Adversarial Deepfake Generation for Detector Misclassification

Sophie Riley Saremsky
Binghamton University
ssarems1@binghamton.edu

Umur Aybars Çiftçi
Binghamton University
uciftci@binghamton.edu

Emily Greene
Binghamton University
egreene4@binghamton.edu

İlke Demir
Intel Corporation
ilke.demir@intel.com

Abstract

With the deep learning era, synthetic content generation has become increasingly easier and popular. Especially for the case of deepfakes, this proliferation causes potentially harmful uses, from fake profiles swaying political opinions to deepfake pornography of celebrities. Similarly, deepfake detection methods are also rising to counter these malicious uses. We would like to contribute to this arms race by introducing adversarial attacks on deepfake detectors to assess their capabilities and limitations. We design our approach as a score-based black-box attack, developing a new loss function and utilizing a light-weight neural network. We evaluate our approach on five different attack and detector models, performing attacks both with the same model and across models to validate its generalizability. We report the reconstruction accuracies of perturbed fake samples, in addition to the misclassification accuracies under stress cases with postprocessing operations. We decrease the fake detection accuracy by over 80% using only perturbed fakes, and extend this up to 93% with postprocessing operations.

1. Introduction

In recent years, generative models, especially in the image domain (e.g., StyleGAN versions [32, 33]), have simplified the creation of realistic faces of people who do not exist [54]. These synthetic faces are frequently used when creating bots or deceptive media designed to be indistinguishable from human beings, especially as profile pictures on social media platforms. Such fake content is called deepfakes, which are fake images, videos, or audio designed to animate existing actors to perform non-existing actions or to generate completely synthetic content around non-existing actors [3]. Usually, the process involves seamlessly stitching a source face over a target face, where the generated image looks indistinguishable from a real face to human eye.

Malicious actors use deepfakes for pornography [56], targeting celebrities and public figures [11]. Political campaigns also exploit fake content in order to sway the views of large portions of the voting population. These efforts, when done successfully, manipulate elections [2,27]. Politicians are often the subject of deepfake videos produced to tarnish their reputation or cause misinformation [53].

Parallel to deepfake generation, machine learning methods are used also for detecting synthetic face images [44]. These cases include searching for forensic traces left in deepfake images during image generation process [23], detecting deepfakes using CNNs to examine artifacts from affine face warping [36], and extracting, segmenting, and classifying facial regions using autoencoders [43]. Although these approaches achieve high accuracies, machine learning methods need more modification as deepfakes get more accurate and realistic.

In contrast to pure learning-based detectors, another branch of deepfake detection covers biological markers. Yang et al. [59] report high accuracy rates analyzing lip features and movements to classify a synthesized video. Demir and Ciftci [17] propose a deepfake detection method that analyzes eye and gaze features to ascertain authenticity.

As deepfakes are developed to fool humans, adversarial attacks on generative models are developed to fool the detection systems by intentionally altering the content [57]. Recent studies have shown that many state-of-the-art detection methods are significantly prone to these adversarial attacks [10, 19, 20, 42]. Even simple postprocessing operations, such as adding a Gaussian Blur, are cheap and effective tools against current deepfake detection methods [21].

In this paper, our main motivation is set as deceiving deepfake detection algorithms to classify fake images are real. Adversarial attacks in deepfakes domain tend to confuse detection systems by establishing classification of all images as fake [29] (as opposed to only fake images to be

classified as fake), however it is harder to change the classification of fake images to reals, especially for biological detectors. In this work, we build a generative network based on a UNet-like [48] encoder decoder architecture, which takes a GAN-generated fake image along with a deepfake detector in a black box setting, having access to only the predicted fakery confidence of a given image. Our generator network outputs an adversarial fake image that makes any fake image appear as real for the given black box detector model. We validate our approach by training five different attack and detector models: DenseNet [28], InceptionResNetV2 [50], InceptionV3 [51], ResNet152V2 [26], and XceptionNet [14]. Our approach confuses every tested detector by more than 80% accuracy in classifying fakes. We perform and document cross-model attacks by pairwise attacker-detector experiments of the mentioned models as an adversary to each other. In addition, we also evaluate the effects of postprocessing (such as blurring or resolution change), in order to evidence that simple models provide powerful modifications as an adversary to fake detection.

