FakeSpotter: A Simple yet Robust Baseline for Spotting AI-Synthesized Fake Faces

Run Wang[1], **Felix Juefei-Xu**[2], **Lei Ma**[3], **Xiaofei Xie**[1], **Yihao Huang**[4], **Jian Wang**[5], **Yang Liu**[1]
[1] **Nanyang Technological University,**[2] **Alibaba Group**
[3] **Kyushu University,**[4] **East Normal University,**[5] **Xiaomi AI Lab**

Abstract

In recent years, generative adversarial networks (GANs) and its variants have achieved unprecedented success in image synthesis. They are widely adopted in synthesizing facial images which brings potential security concerns to humans as the fakes spread and fuel the misinformation. However, robust detectors of these AI-synthesized fake faces are still in their infancy and are not ready to fully tackle this emerging challenge. In this work, we propose a novel approach, named *FakeSpotter*, based on monitoring neuron behaviors to spot AI-synthesized fake faces. The studies on neuron coverage and interactions have successfully shown that they can be served as testing criteria for deep learning systems, especially under the settings of being exposed to adversarial attacks. Here, we conjecture that monitoring neuron behavior can also serve as an asset in detecting fake faces since layer-by-layer neuron activation patterns may capture more subtle features that are important for the fake detector. Experimental results on detecting four types of fake faces synthesized with state-of-the-art GANs (including just released **StyleGAN2** [Karras *et al.*, 2019b] and **DFDC Dataset**[1]) and evading against four perturbation attacks show the effectiveness and robustness of our approach.

1 Introduction

With the remarkable development of AI, particularly GANs, seeing is not believing in nowadays. GANs (*e.g.*, StyleGAN [Karras *et al.*, 2019a], STGAN [Liu *et al.*, 2019], and StarGAN [Choi *et al.*, 2018]) exhibit powerful capabilities in synthesizing human imperceptible fake images and editing images in a natural way. Humans can be easily fooled by these synthesized fake images[2]. Figure 1 presents four typical fake faces synthesized with various GANs. Nobody is likely to distinguish the real and fake face with high confidence.

[1]Deepfakes Detection Challenge Dataset (DFDC Dataset) announced by Facebook.
https://www.kaggle.com/c/deepfake-detection-challenge
[2]https://thispersondoesnotexist.com

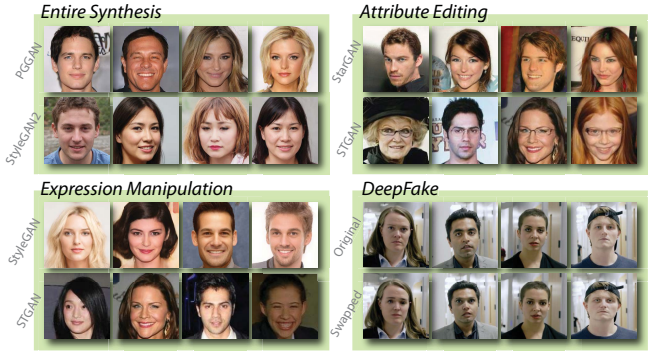


Figure 1: Four types of fake faces synthesized with various GANs. For entire synthesis, the facial image are non-existent faces in the world. For attribute editing, StarGAN changes the color of hair into brown and STAGN wears an eyeglasses. For expression manipulation, both StyleGAN and STGAN manipulate the face with a smile expression. For DeepFake, the data is from the deepfake dataset in FaceForensics++ [Rössler *et al.*, 2019] and they involves face swap.

The AI-synthesized fake faces not only bring funs to users but also raise security and privacy concerns and even panics to everyone including celebrities, politicians, *etc*. Some apps (*e.g.*, FaceApp, Reflect, and ZAO) employ face-synthesis techniques to provide attractive and interesting services such as face swap, facial expression manipulation with several taps on mobile devices. Unfortunately, abusing AI in synthesizing fake images will raise security and privacy concerns like creating fake pornography [Cole, 2018], where a victim's face can be naturally swapped into a naked body and indistinguishable to humans' eyes with several photos [Zakharov *et al.*, 2019]. Politicians will also be confused by fake faces, for example, fake official statements may be announced with nearly realistic facial expressions and body movements by adopting AI-synthesized fake face techniques. Due to the potential and severe threats of fake faces, it is urgent to call for effective techniques to spot fake faces in the wild. In this paper, the AI-synthesized fake face or fake face mean the face is synthesized with GANs unless particular declaration.

Entire face synthesis, **facial attribute editing**, **facial expression manipulation**, and **Deepfake** are four typical fake faces synthesized with various GANs [Stehouwer *et al.*, 2019; Tolosana *et al.*, 2020]. Entire face synthesis means a facial image can be totally synthesized with GANs and the face maybe not existed in the world. Facial attribute edit-

ing manipulates single or several attributes in a face like hair, eyeglass, gender, *etc*. Facial expression manipulation alters one's facial expression or transforms facial expressions among persons. Deepfake is also known as identity swap. It normally swaps synthesized face between different persons and widely applied in producing fake videos [Agarwal *et al.*, 2019]. More recently, there are some work which start to study this topic. But none of these work fully tackle the four types of fake faces and thoroughly evaluate their robustness against perturbation attack with various transformations to present their potentials dealing with fakes in the wild.

