2022 IEEE International Conference on Image Processing (ICIP) | 978-1-6654-9620-9/22/$31.00 ©2022 IEEE | DOI: 10.1109/ICIP46576.2022.9897972

MODEL ATTRIBUTION OF FACE-SWAP DEEPFAKE VIDEOS

Shan Jia[⋆] *Xin Li*[†] *Siwei Lyu*[⋆]

[⋆]University at Buffalo, State University of New York, NY, USA
[†]West Virginia University, WV, USA

ABSTRACT

AI-created face-swap videos, commonly known as *Deepfakes*, have attracted wide attention as powerful impersonation attacks. Existing research on Deepfakes mostly focuses on binary detection to distinguish between real and fake videos. However, it is also important to determine the specific generation model for a fake video, which can help attribute it to the source for forensic investigation. In this paper, we fill this gap by studying the model attribution problem of Deepfake videos. We first introduce a new dataset with DeepFakes from Different Models (DFDM) based on several Autoencoder models. Specifically, five generation models with variations in encoder, decoder, intermediate layer, input resolution, and compression ratio have been used to generate a total of $6,450$ Deepfake videos based on the same input. Then we take Deepfakes model attribution as a multiclass classification task and propose a spatial and temporal attention based method to explore the differences among Deepfakes in the new dataset. Experimental evaluation shows that most existing Deepfakes detection methods failed in Deepfakes model attribution, while the proposed method achieved over 70% accuracy on the high-quality DFDM dataset[1].

Index Terms— Face-swap Deepfakes, Model Attribution, Deepfakes Generation

1. INTRODUCTION

The term *Deepfake*, a portmanteau of 'deep learning' and 'fake', is often used to refer to the AI-synthesized face-swap videos, in which the faces of the original subject (the 'source') are replaced with those of the other person (the 'target') generated using deep learning models with the same facial expressions of the source. Since the nascence of the first Deepfakes in late 2017, they have become increasingly easier to produce thanks to the availability of open-source generation tools. Deepfakes with manipulated identities have raised wide concerns [1], especially when they are weaponized by malicious actors to target individuals or spread disinformation [2].

Accordingly, we have seen a growing research interest in Deepfake detection methods in recent years. The DeepFake detection methods exploit visual/signal artifacts [3, 4, 5, 6] and/or employ novel deep neural networks [7, 8, 9, 10]. Correspondingly, many datasets of Deepfake videos have been created in the past few years, notable examples including Faceforensics++ [11], Celeb-DF [12], DFDC [13], Deeperforensics-1.0 [14], and WildDeepfake [15].

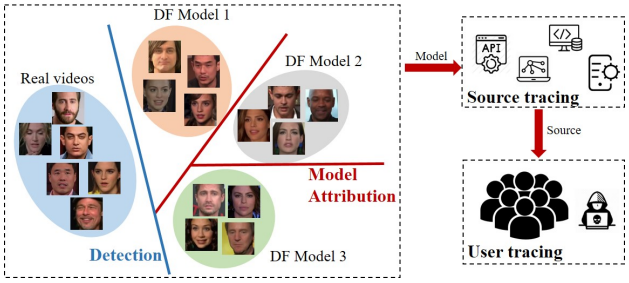

Fig. 1. Different from existing detection methods to distinguish between real and fake videos, Deepfake model attribution aims to identify the generation model of a particular DeepFake video, which may be used to further trace its author and source.

Although the state-of-the-art Deepfake detection methods have demonstrated encouraging performance on benchmark datasets, the specific means by which the DeepFakes were created are also important for forensic analysis (see Fig. 1). For instance, knowing that the synthesis model is a copy of a publicly available tool can narrow down the list of users who have downloaded it. To this end, we need a general and flexible method for Deepfake model attribution to determine which synthesis model was used to create the Deepfake video.

The model attribution problem has been considered in recent studies [16, 17, 18] in the case of Generative Adversarial Network (GAN)-based models. However, the model attribution method designed for GAN images cannot be extended to face-swap Deepfakes videos. The latter is more challenging because the Autoencoder based models to create most Deepfakes [19] will attenuate high-frequency features in the generated video frames, on which the previous methods for GAN model attribution predicate.