2. Related Work

2.1. Deepfake Detection Methods

The growth of generative models triggered the importance of deepfake detection methods. Early models compared face boundary artifacts [36] as state-of-the-art face-swap algorithms only manipulate the inner region of the face, while the outer region is left relatively untouched. For example, Nirkin et al. [45] assert that the inner region of the face (eyes, nose, mouth, facial structure) is very simple in humans, and therefore is easy to modify in a way that still appears realistic. On the other hand, the outer regions of the face (especially the ears and hair) posses much more variation and are therefore more difficult to reconstruct in a convincing way. Another group, Matern et al. [39], claims that the features of the inner face, while easier to manipulate than the outer regions, still leave behind enough artifacts to be distinguishable from real images by a computer. For example, synthetic images frequently suffer from symmetry artifacts, i.e., two vastly different eye colors can co-exist in a face. In reality, this only occurs in about 1% of the population, and often does not affect the entire eye [40]. Matern et al. also note that the areas under the nose and eyes are usually not illuminated properly in deepfakes.

Apart from the structural ways of detecting deepfakes, pure learning-based methods are proposed to utilize the residue of generators. These methods include ResNet [34], InceptionNet [34], XceptionNet [49], compact networks with inception modules [4], shallow networks [52], RNNs [24], two stream networks [60], and ensemble networks [8]. These complex networks are limited to specific generators and their generalizability is limited.

Finally, signals from image space [23], frequency space [55], and biometric space [16] are also demonstrated to be useful in deepfake detection. Our approach primarily focuses on failing pure learning-based detectors, however extensions can be developed for other approaches too.

2.2. Adversarial Attacks

Adversarial attacks are generally broken down into two categories: white box attacks, and black box attacks [57]. In a white box attack, the attacker has access to the model's parameters. In black box attacks, the system is unknown. For the second case, images are generated with the expectation that they transfer effectively to the model in question.

2.2.1 White-Box Attacks

Gradient-based methods are generally the most effective method to-date for white-box attacks [18]. The attacker finds the gradient of the loss function for the input image and modifies it along the same direction. These can be further broken down into two categories: one-shot attacks [13] and iterative attacks [6], depending on how many steps are taken in the direction of the gradient.

The Fast Gradient Sign Method (FGSM) [22] is a frequently used one-shot approach that is computationally inexpensive and has a high amount of success for detecting deepfakes. It works by adding noise in the same direction as the gradient of the cost function of a given dataset. Essentially, it pushes the adversarial sample closer to the classified distributions. Gandhi et al. [20] are able to achieve 95% detection on unperturbed deepfakes, but using the FGSM, they are able to create perturbed images that were only correctly classified 27% of the time. Nasr et al. [41] show that many different architectures are highly susceptible to FGSM white-box attacks, with similar attack accuracies as black-box attacks and other types of white-box attacks.

Projected gradient descent (PGD) is another common white-box attack method. Currently, it is considered as one of the standard methods for large-scale constrained optimization [31]. [47], also analyzes the gradients to determine perturbations. Madry et al. [38] produce results comparing how effectively models can defend against attacks using different training methods. They show that training only with FGSM is not necessarily reliable, while training against a multi-step PGD method leads to increased resistance against attacks due to its robustness.

2.2.2 Black-Box Attacks

Since black-box attacks have less input available than white-box attacks, detection methods are generally computationally expensive and require more resources [30]. As a result, the success of black-box methods is frequently measured in both accuracy and minimum required resources.

Black-box attacks are generally split into three main categories: transfer-based attacks, score-based attacks, and decision-based attacks. Transfer-based attacks work by creating a "substitute model" which is as close to the expected target model as possible. Then adversarial examples are generated against this substitute model. Papernot et al. [46] create an adversarial example trained on a surrogate model which can mislead the target model. In DeepMisR [7], having access to the training data, authors use adversarial examples generated against the substitute model by a white-box attack to deceive a target model using transferability of the adversarial example. In score-based attacks, the only accessible knowledge cover input images for the attacked model and the relative confidence scores as output. The attack can still attempt to estimate the gradient through the information available from the model. Gradient estimation is a common score-based technique used for black-box attacks. Chen et al. [12] propose a method referred to as Zeroth Order Optimization (ZOO) to perform gradient estimation. The authors observed that their method is as effective at detecting deepfakes as most white-box attack methods. Decision-based attacks are the most restrictive type of attack with respect to how much information is available. The attacker is only able to access discrete hard-label predictions about the model. While this is the most restrictive type of a black-box attack, it is also the most representative of real-world scenarios where statistics like confidence intervals are rarely available to the attack model. In [9], starting with large adversarial perturbations and iteratively reducing them, Brendel et al. are able to generate smaller adversarial perturbations. The perturbations stayed within the adversarial region while neared the decision boundary. Liu et al [37], proposed a geometry-inspired decision-based attack to reduce the number of queries by constraining adversarial perturbations to low frequency subspace in order to fool commercial image recognition systems.

