2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops

On the Robustness and Generalizability of Face Synthesis Detection Methods

Johan Sabel
Swedish Defence Research Agency (FOI)
johan.sabel@foi.se

Fredrik Johansson
Swedish Defence Research Agency (FOI)
fredrik.johansson@foi.se

Abstract

In recent years, significant progress has been made within human face synthesis. It is now possible, and easy for anyone, to generate credible high-resolution images of non-existing people. This calls for effective detection methods. In this paper, three state-of-the-art deep learning-based methods are evaluated with respect to their robustness and generalizability, which are two factors that must be taken into consideration for methods intended to be deployed in the wild. The robustness experiments show that it is possible to achieve near-perfect performance when discriminating between real and synthetic facial images that have been post-processed heavily with various perturbation techniques; especially when similar perturbations are incorporated during training of the detection models. The generalization experiments show that already trained detection models can achieve high performance on images from sources not known during training, provided that the models are fine-tuned on such images. One model achieved an average accuracy of 96.8% after being fine-tuned on 3 training images from each unknown source considered (one real and one synthetic source). However, additional images were required when fine-tuning using a different approach aimed at preventing catastrophic forgetting. Furthermore, in general, no method generalized well without fine-tuning. Hence, the limited generalization capability remains a shortcoming that must be overcome before the detection methods can be utilized in the wild.

1. Introduction

The rapid development of new Generative Adversarial Network (GAN) architectures [9, 10, 11] has pushed state-of-the-art for image synthesis to levels where synthetic images are often perceived as authentic by humans [18, 13]. One well-known example is images of faces generated by StyleGAN2 [11] (see Figure 1). The process of creating these images is commonly referred to as *entire face synthesis* since the GAN generates portraits of people who have never existed in the real world. While image synthesis

Figure 1: Synthetic images generated by StyleGAN2.

brings many benefits, such as the possibility to artificially augment datasets, it also poses a threat when people no longer are able to assess the authenticity of images. GAN-generated images might be used by malicious actors to deceive and take advantage of others; both individuals and organizations. An example is fake profiles on social media platforms, which can be used for fraudulent purposes and facilitate the spread of false information [24]. Hence, it is important to develop effective methods to spot fake images.

Several existing deep learning-based methods do achieve near-perfect results for synthetic face detection when evaluated on images held-out from the datasets used for training [5, 18, 13]. However, a classifier should ideally generalize to out-of-distribution images from unknown sources as well. Otherwise, it will fail to detect images from GANs not encountered during training, or return false positives when fed with real images from arbitrary sources. It should also be robust to image perturbations such as noise, blur, compression, and resizing. Perturbations do occur natu-

978-1-6654-4899-4/21/$31.00 ©2021 IEEE
DOI 10.1109/CVPRW53098.2021.00107

rally, e.g., when editing and uploading images to social media platforms, but could as well be the result of thoughtful attempts to deceive detection systems. In other words, the usefulness of a detection method might be diminished in real-world applications, i.e., in unconstrained *in-the-wild* scenarios, if it lacks sufficient robustness and generalizability. In this paper, the following contributions are made:

- An extensive study of three existing state-of-the-art detection methods based on Convolutional Neural Networks (CNNs) is presented, focusing on their perturbation robustness and generalizability when discriminating between real facial images and synthetic facial images generated by high-resolution GANs, including StyleGAN2.
- The generalizability of detection methods is evaluated in settings where *all* test images have been collected from out-of-distribution datasets; not only the synthetic images. So far, most existing work has primarily focused on generalization experiments where all or part of the real test images originate from a held-out subset of the dataset used for training each detection model.
- The experiments show that it is possible to fine-tune models on a limited number of out-of-distribution images, and significantly increase detection performance on the corresponding target datasets, while still maintaining high performance on held-out images from the datasets originally used for training.

2. Related Work

Several deep learning-based methods have been developed to detect fake images. This section summarizes existing work relevant to this paper, i.e., studies which propose novel detection methods or investigate the robustness or generalizability of such methods. Section 2.1 is dedicated to studies that partly, or entirely, focus on the detection of entire face synthesis. Section 2.2 describes studies that focus on related forgeries such as face swapping, facial attribute manipulation, facial expression manipulation, image-to-image translation not involving faces, and entire image synthesis not involving faces.

2.1. Detection of Entire Face Synthesis

Recent studies suggest that each GAN leaves a unique fingerprint in the images that it generates, and that the fingerprint depends on parameters such as GAN architecture, training set, and random initialization seed [28, 15]. Yu et al. [28] train a fingerprint-based attribution classifier to distinguish between real and synthesized images, and attribute the latter to their GAN sources. The authors suggest a learning-based fingerprint formulation, as opposed to the hand-crafted one suggested by Marra et al. [15]. Yu et al. also show that their classifier becomes more robust to various perturbations, but not perfect, after fine-tuning on perturbed images. For entire face synthesis, they only consider GANs trained on the CelebA dataset [12].

Neves et al. [18] remove the GAN fingerprint from images by passing them through an autoencoder trained to reconstruct real images. Consequently, some detection methods classify synthetic images as real. The authors also show that the detection performance decreases when the test set contains images of lower resolution than the training set, JPEG compressed images, or out-of-distribution images.

Wang et al. [25] monitor layer-wise neuron behavior in deep facial recognition systems with a shallow binary classifier. Specifically, the input to the classifier consists of feature vectors, in which each element corresponds to the number of activated neurons in a specific layer of the facial recognition system. In other words, the classifier is fed with "the general neuron behavior rather than the ad-hoc raw pixels" [25]. This improves robustness since the method does not rely on easily perturbed raw pixels as input. However, the performance is not satisfactory when discriminating between unperturbed StyleGAN2 [11] and FFHQ [10] images, where StyleGAN2 has been trained on the FFHQ dataset.