In this work, we provide a new method for the Deepfakes model attribution. As the majority of face-swap videos are created with tools (i.e., [20, 21]] based on the autoencoder models [19], which have higher quality in video than swaps generated with GAN-based models and require much less fine-tuning [19], we focus on such models in this work. We address two questions: 1) do different generation mod-

[1]The DFDM dataset and codes are available from `https://github.com/shanface33/Deepfake_Model_Attribution`

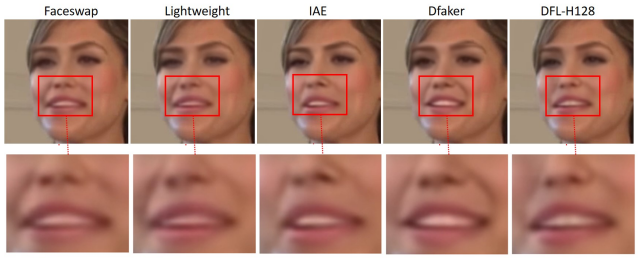

Fig. 2. Examples of Deepfakes generated by different models in our DFDM dataset. Visible but subtle visual differences can be observed in face regions. Best view when zoom in.

els result in visually similar but statistically distinguishable Deepfake videos? and 2) since several open-source generation models have been widely used to create Deepfake videos, are the differences among variations of the same generation model consistent and detectable from the input video?

To answer these questions, we first construct a new dataset as *DeepFakes generated from Different Models* (DFDM). Specifically, based on the most popular off-the-shelf software, *Facewap* [20],we have created five categories of balanced Deepfake videos using Autoencoder models with variations in the encoder, decoder, intermediate layer, and input resolution. For the second question, we formulate the face-swap Deepfakes model attribution as a multi-classification problem and design a novel Deepfakes model attribution method by extracting subtle and discriminative features based on spatial and temporal attention.

The main contributions of our work can be summarized as follows. 1) We aim to tackle the problem of fine-grained model attribution for face-swap Deepfake videos. To the best of our knowledge, this is the first method to solve this problem. 2) We provide a new dataset DFDM that highlights subtle visual differences caused by different Deepfakes generation models and indistinguishable to human eyes (see Fig.2). 3) A simple and effective Deepfakes model attribution method based on *spatial and temporal attention*, named DMA-STA, is designed and evaluated on the DFDM dataset, achieving over 70% accuracy in identifying the high-quality Deepfakes.

2. RELATED WORK

Model attribution. Model attribution aims to identify the specific model behind a synthetic image or video [18]. With an increasing number of GAN models introduced in image manipulation, several studies [16, 22, 17, 18, 23] have investigated GAN model attribution in fake images. These methods propose to extract unique artificial fingerprints left by different GAN models and perform multi-class classification for GAN model attribution. However, as far as we know, there is no previous work on Deepfakes model attribution method that works for face-swap videos. Although GAN models can be used for Deepfakes generation, they seem to work well in limited settings with even lighting conditions [19]. Consequently, most publicly available face-swap Deepfakes were created with Deepfake Autoencoder (DFAE) methods.

Table 1. Comparison of various Deepfake datasets

Dataset	#Deepfakes	#Sub	#Models	Model label?
UADFV [4]	49	49	1	-
Deepfake-TIMIT [27]	640	43	1	-
FaceForensics++ [11]	3,000	977	1	-
DFDC Preview [13]	5,244	66	2	No
DeepfakeDetection [28]	3,068	28	-	No
Celeb-DF [12]	5,639	59	1	-
DeeperForensics-1.0 [14]	10,000	100	1	-
DFDC [19]	104,500	960	6	No
WildDeepfake [15]	3,509	-	-	No
DFDM (ours)	**6,450**	**59**	**5**	**Yes**

Will variant DFAE models differentiate Deepfakes attribution? Will image-based GAN model attribution also work for DFAE with videos? Answering these questions is valuable for research on Deepfake model attribution.

Deepfakes detection. Research on Deepfake detection involves both data generation and method design. There have been several publicly available datasets with both real video and Deepfakes proposed for binary detection. Most of them created *face-swap* Deepfakes with one manipulation model, or provided imbalanced Deepfakes without model annotations (as shown in Table 1). Based on these datasets, many Deepfake detection methods have been developed. One popular category explores various artifacts, including head poses [4], emotion perception [5] etc., and signal-level artifacts in visual cues [3, 5, 6], and frequency domain [24], etc. Another category employs deep neural networks, such as Capsule network [8], ensemble of CNNs [25], and attention schemes [26]. For Deepfake model attribution, artifacts-based detection methods may fail due to the fact that the artifacts to distinguish Deepfakes from real videos exist in all Deepfakes from different models. In this paper, we will design a deep neural network-based method to learn subtle differences in Deepfake model attribution, and also evaluate how Deepfake detection methods will perform in model attribution task.