For our system, we assume that the only accessible knowledge is the probability that an image is real or fake, which corresponds to the score-based black-box attack model. This is inline with the real life scenario; many online detectors used by companies will not output the classification result, but only the class probabilities.

3. System Overview

We design our attack model based on detection accuracies of different deepfake detectors, between each pair of attacker and detector models. In order to further show that simple and less computationally expensive methods can still be fairly effective at detecting synthetically generated images, we fortify the analysis with the detection results with simple image modifications like blurring and resizing.

When it comes to traditional image authenticity classification networks, outputs tend to be binary; either real or fake. These binary outputs range from 0 to 1 depending on the authenticity of the image. As synthetic images can be completely synthetic based on a distribution or partially modified as in reanimation cases, the detector model needs to give a number representing how real it thinks the image is. One of the common ways to provide this information is to use a floating point output between 0 to 1, that contains the authenticity percentage of the image where 1 means 100% real and 0 means 100% fake. An example of this percentage-based detection model can be found in commercially available deepfake detectors [1]. In our attack model, we assume we have a detector model as a blackbox that is trained to identify synthetic fake images and we do not have any information about it. We can only interact with the model by input output pairs, where the output represents the relative authenticity value reported by the specific detector.

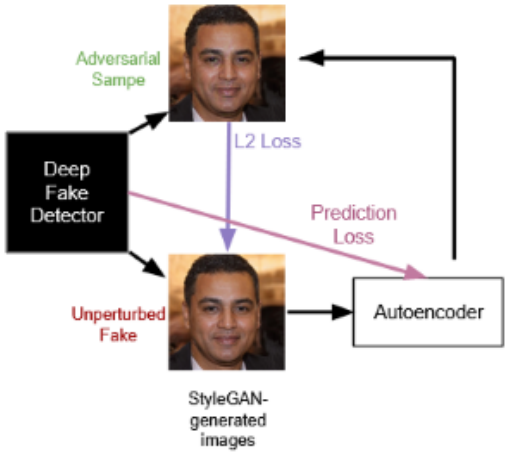

Figure 1. **System Overview.** Fake samples from the dataset are trained in an encoder-decoder architecture with perturbation and reconstruction losses.

In our generative model, we employ a UNet-style [48] encoder-decoder based generative adverserial network to refine fake images by introducing perturbations to fool detector models into classifying them as real. Figure 1 shows a general overview of our adversarial generation. The reason we choose UNet as a base is for its ability to keep the structural integrity of an image without much effort, thanks to its design to fuse layer information with its spatially complementing layers. Instead of targeting the detector models with a model-specific noise that can be added to every image, we follow the path of predicting image-specific noise with this adversarial training process. This also enables adversarial images that fail one detector model to be able to fool others too. Therefore, our model takes a synthetic fake image and modifies it in a way that minimizes the reconstruction error between the modified fake image and the provided fake image, while maximizing the detection error.

Our encoder decoder architecture has three parts: encoder, decoder, and ending. The encoder and the corresponding decoder contain four CNN segments (eight in total) where each segment consists of two convolutional layers with ReLU activations [5], followed by max pooling and upsampling on the encoder part, convolution with a ReLU activation, and ends with concatenation of layers on the decoder part. In our concatenation layer in the decoder segment, we combine information from the second convolutional layer of each corresponding encoder segment. After our encoder and decoder, we add two more convolutional layers with ReLU activations and one convolutional layer with sigmoid activation. We use Kaiming He initialization [25] for every layer of our generator network.