In this paper, we propose a novel approach, named FakeSpotter, which detects fake faces by monitoring neuron behaviors of deep face recognition (FR) systems with a simple binary-classifier. Specifically, FakeSpotter leverages the power of deep FR systems in learning the representations of faces and the capabilities of neurons in monitoring the layer-by-layer behaviors. The neuron activation behaviors can capture more subtle differences for distinguishing between real and fake faces.

To evaluate the effectiveness of our approach in detecting fake faces and its robustness against various perturbation attacks. We collect numerous high-quality fake faces produced with state-of-the-art GANs. For example, our entire synthesized fake faces are generated with just released StyleGAN2 [Karras *et al.*, 2019b], facial attributes are edited with newest STGAN [Liu *et al.*, 2019], *Deepfake* is from publicly datasets (*e.g.*, FaceForensics++ [Rössler *et al.*, 2019] and *Celeb-DF* [Yuezun Li and Lyu, 2019]) and a real Deepfake detection competition dataset, DFDC, announced by Facebook. Experiments are evaluated on our collected four types of high-quality fake faces and results demonstrated the effectiveness of FakeSpotter in spotting fake faces and its robustness in tackling four perturbation attacks (including **adding noise**, **blur**, **compression**, and **resizing**). FakeSpotter also outperforms prior work AutoGAN [Zhang *et al.*, 2019] and gives an average detection accuracy more than 90% on the four types of fake faces. The average performance measured by AUC score is decreased less than 3.77% in tackling the four perturbation attacks under various intensities. Our main contributions are summarized as follows.

- **New observation of neurons in spotting AI-synthesized fake faces.** We observe that layer-by-layer neuron behaviors can be served as an asset for distinguishing fake faces. Additionally, it can also robust against perturbation attacks.
- **Presenting a new insight for spotting AI-synthesized fake faces by monitoring neuron behaviors.** We propose the first neuron coverage based fake detection approach that monitors the layer-by-layer neuron behaviors in deep FR systems. Our approach provides a novel insight for spotting AI fakes with neuron coverage techniques.
- **Performing the first comprehensive evaluation on four typical AI-synthesized fake faces and robustness against four common perturbation attacks.** Experiments are conducted on our collected high-quality fake faces synthesized with state-of-the-art GANs and real dataset like DFDC. Experimental results demonstrated the effectiveness and robustness of our approach.

2 Related Work

2.1 Image Synthesis

GAN has achieved impressive progress in image synthesis [Zhu *et al.*, 2017; Yi *et al.*, 2017] which is the most well-studied area of the applications of GAN since it is first proposed in 2014 [Goodfellow *et al.*, 2014]. The generator in GAN learns to produce synthesized samples that are almost identical to real samples, while the discriminator learns to differentiate them. Recently, various GANs are proposed for facial images synthesis and manipulation.

In entire face synthesis, PGGAN [Karras *et al.*, 2017] and StyleGAN, created by NVIDIA, produce faces in large resolution with unprecedented quality and synthesize non-existent faces in the world. STGAN and StarGAN focus on face editing which manipulates the attributes and expressions of humans' faces. For example, change the color of hair, wear an eyeglasses, and laugh with a smile or feared expression, *etc*. *FaceApp* and *FaceSwap*[3] employ GANs to generate *Deepfake* which involves identity swap.

Currently, GANs can be well applied in synthesizing entire fake faces, editing facial attributes, manipulating facial expressions, and swapping identities among persons (also known as Deepfake). Fake faces synthesized with state-of-the-art GANs are almost indistinguishable to humans. We are living in a world where we cannot believe our eyes anymore.

2.2 Fake Face Detection

Some researchers employ traditional forensics-based techniques to spot fake faces/images. These work inspect the disparities in pixel-level between real and fake images, however, they are susceptible to perturbation attack like compression which is common in producing videos with still images [Böhme and Kirchner, 2013]. Another line in detecting fake images is leveraging the power of Deep Neural Networks (DNNs) in learning the differences between real and fake, but they are also vulnerable to perturbation attack like adding human-imperceptible noise [Goodfellow *et al.*, 2015].

In forensics-based fake detection, Nataraj *et al*. [Nataraj *et al.*, 2019] compute the image co-occurrence matrics on RGB channels and employ a DNN model to learn the representation of co-occurrence matrics. McCloskey *et al*. [McCloskey and Albright, 2018] observe that the frequency of saturated pixels in GAN-synthesized fake images is limited as the generator's internal values are normalized and the formation of a color image is vastly different from real images which are sensitive to spectral. Different from forensics-based fake detection, Stehouwer *et al*. [Stehouwer *et al.*, 2019] introduce an attention-based layer in convolutional neural networks (CNNs) to improve fake identification performance. Wang *et al*. [Wang *et al.*, 2019] use ResNet-50 to train a binary-classifier for CNN-synthesized images detection. AutoGAN [Zhang *et al.*, 2019] trains a classifier to identify the artifacts inducted in the up-sampling component of GAN.