3. METHOD

3.1. The DFDM dataset

As there are no existing datasets with labeled Deepfakes generated from various models, we create a new DFDM dataset for Deepfakes model attribution. Considering that most public Deepfakes were produced based on Autoencoder architectures [19, 15], we focus on Autoencoder models for high-quality Deepfakes generation.

We use the most popular open-source software *Facewap* [20] with several optional DFAE models, including the original model (Faceswap) created by the reddit user, and models from *DeepFaceLab* [21]. As the first attempt in Deepfake video model attribution, we elaborately selected five models based on the following criteria: use the original Faceswap model as the baseline to select models with only one variation (in encoder, decoder, intermediate layer, and input resolution, respectively) for the most subtle model attribution. Table 2 shows the details of the five models, including Faceswap [20],

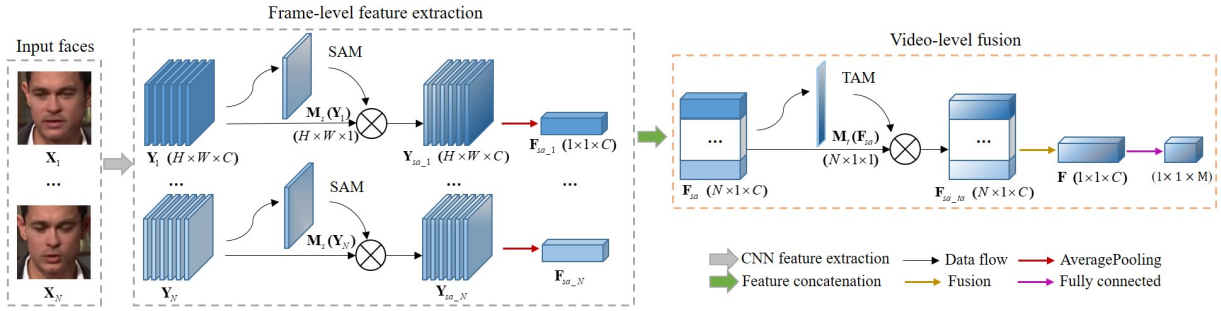

Fig. 3. Framework of the proposed DMA-STA method. Two major components are included: 1) Feature extraction from multiple single frames based on SAM (Spatial Attention Map); 2) Video-level fusion module based on TAM (Temporal Attention Map).

Table 2. Structures of Deepfakes generation models used in the proposed DFDM dataset

Model	Input	Output	Encoder	Decoder	Variation
Faceswap (baseline)	64	64	4Conv+1Ups	3Ups+1Conv	/
Lightweight	64	64	3Conv+1Ups	3Ups+1Conv	Encoder
IAE	64	64	4Conv	4Ups+1Conv	Intermediate layers; Shared Encoder&Decoder
Dfaker	64	128	4Conv+1Ups	4Ups+3Resi-dual+1Conv	Decoder
DFL-H128	128	128	4Conv+1Ups	3Ups+1Conv	Input resolution

'Conv' - convolution layer, and 'Ups' - upscaling function.

Lightweight [20], IAE [20], Dfaker [29], and DFL-H128 [30]. Note that other models provided by *Facewap* have over one variation with each other, which we believe are easier to identify than the five models we selected here.

To generate videos with high diversity, we chose the real videos in the Celeb-DF dataset [12], which contains 590 YouTube interviews of 59 celebrities. We used the S3FD detector [31] and FAN face aligner [32] for face extraction, trained each model with 100,000 iterations, and generated the final Deepfake videos in MPEG4.0 format. Three H.264 compression rates are considered to get videos with different qualities, including lossless with the constant rate factor (crf) as 0, high quality with crf as 10, and low quality with crf as 23. Totally, 6,450 Deepfakes have been created. Fig. 2 shows the face examples in Deepfakes from five models based on the same training data. The visual differences in face regions demonstrate evidence of model attribution artifacts.

