Artifact Regularization based on Fourier Transform for Fine-Tuning of GANs

Amir Hadžić*
ETH Zurich
hadzica@ethz.ch

Jonas Holzem*
ETH Zurich
jholzem@ethz.ch

Maximilian Schaller*
ETH Zurich
mschaller@ethz.ch

Oliver Steffen*
ETH Zurich
steffeol@ethz.ch

***Abstract*—Recent research has shown that GAN-generated images are easy to detect for dedicated detection algorithms. One form of anomalies on which they rely are artifacts in the Fourier representation of synthesized images. As the goal for image synthesis is to produce images which are indistinguishable from real ones, this is a major drawback of GAN-generated images. In this work, we propose a novel training framework which penalizes synthesis artifacts by computing the dissimilarity between synthesized and real images in the Fourier domain. We investigate in which frequency range the Fourier spectra differ the most and show that this dissimilarity can be reduced by using an appropriate training strategy. A key contribution of our work is the reduction of the detection accuracy by more than 30 % only through fine-tuning of the generator, without applying additional post-processing to the output.**

I. Introduction

Generative Adversarial Networks (GANs) [1] are powerful generative models which can be used for various tasks, such as image synthesis and image-to-image translation [2], [9], [10]. A much-noticed state-of-the-art model is StyleGAN, which can, among other things, synthesize photo-realistic images of human faces. Since such capabilities might lead to misuse, a recent research trend is the detection of GAN-generated images [4], [5]. It has been shown that dedicated classifiers can detect GAN-generated images with accuracies above 90% by relying on anomalies in the Fourier representation of artificially synthesized images. These artifacts are presumed to emerge during the convolutional upsampling in the generator of GANs, which leads to unnatural local peaks in the frequency spectrum of the resulting output image [5].

However, sophisticated detection methods go hand in hand with the goal to develop GANs that produce detection-evasive outputs. Such approaches include the usage of autoencoders to remove generation artifacts [7], adding noise to the images [19], [20], and matching the Fourier representation of generated and real images using additional learning or scaling methods [6], [8]. Although these methods made progress in detection-evasion, a limitation is the requirement of additional resources to post-process the output of GANs. In contrast, to the best of our knowledge, there exists no approach which tries to improve the generator of the GAN itself by taking artifacts in the Fourier domain into account.

In this work, we investigate a strategy to modify the generator of a GAN by focusing on anomalies in the Fourier representation of its synthesized images, such that the modified generator becomes more detection-evasive. In particular, we make the following contributions:

- We analyze the frequency-dependent deviations in the Fourier representation of generated and real images. This way, a relevant range in which such deviations occur is estimated.
- An additional Fourier layer is introduced which computes a Fourier loss term ℓ_{F}. This term penalizes the typical generation artifacts. Two different variants for ℓ_{F} are elaborated.
- We develop a training framework which optimizes the generator of a GAN. The framework ensures improved detection evasiveness without additional post-processing of synthesized images.

II. Models and Methods

This section first gives an overview of recent research outcomes in related fields. Second, our framework for artifact regularization is introduced. Third, the experimental design used to evaluate the framework is explained.

A. Related Work

Image Synthesis using GANs. As stated in Section I, present-day's GANs are often employed to synthesize images. This capability is of great use in numerous fields. For instance, in the medical domain, GANs are used for data augmentation in machine learning applications [12] as well as for inference tasks in order to avoid harmful radiation [13]. Nowadays, GANs can also generate images from text input [18]. However, traditional approaches do not incorporate any conditioning on image features. The much-noticed StyleGAN model overcomes this issue by automatically separating high-level attributes and stochastic variation [3]. This way, photo-realistic images can be synthesized and edited by only adjusting the features of interest.