During training, our loss function has two components. First, we optimize for the mean square error between the perturbed fake image and the original fake image to ensure our generated image looks similar to the fake image. Second, we optimize for the prediction from the target detector model. We notice that the contributions of loss terms are not balanced and the prediction loss ends up dominating the reconstruction loss and creates visible changes. We balance it with an empirically determined weight of 0.001 to keep the image similar to the unperturbed fake image. This results in our final loss to have the following form:

$$L = MSE(I, G(I)) + 0.001 * D(G(I)) \quad (1)$$

where I is the provided fake image that will be modified to pass as real, $G(I)$ is the generated image by our network and $D(G(I))$ is the deepfake detector's prediction response.

4. Results

Both our detector models and our framework is implemented using keras [15] library. The training for the adversarial generation is done on an NVIDIA 1060GTX for 200 epochs, where Adam [35] is chosen as the optimizer with a learning rate of 1e-4. Our dataset contains 140,000 faces [58] with 70,000 reals / 70,000 fakes generated by StyleGAN [32], of which we use 60,000 real / 60,000 fake images for training detector models and 10,000 real / 10,000 fake images of diverse people for testing our generator.

Model	Fake Images	Real Images
DenseNet [28]	0.6%	99.5%
Inc.ResNet [34]	0.6%	99.7%
InceptionV3 [51]	0.8%	99.8%
ResNet152V2 [26]	2.1%	99.7%
XceptionNet [14]	0.5%	99.7%

Table 1. **Real detection accuracies** of unmodified fake and real images as the baseline for each deepfake detector.

We document the initial **real** detection accuracies of five detectors on our dataset in Table 1. In other words, more than 99% of real images are detected as real and almost none of the fake images are detected as real. This exercise verifies that selected detectors are suitable by working almost perfectly on our dataset for deepfake detection.

Model	[28]	[34]	[51]	[26]	[14]
DenseNet [28]	**79.9**	72.0	78.6	80.4	64.4
Inc.ResNet [34]	76.7	**83.6**	83.8	75.4	72.5
InceptionV3 [51]	71.4	81.2	**87.5**	71.1	73.5
ResNet [26]	77.2	73.3	73.1	**84.9**	63.9
XceptionNet [14]	82.5	83.2	87.5	82.7	**80.9**

Table 2. **Real detection accuracies of perturbed fakes** demonstrating adversarial misclassification both per-model (diagonal) and cross-model (others).

To measure the success of our adversarial samples, we run our framework on the aforementioned dataset, and do the same evaluation on the perturbed fake images. As our main motivation is to confuse detectors to classify fakes as reals, we document only **real** classification accuracy on **fake** samples (Tab 2). The accuracies in bold indicate the main accuracies when the training and testing models are the same, for example, DenseNet fake detection accuracy is **reduced from 99.4% to 20.1%**, comparing the first cells of Table 1 and 2. All detector models show high accuracies (approximately or greater than 80%) of being able to misclassify the adversarial images.

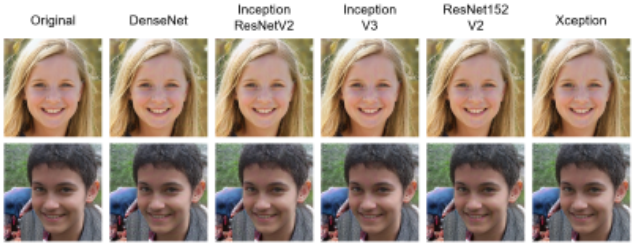

Figure 2. **Sample fake images** with adversarial perturbations generated to target different models.

In addition, we show example adversarial fake images created by our algorithm from the fake samples in the database per different detectors in Figure 2, supporting that there is no significant visual artifacts introduced by the adversarial training process.

