Hugo C de Albuquerque. Locally gan-generated face detection based on an improved xception. *Information Sciences*, 572:16–28, 2021. 3

[13] Haoxin Chen, Menghan Xia, Yingqing He, Yong Zhang, Xiaodong Cun, Shaoshu Yang, Jinbo Xing, Yaofang Liu, Qifeng Chen, Xintao Wang, Chao Weng, and Ying Shan. Videocrafter1: Open diffusion models for high-quality video generation, 2023. 2, 3, 4, 5, 8

[14] 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, 2024. 2, 5, 8

[15] Jiaxuan Chen, Jieteng Yao, and Li Niu. A single simple patch is all you need for ai-generated image detection. *arXiv preprint arXiv:2402.01123*, 2024. 1, 3

[16] Harry Cheng, Yangyang Guo, Tianyi Wang, Liqiang Nie, and Mohan Kankanhalli. Diffusion facial forgery detection. *arXiv preprint arXiv:2401.15859*, 2024. 3

[17] Yunjey Choi, Minje Choi, Munyoung Kim, Jung-Woo Ha, Sunghun Kim, and Jaegul Choo. Stargan: Unified generative adversarial networks for multi-domain image-to-image translation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 8789–8797, 2018. 2, 3, 4

[18] François Chollet. Xception: Deep learning with depthwise separable convolutions. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1251–1258, 2017. 3, 6

[19] Riccardo Corvi, Davide Cozzolino, Giada Zingarini, Giovanni Poggi, Koki Nagano, and Luisa Verdoliva. On the detection of synthetic images generated by diffusion models. In *ICASSP 2023-2023 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 1–5. IEEE, 2023. 1, 2, 3, 4, 5, 6, 8

[20] Prafulla Dhariwal and Alexander Nichol. Diffusion models beat gans on image synthesis. In *Advances in Neural Information Processing Systems*, pages 8780–8794. Curran Associates, Inc., 2021. 2

[21] Patrick Esser, Robin Rombach, and Bjorn Ommer. Taming transformers for high-resolution image synthesis. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 12873–12883, 2021. 2

[22] Chin-Shyurng Fahn and Tzu-Chin Wu. A deep-neural-network-based approach to detecting forgery images generated from various generative adversarial networks. In *2022 International Conference on Machine Learning and Cybernetics (ICMLC)*, pages 115–123, 2022. 1, 3

[23] Shengbang Fang, Tai D Nguyen, and Matthew C Stamm. Open set synthetic image source attribution. In *34th British Machine Vision Conference 2023, BMVC 2023, Aberdeen, UK, November 20-24, 2023*. BMVA, 2023. 1, 2, 3, 6, 8

[24] NA Fotedar and JH Wang. Bumblebee: Text-to-image generation with transformers. In *Proceedings of the 2019 IEEE International Conference on Image Processing (ICIP)*, pages 3465–3469, 2019. 2

[25] Songwei Ge, Thomas Hayes, Harry Yang, Xi Yin, Guan Pang, David Jacobs, Jia-Bin Huang, and Devi Parikh. Long video generation with time-agnostic vqgan and time-sensitive transformer. In *European Conference on Computer Vision*, pages 102–118. Springer, 2022. 2

[26] Michael Goebel, Lakshmanan Nataraj, Tejaswi Nanjundaswamy, Tajuddin Manhar Mohammed, Shivkumar Chandrasekaran, and BS Manjunath. Detection, attribution and localization of gan generated images. *arXiv preprint arXiv:2007.10466*, 2020. 3

[27] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. *Advances in neural information processing systems*, 27, 2014. 2

[28] Zhiqing Guo, Gaobo Yang, Jiyou Chen, and Xingming Sun. Fake face detection via adaptive manipulation traces extraction network. *Computer Vision and Image Understanding*, 204:103170, 2021. 3

[29] Moayed Haji-Ali, Guha Balakrishnan, and Vicente Ordonez. Elasticdiffusion: Training-free arbitrary size image generation. *arXiv preprint arXiv:2311.18822*, 2023. 2