Detection of Synthesized Images. Despite many useful applications, the progress in the field of image synthesis brings not only benefits but also the risk of intentional data manipulation, e.g., to spread disinformation. Therefore, advances in the field of detecting artificially synthesized images were sought [4], [5], [14]–[17]. While some of the approaches focus on the pixel-space and aim to identify cues for artificial synthesis based on the colors of images [14], [15], other investigations

* All authors have contributed equally.

have shown that images can be accurately classified as real or fake by using the frequency spectra of images as input [5]. Wang et al. presented a model which also relies on these artifacts in the frequency domain and which is known to generalize well for many GAN architectures [4]. Therefore, our work uses this model to evaluate the detection evasion capacity of different fine-tuned versions of StyleGAN.

Detection Evasion. A very recent trend is to investigate ways to evade the detection of synthesized images. One approach is the installment of an additional autoencoder which has been trained on real images. When synthesized images are sent through the encoder, it removes anomalies which are typical for GAN-generated images but keeps the visual quality of the resulting image [7]. The addition of noise also introduces difficulties for detection algorithms [19], [20]. Moreover, there exist approaches which specifically focus on the anomalies in the Fourier representations. For instance, shallow reconstruction of generated images through dictionary learning is suggested [8]. Another recently proposed idea is to scale the Fourier representation of a generated image to match it with the representation of a real image, before transforming it back to the pixel-space [6]. In contrast to related work in the field, our work aims at removing artifacts in the Fourier domain exclusively by fine-tuning the generator.

B. Artifact Regularization Framework

We fine-tune StyleGAN by adding a regularization term to the original loss function. This regularization term penalizes differences of real and generated images in the Fourier domain, which are investigated explicitly in Section III. For a meaningful comparison, every generated image should be associated with a real image. To achieve this, we use the recently introduced in-domain GAN inversion technique [11] to obtain latent space encodings which are accurate representations of real images, not only concerning pixel reconstruction but also semantically. These encodings are retrieved by passing real images through a specifically trained encoder network. Subsequently, these latent codes are passed through the StyleGAN generator to generate images which are associated with every real image. Figure 1 shows the overall architecture.

To enable efficient and effective fine-tuning of the generator, we make several assumptions and according design choices. First, we assume that the fine-tuning of the generator causes only small changes in its weights, such that the discriminator weights remain applicable. Therefore, the discriminator is kept fixed during training, and no real images are fed to it.

Second, note that the utilized in-domain GAN inversion technique [11] requires the availability of the generator for optimizing i. a. the pixel reconstruction loss of real and generated images with the latent code as an optimization variable. Again, expecting that the fine-tuning leads to minor changes of the generator weights, we suppose that it is reasonable to pre-optimize the latent code z for the original generator weights. This assumption implies that after training, the difference of both Fourier representations is still dominated by generator artifacts rather than by the semantics of the image. With this

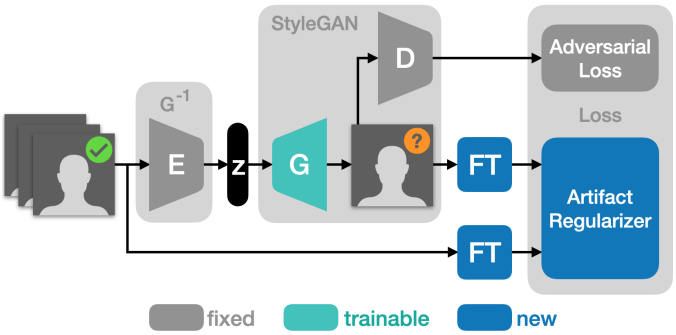


Fig. 1: Architecture of our framework. The encoder E performs the in-domain GAN inversion of StyleGAN's generator G. We use pre-trained models of E, G, and the discriminator D from [11]. The artifact regularizer quantifies the dissimilarity utilizing the Fourier transforms (FTs) of pairs of generated and real images.

assumption, we generate a data set of 9834 pairs of real images from the FFHQ data set [3] down-sampled to a resolution of 256 by 256 pixels. Then, the corresponding latent codes z are optimized for the original generator with the GAN-inversion algorithm. Through this design choice, we speed-up the training process by two orders of magnitude. During training, we pass the latent codes z through the generator to obtain corresponding synthesized images.