3.2. The DMA-STA method

To explore if the observed artifacts among different Deepfakes are consistent and detectable, we design a model attribution method (named DMA-STA) by fusing spatial (frame-level) and temporal (video-level) attention schemes for discriminative feature extraction.

Frame-level feature extraction. We first extract frame-level features from N cropped faces $\{\mathbf{X}_i, i \in [1, N]\}$. As shown in Fig.3, a CNN model is employed to automatically extract high-level representations $\{\mathbf{Y}_i \in \mathbb{R}^{H\times W\times C}\}$ from each face $\mathbf{X}_i$. Next, the 2D Spatial Attention Map (SAM) $\{\mathbf{M}_s(\mathbf{Y}_i) \in \mathbb{R}^{H\times W\times 1}\}$ is obtained using the lightweight and flexible convolutional block attention module proposed by [33]. Specifically, the SAM is computed as:

$$\mathbf{M}_s(\mathbf{Y}_i) = \sigma(f^{7\times 7}([AvgPool(\mathbf{Y}_i)]; [MaxPool(\mathbf{Y}_i)])) \quad (1)$$

where σ denotes the sigmoid function to obtain probabilities to weigh the feature maps, $f^{7\times 7}$ is a convolution operation with the filter size of 7×7, $AvgPool$ and $MaxPool$ represent the average pooling and max pooling respectively to highlight the information regions, and ';' means feature concatenation. Based on the SAM, the adaptive features $\{\mathbf{Y}_{sa\_i} \in \mathbb{R}^{H\times W\times C}\}$ are obtained by,

$$\mathbf{Y}_{sa\_i} = \mathbf{M}_s(\mathbf{Y}_i) \otimes \mathbf{Y}_i \quad (2)$$

where $\otimes$ represents the element-wise multiplication. Note that the SAM can be integrated into each subblock in CNN, such as the ResBlock in ResNet. To aggregate the frame features, extracted features $\mathbf{Y}_{sa\_i}$ are finally fed into the average pooling layer, outputting the final frame features $\{\mathbf{F}_{sa\_i} \in \mathbb{R}^{1\times 1\times C}\}$.

Video-level fusion. Previous studies on video-level Deepfakes detection mostly fused multi-frame features based on averaging the network predictions (score fusion) [3, 7, 8, 27, 34]. To improve the classification performance, we introduce the temporal attention map (TAM) to adaptively aggregate the features from different frames. The frame features $\mathbf{F}_{sa\_i}$ from N faces are first concatenated to get the multi-frame representations $\{\mathbf{F}_{sa} \in \mathbb{R}^{N\times 1\times C}\}$, which are fused to get the adaptive representation $\{\mathbf{F}_{sa\_ta} \in \mathbb{R}^{N\times 1\times C}\}$, computed as,

$$\mathbf{F}_{sa\_ta} = \mathbf{M}_t(\mathbf{F}_{sa}) \otimes \mathbf{F}_{sa} \quad (3)$$

where $\{\mathbf{M}_t(\mathbf{F}_{sa}) \in \mathbb{R}^{N\times 1\times 1}\}$ is the temporal attention map, using the similar structure to SENet [35],

$$\mathbf{M}_t(\mathbf{F}_{sa}) = \sigma f_2(\delta f_1[AvgPool(\mathbf{F}_{sa})]) \quad (4)$$

where δ is the ReLU function, and f_1 and f_2 are two fully connected layers. Let the elements in the j channel of $\mathbf{F}_{sa\_ta}$ and $\mathbf{M}_t(\mathbf{F}_{sa})$ as $f_{i,j}$ and m_i, respectively; then the final representation $\{\mathbf{F} \in \mathbb{R}^{1\times 1\times C}\}$ is formulated as Equation (5), which is finally fed into the fully connected layer for classification (outputting the class probabilities for M category of Deepfakes).

$$\mathbf{F}_j = \frac{\sum_{i=1}^{N} f_{i,j}}{\sum_{i=1}^{N} m_i} \quad (j \in [1, C]) \quad (5)$$

The cross-entropy loss is used to train the network. Different from existing Deepfake detection methods, we design

the attention mechanism in both frame and video level for subtle difference extraction in model attribution. When compared to previous fingerprint based GAN model attribution studies, the DMA-STA method directly takes the whole faces as input instead of designing auxiliary artifacts for the multi-classification based task, so that the differences among different Deepfakes can be learned effectively and automatically.