[30] Ali Hatamizadeh, Jiaming Song, Guilin Liu, Jan Kautz, and Arash Vahdat. Diffit: Diffusion vision transformers for image generation. *arXiv preprint arXiv:2312.02139*, 2023. 2

[31] 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*, pages 770–778, 2016. 3, 4, 6

[32] Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. *Advances in Neural Information Processing Systems*, 33:6840–6851, 2020. 2

[33] Jonathan Ho, Chitwan Saharia, William Chan, David J Fleet, Mohammad Norouzi, and Tim Salimans. Cascaded diffusion models for high fidelity image generation. *J. Mach. Learn. Res.*, 23(47):1–33, 2022. 2

[34] Wenyi Hong, Ming Ding, Wendi Zheng, Xinghan Liu, and Jie Tang. Cogvideo: Large-scale pretraining for text-to-video generation via transformers. *arXiv preprint arXiv:2205.15868*, 2022. 2, 3, 4, 5, 8

[35] Brian Hosler, Owen Mayer, Belhassen Bayar, Xinwei Zhao, Chen Chen, James A Shackleford, and Matthew Christopher Stamm. A video camera model identification system using deep learning and fusion. In *ICASSP 2019-2019 IEEE international conference on acoustics, speech and signal processing (ICASSP)*, pages 8271–8275. IEEE, 2019. 7

[36] Brian C Hosler, Xinwei Zhao, Owen Mayer, Chen Chen, James A Shackleford, and Matthew C Stamm. The video authentication and camera identification database: A new database for video forensics. *IEEE Access*, 7:76937–76948, 2019. 3, 5

[37] Gao Huang, Zhuang Liu, Laurens Van Der Maaten, and Kilian Q Weinberger. Densely connected convolutional networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4700–4708, 2017. 3, 6

[38] Nils Hulzebosch, Sarah Ibrahimi, and Marcel Worring. Detecting cnn-generated facial images in real-world scenarios. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*, 2020. 3

[39] Yuming Jiang, Shuai Yang, Tong Liang Koh, Wayne Wu, Chen Change Loy, and Ziwei Liu. Text2performer: Text-driven human video generation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 22747–22757, 2023. 2

[40] Tero Karras, Timo Aila, Samuli Laine, and Jaakko Lehtinen. Progressive growing of gans for improved quality, stability, and variation. In *International Conference on Learning Representations*, 2018. 2, 3, 4

[41] Tero Karras, Samuli Laine, and Timo Aila. A style-based generator architecture for generative adversarial networks. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 4401–4410, 2019.

[42] Tero Karras, Samuli Laine, Miika Aittala, Janne Hellsten, Jaakko Lehtinen, and Timo Aila. Analyzing and improving the image quality of stylegan. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 8110–8119, 2020.

[43] Tero Karras, Miika Aittala, Samuli Laine, Erik Härkö nen, Janne Hellsten, Jaakko Lehtinen, and Timo Aila. Alias-free generative adversarial networks. *Advances in Neural Information Processing Systems*, 34:852–863, 2021. 2, 3, 4

[44] Matthias Kirchner and Jessica Fridrich. On detection of median filtering in digital images. In *Media forensics and security II*, pages 371–382. SPIE, 2010. 4

[45] Pika Labs. https://pika.art/. 2, 4, 5, 8

[46] Sangyup Lee, Shahroz Tariq, Youjin Shin, and Simon S Woo. Detecting handcrafted facial image manipulations and gan-generated facial images using shallow-fakefacenet. *Applied soft computing*, 105:107256, 2021. 3

[47] Haodong Li, Han Chen, Bin Li, and Shunquan Tan. Can forensic detectors identify gan generated images? In *2018 Asia-Pacific Signal and Information Processing Association Annual Summit and Conference (APSIPA ASC)*, pages 722–727. IEEE, 2018. 1, 2, 3

[48] Weichuang Li, Peisong He, Haoliang Li, Hongxia Wang, and Ruimei Zhang. Detection of gan-generated images by estimating artifact similarity. *IEEE Signal Processing Letters*, 29:862–866, 2022. 3