Cozzolino et al. [4] show impressive results when evaluating their autoencoder-based detection method on out-of-distribution images. The latent space of the autoencoder is split into two parts to disentangle real and fake images. One model achieves high accuracy when tested on StyleGAN [10] and CelebA-HQ [9] images (where StyleGAN has been trained on the CelebA-HQ dataset), after being trained on CycleGAN [30] images (image-to-image translation) and the corresponding real images. The generalization capability is further improved after fine-tuning on just a few images from StyleGAN and CelebA-HQ. However, the transferability *between* GANs used for entire face synthesis is not investigated, nor how fine-tuning affects performance on the original test set containing CycleGAN images.

Marra et al. [16] propose an incremental learning method for detecting fake images from new GAN architectures without reducing performance on already known architectures. The focus is not on detecting samples from completely unknown GAN architectures, but rather on preventing catastrophic forgetting when an already trained classifier is adapted to handle additional architectures from which training samples are available. Hence, the experiments are not conducted in zero- or few-shot settings. Adaptation to new classes, without the need to re-train the classifier on the entire dataset, is enabled by keeping a relatively small number of training samples from the old classes while including additional samples from the new classes. The authors use an Xception network [3] (pre-trained on the ImageNet dataset [6]) as feature extractor and train an incremental classifier based on a version of the iCaRL algorithm [20].

Liu et al. [13] insert Gram blocks into a backbone CNN (ResNet-18 [7] pre-trained on ImageNet), where a Gram-

matrix layer extracts global texture features. This supposedly helps increase detection performance since the authors show that image textures, such as regions containing skin and hair, provide the most discriminative information for detection, as opposed to shape and color artifacts. Similar to most CNNs, Gram-Net outperforms human subjects in detecting images generated by ProGAN [9] and StyleGAN. It is also rather robust against various perturbations and generalizes well to synthetic images from unseen distributions. However, the validation set used for model selection contains 100 images from each GAN and real source considered in the study. Despite this, the detection method does in general not achieve satisfactory performance when evaluated on test sets containing both synthetic *and* real images from unseen distributions. For high-resolution GANs, training on FFHQ and StyleGAN (trained on FFHQ) and testing on CelebA-HQ and StyleGAN (trained on CelebA-HQ) gives the best accuracy, which is still rather low [13].

Hulzebosch et al. [8] compare the Xception network and the method proposed by Cozzolino et al. [4] in more realistic scenarios. In some cases, the latter method does generalize to samples from unknown GAN architectures. However, in those experiments the real test images originate from the same distribution as the real images used for training. The authors do conduct one experiment where both real and synthetic test images are out-of-distribution. Specifically, the methods are trained on CelebA-HQ and StyleGAN (trained on CelebA-HQ) and then tested on FFHQ and StyleGAN (trained on FFHQ), and vice versa. In both cases, no method achieves satisfactory performance. The authors also investigate how performance, measured on out-of-distribution images and held-out images post-processed with JPEG compression or blur, is affected by training on images post-processed using high-pass filters, co-occurrence matrices, or color transformations. In general, none of these post-processing techniques renders good performance in multiple experimental settings, although performance improvement (and impairment) can be observed in some cases. StyleGAN2 is not considered in the experiments, and model adaptation through fine-tuning is not investigated.

2.2. Detection of Other Forgeries

Rössler et al. [21] suggest using transfer learning with an Xception network pre-trained on ImageNet and fine-tuned on domain-specific data, i.e., real and manipulated images of faces extracted from video frames. In this case, fine-tuning is performed using large datasets. The network outperforms human subjects, achieves state-of-the-art performance on raw videos and maintains reasonable performance on compressed videos. Marra et al. [14] conduct similar experiments and show that robustness can be increased by training detection models on compressed images.

Nataraj et al. [17] feed a CNN classifier with pixel co-occurrence matrices computed on the RGB channels of the input. The proposed method exhibits near-perfect performance both when tested on held-out images and out-of-distribution images. However, the transferability is only tested between two architectures [30, 2] used for image-to-image translation. Finally, the authors show that training on JPEG compressed images improves performance when the held-out test images also have been compressed.

Zhang et al. [29] improve the generalization capability of a binary classifier (ResNet-34 [7] pre-trained on ImageNet) by using the image frequency spectrum as input instead of RGB pixels. They also train a GAN simulator, AutoGAN, to synthesize artifacts common for similar GAN architectures. Hence, the classifier does not rely on fake images during training since real images instead can be fed through AutoGAN. The classifier also becomes more robust to JPEG compression and resizing after being trained on images incorporating these perturbations. Although the proposed method shows promising results when tested on architectures similar to AutoGAN, it does not generalize as well to drastically different architectures.

Dang et al. [5] utilize learned attention maps to localize manipulated regions in images of faces, which makes it possible for detection networks to focus on the most discriminative regions. Specifically, an attention-based layer is inserted into a backbone Xception network that has been pre-trained on ImageNet (and is later fine-tuned on domain-specific data). The input features to the attention-based layer are multiplied with an estimated image-specific attention map, and then fed back into the backbone. In other words, the attention-based layer produces refined feature maps which supposedly help increase detection performance. The authors do consider detection of entire face synthesis, but the attention mechanism seems to be most effective for other forgery methods where only parts of images have been manipulated.