4. EXPERIMENTS

4.1. Experimental Settings

We use the ResNet-50 [36] as the CNN feature extractor in DMA-STA.The videos in DFDM are randomly split into a training set (70%) and a testing set (30%). To balance the distribution of faces in frame selection, we selected 10 frames of each video by periodic sampling for the proposed video-level attribution scheme. Note that our experimental results show that the performance improves clearly when increasing the frame number from 1 to 10, while becomes stable for over 10 images. Consequently, the network takes the cropped faces in $224 \times 224 \times 3 \times 10$ as input. To train the network, the optimizer is set to SDG with the weight decay of 5×10^{-4}, and momentum equal to 0.9. The initial learning rate of 0.01 is divided by 10 every 40 epochs to 0.0001. A batch size of 10 is used for training with 300 epochs.

Several existing classification methods with open-source codes are re-trained and evaluated on the DFDM dataset, including the CNN based detection methods [7, 8, 11, 9], artifacts based detection [3, 24]; and GAN image model attribution methods [22, 23]. Classification accuracy is used as the evaluation metric. For a fair comparison, we implemented all these methods in video-level fusion using the same frame number (10) as the proposed DMA-STA.

4.2. Comparison with existing methods

Attention schemes. Table 3 compares our method with different attention schemes on the high-quality videos in DFDM. Our proposed DMA-STA achieves the best overall accuracy of 71.94%. Further, we can observe that these methods show better performance in identifying Deepfakes generated by models with decoder variations, i.e., Dfaker, and DFL-H128.

Table 3. Comparison of different attention schemes (%)

Attention	FS	LW	IAE	Dfaker	DFL	Overall
ResNet-50 [36]	54.84	57.36	70.54	89.92	70.54	68.02
CBAM [33]	52.42	63.57	69.77	84.50	74.42	68.53
SAM+FA [37]	64.34	42.64	76.61	74.42	79.07	66.82
SAM+Ave	58.87	51.16	76.74	80.62	79.07	68.84
Ours: DMA-STA	63.57	58.91	66.67	82.95	87.60	**71.94**

FS: Faceswap model, LW: Lightweight model, DFL: DFL-H128 model.

Classification methods. We conducted comparison experiments (all based on 10-frame feature fusion) to show how Deepfake detection and GAN model attribution methods perform in identifying Deepfakes on the high-quality DFDM dataset. Table 4 shows that most methods fail in identifying Deepfakes, with an overall accuracy less than 25%, indicating the weak or inconsistent artifacts/noise patterns among different Deepfakes. Benefiting from the rich representation of the residual network and adaptive feature learning of attention mechanisms, our method achieves an overall accuracy of 71.94%, 16% higher than the Capsule network.

Table 4. Comparison of different methods on DFDM (%)

Method	FS	LW	IAE	Dfaker	DFL	Overall
MesoInception [7]	6.98	2.33	79.07	79.07	4.65	20.93
Xception [11]	0.77	0	12.40	12.40	19.38	20.93
R3D [9]	27.13	25.58	15.5	20.16	18.61	21.40
DSP-FWA [3]	17.05	7.75	43.41	40.31	8.87	23.41
DFT-spectrum [24]	99.92	3.26	0.23	27.21	48.91	35.91
DnCNN [23]	2.33	0	0	7.75	99.22	21.86
GAN_Fingerprint [22]	20.16	22.48	54.26	21.71	26.36	28.99
Capsule[8]	32.56	42.64	69.77	73.64	58.91	55.50
Ours: DMA-STA	63.57	58.91	66.67	82.95	87.60	**71.94**

Table 5. Comparison on different video qualities (%)

Train→Test	FS	LW	IAE	Dfaker	DFL	Overall
NoL→NoL	52.42	54.26	82.17	94.6	86.82	**73.64**
NoL→HQ	41.09	42.64	75.00	61.24	83.72	**60.62**
NoL→LQ	34.88	41.09	33.59	4.65	3.88	**23.57**
HQ→HQ	63.57	58.91	66.67	82.95	87.60	**71.94**
HQ→NoL	58.14	45.74	82.03	86.82	82.17	**70.85**
HQ→LQ	20.31	27.91	53.49	56.59	36.43	**38.91**
LQ→LQ	32.26	25.58	59.69	72.09	69.77	**51.63**
LQ→NoL	50.00	22.48	78.29	59.69	52.71	**52.56**
LQ→HQ	40.62	28.68	67.44	72.09	61.24	**53.95**

LQ, HQ, and NoL denote Deepfakes with crf of 23, 10 and 0, respectively.