Other work explore various *ad-hoc* features to investigate artifacts in images for differentiating real and synthesized facial images. For example, mismatched facial landmark points

[3]https://github.com/deepfakes/faceswap

[Yang *et al.*, 2019], fixed size of facial area [Li and Lyu, 2018], and unique fingerprints of GANs [Zhang *et al.*, 2019; Yu *et al.*, 2019], *etc*. These approaches will be invalid in dealing with improved or advanced GANs.

Existing works are sensitive to perturbation attacks, but robustness is important for a fake detector deployed in the wild.

3 Our Method

In this section, we first give our basic insight and present an overview of FakeSpotter in spotting fake faces by monitoring neuron behaviors. Then, a neuron coverage criteria *MNC* is proposed for capturing the layer-by-layer neuron activation behaviors. Finally, FakeSpotter differentiates four types of fake faces with a simple binary-classifier.

3.1 Insight

Neuron coverage techniques are widely adopted for investigating the internal behaviors of DNNs and play an important role in assuring the quality and security of DNNs. It explores activated neurons whose output values are larger than a threshold. The activated neurons serve as another representation of inputs that preserves the learned layer-by-layer representations in DNN models.

Studies shown that activated neurons exhibit strong capabilities in capturing more subtle features of inputs that are important for study the intrinsic of inputs. DeepXplore [Pei *et al.*, 2017] first introduces neuron coverage as metrics for DNN testing to assure their qualities. Some work exploit the critical activated neurons in layers to detect adversarial examples for securing DNNs [Ma *et al.*, 2019; Tao *et al.*, 2018].

Our work motivated by the power of layer-wise activated neurons in capturing the subtle features of inputs which could be used for hunting the differences between real and synthesized facial images. Based on this insight, we propose FakeSpotter by monitoring neuron behaviors in deep FR systems (*e.g.*, VGG-Face) for fake faces detection. Deep FR systems achieve incredible progress in face recognition but vulnerable to recognizing fake faces [Korshunov and Marcel, 2018]. In Figure 2, we present an overview of FakeSpotter using layer-wise neuron behavior as features with a simple binary-classifier to identify real and fake faces.

3.2 Monitoring Neuron Behaviors

Neuron is the basic unit in DNNs and the final layer neuron outputs are employed for prediction. Given an input of trained DNN, the activation function ϕ (*e.g.*, Sigmoid, ReLU) computes the output value of neurons with connected neurons x_i in the previous layers, weights matrix W_i^k, and bias b_j. Activated neurons in each individual layers are determined by whether the output value is higher than a threshold ξ.

In this work, we propose a new neuron coverage criteria, named *MNC*, for determining the threshold ξ as existing approaches [Ma *et al.*, 2018] in calculating threshold ξ are mostly designed for testing DNNs and not applicable for fake detection. Pei *et al.* [Pei *et al.*, 2017] define a global threshold for activating neurons in all layers which is too rough.

In DNNs, each layer plays their own unique role in learning the representations of inputs [Mahendran and Vedaldi, 2015].

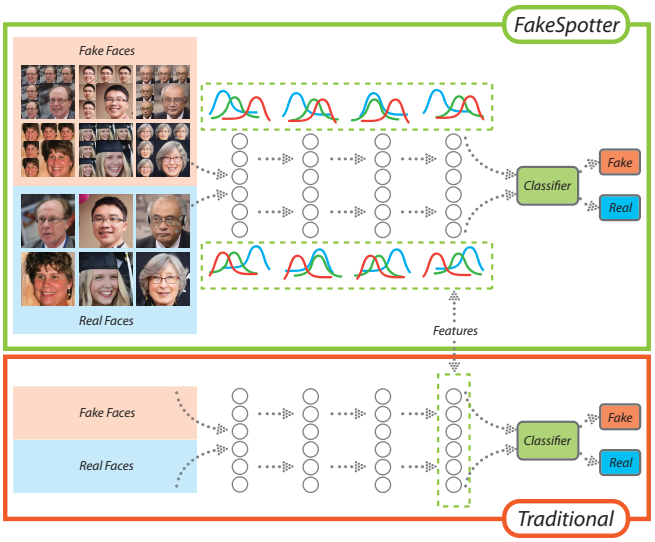


Figure 2: An overview of the proposed fake face detection method, FakeSpotter. Compared to the traditional learning based method (shown at the bottom), the FakeSpotter uses layer-wise neuron behavior as features, as opposed to final-layer neuron output. Our approach uses a shallow neural network as classifier while traditional methods rely on deep neural networks in classification.

Here, we introduce another strategy by specifying a threshold ξ_l for each layer l. The threshold ξ_l is an average value of neuron outputs in each layer for given training inputs. The layer l is the convolutional and full-connected layers which are valuable layers preserving more representation information in the model. Specifically, we calculate the threshold ξ_l for each layer by the following formula:

$$\xi_l = \frac{\sum_{n \in N, t \in \mathcal{T}} \delta(n,t)}{|N| \cdot |\mathcal{T}|} \tag{1}$$

where N represents a set of neurons in the lth layer and $|N|$ is the total number of neurons in the N, $\mathcal{T}$=$\{t_1, t_2, ..., t_k\}$ is a set of training inputs and $|\mathcal{T}|$ indicates the number of training inputs in $\mathcal{T}$, $\delta(n,t)$ calculates the neurons output value where n is the neuron in N and t denotes the input in $\mathcal{T}$. Finally, our neuron coverage criteria *MNC* determines whether a neuron in the lth layer is activated or not by checking whether its output value is higher than the threshold ξ_l. We define the neuron coverage criteria *MNC* for each layer l as follows:

$$MNC(l,t) = |\{n | \forall n \in l, \delta(n,t) > \xi_l\}| \tag{2}$$

where t represents the input, n is the neuron in layer l, δ is a function for computing the neuron output value, ξ_l is the threshold of the lth layer calculated by formula (1).