4.1. Effects of Post-processing

We explore the effects of blurring and resizing operations on the adversarially created samples. To have a fair comparison; we first apply Gaussian blur with 3x3, 5x5, and 7x7 kernels, and resizing to +-10% of the original size; to the unmodified real and fake samples in the dataset 3. We observe

	Real					Fake				
	3x3B	5x5B	7x7B	R90	R110	3x3B	5x5B	7x7B	R90	R110
DenseNet [28]	99.6%	99.7%	99.7%	99.7%	99.7%	1.2%	2.0%	4.4%	1.7%	1.1%
InceptionResNet [34]	99.8%	99.9%	99.9%	99.7%	99.7%	1.4%	2.8%	10.1%	1.6%	1.1%
InceptionV3 [51]	99.9%	99.9%	99.9%	99.9%	99.9%	1.7%	3.7%	12.3%	2.2%	1.6%
ResNet152V2 [26]	99.7%	99.8%	99.9%	99.8%	99.7%	2.9%	4.3%	8.1%	3.5%	2.4%
XceptionNet [14]	99.7%	99.7%	99.8%	99.7%	99.7%	0.8%	1.3%	3.0%	1.3%	0.8%

Table 3. **Effects of post processing** for the real classification accuracies of the **unmodified** (or original) real and fake images. 3x3B, 5x5B, and 7x7B correspond to Gaussian blur kernel sizes. R90 and R110 represent 10% down and upsampling of the image.

	Real					Fake				
	3x3B	5x5B	7x7B	R90	R110	3x3B	5x5B	7x7B	R90	R110
DenseNet [28]	94.7%	95.8%	97.1%	93.5%	93.2%	86.9%	89.0%	91.4%	85.5%	84.7%
InceptionResNet [34]	99.1%	99.6%	99.8%	98.3%	98.3%	86.5%	87.8%	90.4%	85.9%	85%
InceptionV3 [51]	99.4%	99.6%	99.8%	99.4%	99.3%	90.3%	91.3%	93.2%	90.0%	89.1%
ResNet152V2 [26]	95.5%	96.4%	97.6%	94.5%	94.4%	87.4%	88.7%	90.05%	86.1%	85.8%
XceptionNet [14]	94.7%	94.9%	95.4%	94.9%	94.2%	81.5%	82.8%	85.3%	83%	80.9%

Table 4. **Effects of post processing** for the real classification accuracies of the **adversarial** (or original) real and fake images. 3x3B, 5x5B, and 7x7B correspond to Gaussian blur kernel sizes. R90 and R110 represent 10% down and upsampling of the image.

that the real classification accuracy does not drop significantly compared to table 1, as the change is in the order of 1%. Next, we apply the same set of operations to the adversarially perturbed fake images (Table 4). We note that the misclassification increases even more than that of 2, concluding that performing post processing operations quantitatively lowers the correct classification accuracy on synthesized fake images. Sample blurred images corresponding to this experiment are shown in Figure 3.

Figure 3. **Example images with Gaussian blur** before being altered for adversarial attacks.

4.2. Reconstruction Fidelity

Following our visual results, we also measure the reconstruction quality of the adversarial samples to stay as loyal as possible to the original fake images. Table 5 documents average scores for various image metrics, for the adversarial counterparts of all the images in the dataset. We conclude that there is not too much noise introduced (low PSNR), the reconstruction is accurate (low RMSE), and adversarial images are structurally similar to original fakes (high SSIM).

Model	PSNR	RMSE	SSIM
DenseNet [28]	35.4007	4.3709	0.9495
InceptionResNet [34]	34.8324	4.6583	0.9487
InceptionV3 [51]	35.1660	4.4893	0.9484
ResNet152V2 [26]	35.3268	4.4004	0.9490
XceptionNet [14]	35.2031	4.4711	0.9482

Table 5. **Image space metrics** for fake images generated to target different models, evaluated by noise, accuracy, and structure.

4.3. Cross-Model Attacks

Generalizability of adversarial attacks is one of the most important features to make it adopted widely. In order to assess that, we conduct a cross-model experiment where our system is trained for a specific model and tested on another model. Table 2 lists the results of the detector networks on our altered fake images along with cross-model

prediction results between detector models (accuracies on non-diagonal cells). Similar to Section 4, the results indicate the percentage of altered fake images classified as real. We can securely conclude that even if the model is trained for targeting a single detector model, it still works for other never-before-seen models with only minimal accuracy drops. Moreover, the accuracy even improved for one case; DenseNet detector model had 79.9% accuracy when tested with the DenseNet attack model, but an 82.5% accuracy when tested with the XceptionNet attack model.