4.3. Comparison on different video qualities

Table 5 compares the model attribution performance of our method under both intra- and cross-quality testing scenarios on the DFDM. The video compression shows a significant influence on the attribution results, with the overall accuracy ranging from 23.57% to 73.64%. Furthermore, it can be observed that training on videos with lower qualities and testing on videos with higher quality results in better performance than vice versa, e.g., HQ $\rightarrow$ NoL with 70.85% overall accuracy, comparable to HQ $\rightarrow$ HQ results with 71.94% (same trends in LQ $\rightarrow$ NoL and LQ $\rightarrow$ HQ cases).

5. CONCLUSIONS

In this work, we have made the first attempt to explore the Deepfakes model attribution of different Autoencoder structures by creating a new dataset (DFDM) with Deepfakes from five models and designing a model attribution method for Deepfakes identification (DMA-STA). Although evaluation results show that several state-of-the-art Deepfake detection or GAN image attribution methods failed in identifying Deepfake videos, the proposed method based on both spatial and temporal attention achieves over 70% accuracy on higher-quality Deepfakes identification. Our future work includes adding more Deepfakes generation models and enhancing the robustness of the method to video compression.

Acknowledgement. This work is supported by the US DARPA Semantic Forensic (SemaFor) program, under Contract No. HR001120C0123, and NSF CITeR Award 20f1x5 "Deepfake Video Fingerprinting".

6. REFERENCES

[1] Siwei Lyu, "Deepfake detection: Current challenges and next steps," in *2020 IEEE International ECCV on Multimedia & Expo Workshops (ICMEW)*. IEEE, 2020, pp. 1–6.

[2] Bobby Chesney and Danielle Citron, "Deep fakes: A looming challenge for privacy, democracy, and national security," *Calif. L. Rev.*, vol. 107, pp. 1753, 2019.

[3] Yuezun Li and Siwei Lyu, "Exposing deepfake videos by detecting face warping artifacts," in *CVPRW*, 2019, pp. 46–52.

[4] Xin Yang, Yuezun Li, and Siwei Lyu, "Exposing deep fakes using inconsistent head poses," in *ICASSP 2019*. IEEE, 2019, pp. 8261–8265.

[5] Trisha Mittal, Uttaran Bhattacharya, et al., "Emotions don't lie: An audio-visual deepfake detection method using affective cues," in *Proceedings of the 28th ACM International Conference on Multimedia*, 2020, pp. 2823–2832.

[6] Luca Guarnera, Oliver Giudice, and Sebastiano Battiato, "Deepfake detection by analyzing convolutional traces," in *CVPRW*, 2020, pp. 666–667.

[7] Darius Afchar, Vincent Nozick, et al., "Mesonet: a compact facial video forgery detection network," in *2018 IEEE International Workshop on Information Forensics and Security (WIFS)*. IEEE, 2018, pp. 1–7.

[8] Huy H Nguyen, Junichi Yamagishi, and Isao Echizen, "Capsule-forensics: Using capsule networks to detect forged images and videos," in *ICASSP*. IEEE, 2019, pp. 2307–2311.

[9] Oscar de Lima, Sean Franklin, et al., "Deepfake detection using spatiotemporal convolutional networks," *arXiv preprint arXiv:2006.14749*, 2020.

[10] Minha Kim, Shahroz Tariq, and Simon S Woo, "Fretal: Generalizing deepfake detection using knowledge distillation and representation learning," in *CVPR*, 2021, pp. 1001–1012.

[11] Andreas Rossler, Davide Cozzolino, et al., "Faceforensics++: Learning to detect manipulated facial images," in *ICCV*, 2019, pp. 1–11.

[12] Yuezun Li, Xin Yang, et al., "Celeb-df: A large-scale challenging dataset for deepfake forensics," in *CVPR*, 2020, pp. 3207–3216.