3.3 Detecting Fake Faces

As described above, we capture the layer-by-layer activated neurons with our proposed *MNC*. Then, we train a simple binary-classifier with shallow neural networks to predict real and fake faces. The input of our classifier is the *general* neuron behavior rather than the *ad-hoc* raw pixels like some traditional image classification models. Raw pixels could be easily perturbed by malicious attackers intentionally and trigger erroneous outputs.

Algorithm 1: Algorithm for detecting fake faces with neuron coverage in deep FR systems.

Input : Training dataset of fake and real faces $\mathcal{T}$, Test dataset of fake and real faces $\mathcal{D}$, Pre-trained deep FR model $\widetilde{M}$

Output: Label tag

L is the convolutional and full-connected layers in $\widetilde{M}$.
▷ Determine the threshold of neuron activation for each layer.
for $t \in \mathcal{T}$ **do**
    N is a set of neurons in the lth layer of $\widetilde{M}$.
    S saves neuron output value for a given input t.
    for $l \in$ L, $n \in N$ **do**
        $\mathrm{S}_l = \sum \delta(n, t)$
    $\xi_l = \frac{1}{|L|} \cdot \mathrm{S}$
▷ Train a binary-classifier for detecting fake/real faces.
V saves activated neurons in L.
for $t \in \mathcal{T}$ **do**
    for $l \in$ L, $n \in N$ **do**
        if $\delta(n, t) > \xi_l$ **then**
            $\mathrm{V}_l \leftarrow n$
Train a binary-classifier $\widetilde{C}$ with inputs V.
▷ Predict whether a face from test dataset $\mathcal{D}$ is real or fake.
for $d \in \mathcal{D}$ **do**
    $tag \leftarrow \mathrm{argmax}\, \widetilde{C}(d)$
return tag

Algorithm 1 describes the procedure of fake face detection. First, the thresholds for determining neuron activation in each layer are identified by our proposed neuron coverage criteria *MNC* with fake and real faces as training dataset, denoted as $\mathcal{T}$. Then, a feature vector for each input face is formed as the number of activated neurons in each layer. Let F=$\{f_1, f_2, ..., f_i, ..., f_m\}$ and R=$\{r_1, r_2, ..., r_j, ..., r_m\}$ represent the feature vector of fake and real input faces respectively, where f_i and r_j are the number of activated neurons in the ith and jth layer, m is the total number of layers in deep FR system. Finally, we train a supervised binary-classifier, denoted as $\widetilde{C}$, by receiving the formed feature vectors of fake and real faces as inputs to predict the input is real or fake.

In prediction, an input face should be processed by a deep FR system to extract the neuron coverage behaviors with our proposed criteria *MNC*, actually the activated neurons in each layer. The activated neurons are formed as a feature to represent the input face. Then, the trained binary-classifier predicts whether the input is a real or fake face.

4 Experiments

In this section, we conduct experiments to evaluate the effectiveness of FakeSpotter in spotting four types of fake faces produced with state-of-the-art techniques and investigate its robustness against four common perturbation attacks. We present the experimental results of detection performance with a comparison of recently published work AutoGAN [Zhang *et al.*, 2019] in Section 4.2 and robustness analysis in Section 4.3. In Section 4.4, we give the comparison results in detecting a publicly DeepFake dataset *Celeb-DF*.

Table 1: Statistics of collected fake faces dataset. Column *Manipulation* indicates the manipulated region in face. Column *Real Source* denotes the source of real face for producing fake faces. Last column *Collection* means the way of producing fake faces, synthesized by ourselves or collected from publicly dataset. *F.F.* ++ denotes FaceForensics++ dataset.

Fake Faces	GAN Type	Manipulation	Real Source	Collection
Entire Synthesis	PGGAN	full	CelebA	self-synthesis
	StyleGAN2	full	FFHQ	officially released
Attribute Editing	StarGAN	brown-hair	CelebA	self-synthesis
	STGAN	eyeglasses	CelebA	self-synthesis
Expression Manipulation	StyleGAN	ctrl. smile intensity	FFHQ	self-synthesis
	STGAN	smile	CelebA	self-syntheis
DeepFake	F. F. ++	face swap	unknown	FaceForensics++
	DFDC	face/voice swap	unknown	Kaggle dataset
	Celeb-DF	face swap	YouTube	Celeb-DF(V2)

4.1 Experimental Setup

• **Data Collection.** In our experiments, real face samples are collected from CelebA [Liu *et al.*, 2015] and Flicker-Faces-HQ (FFHQ)[4] since they own a good diversity. Otherwise, we also utilize original real images provided by publicly dataset FaceForensics++, DFDC, and *Celeb-DF*.