For the regularization, we consider two kinds of measures for the comparison of Fourier spectra. To this end, we compute the magnitude spectra F_r and F_g corresponding to real and generated images, respectively, based on the discrete Fourier transform. On the one hand, we utilize the Frobenius norm $\ell_{\mathrm{F,n}}$ of their difference,

$$\ell_{\mathrm{F,n}} = \| \mathcal{T}\left(F_\mathrm{g}\right) - \mathcal{T}\left(F_\mathrm{r}\right) \|_\mathrm{F}, \tag{1}$$

where $\mathcal{T}$ denotes the truncation operation used to cut-off frequencies lower than a certain threshold. Section III provides details on how this threshold is chosen. As an alternative, we consider the cosine dissimilarity $\ell_{\mathrm{F,cos}}$ of the truncated and vectorized spectra,

$$\ell_{\mathrm{F,cos}} = 1 - \frac{g \cdot r}{\|g\| \|r\|}, \tag{2}$$

$$\text{with} \quad g = \mathcal{V}\left(\mathcal{T}\left(F_\mathrm{g}\right)\right), \quad r = \mathcal{V}\left(\mathcal{T}\left(F_\mathrm{r}\right)\right),$$

where $\|.\|$ denotes the euclidean norm, $\cdot$ denotes the inner product and $\mathcal{V}$ declares the vectorization operation. Note that $\ell_{\mathrm{F,cos}} \in [0, 1]$ because g and r are component-wise non-negative as they contain amplitudes of spectra. Finally, we multiply $\ell_{\mathrm{F,n}}$ or $\ell_{\mathrm{F,cos}}$ with a regularization factor λ and add it to the original loss function of StyleGAN. In the following, we use the term *Fourier loss* equivalently to Fourier dissimilarity value.

C. Experimental Design

Dissimilarity Analysis in the Frequency Domain. We analyze the frequency regions in which most of the dissimilarity occurs by comparing a real image with the corresponding image which results from first inverting and then reconstructing the original one. The goal of this analysis is to gain knowledge about which frequency interval should be accounted for in the Fourier regularization loss. Specifically, we calculate the dissimilarity in the Fourier representation of 1'000 pairs of real and synthesized images and identify appropriate truncation thresholds for the dissimilarity measures introduced in Section II-B.

Evaluation of Training Configurations. To evaluate different training configurations, we apply the convolutional fake detection classifier according to [4] to 10'000 images which are synthesized by each fine-tuned generator for every training configuration. The corresponding fake detection accuracy values serve as a validation measure. We compare three overall training configurations in terms of the composition of the loss function. First, we only consider the Fourier regularization term for the loss. Second, we investigate a weighted combination of the Fourier loss and the original adversarial loss. Third, we analyze only back-propagating the adversarial loss. For each of these configurations, we search for appropriate hyper-parameters in terms of regularization measure ℓ_{F}, regularization factor λ, learning rate η, and number of epochs N_{epochs}, using an iteratively refined grid search.

III. Results

First, we show how the dissimilarity behaves in the frequency domain and deduce consequences for the truncation operation introduced in Equation (1). Second, we present the evolution of the loss values during training and the corresponding values of the fake detection accuracy. Third, we show the resulting generated images and their associated Fourier spectra.

A. Analysis of Dissimilarity in Frequency Domain

Figure 2 shows pairs of real and generated images and their corresponding Fourier spectra on the left and right, respectively. In the spectra, the yellow color indicates high magnitudes, while the dark blue color represents little frequency content. For the upper pair, the spectrum of the generated image contains a ray-like shape that points from the DC-gain towards high frequencies, which is less pronounced in the real image's spectrum. While the spectra of the second pair are visually almost indistinguishable, the spectrum of the lower generated image contains granular peaks at high frequency in contrast to the more homogeneous real spectrum. These differences are assumed to be the generation artifacts even though a higher image resolution would probably lead to more distinct peaks.