[49] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. Microsoft coco: Common objects in context. In *Computer Vision–ECCV 2014: 13th European Conference, Zurich, Switzerland, September 6-12, 2014, Proceedings, Part V 13*, pages 740–755. Springer, 2014. 2

[50] Ze Liu, Yutong Lin, Yue Cao, Han Hu, Yixuan Wei, Zheng Zhang, Stephen Lin, and Baining Guo. Swin transformer: Hierarchical vision transformer using shifted windows. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 10012–10022, 2021. 3, 6

[51] Peter Lorenz, Ricard L. Durall, and Janis Keuper. Detecting images generated by deep diffusion models using their local intrinsic dimensionality. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV) Workshops*, pages 448–459, 2023. 2

[52] Xin Ma, Yaohui Wang, Gengyun Jia, Xinyuan Chen, Ziwei Liu, Yuan-Fang Li, Cunjian Chen, and Yu Qiao. Latte: Latent diffusion transformer for video generation. *arXiv preprint arXiv:2401.03048*, 2024. 2

[53] Yishu Malhotra. Image forgery detection using textural features and deep learning. 2021. 1, 3

[54] Francesco Marra, Diego Gragnaniello, Davide Cozzolino, and Luisa Verdoliva. Detection of gan-generated fake images over social networks. In *2018 IEEE conference on multimedia information processing and retrieval (MIPR)*, pages 384–389. IEEE, 2018. 2, 3

[55] Francesco Marra, Diego Gragnaniello, Luisa Verdoliva, and Giovanni Poggi. Do gans leave artificial fingerprints? In *2019 IEEE Conference on Multimedia Information Processing and Retrieval (MIPR)*, pages 506–511, 2019. 2

[56] Brandon B May, Kirill Trapeznikov, Shengbang Fang, and Matthew Stamm. Comprehensive dataset of synthetic and manipulated overhead imagery for development and evaluation of forensic tools. In *Proceedings of the 2023 ACM Workshop on Information Hiding and Multimedia Security*, pages 145–150, 2023. 2

[57] Owen Mayer, Belhassen Bayar, and Matthew C Stamm. Learning unified deep-features for multiple forensic tasks. In *Proceedings of the 6th ACM workshop on information hiding and multimedia security*, pages 79–84, 2018. 7

[58] Owen Mayer, Brian Hosler, and Matthew C Stamm. Open set video camera model verification. In *ICASSP 2020-2020 IEEE international conference on acoustics, speech and signal processing (ICASSP)*, pages 2962–2966. IEEE, 2020. 7

[59] Kunj Bihari Meena and Vipin Tyagi. A deep learning based method to discriminate between photorealistic computer generated images and photographic images. In *Advances in Computing and Data Sciences: 4th International Conference, ICACDS 2020, Valletta, Malta, April 24–25, 2020, Revised Selected Papers 4*, pages 212–223. Springer, 2020. 3

[60] Zhongjie Mi, Xinghao Jiang, Tanfeng Sun, and Ke Xu. Gan-generated image detection with self-attention mechanism against gan generator defect. *IEEE Journal of Selected Topics in Signal Processing*, 14(5):969–981, 2020. 2

[61] Midjourney. Midjourney ai - free image generator. 2

[62] Runway ML. Advancing creativity with artificial intelligence. 2

[63] Mathew Monfort, SouYoung Jin, Alexander Liu, David Harwath, Rogerio Feris, James Glass, and Aude Oliva. Spoken moments: Learning joint audio-visual representations from video descriptions. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 14871–14881, 2021. 3, 5

[64] Verya Monjezi, Ashutosh Trivedi, Gang Tan, and Saeid Tizpaz-Niari. Information-theoretic testing and debugging of fairness defects in deep neural networks. In *2023 IEEE/ACM 45th International Conference on Software Engineering (ICSE)*, pages 1571–1582. IEEE, 2023. 2