5. Conclusion

We present an adversarial generation framework for creating perturbed fake images to deceive deepfake detectors. Our model uses a new loss function and utilizes a generic UNet [48] model to create adversarial fakes. We compare the detection accuracy of five deepfake detectors before and after the adversarial attacks, perform cross-model attacks for generalizability, measure the reconstruction quality for image fidelity, and analyze the effects of post-processing artifacts on the misclassification results. Ultimately, our results show that, training robust black-box models is highly successful at creating adversarial images to trick the detectors, and operations like resizing or blurring an image (which are computationally much less expensive) still have fairly successful outcomes on top of the adversarial generation. In future, we would like to extend our work to trick other models such as biological and spatiotemporal detectors. As those detectors depend on the preservation of authentic information in real images (as opposed to current detectors depending on artifacts of fake images), we would like to solve this new challenge with additional loss terms to fake the authenticity clues.

References

[1] Deepware. deepware.ai. Accessed: 2022-03-08. 3

[2] The rise of the deepfake and the threat to democracy. https://www.theguardian.com/technology/ng-interactive/2019/jun/22/the-rise-of-the-deepfake-and-the-threat-to-democracy. Accessed: 2022-03-08. 1

[3] Sally Adee. What are deepfakes and how are they created?, Jun 2021. 1

[4] D. Afchar, V. Nozick, J. Yamagishi, and I. Echizen. Mesonet: a compact facial video forgery detection network. In *IEEE International Workshop on Information Forensics and Security (WIFS)*, pages 1–7, Dec 2018. 2

[5] Abien Fred Agarap. Deep learning using rectified linear units (relu). *arXiv preprint arXiv:1803.08375*, 2018. 4

[6] Rima Alaifari, Giovanni S Alberti, and Tandri Gauksson. Adef: an iterative algorithm to construct adversarial deformations. *arXiv preprint arXiv:1804.07729*, 2018. 2

[7] Song Bai, Yingwei Li, Yuyin Zhou, Qizhu Li, and Philip HS Torr. Metric attack and defense for person re-identification. 2019. 3

[8] Nicolo Bonettini, Edoardo Daniele Cannas, Sara Mandelli, Luca Bondi, Paolo Bestagini, and Stefano Tubaro. Video face manipulation detection through ensemble of cnns. In *2020 25th International Conference on Pattern Recognition (ICPR)*, pages 5012–5019. IEEE, 2021. 2

[9] Wieland Brendel, Jonas Rauber, and Matthias Bethge. Decision-based adversarial attacks: Reliable attacks against black-box machine learning models. *arXiv preprint arXiv:1712.04248*, 2017. 3

[10] Nicholas Carlini and Hany Farid. Evading deepfake-image detectors with white- and black-box attacks. *CoRR*, abs/2004.00622, 2020. 1

[11] Angela Chen. Forget fake news-nearly all deepfakes are being made for porn, Apr 2020. 1

[12] Pin-Yu Chen, Huan Zhang, Yash Sharma, Jinfeng Yi, and Cho-Jui Hsieh. Zoo. *Proceedings of the 10th ACM Workshop on Artificial Intelligence and Security*, Nov 2017. 3

[13] Xuesong Chen, Xiyu Yan, Feng Zheng, Yong Jiang, Shu-Tao Xia, Yong Zhao, and Rongrong Ji. One-shot adversarial attacks on visual tracking with dual attention. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10176–10185, 2020. 2

[14] François Chollet. Xception: Deep learning with depthwise separable convolutions. *CoRR*, abs/1610.02357, 2016. 2, 4, 5

[15] François Chollet et al. Keras. https://github.com/fchollet/keras, 2015. 4

[16] Umur Aybars Ciftci, Ilke Demir, and Lijun Yin. Fakecatcher: Detection of synthetic portrait videos using biological signals. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2020. 2

[17] Ilke Demir and Umur Aybars Ciftci. *Where Do Deep Fakes Look? Synthetic Face Detection via Gaze Tracking*. Association for Computing Machinery, New York, NY, USA, 2021. 1

[18] Javid Ebrahimi, Anyi Rao, Daniel Lowd, and Dejing Dou. Hotflip: White-box adversarial examples for text classification. *arXiv preprint arXiv:1712.06751*, 2017. 2

[19] Steven Lawrence Fernandes and Sumit Kumar Jha. Adversarial attack on deepfake detection using rl based texture patches. In *ECCV Workshops (1)*, pages 220–235, 2020. 1