The Frobenius norm and cosine dissimilarity values of truncated spectra are plotted for different truncation thresholds in Figure 3. The plots show the average dissimilarity values

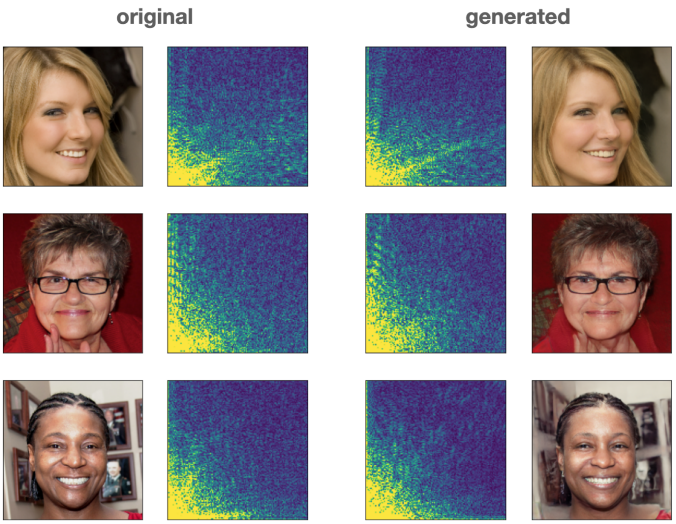


Fig. 2: Real images and their spectra on the left, generated images and their spectra on the right. The frequency equals zero at the bottom left corner, whereas the highest frequencies are visible on the right and top of the spectra.

over 1'000 pairs of real and generated images for every truncation threshold. Note that the Nyquist frequency for the two-dimensional discrete Fourier transform is 256 / 2 = 128 for the investigated images with a resolution of 256 by 256 pixels. For instance, a truncation threshold of 64 means that only the top-right quarter of the spectra visualized in Figure 2 is considered.

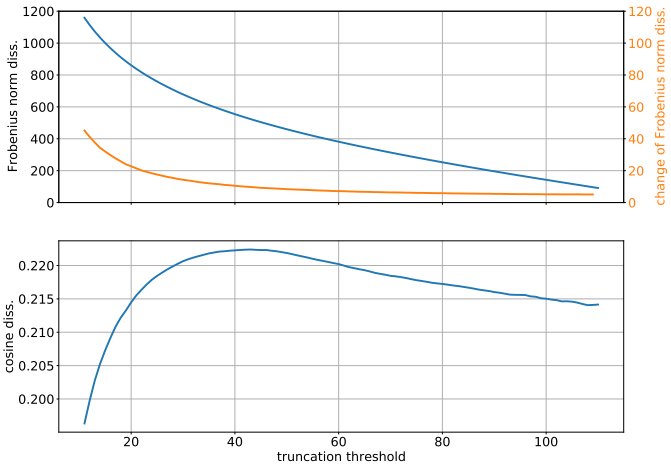
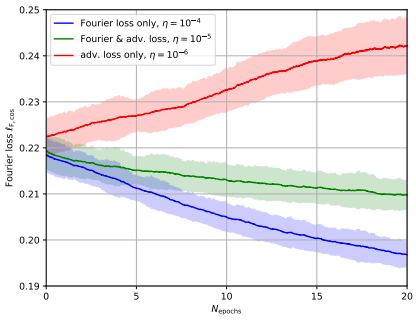
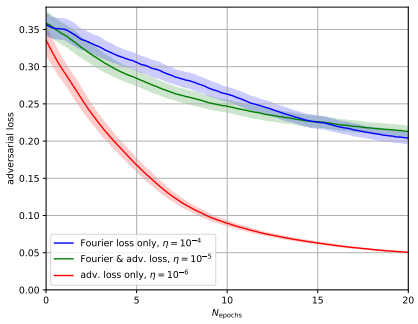


Fig. 3: Fourier dissimilarity values for different truncation thresholds, averaged over 1'000 pairs of real and generated images.