[65] Lakshmanan Nataraj, Tajuddin Manhar Mohammed, Shivkumar Chandrasekaran, Arjuna Flenner, Jawadul H Bappy, Amit K Roy-Chowdhury, and BS Manjunath. Detecting gan generated fake images using co-occurrence matrices. *arXiv preprint arXiv:1903.06836*, 2019. 2

[66] Tai D Nguyen, Shengbang Fang, and Matthew C Stamm. Videofact: Detecting video forgeries using attention, scene context, and forensic traces. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 8563–8573, 2024. 3

[67] Niki Parmar, Ashish Vaswani, Jakob Uszkoreit, Lukasz Kaiser, Noam Shazeer, Alexander Ku, and Dustin Tran. Image transformer. In *International conference on machine learning*, pages 4055–4064. PMLR, 2018. 2

[68] Kha-Luan Pham, Khanh-Mai Dang, Loi-Phat Tang, and Thanh-Nhan Nguyen. Gan generated portraits detection using modified vgg-16 and efficientnet. In *2020 7th NAFOSTED Conference on Information and Computer Science (NICS)*, pages 344–349. IEEE, 2020. 3

[69] Weize Quan, Kai Wang, Dong-Ming Yan, and Xiaopeng Zhang. Distinguishing between natural and computer-generated images using convolutional neural networks. *IEEE Transactions on Information Forensics and Security*, 13(11): 2772–2787, 2018. 2

[70] Weize Quan, Pengfei Deng, Kai Wang, and Dong-Ming Yan. Cgformer: Vit-based network for identifying computer-generated images with token labeling. *IEEE Transactions on Information Forensics and Security*, 19:235–250, 2024. 2

[71] Ruslan Rakhimov, Denis Volkhonskiy, Alexey Artemov, Denis Zorin, and Evgeny Burnaev. Latent video transformer. *arXiv preprint arXiv:2006.10704*, 2020. 2

[72] Aditya Ramesh, Prafulla Dhariwal, Alex Nichol, Casey Chu, and Mark Chen. Hierarchical text-conditional image generation with clip latents. *arXiv preprint arXiv:2204.06125*, 1 (2):3, 2022. 2

[73] Apoorva Rauniyar, Aryan Raj, Ashish Kumar, Ashish Kumar Kandu, Astha Singh, and Anjani Gupta. Text to image generator with latent diffusion models. In *2023 International Conference on Computational Intelligence, Communication Technology and Networking (CICTN)*, pages 144–148, 2023. 4

[74] Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 10684–10695, 2022. 2, 3, 4

[75] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2014. 3, 6

[76] 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. 2

[77] Sergey Sinitsa and Ohad Fried. Deep image fingerprint: Towards low budget synthetic image detection and model lineage analysis. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 4067–4076, 2024. 1, 2, 3, 6

[78] Snehith.K, Anuradha.G, Vemuri, Hari Krishna.N, Sathishkumar Veerappampalayam Easwaramoorthy, and Laith Abualigah. *Detection of GAN Generated Fake Satellite Images Using Deep Learning*. 2023. 1, 3

[79] CS Sychandran and R Shreelekshmi. A hybrid xception-ensemble model for the detection of computer generated images. In *2022 IEEE International Conference on Artificial Intelligence in Engineering and Technology (IICAIET)*, pages 1–6. IEEE, 2022. 3

[80] Francesca Uccheddu, Alessia De Rosa, Alessandro Piva, and Mauro Barni. Detection of resampled images: performance analysis and practical challenges. In *2010 18th*

European Signal Processing Conference, pages 1675–1679. IEEE, 2010. 4

[81] Danial Samadi Vahdati and Matthew C Stamm. Detecting gan-generated synthetic images using semantic inconsistencies. *Electronic Imaging*, 35:1–6, 2023. 2

[82] Jinwei Wang, Ting Li, Xiangyang Luo, Yun-Qing Shi, and Sunil Kr. Jha. Identifying computer generated images based on quaternion central moments in color quaternion wavelet domain. *IEEE Transactions on Circuits and Systems for Video Technology*, 29(9):2775–2785, 2019. 2