Wang et al. [26] train a binary classifier (ResNet-50 [7] pre-trained on ImageNet) on real images from 20 LSUN object categories [27], and synthetic images generated by 20 ProGAN models; each one trained on one of the LSUN categories. In general, aggressive post-processing of the training images improves the generalization capability and also makes the classifier robust when tested on images, including out-of-distribution images, incorporating perturbations similar to those imposed during training. For entire image synthesis not involving faces, the classifier does generalize to StyleGAN and StyleGAN2 with high average precision but slightly lower accuracy. However, these GANs have in most cases been trained on one of the LSUN categories used for training the classifier, where images from the same category are used as real samples at test time. In other words, some of the real image sources are used during both training and testing of the classifier.

3. Detection Methods

In this paper, three deep learning-based detection methods shown to perform well in previous studies were selected for experimental assessment. This section provides a brief description of each method. More details about the network architectures and training procedures can be found in Section A of the Supplemental Material.

3.1. Fingerprint-based Network

The proposed method of Yu et al. [28] is designed to utilize fingerprints, since it has been suggested that each GAN leaves a unique fingerprint in the images that it generates. They train a model to visualize both GAN model fingerprints and image fingerprints as images. The fingerprints are then multiplied pixel-wise to measure their correlation. This makes it possible to attribute an image to a specific GAN model without having access to the model itself, since all fingerprints are learned directly from images.

Although the detection model is able to both visualize fingerprints and classify images, the authors use another CNN-based *attribution network* for classification since it is faster to train. The learned GAN model fingerprints are represented by the weights of the fully-connected output layer (one $1 \times 1 \times 512$ weight tensor for each real and synthetic class), while the image fingerprint is represented by the input features to the output layer (one $1 \times 1 \times 512$ tensor). In this paper, the attribution network (referred to as the *fingerprint-based network*, or simply *Fingerprint*) was selected for experimental assessment.

3.2. Xception Network

The Xception architecture, introduced by Chollet [3], is inspired by the Inception network [22] and based on the idea of mapping cross-channel correlations and spatial correlations independently with multiple filters instead of jointly with a single filter. The Inception modules [23] are replaced with *modified depthwise separable convolutions*, in which a 1×1 pointwise convolution is followed by a depthwise convolution (channel-wise spatial convolution). In this paper, the Xception network was fine-tuned on domain-specific data (large datasets of real and synthetic faces) after originally being trained on the ImageNet dataset [6] of 1,000 classes. This transfer learning approach has proven to be successful in previous studies [21, 14, 18, 5, 16].

3.3. ForensicTransfer Network

The autoencoder-based detection method, ForensicTransfer, proposed by Cozzolino et al. [4] has shown promising results regarding generalizability. Each input sample to the encoder is represented by a six-channel residual image, which is obtained from the original and transposed input image by applying a high-pass filter to each RGB color channel. The encoder outputs a latent representation $\mathbf{h}$ (128 feature maps), which is split into the disjoint parts $\mathbf{h}_0$ and $\mathbf{h}_1$, where each part contains 64 feature maps. Here, each feature map is associated with a class $k \in \{0, 1\}$. In this paper, $\mathbf{h}_0$ would represent the real class ($k = 0$), and $\mathbf{h}_1$ the synthetic class ($k = 1$).

During training, the output from the encoder is fed to a custom *selection* layer, which fills all feature maps in $\mathbf{h}_{1-k}$ with zeros provided that the input sample belongs to class k. Hence, the decoder has to reconstruct the residual only from the information in $\mathbf{h}_k$. Consequently, the encoder is trained to encode samples in that part of the latent space. Put differently, $\mathbf{h}_k$ should be *activated* by the encoder if and only if the input sample belongs to class k. Based on the activations in the feature maps of each part, it is possible to determine whether unseen test samples are closer to the real or synthetic images of the training set.

4. Experiments

A number of experiments were undertaken to study the robustness and generalizability of the methods described in Section 3. In each experiment, the precision, recall, accuracy, F1-score, AUROC, and average precision were computed to quantify the performance. The accuracy and AUROC are presented in the main paper, while the other metrics can be found in Section B of the Supplemental Material.

4.1. Datasets

As shown in Table 1, real images were collected from the CelebA-HQ [9] and FFHQ [10] datasets, while synthetic images were generated by officially released ProGAN [9], StyleGAN [10], and StyleGAN2 [11] models. Specifically, ProGAN had been trained on CelebA-HQ while both StyleGAN and StyleGAN2 had been trained on FFHQ. For StyleGAN and StyleGAN2, the amount of variation is controlled by adjusting the style scale $\psi \in [0, 1]$ [10]. $\psi = 0$ generates the average face while $\psi = 1$ generates images of high variation, many of which tend to look unrealistic since they are not well-represented in the training data. Therefore, $\psi = 0.5$ was used since this is a good trade-off between quality and variation. All CelebA-HQ and FFHQ images have a resolution of 1024×1024, which is also the default resolution when generating faces with ProGAN, StyleGAN,

Dataset	Real	Synthetic	# Training	# Validation	# Test
CelebA-HQ	✓		15,750	5,250	9,000
FFHQ	✓		15,750	5,250	9,000
ProGAN		✓	15,750	5,250	9,000
StyleGAN		✓	15,750	5,250	9,000
StyleGAN2		✓	15,750	5,250	9,000

Table 1: Real and synthetic data, with no overlap between the sets.

and StyleGAN2. Therefore, 1024×1024 was chosen as the default resolution in the experiments.

4.2. Perturbation Robustness

Augmentations in the form of perturbations were applied to the test sets listed in Table 1 to evaluate detection model robustness in unconstrained scenarios where post-processed images occur. Two different types of experiments were conducted: one using random augmentations, thus introducing a variety of low- and high-intensity perturbations, and the other using gradually increasing perturbation intensities.