To ensure the diversity and high-quality of our fake face dataset, we use the newest GANs for synthesizing fake faces (like StyleGAN2, STGAN), publicy dataset (*e.g.*, FaceForensics++, *Celeb-DF*), and real dataset (such as DFDC dataset announced by Facebook for competition use). The DFDC dataset is the officially released version rather than the preview edition. Table 1 presents the statistics of our collected fake face dataset.

•**Implementation Details.** We design a shallow neural network with merely five full-connected layers as our binary-classifier for spotting fakes. The optimizer is SGD with momentum 0.9 and the starting learning rate is 0.0001, with a decay of $1e$-6. The loss function is binary cross-entropy.

In monitoring neuron behaviors with *MNC*, we utilize VGG-Face[5] with ResNet50 as backend architecture for capturing activated neurons as it can well balance detection performance and computing overhead. Our approach is generic to FR systems, which could be easily extended to other deep FR systems. In evaluating the robustness in tackling perturbation attacks, we select four common transformations, namely *compression*, *resizing*, *adding noise*, and *blur*.

•**Training and Test Dataset.** In the training dataset $\mathcal{T}$, we train the model with 5,000 real and 5,000 fake fakes for individual GAN. In the test dataset $\mathcal{D}$, we use 1,000 real and 1,000 fake faces for evaluation. The training and test dataset are based on different identities. The training dataset $\mathcal{T}$ and test dataset $\mathcal{D}$ are employed for evaluating the effectiveness and robustness of FakeSpotter. The *Celeb-DF* dataset provides another independent training and test dataset for comparing the performance with existing thirteen methods in detecting fake videos on *Celeb-DF*.

•**Evaluation Metrics.** In spotting real and fake faces, we adopt eight popular metrics to get a comprehensive performance evaluation of FakeSpotter. Specifically, we report pre-

[4]https://github.com/NVlabs/ffhq-dataset

[5]https://github.com/rcmalli/keras-vggface

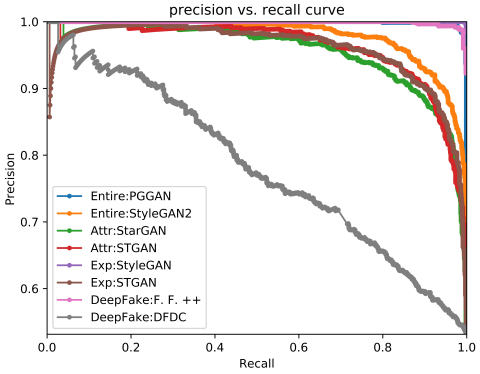


Figure 3: Precision and recall curves of the four types of fake faces. The curve computes precision-recall pairs for different probability thresholds. Higher AP means that FakeSpotter achieves a better balance between precision and recall on this type.

cision, recall, F1-score, accuracy, AP (average precision), AUC (area under curve) of ROC (receiver operating characteristics), FPR (false positive rate), and FNR (false negative rate), respectively. Additionally, we utilize AUC as metrics to evaluate the performance of FakeSpotter in tackling various perturbation attacks.

All our experiments are conducted on a server running Ubuntu 16.04 system on a total 24 cores 2.20GHz Xeon CPU with 260GB RAM and two NVIDIA Tesla P40 GPUs with 24GB memory for each.

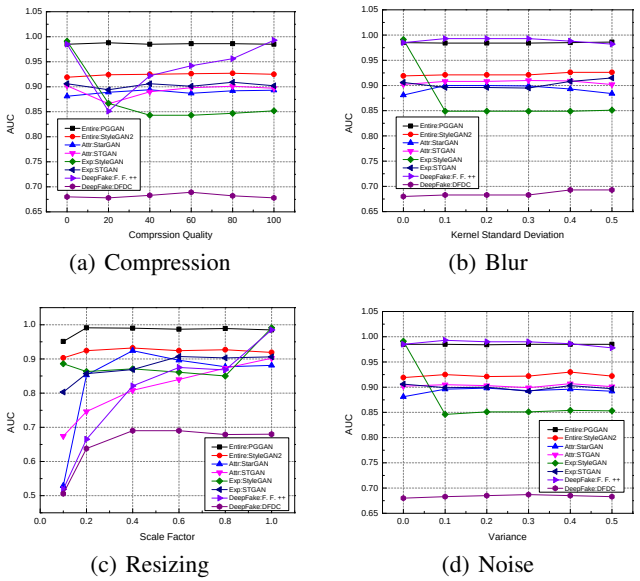
4.2 Detection Performance

In evaluating the performance of FakeSpotter in detecting fake faces and its generalization to different GANs. We select four totally different types of fake faces synthesized with various GANs and compare with prior work AutoGAN. To get a comprehensive performance evaluation, we use eight different metrics to report the detection rate and false alarm rate.

Table 2 presents us the performance of FakeSpotter and prior work AutoGAN in detecting fake faces measured by eight different metrics. AutoGAN is a recent open source work leveraging the artifacts existed in GAN-synthesized images and detect the fake image with a deep neural network based classifier. Furthermore, to illustrate the performance of FakeSpotter in balancing the precision and recall, we present the precision and recall curves in Figure 3 as well.