[13] Brian Dolhansky, Russ Howes, et al., "The deepfake detection challenge (dfdc) preview dataset," *arXiv preprint:1910.08854*, 2019.

[14] Liming Jiang, Ren Li, et al., "Deeperforensics-1.0: A large-scale dataset for real-world face forgery detection," in *CVPR*, 2020, pp. 2889–2898.

[15] Bojia Zi, Minghao Chang, et al., "Wilddeepfake: A challenging real-world dataset for deepfake detection," in *Proceedings of the 28th ACM International Conference on Multimedia*, 2020, pp. 2382–2390.

[16] Ning Yu, Larry S Davis, and Mario Fritz, "Attributing fake images to gans: Learning and analyzing gan fingerprints," in *ICCV*, 2019, pp. 7556–7566.

[17] Michael Goebel, Lakshmanan Nataraj, et al., "Detection, attribution and localization of gan generated images," *Electronic Imaging*, vol. 2021, no. 4, pp. 276–1, 2021.

[18] Sharath Girish, Saksham Suri, et al., "Towards discovery and attribution of open-world gan generated images," *arXiv preprint:2105.04580*, 2021.

[19] Brian Dolhansky, Joanna Bitton, et al., "The deepfake detection challenge (dfdc) dataset," *arXiv preprint arXiv:2006.07397*, 2020.

[20] "Faceswap," `https://github.com/deepfakes/faceswap`.

[21] "Deepfacelab," `https://github.com/iperov/DeepFaceLab`.

[22] Francesco Marra, Diego Gragnaniello, et al., "Do gans leave artificial fingerprints?," in *2019 IEEE Conference on Multimedia Information Processing and Retrieval (MIPR)*. IEEE, 2019, pp. 506–511.

[23] Vishal Asnani, Xi Yin, et al., "Reverse engineering of generative models: Inferring model hyperparameters from generated images," *arXiv preprint arXiv:2106.07873*, 2021.

[24] Ricard Durall, Margret Keuper, et al., "Unmasking deepfakes with simple features," *arXiv preprint arXiv:1911.00686*, 2019.

[25] Nicolò Bonettini, Edoardo Daniele Cannas, et al., "Video face manipulation detection through ensemble of cnns," in *2020 25th International Conference on Pattern Recognition (ICPR)*. IEEE, 2021, pp. 5012–5019.

[26] Hanqing Zhao, Wenbo Zhou, et al., "Multi-attentional deepfake detection," in *CVPR*, 2021, pp. 2185–2194.

[27] Pavel Korshunov and Sébastien Marcel, "Vulnerability assessment and detection of deepfake videos," in *2019 International Conference on Biometrics (ICB)*. IEEE, 2019, pp. 1–6.

[28] Nick Dufour and Andrew Gully, "Contributing data to deepfake detection research," in *Google AI Blog*, 2019.

[29] "Dfaker," `https://github.com/dfaker/df`.

[30] Ivan Perov, Daiheng Gao, et al., "Deepfacelab: Integrated, flexible and extensible face-swapping framework," *arXiv preprint arXiv:2005.05535*, 2020.

[31] Shifeng Zhang, Xiangyu Zhu, et al., "S3fd: Single shot scale-invariant face detector," in *ICCV*, 2017, pp. 192–201.

[32] Adrian Bulat and Georgios Tzimiropoulos, "How far are we from solving the 2d & 3d face alignment problem?(and a dataset of 230,000 3d facial landmarks)," in *ICCV*, 2017, pp. 1021–1030.

[33] Sanghyun Woo, Jongchan Park, et al., "Cbam: Convolutional block attention module," in *ECCV*, 2018, pp. 3–19.

[34] Jiaming Li, Hongtao Xie, et al., "Frequency-aware discriminative feature learning supervised by single-center loss for face forgery detection," in *CVPR*, 2021, pp. 6458–6467.

[35] Jie Hu, Li Shen, and Gang Sun, "Squeeze-and-excitation networks," in *CVPR*, 2018, pp. 7132–7141.

[36] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun, "Deep residual learning for image recognition," in *CVPR*, 2016, pp. 770–778.

[37] Debin Meng, Xiaojiang Peng, et al., "Frame attention networks for facial expression recognition in videos," in *2019 ICIP*. IEEE, 2019, pp. 3866–3870.