Random Perturbations. The dataset combinations considered in each random perturbation experiment are shown in Table 2. Here, all collected images were used. Hence, each training, validation, and test set contained 31,500, 10,500, and 18,000 images, respectively. Four augmentation techniques were applied in isolation to all test images (i.e., one augmentation technique per experiment):

- I.i.d. *Gaussian noise* with zero mean was added. The standard deviation was randomly sampled from $U[8, 16]$, and all pixel values were clipped to the $[0, 255]$ range.
- *Gaussian blur* was applied using OpenCV [1]. The kernel size was randomly sampled from $\{7, 9, 11, ..., 19\}$.
- *JPEG compression* was applied using OpenCV. The quality was randomly sampled from $U[10, 75]$, where 0 and 100 are the lowest and highest qualities, respectively.
- *Resizing* was performed using bilinear interpolation in OpenCV. The resolution was randomly sampled from $\{75^2, 76^2, 77^2, ..., 299^2\}$. 299×299 was chosen as the upper bound since it is the highest input resolution of any of the detection networks studied.

Note that the standard deviation, kernel size, quality, and resolution were sampled for *each individual* test image.

In a final experiment, the test images were randomly augmented with combinations of the four techniques mentioned above. Specifically, each augmentation operation was applied with 50% probability in random order. When applying Gaussian blur, the minimum and maximum kernel sizes (7x7 and 19x19, respectively) were scaled by the downsizing factor of the image resolution and then rounded to the nearest odd integer. For Gaussian noise and JPEG compression, a basic linear interpolation scheme was introduced to ensure reasonable perturbation intensities at all resolutions.

Training		Validation		Test	
Real	Synthetic	Real	Synthetic	Real	Synthetic
CelebA-HQ	ProGAN	CelebA-HQ	ProGAN	CelebA-HQ	ProGAN
CelebA-HQ	StyleGAN	CelebA-HQ	StyleGAN	CelebA-HQ	StyleGAN
CelebA-HQ	StyleGAN2	CelebA-HQ	StyleGAN2	CelebA-HQ	StyleGAN2
FFHQ	ProGAN	FFHQ	ProGAN	FFHQ	ProGAN
FFHQ	StyleGAN	FFHQ	StyleGAN	FFHQ	StyleGAN
FFHQ	StyleGAN2	FFHQ	StyleGAN2	FFHQ	StyleGAN2

Table 2: Datasets used when evaluating model robustness.

		Lower Bound		Upper Bound	
Augmentation	Parameter	107^2	1024^2	107^2	1024^2
Gaussian Noise	Standard Deviation	3.3	8	6.5	16
JPEG Compression	Quality	75	75	17	10

Table 3: Bounds set for the noise and compression augmentation operations. Intermediate lower bounds were obtained by interpolating between the lower bounds at 107×107 and 1024×1024, while intermediate upper bounds were obtained by interpolating between the upper bounds at 107×107 and 1024×1024.

Specifically, upper and lower bounds were set at the minimum resolution (in this case 107×107) in addition to the bounds already set at the default resolution of 1024×1024. The bounds at the intermediate resolutions were obtained by interpolating between the corresponding bounds at the minimum and maximum (default) resolutions. Table 3 summarizes this information.

In general, real images incorporate more detail and variety in their background compared to synthetic images. Therefore, all images were cropped by a face detector [19] to prevent the detection models from potentially relying too much on this information. It is also reasonable to work with limited background information since one cannot expect to only find uncropped images in the wild. As shown in Figure 2, a rather conservative crop was used in order to include as much of the head as possible while excluding the majority of the background. It should be kept in mind that image augmentation was performed *before* cropping, which would

(a) Original unperturbed images.

(b) Cropped images.

Figure 2: Random samples from the experimental datasets.

also be the case if the detection models were to be deployed in the wild. Hence, the minimum image resolution was even smaller than 75×75 in practice.

All detection networks were trained separately on unperturbed images (no augmentation) and randomly augmented images (cropping was used in both cases). In the latter case, the training and validation images were augmented with combinations of the four operations as described above.

The results of the experiments are presented in Figure 3. As expected, on average, all networks achieved near-perfect performance on unperturbed images (No Aug.). One can also observe that the fingerprint-based network proved to be robust against all perturbations no matter how it was trained. On average, the accuracy of the Xception network trained on unperturbed images deteriorated significantly for noise, compression, and random combinations of perturbations. However, AUROC always remained rather high. In other words, the Xception models still separated the classes well, but not always at the desired probability threshold. The accuracy was completely recovered after training on randomly augmented images. Finally, the performance of ForensicTransfer trained on unperturbed images was impaired by noise, blur, compression, and random combinations; both with respect to accuracy and AUROC in most cases. However, the performance improved significantly after training on randomly augmented images, although ForensicTransfer could not match the robustness of the other two methods.

The performance on each individual test set of Table 2, and additional evaluation metrics, are presented in Tables B1–B12 in Section B of the Supplemental Material.

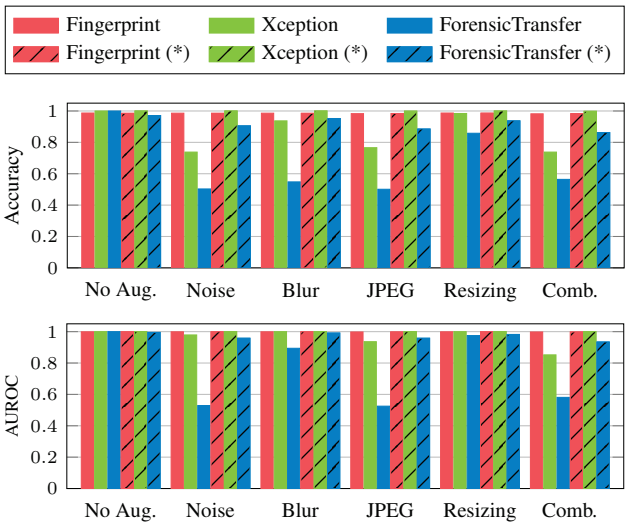


Figure 3: Performance on held-out images augmented in various ways, averaged over all test sets of Table 2. The detection networks shown were trained separately on unperturbed and randomly augmented (*) images. *Comb.* stands for *Random Combinations*.