Experimental results show that FakeSpotter outperforms prior work AutoGAN and achieves attractive performance with high detection rate and low false alarm rate in spotting the four typical fake faces synthesized with state-of-the-art GANs. We also find that FakeSpotter achieves a better balance between precision and recall on four types of fake faces from Figure 3. Further, we also observed some interesting findings from Table 2.

Firstly, fake faces synthesized with advanced GANs are difficult to be spotted by FakeSpotter, for example in entire synthesis, FakeSpotter detects PGGAN with an accuracy of 98.6%, but gives an accuracy of 91.8% on StyleGAN2 (the best performed GAN in entire synthesis and just released by NVIDIA). Secondly, entire face synthesis is easily spotted than partial manipulation fake faces which maybe preserve less fake footprints. These two findings hint that



(a) Compression (b) Blur

(c) Resizing (d) Noise

Figure 4: Four types of perturbation attacks under different intensities.

well-designed GANs and minor manipulations could produce more realistic and hardly spotted fake faces.

In Table 2, the performance of FakeSpotter in detecting DFDC is not as good as other types of fake faces since fake faces in DFDC could be either a face swap or voice swap (or both) claimed by Facebook. In our experiments, some false alarms could be caused by the voice swap which is out the scope of FakeSpotter. A potential idea to detect fakes randomly combining face and voice swap is inferring the characteristic physical features of faces from voice.

4.3 Robustness Analysis

Robustness analysis aims at evaluating the capabilities of FakeSpotter against perturbation attacks since image transformations are common in the wild, especially in creating fake videos. The transformations should be less sensitive to human eyes. In this section, we mainly discuss the performance of FakeSpotter in tackling four different perturbation attacks under various intensities. We utilize AUC as metrics for performance evaluation as it is an important metrics for evaluating the performance of classifier. Figure 4 plots the experimental results of FakeSpotter against the four perturbation attacks.

In Figure 4, the compression quality measures the intensity of compression, the maximum value and minimum value are 100 and 0, respectively. Blur means we employing Gaussina blur to faces. The value of gaussian kernel standard is adjusted to control the intensity of blur while maintaining the gaussian kernel size to $(3, 3)$ unchanged. In resizing, scale factor is used for controlling the size of image in horizontal and vertical axis. We add gaussian-distributed additive noise to produce images with noise and the variance is used for controlling the intensity of noise.

Experimental results demonstrated the robustness of FakeSpotter in tackling the four common perturbation attacks. We find that the AUC score of FakeSpotter maintains a minor fluctuation range when the intensity of perturbation at-

Table 2: Performance of FakeSpotter (*F. S.*) and AutoGAN (*A. G.*) in spotting the four types of fake faces. PGGAN and StyleGAN2 produce entire synthesized facial images. In attribute editing, StarGAN manipulates the color of hair with brown, STGAN manipulates face by wearing an eyeglasses. In Expression manipulation, StyleGAN and STGAN manipulate the expression of faces with smile while StyleGAN can control the intensity of smile. Average performance is an average results over the fake faces. Here, we provide two kinds of average performance, average performance on still images (including the first three types of fake faces) and all the four types of fake faces.

Fake Faces	GAN	precision		recall		F1		accuracy		AP		AUC		FPR		FNR	
		F. S.	A. G.	F. S.	A. G.	F. S.	A. G.	F. S.	A. G.	F. S.	A. G.	F. S.	A. G.	F. S.	A. G.	F. S.	A. G.
Entire Synthesis	PGGAN	0.986	0.926	0.987	0.974	0.986	0.949	0.986	0.948	0.979	0.915	0.985	0.948	0.013	0.026	0.016	0.078
	StyleGAN2	0.912	0.757	0.924	0.663	0.918	0.707	0.919	0.725	0.881	0.670	0.919	0.725	0.076	0.337	0.087	0.213
Attribute Editing	StarGAN	0.901	0.690	0.865	0.567	0.883	0.622	0.88	0.656	0.851	0.608	0.881	0.656	0.135	0.433	0.104	0.255
	STGAN	0.885	0.555	0.918	0.890	0.901	0.683	0.902	0.588	0.852	0.549	0.902	0.588	0.082	0.11	0.114	0.715
Expression Manipulation	StyleGAN	1.0	0.736	0.983	0.920	0.991	0.818	0.991	0.795	0.992	0.717	0.991	0.795	0.017	0.08	0.0	0.33
	STGAN	0.898	0.0	0.913	0.0	0.905	0.0	0.906	0.5	0.863	0.5	0.906	0.5	0.087	1.0	0.102	0.0
DeepFake	FaceForensics++	0.978	0.508	0.992	0.629	0.985	0.562	0.985	0.511	0.973	0.505	0.985	0.511	0.008	0.371	0.021	0.608
	DFDC	0.691	0.536	0.719	1.0	0.705	0.698	0.682	0.536	0.645	0.536	0.680	0.5	0.281	0.0	0.359	1.0
Average Performance (first three types)		0.930	0.611	0.932	0.669	0.931	0.630	0.931	0.702	0.903	0.660	0.931	0.702	0.068	0.331	0.071	0.265
Average Performance (all four types)		0.906	0.589	0.913	0.705	0.909	0.630	0.906	0.657	0.880	0.625	0.906	0.653	0.087	0.295	0.10	0.40