[83] Sheng-Yu Wang, Oliver Wang, Richard Zhang, Andrew Owens, and Alexei A Efros. Cnn-generated images are surprisingly easy to spot... for now. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 8695–8704, 2020. 2

[84] Jianfeng Xiang, Jiaolong Yang, Binbin Huang, and Xin Tong. 3d-aware image generation using 2d diffusion models. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 2383–2393, 2023. 2

[85] Shivangi Yadav, Cunjian Chen, and Arun Ross. Synthesizing iris images using rasgan with application in presentation attack detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*, 2019. 2

[86] Wilson Yan, Yunzhi Zhang, Pieter Abbeel, and Aravind Srinivas. Videogpt: Video generation using vq-vae and transformers. *arXiv preprint arXiv:2104.10157*, 2021. 2

[87] Wilson Yan, Danijar Hafner, Stephen James, and Pieter Abbeel. Temporally consistent transformers for video generation. In *Proceedings of the 40th International Conference on Machine Learning*, pages 39062–39098. PMLR, 2023. 2

[88] Pengpeng Yang, Daniele Baracchi, Rongrong Ni, Yao Zhao, Fabrizio Argenti, and Alessandro Piva. A survey of deep learning-based source image forensics. *Journal of Imaging*, 6(3):9, 2020. 3

[89] Ye Yao, Zhuxi Zhang, Xuan Ni, Zhangyi Shen, Linqiang Chen, and Dawen Xu. Cgnet: Detecting computer-generated images based on transfer learning with attention module. *Signal Processing: Image Communication*, 105:116692, 2022. 3

[90] Fisher Yu, Ari Seff, Yinda Zhang, Shuran Song, Thomas Funkhouser, and Jianxiong Xiao. Lsun: Construction of a large-scale image dataset using deep learning with humans in the loop. *arXiv preprint arXiv:1506.03365*, 2015. 2

[91] Lijun Yu, Yong Cheng, Kihyuk Sohn, José Lezama, Han Zhang, Huiwen Chang, Alexander G. Hauptmann, Ming-Hsuan Yang, Yuan Hao, Irfan Essa, and Lu Jiang. Magvit: Masked generative video transformer. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 10459–10469, 2023. 2

[92] Bowen Zhang, Shuyang Gu, Bo Zhang, Jianmin Bao, Dong Chen, Fang Wen, Yong Wang, and Baining Guo. Styleswin: Transformer-based gan for high-resolution image generation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 11304–11314, 2022. 2

[93] Kejun Zhang, Yu Liang, Jianyi Zhang, Zhiqiang Wang, and Xinxin Li. No one can escape: A general approach to detect tampered and generated image. *IEEE Access*, 7:129494–129503, 2019. 2

[94] Xu Zhang, Svebor Karaman, and Shih-Fu Chang. Detecting and simulating artifacts in gan fake images. In *2019 IEEE international workshop on information forensics and security (WIFS)*, pages 1–6. IEEE, 2019. 1, 2, 3, 4

[95] Junjie Zhao, Junfeng Wu, James Msughter Adeke, Sen Qiao, and Jinwei Wang. Detecting high-resolution adversarial images with few-shot deep learning. *Remote Sensing*, 15(9): 2379, 2023. 1, 3

[96] Jun-Yan Zhu, Taesung Park, Phillip Isola, and Alexei A Efros. Unpaired image-to-image translation using cycle-consistent adversarial networks. In *Proceedings of the IEEE international conference on computer vision*, pages 2223–2232, 2017. 2, 3, 4

[97] Mingjian Zhu, Hanting Chen, Mouxiao Huang, Wei Li, Hailin Hu, Jie Hu, and Yunhe Wang. Gendet: Towards good generalizations for ai-generated image detection. *arXiv preprint arXiv:2312.08880*, 2023. 1, 2, 3

[98] Yuanzhi Zhu, Zhaohai Li, Tianwei Wang, Mengchao He, and Cong Yao. Conditional text image generation with diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 14235–14245, 2023. 2