Strictly Controlled Perturbations. One should keep in mind that the purpose of the previous perturbation experiments was not to push the detection methods to their limits, but rather to evaluate them on data incorporating perturbations more representative of in-the-wild scenarios. Therefore, the previous experiments were complemented with a strictly controlled experiment in which perturbation intensities were gradually increased for the test set. Only images from two datasets were considered, namely FFHQ (real) and StyleGAN2 (synthetic). They were assumed to constitute the most difficult case for a human subject, with respect to distinguishability, since FFHQ offers higher quality than CelebA-HQ while StyleGAN2 yields state-of-the-art results for image synthesis. Furthermore, StyleGAN2 had been trained on FFHQ. Here, $1{,}000$ test images were randomly selected from each dataset. As shown in Figure 4, the fingerprint-based network once again proved to be robust no matter how it was trained; even when moving beyond the upper bounds specified for each augmentation operation in the previous experiments. The Xception network also exhibited high performance when trained on randomly augmented images. ForensicTransfer did not achieve as high performance as the other two methods, although it became more robust after being trained on augmented images.

4.3. Generalizability

The purpose of the generalization experiments was to evaluate detection model performance on out-of-distribution images from unseen datasets instead of held-out images from the datasets used for training. In some of these experiments, models were fine-tuned on a small number of samples from a previously unseen training set in an attempt to further improve performance on the corresponding test set. This would provide insights about the possibility of adapting models when only a few samples are available from an undisclosed GAN which cannot easily be accessed to generate more data. It would also help answer whether models can be continuously fine-tuned to better handle real images collected from arbitrary sources in the wild.

No Fine-tuning. In the first generalization experiment, all models from the experiments in Section 4.2 were reused, including those trained on randomly augmented images. The same goes for the test sets containing $18{,}000$ samples each. However, none of the test images were augmented in this experiment (except for cropping), and the models were not fine-tuned. As shown in Table 4 and Table 5, on average, the detection networks did not generalize well to out-of-distribution images; neither when trained on unperturbed images nor randomly augmented images. The performance was often even worse than random guessing; sometimes with accuracy well below 0.5, which means that a significant number of both real and synthetic images were incorrectly classified (Table B13 and Table B14 in Section B

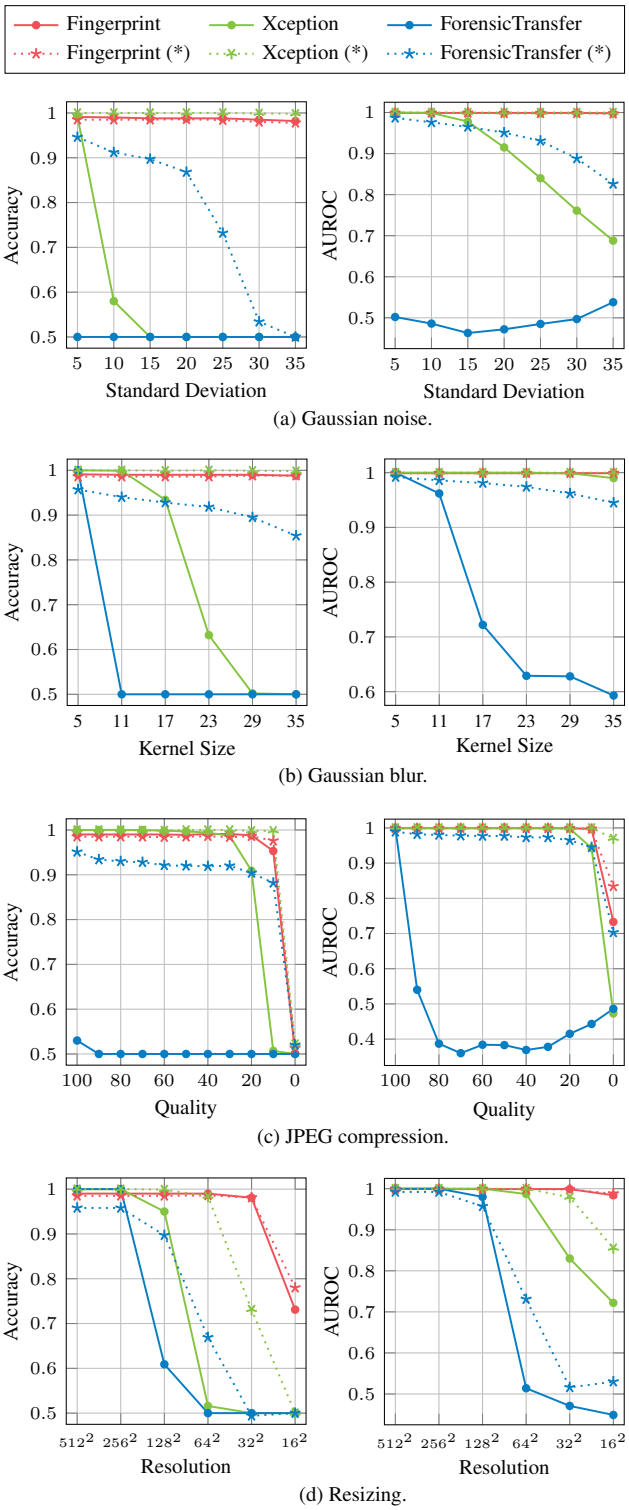


Figure 4: Performance of models trained on FFHQ / StyleGAN2 and tested on held-out images augmented in various ways. The detection networks shown were trained separately on unperturbed and randomly augmented (*) images. Note that the scale of the axis is not uniform for *Resolution*.