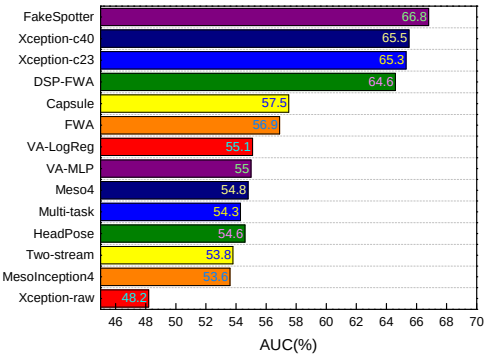


Figure 5: AUC score of various methods on *Cele-DF(V2)* dataset.

tacks increased. The average AUC score of all the four types of fake faces decreased less than 3.77% on the four perturbation attacks under five different intensities.

4.4 Performance on *Celeb-DF(v2)*

Celeb-DF [Yuezun Li and Lyu, 2019] is another large-scale DeepFake video dataset with many different subjects (*e.g.*, ages, ethic groups, gender) and contains more than 5,639 high-quality fake videos. In their project website, they provide some comparison results of existing video detection methods on several DeepFake videos including *Celeb-DF*. There are two versions of *Celeb-DF* dataset, *Celeb-DF(v1)* and *Celeb-DF(v2)* dataset, a superset of *Celeb-DF(v1)*.

We use *Celeb-DF(v2)* dataset for demonstrating the effectiveness of FakeSpotter further and get a more comprehensive comparison with existing work on fake video detection. We also utilize AUC score as metrics for evaluating our approach FakeSpotter as AUC score is served as the metrics in *Celeb-DF* project for comparing with various methods. Figure 5 shown the performance of FakeSpotter in spotting fake videos on *Celeb-DF(v2)*. Experimental results shown that FakeSpotter reaches an AUC score 66.8% on the test dataset provided in *Celeb-DF(v2)* and outperforms all the existing work listed.

According to the experimental results in Figure 5, fake video detection is still a challenge. Especially, some high-quality fake videos utilize various unknown techniques.

4.5 Discussion

Our approach achieves impressive results in detecting various types of fake faces and robust against several common perturbation attacks. However, there are also some limitations of our proposed approach.

The performance of FakeSpotter in spotting DFDC is not ideal as other types of fake faces. One of the main reason is that fake faces in DFDC involves two different domain fake, face swap and voice swap. However, our approach only focus on facial images without any consideration of voice. This also reminds us that producing fake multimedia by incorporating various seen and unseen techniques may be a trend in the future, which poses a big challenge to community and call for effective approaches fighting against these terrifying fakes.

5 Conclusion and Future Research Directions

We propose FakeSpotter, the first neuron coverage based approach for fake faces detection. We perform an extensive evaluation of FakeSpotter in fake detection on four typical fake faces which are synthesized with state-of-the-art GANs (including StyleGAN2, DFDC dataset) to corroborate its high detection rates and low false alarm rates. FakeSpotter also outperforms prior work AutoGAN. Furthermore, our approach also show its robustness in tackling four common perturbation attacks. Our neuron coverage based approach presents a new insight for detecting fakes, which could be extended to other field like fake speech detection.

Everyone will be the victims due to the rapidly development of AI techniques in producing fakes (*e.g.*, fake speech, fake videos). The arms race between producing and fighting fakes is on the endless road, and powerful weapons should be developed for protecting us against AI risks. However, a publicly database with benchmark containing diverse high-quality fake faces produced by state-of-the-art GANs is still lacking in community. Beyond detection, provenance is another issue should be considered in fighting fakes. For provenance, we need to answer the following three questions, namely who produced the fake, how to produce the fake, and where is the tampered region in fake. On the attack side, new metrics are needed for evaluating the quality of GANs in image synthesis.

References

[Agarwal *et al.*, 2019] Shruti Agarwal, Hany Farid, Yuming Gu, Mingming He, Koki Nagano, and Hao Li. Protecting world leaders against deep fakes. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, pages 38–45, 2019.

[Böhme and Kirchner, 2013] Rainer Böhme and Matthias Kirchner. Counter-forensics: Attacking image forensics. In *Digital image forensics*, pages 327–366. Springer, 2013.

[Choi *et al.*, 2018] 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.

[Cole, 2018] Samantha Cole. We Are Truly F—ed: Everyone Is Making AI-Generated Fake Porn Now. https://www.vice.com/en_us/article/bjye8a/reddit-fake-porn-app-daisy-ridley/, 2018. (Jan 25 2018).

[Goodfellow *et al.*, 2014] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. In *Advances in neural information processing systems*, pages 2672–2680, 2014.

[Goodfellow *et al.*, 2015] Ian Goodfellow, Jonathon Shlens, and Christian Szegedy. Explaining and harnessing adversarial examples. In *International Conference on Learning Representations*, 2015.

[Karras *et al.*, 2017] Tero Karras, Timo Aila, Samuli Laine, and Jaakko Lehtinen. Progressive growing of GANs for improved quality, stability, and variation. *arXiv preprint arXiv:1710.10196*, 2017.