[20] Apurva Gandhi and Shomik Jain. Adversarial perturbations fool deepfake detectors. *CoRR*, abs/2003.10596, 2020. 1, 2

[21] Akhil Goel, Anirudh Singh, Akshay Agarwal, Mayank Vatsa, and Richa Singh. Smartbox: Benchmarking adversarial detection and mitigation algorithms for face recognition. In *2018 IEEE 9th International Conference on Biometrics Theory, Applications and Systems (BTAS)*, pages 1–7. IEEE, 2018. 1

[22] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. In Z. Ghahramani, M. Welling, C. Cortes, N. Lawrence, and K. Q. Weinberger, editors, *Advances in Neural Information Processing Systems*, volume 27. Curran Associates, Inc., 2014. 2

[23] Luca Guarnera, Oliver Giudice, and Sebastiano Battiato. Deepfake detection by analyzing convolutional traces. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops*, pages 666–667, 2020. 1, 2

[24] D. Güera and E. J. Delp. Deepfake video detection using recurrent neural networks. In *15th IEEE International Conference on Advanced Video and Signal Based Surveillance (AVSS)*, pages 1–6, Nov 2018. 2

[25] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Delving deep into rectifiers: Surpassing human-level performance on imagenet classification. In *Proceedings of the IEEE international conference on computer vision*, pages 1026–1034, 2015. 4

[26] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Identity mappings in deep residual networks. *CoRR*, abs/1603.05027, 2016. 2, 4, 5

[27] Philip N. Howard. How political campaigns weaponize social media bots, Jul 2021. 1

[28] Gao Huang, Zhuang Liu, and Kilian Q. Weinberger. Densely connected convolutional networks. *CoRR*, abs/1608.06993, 2016. 2, 4, 5

[29] Xiaowei Huang, Daniel Kroening, Wenjie Ruan, James Sharp, Youcheng Sun, Emese Thamo, Min Wu, and Xinping Yi. A survey of safety and trustworthiness of deep neural networks: Verification, testing, adversarial attack and defence, and interpretability. *Computer Science Review*, 37:100270, 2020. 1

[30] Andrew Ilyas, Logan Engstrom, Anish Athalye, and Jessy Lin. Black-box adversarial attacks with limited queries and information. In *International Conference on Machine Learning*, pages 2137–2146. PMLR, 2018. 2

[31] Wei Jin, Yaxin Li, Han Xu, Yiqi Wang, Shuiwang Ji, Charu Aggarwal, and Jiliang Tang. Adversarial attacks and defenses on graphs: A review, a tool and empirical studies. *arXiv preprint arXiv:2003.00653*, 2020. 2

[32] 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 (CVPR)*, June 2019. 1, 4

[33] 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. 1

[34] A. Khodabakhsh, R. Ramachandra, K. Raja, P. Wasnik, and C. Busch. Fake face detection methods: Can they be generalized? In *2018 International Conference of the Biometrics Special Interest Group (BIOSIG)*, pages 1–6, Sep. 2018. 2, 4, 5

[35] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014. 4

[36] Yuezun Li and Siwei Lyu. Exposing deepfake videos by detecting face warping artifacts. *CoRR*, abs/1811.00656, 2018. 1, 2

[37] Yujia Liu, Seyed-Mohsen Moosavi-Dezfooli, and Pascal Frossard. A geometry-inspired decision-based attack. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 4890–4898, 2019. 3

[38] Aleksander Madry, Aleksandar Makelov, Ludwig Schmidt, Dimitris Tsipras, and Adrian Vladu. Towards deep learning models resistant to adversarial attacks, Sep 2019. 2

[39] Falko Matern, Christian Riess, and Marc Stamminger. Exploiting visual artifacts to expose deepfakes and face manipulations. In *2019 IEEE Winter Applications of Computer Vision Workshops (WACVW)*, pages 83–92, 2019. 2

[40] Remy Melina. Why do some people have differently colored eyes?, Jan 2011. 2

[41] Milad Nasr, Reza Shokri, and Amir Houmansadr. Comprehensive privacy analysis of deep learning: Passive and active white-box inference attacks against centralized and federated learning, 2019. 2