Training		Test		Accuracy			AUROC		
Real	Synthetic	Real	Synthetic	F.P.	X.C.	F.T.	F.P.	X.C.	F.T.
CelebA-HQ	ProGAN	FFHQ	StyleGAN	0.491	0.510	**0.573**	0.394	**0.797**	0.666
CelebA-HQ	ProGAN	FFHQ	StyleGAN2	0.488	0.511	**0.573**	0.265	**0.821**	0.818
CelebA-HQ	StyleGAN	FFHQ	ProGAN	**0.417**	0.318	0.134	**0.205**	0.099	0.108
CelebA-HQ	StyleGAN	FFHQ	StyleGAN2	**0.848**	0.623	0.589	0.923	0.672	**0.933**
CelebA-HQ	StyleGAN2	FFHQ	ProGAN	**0.461**	0.224	0.281	0.235	0.053	**0.259**
CelebA-HQ	StyleGAN2	FFHQ	StyleGAN	**0.796**	0.672	0.755	0.892	0.737	**0.927**
FFHQ	ProGAN	CelebA-HQ	StyleGAN	**0.370**	0.244	0.365	**0.080**	0.048	0.004
FFHQ	ProGAN	CelebA-HQ	StyleGAN2	**0.368**	0.244	0.365	**0.043**	0.026	**0.043**
FFHQ	StyleGAN	CelebA-HQ	ProGAN	0.503	**0.515**	0.500	0.539	0.831	**0.842**
FFHQ	StyleGAN	CelebA-HQ	StyleGAN2	0.851	0.499	**0.994**	0.976	0.526	**1.000**
FFHQ	StyleGAN2	CelebA-HQ	ProGAN	**0.507**	0.501	0.500	0.579	0.685	**0.982**
FFHQ	StyleGAN2	CelebA-HQ	StyleGAN	0.705	0.501	**0.736**	0.902	0.412	**0.985**
			Average	**0.567**	0.447	0.530	0.503	0.476	**0.631**

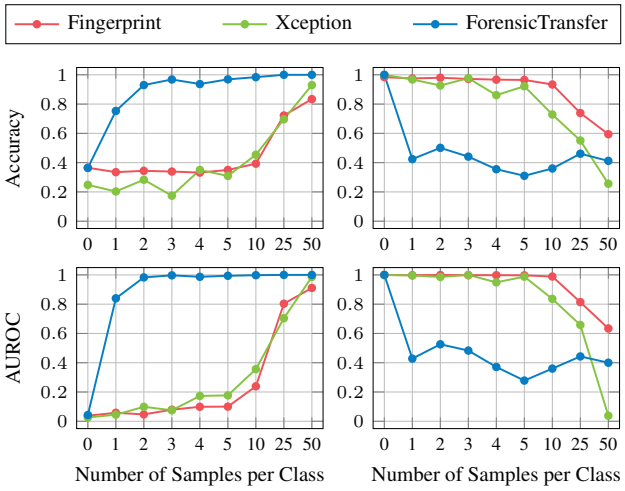
Table 4: Performance of detection models trained on unperturbed images and tested on unperturbed out-of-distribution images. The fingerprint-based network, Xception network, and ForensicTransfer network are abbreviated as F.P., X.C., and F.T., respectively.

Training		Test		Accuracy			AUROC		
Real	Synthetic	Real	Synthetic	F.P.	X.C.	F.T.	F.P.	X.C.	F.T.
CelebA-HQ	ProGAN	FFHQ	StyleGAN	0.486	0.501	**0.604**	0.349	**0.759**	0.609
CelebA-HQ	ProGAN	FFHQ	StyleGAN2	0.484	0.500	**0.617**	0.216	0.236	**0.619**
CelebA-HQ	StyleGAN	FFHQ	ProGAN	0.419	**0.487**	0.141	0.208	**0.283**	0.042
CelebA-HQ	StyleGAN	FFHQ	StyleGAN2	**0.860**	0.577	0.619	**0.935**	0.762	0.821
CelebA-HQ	StyleGAN2	FFHQ	ProGAN	0.447	**0.480**	0.255	**0.256**	0.160	0.087
CelebA-HQ	StyleGAN2	FFHQ	StyleGAN	**0.804**	0.752	0.706	0.884	**0.900**	0.836
FFHQ	ProGAN	CelebA-HQ	StyleGAN	0.328	**0.457**	0.098	**0.082**	0.066	0.013
FFHQ	ProGAN	CelebA-HQ	StyleGAN2	0.325	**0.457**	0.098	**0.055**	0.014	0.019
FFHQ	StyleGAN	CelebA-HQ	ProGAN	0.504	**0.509**	0.484	0.549	**0.742**	0.393
FFHQ	StyleGAN	CelebA-HQ	StyleGAN2	**0.859**	0.522	0.763	**0.971**	0.831	0.886
FFHQ	StyleGAN2	CelebA-HQ	ProGAN	**0.516**	0.501	0.497	0.580	**0.614**	0.481
FFHQ	StyleGAN2	CelebA-HQ	StyleGAN	**0.747**	0.597	0.699	0.897	**0.927**	0.814
			Average	**0.565**	0.528	0.465	0.499	**0.525**	0.468

Table 5: Performance of detection models trained on randomly augmented images and tested on unperturbed out-of-distribution images.