[Karras *et al.*, 2019a] Tero Karras, Samuli Laine, and Timo Aila. A style-based generator architecture for generative adversarial networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4401–4410, 2019.

[Karras *et al.*, 2019b] Tero Karras, Samuli Laine, Miika Aittala, Janne Hellsten, Jaakko Lehtinen, and Timo Aila. Analyzing and improving the image quality of stylegan. *arXiv preprint arXiv:1912.04958*, 2019.

[Korshunov and Marcel, 2018] Pavel Korshunov and Sébastien Marcel. Deepfakes: a new threat to face recognition? assessment and detection. *arXiv preprint arXiv:1812.08685*, 2018.

[Li and Lyu, 2018] Yuezun Li and Siwei Lyu. Exposing deepfake videos by detecting face warping artifacts. *arXiv preprint arXiv:1811.00656*, 2, 2018.

[Liu *et al.*, 2015] Ziwei Liu, Ping Luo, Xiaogang Wang, and Xiaoou Tang. Deep learning face attributes in the wild. In *Proceedings of International Conference on Computer Vision (ICCV)*, December 2015.

[Liu *et al.*, 2019] Ming Liu, Yukang Ding, Min Xia, Xiao Liu, Errui Ding, Wangmeng Zuo, and Shilei Wen. STGAN: A unified selective transfer network for arbitrary image attribute editing. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3673–3682, 2019.

[Ma *et al.*, 2018] Lei Ma, Felix Juefei-Xu, Fuyuan Zhang, Jiyuan Sun, Minhui Xue, Bo Li, Chunyang Chen, Ting Su, Li Li, Yang Liu, et al. DeepGauge: Multi-granularity testing criteria for deep learning systems. In *Proceedings of the 33rd ACM/IEEE International Conference on Automated Software Engineering*, pages 120–131. ACM, 2018.

[Ma *et al.*, 2019] Shiqing Ma, Yingqi Liu, Guanhong Tao, Wen-Chuan Lee, and Xiangyu Zhang. NIC: Detecting adversarial samples with neural network invariant checking. In *Proceedings of the Network and Distributed System Security Symposium*, 2019.

[Mahendran and Vedaldi, 2015] Aravindh Mahendran and Andrea Vedaldi. Understanding deep image representations by inverting them. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2015.

[McCloskey and Albright, 2018] Scott McCloskey and Michael Albright. Detecting GAN-generated imagery using color cues. *arXiv preprint arXiv:1812.08247*, 2018.

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

[Pei *et al.*, 2017] 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. ACM, 2017.

[Rössler *et al.*, 2019] 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.

[Stehouwer *et al.*, 2019] Joel Stehouwer, Hao Dang, Feng Liu, Xiaoming Liu, and Anil Jain. On the detection of digital face manipulation. *arXiv preprint arXiv:1910.01717*, 2019.

[Tao *et al.*, 2018] Guanhong Tao, Shiqing Ma, Yingqi Liu, and Xiangyu Zhang. Attacks meet interpretability: Attribute-steered detection of adversarial samples. In *Advances in Neural Information Processing Systems*, pages 7717–7728, 2018.

[Tolosana *et al.*, 2020] Ruben Tolosana, Ruben Vera-Rodriguez, Julian Fierrez, Aythami Morales, and Javier Ortega-Garcia. Deepfakes and beyond: A survey of face manipulation and fake detection. *arXiv preprint arXiv:2001.00179*, 2020.

[Wang *et al.*, 2019] Sheng-Yu Wang, Oliver Wang, Richard Zhang, Andrew Owens, and Alexei A Efros. CNN-generated images are surprisingly easy to spot... for now. *arXiv preprint arXiv:1912.11035*, 2019.

[Yang *et al.*, 2019] Xin Yang, Yuezun Li, and Siwei Lyu. Exposing deep fakes using inconsistent head poses. In *ICASSP 2019-2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 8261–8265. IEEE, 2019.

[Yi *et al.*, 2017] Zili Yi, Hao Zhang, Ping Tan, and Minglun Gong. Dualgan: Unsupervised dual learning for image-to-image translation. In *Proceedings of the IEEE international conference on computer vision*, pages 2849–2857, 2017.

[Yu *et al.*, 2019] Ning Yu, Larry S Davis, and Mario Fritz. Attributing fake images to gans: Learning and analyzing gan fingerprints. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 7556–7566, 2019.

[Yuezun Li and Lyu, 2019] Pu Sun Honggang Qi Yuezun Li, Xin Yang and Siwei Lyu. Celeb-DF: A new dataset for deepfake forensics. *arXiv preprint arXiv:1909.12962*, 2019.

[Zakharov *et al.*, 2019] Egor Zakharov, Aliaksandra Shysheya, Egor Burkov, and Victor Lempitsky. Few-shot adversarial learning of realistic neural talking head models. *arXiv preprint arXiv:1905.08233*, 2019.

[Zhang *et al.*, 2019] Xu Zhang, Svebor Karaman, and Shih-Fu Chang. Detecting and simulating artifacts in gan fake images. *arXiv preprint arXiv:1907.06515*, 2019.

[Zhu *et al.*, 2017] 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.