[42] Paarth Neekhara, Shehzeen Hussain, Malhar Jere, Farinaz Koushanfar, and Julian J. McAuley. Adversarial deepfakes: Evaluating vulnerability of deepfake detectors to adversarial examples. *CoRR*, abs/2002.12749, 2020. 1

[43] Huy H. Nguyen, Fuming Fang, Junichi Yamagishi, and Isao Echizen. Multi-task learning for detecting and segmenting manipulated facial images and videos. *CoRR*, abs/1906.06876, 2019. 1

[44] Thanh Thi Nguyen, Cuong M. Nguyen, Dung Tien Nguyen, Duc Thanh Nguyen, and Saeid Nahavandi. Deep learning for deepfakes creation and detection. *CoRR*, abs/1909.11573, 2019. 1

[45] Yuval Nirkin, Lior Wolf, Yosi Keller, and Tal Hassner. Deepfake detection based on the discrepancy between the face and its context, Aug 2020. 2

[46] Nicolas Papernot, Patrick McDaniel, Ian Goodfellow, Somesh Jha, Z Berkay Celik, and Ananthram Swami. Practical black-box attacks against machine learning. In *Proceedings of the 2017 ACM on Asia conference on computer and communications security*, pages 506–519, 2017. 3

[47] Kexin Pei, Yinzhi Cao, Junfeng Yang, and Suman Jana. Deepxplore: Automated whitebox testing of deep learning systems. In *proceedings of the 26th Symposium on Operating Systems Principles*, pages 1–18, 2017. 2

[48] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. In *International Conference on Medical image computing and computer-assisted intervention*, pages 234–241. Springer, 2015. 2, 3, 6

[49] Andreas Rössler, Davide Cozzolino, Luisa Verdoliva, Christian Riess, Justus Thies, and Matthias Nießner. FaceForensics++: Learning to detect manipulated facial images. In *International Conference on Computer Vision (ICCV)*, 2019. 2

[50] Christian Szegedy, Sergey Ioffe, and Vincent Vanhoucke. Inception-v4, inception-resnet and the impact of residual connections on learning. *CoRR*, abs/1602.07261, 2016. 2

[51] Christian Szegedy, Vincent Vanhoucke, Sergey Ioffe, Jonathon Shlens, and Zbigniew Wojna. Rethinking the inception architecture for computer vision. *CoRR*, abs/1512.00567, 2015. 2, 4, 5

[52] Shahroz Tariq, Sangyup Lee, Hoyoung Kim, Youjin Shin, and Simon S. Woo. Detecting both machine and human created fake face images in the wild. In *Proceedings of the 2Nd International Workshop on Multimedia Privacy and Security*, MPS '18, pages 81–87, New York, NY, USA, 2018. ACM. 2

[53] Rob Toews. Deepfakes are going to wreak havoc on society. we are not prepared., Dec 2021. 1

[54] Ruben Tolosana, Ruben Vera-Rodriguez, Julian Fierrez, Aythami Morales, and Javier Ortega-Garcia. Deepfakes and beyond: A survey of face manipulation and fake detection. *Information Fusion*, 64:131–148, 2020. 1

[55] Sheng-Yu Wang, Oliver Wang, Richard Zhang, Andrew Owens, and Alexei A Efros. Cnn-generated images are surprisingly easy to spot...for now. In *CVPR*, 2020. 2

[56] Tamsin Selbie amp; Craig Williams. Deepfake pornography could become an 'epidemic', expert warns, May 2021. 1

[57] Rey Reza Wiyatno, Anqi Xu, Ousmane Dia, and Archy de Berker. Adversarial examples in modern machine learning: A review. *arXiv preprint arXiv:1911.05268*, 2019. 1, 2

[58] Xhlulu. 140k real and fake faces, Feb 2020. 4

[59] Chen-Zhao Yang, Jun Ma, Shilin Wang, and Alan Wee-Chung Liew. Preventing deepfake attacks on speaker authentication by dynamic lip movement analysis. *IEEE Transactions on Information Forensics and Security*, 16:1841–1854, 2021. 1

[60] P. Zhou, X. Han, V. I. Morariu, and L. S. Davis. Two-stream neural networks for tampered face detection. In *2017 IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, pages 1831–1839, July 2017. 2