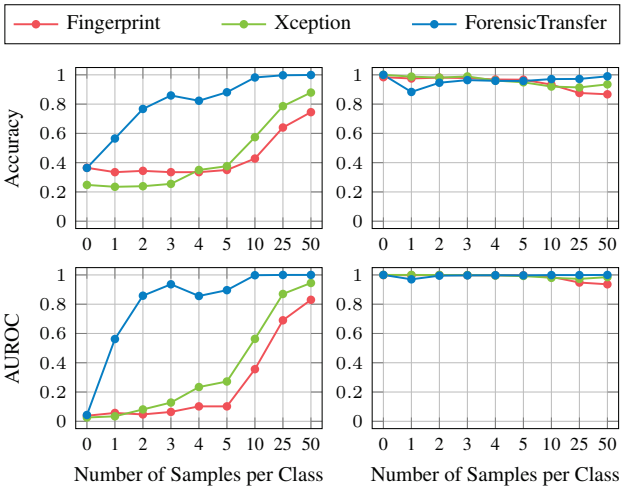
of the Supplemental Material provide additional evaluation metrics for analyzing the performance on real and synthetic images, respectively). However, in some cases the networks did achieve reasonable performance. For instance, this was the case for all models trained on CelebA-HQ / StyleGAN2 and tested on FFHQ / StyleGAN (see Table 4 and Table 5). ForensicTransfer even achieved near-perfect performance when tested on CelebA-HQ / StyleGAN2 after being trained on unperturbed FFHQ / StyleGAN images (see Table 4).

Target Fine-tuning. In the second experiment, models originally trained on FFHQ / ProGAN (source) were fine-tuned on CelebA-HQ / StyleGAN2 (target). As shown in Table 4 and Table 5, all these models exhibited low performance on the target test set before fine-tuning. Hence, the chosen datasets offered room for improvement and constituted a realistic scenario in which models, given only a few samples, had to be adapted to a new, possibly undisclosed, GAN and real images of unknown origin. Since training on randomly augmented images did not seem to improve generalizability in the previous experiment, only the three models (one for each detection network) that had been trained on unperturbed FFHQ and ProGAN images were fine-tuned. As before, the target test set only contained unperturbed images. Therefore, the models were fine-tuned on unperturbed images from the corresponding target training set.



(a) Target. (b) Source.

Figure 5: Source and target performance of models trained on FFHQ / ProGAN (source) and fine-tuned on CelebA-HQ / StyleGAN2 (target), averaged over 10 runs. The metrics are plotted with respect to the number of target training samples per class used for fine-tuning. All detection models were trained and tested on unperturbed images. Note that the scale of the axis is not uniform for *Number of Samples per Class*.



(a) Target. (b) Source.

Figure 6: Source and target performance of models trained on FFHQ / ProGAN (source) and fine-tuned on both CelebA-HQ / StyleGAN2 (target) and FFHQ / ProGAN (source), averaged over 10 runs. The metrics are plotted with respect to the number of target training samples per class used for fine-tuning. All models were trained and tested on unperturbed images. Note that the scale of the axis is not uniform for *Number of Samples per Class*.

The number of training and validation samples used for fine-tuning was gradually increased. For each incrementation, all samples were replaced with new samples randomly selected from the target training and validation sets. The number of validation samples was set to three-fifths of the number of training samples, rounded up to the nearest integer. Finally, the model performance was averaged over 10 runs, using new training and validation samples every time.

As shown in Figure 5, the models were evaluated on both the source (FFHQ / ProGAN) and target (CelebA-HQ / StyleGAN2) test sets. Here, 1,000 test images had been randomly sampled from each dataset (i.e., 4,000 in total) to speed up evaluation. As shown in Figure 5a, the performance on the target test set increased significantly when fine-tuning the models. ForensicTransfer outperformed the other methods with an average accuracy of 75.3% at 1 training sample per class, and 96.8% at 3 samples. At 2 samples, its average AUROC reached 98.3%. However, as shown in Figure 5b, the performance on the source test set deteriorated for all methods as a result of fine-tuning on target data.

Source and Target Fine-tuning. The third experiment aimed at solving the problem of catastrophic forgetting by also including images from the source training set during fine-tuning. In other words, the exact same experiment was repeated, but only after including randomly selected training and validation images from the original datasets in addition to the out-of-distribution images used in the previous experiment. Hence, the models were fine-tuned on an equal number of images from the source and target datasets.

As shown in Figure 6, the performance remained high on the source test set after fine-tuning. However, ForensicTransfer needed more samples than before to achieve near-perfect performance on the target test set.

5. Conclusion

In this paper, three methods were evaluated regarding their ability to detect GAN-generated faces in the wild. All methods proved to be rather robust against common image perturbations, provided that similar perturbations were incorporated during training. One method (ForensicTransfer) also exhibited high transferability when fine-tuned on a small number of out-of-distribution images. However, no method generalized well before fine-tuning. This is problematic since the number of image sources is large in the wild, i.e., it is impractical to *solely* rely on fine-tuning other than in clearly defined use cases where the number of image sources is limited. Catastrophic forgetting also proved to be a problem, and there is no guarantee that the less naive fine-tuning approach will be able to prevent it if many more image sources are to be considered. Hence, this paper confirms what has been indicated in previous studies; namely that current detection methods tend to lack sufficient generalizability. There is a need for further research about new methods and training procedures to enhance performance.

References

[1] Gary Bradski. The opencv library. *Dr. Dobb's Journal of Software Tools (DDJ)*, 2000. 5

[2] 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 (CVPR)*, 2018. 3

[3] François Chollet. Xception: Deep learning with depthwise separable convolutions. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017. 2, 4

[4] Davide Cozzolino, Justus Thies, Andreas Rössler, Christian Riess, Matthias Nießner, and Luisa Verdoliva. Forensictransfer: Weakly-supervised domain adaptation for forgery detection. *arXiv preprint arXiv:1812.02510*, 2018. 2, 3, 4

[5] Hao Dang, Feng Liu, Joel Stehouwer, Xiaoming Liu, and Anil K. Jain. On the detection of digital face manipulation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020. 1, 3, 4

[6] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2009. 2, 4

[7] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016. 2, 3

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

[9] Tero Karras, Timo Aila, Samuli Laine, and Jaakko Lehtinen. Progressive growing of gans for improved quality, stability, and variation. In *Proceedings of the International Conference on Learning Representations (ICLR)*, 2018. 1, 2, 3, 4

[10] 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 (CVPR)*, 2019. 1, 2, 4

[11] 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 Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020. 1, 2, 4

[12] Ziwei Liu, Ping Luo, Xiaogang Wang, and Xiaoou Tang. Deep learning face attributes in the wild. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, 2015. 2

[13] Zhengzhe Liu, Xiaojuan Qi, and Philip H.S. Torr. Global texture enhancement for fake face detection in the wild. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020. 1, 2, 3

[14] Francesco Marra, Diego Gragnaniello, Davide Cozzolino, and Luisa Verdoliva. Detection of gan-generated fake images over social networks. In *Proceedings of the IEEE Conference on Multimedia Information Processing and Retrieval (MIPR)*, 2018. 3, 4

[15] Francesco Marra, Diego Gragnaniello, Luisa Verdoliva, and Giovanni Poggi. Do gans leave artificial fingerprints? In *Proceedings of the IEEE Conference on Multimedia Information Processing and Retrieval (MIPR)*, 2019. 2

[16] Francesco Marra, Cristiano Saltori, Giulia Boato, and Luisa Verdoliva. Incremental learning for the detection and classification of gan-generated images. In *Proceedings of the IEEE International Workshop on Information Forensics and Security (WIFS)*, 2019. 2, 4

[17] Lakshmanan Nataraj, Tajuddin Manhar Mohammed, Shivkumar Chandrasekaran, Arjuna Flenner, Jawadul H. Bappy, Amit K. Roy-Chowdhury, and B. S. Manjunath. Detecting gan generated fake images using co-occurrence matrices. *Journal of Electronic Imaging (JEI)*, 2019(5):532–1–532–7, 2019. 3

[18] João C. Neves, Ruben Tolosana, Ruben Vera-Rodriguez, Vasco Lopes, Hugo Proença, and Julian Fierrez. Ganprintr: Improved fakes and evaluation of the state of the art in face manipulation detection. *IEEE Journal of Selected Topics in Signal Processing (JSTSP)*, 14(5):1038–1048, 2020. 1, 2, 4

[19] Opencv face detector. *GitHub [Online]. Available: https://github.com/opencv/opencv/tree/master/samples/dnn/face_detector*. [Accessed 6 April 2020]. 5

[20] Sylvestre-Alvise Rebuffi, Alexander Kolesnikov, Georg Sperl, and Christoph H. Lampert. icarl: Incremental classifier and representation learning. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017. 2

[21] Andreas Rössler, Davide Cozzolino, Luisa Verdoliva, Christian Riess, Justus Thies, and Matthias Nießner. Faceforensics++: Learning to detect manipulated facial images. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, 2019. 3, 4

[22] Christian Szegedy, Wei Liu, Yangqing Jia, Pierre Sermanet, Scott Reed, Dragomir Anguelov, Dumitru Erhan, Vincent Vanhoucke, and Andrew Rabinovich. Going deeper with convolutions. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2015. 4

[23] Christian Szegedy, Vincent Vanhoucke, Sergey Ioffe, Jonathon Shlens, and Zbigniew Wojna. Rethinking the inception architecture for computer vision. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016. 4

[24] Taylor Hatmaker. Chinese propaganda network on facebook used ai-generated faces. *TechCrunch [Online]. Available: https://techcrunch.com/2020/09/22/facebook-gans-takes-down-networks-of-fake-accounts-originating-in-china-and-the-philippines/*, 2020. 1

[25] Run Wang, Felix Juefei-Xu, Lei Ma, Xiaofei Xie, Yihao Huang, Jian Wang, and Yang Liu. Fakespotter: A simple yet robust baseline for spotting ai-synthesized fake faces. In *Proceedings of the Twenty-Ninth International Joint Conference on Artificial Intelligence (IJCAI)*, 2020. 2

[26] Sheng-Yu Wang, Oliver Wang, Richard Zhang, Andrew Owens, and Alexei A. Efros. Cnn-generated images are

surprisingly easy to spot... for now. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020. 3

[27] Fisher Yu, Ari Seff, Yinda Zhang, Shuran Song, Thomas Funkhouser, and Jianxiong Xiao. Lsun: Construction of a large-scale image dataset using deep learning with humans in the loop. *arXiv preprint arXiv:1506.03365*, 2015. 3

[28] Ning Yu, Larry Davis, and Mario Fritz. Attributing fake images to gans: Learning and analyzing gan fingerprints. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, 2019. 2, 4

[29] Xu Zhang, Svebor Karaman, and Shih-Fu Chang. Detecting and simulating artifacts in gan fake images. In *Proceedings of the IEEE International Workshop on Information Forensics and Security (WIFS)*, 2019. 3

[30] 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 (ICCV)*, 2017. 2, 3