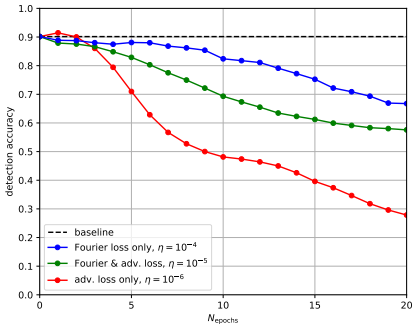
The upper plot of Figure 3 displays the Frobenius norm dissimilarity $\ell_{\mathrm{F,n}}$ with respect to different truncation thresholds, together with the change of the latter from each integer threshold value to the next. Similarly, the lower plot visualizes the cosine dissimilarity $\ell_{\mathrm{F,cos}}$, with a maximum at a threshold value of about 40. For lower threshold values, the generally



(a) Fourier loss $\ell_{\mathrm{F,cos}}$ over the number of training epochs N_{epochs} for different configurations of the loss function. The trajectories are smoothed with a uniform average window. One corresponding standard deviation is represented above and below the mean by the shaded areas.



(b) Adversarial loss over the number of training epochs N_{epochs} for different configurations of the loss function. The trajectories are smoothed with a uniform average window. One corresponding standard deviation is represented above and below the mean by the shaded areas.



(c) Detection accuracy over the number of full training epochs N_{epochs} for different configurations of the loss function.

Fig. 4: Analysis of different training configurations.

large low-frequency components dominate the spectra. As the low-frequency content is similar for real and generated images, the cosine dissimilarity value is low. For truncation frequencies above the maximizer, the frequencies which correspond to a considerable amount of artifacts may be missed. Therefore, we choose a threshold of 40 for our subsequent analyses. We adopt this value also when utilizing the Frobenius norm dissimilarity $\ell_{\mathrm{F,n}}$. For higher threshold values, the change of $\ell_{\mathrm{F,n}}$ settles at a nearly constant value, which is related to the shrinking dimension of the vectorized spectra.

B. Comparison of Training Configurations

After multiple iterations of the grid search described in Section II-C, for fine-tuning with only the Fourier loss, we found $\eta = 10^{-4}$ to be an appropriate learning rate. We multiply the raw Fourier loss $\ell_{\mathrm{F,cos}}$ with a factor $\lambda = 10^3$, only affecting the length of the gradient in this case. When back-propagating a weighted combination of Fourier loss and adversarial loss, we found $\eta = 10^{-5}$ and $\lambda = 10^3$ to be suitable. For solely using the adversarial loss during training, we use a learning rate of $\eta = 10^{-6}$. Due to the different nature of Fourier loss and adversarial loss, it is necessary to choose individual learning rates to compensate for the different scales of the learning dynamics.

Figure 4a shows the evolution of the Fourier loss for the three loss function configurations with the cosine dissimilarity measure $\ell_{\mathrm{F,cos}}$. We found qualitatively similar results for the Frobenius norm dissimilarity $\ell_{\mathrm{F,n}}$. While only optimizing the adversarial loss leads to an increasing value of the Fourier loss, the latter decreases when the Fourier dissimilarity measure is part of the loss function.

Figure 4b contains the training trajectories of the adversarial loss. Although the learning rate is the lowest for back-propagating only the adversarial loss itself, it decreases the fastest, compared to having the Fourier dissimilarity within the loss function. Still, the adversarial loss decreases when the loss function is computed exclusively with the Fourier dissimilarity measure.

The resulting fake detection accuracy values are shown in Figure 4c for every full epoch. The reproduced baseline accuracy value according to [4] is 90.15 %. The accuracy values converge at around 60 % with only the Fourier loss being optimized. Further, we observed that it is possible to reduce the accuracy to zero if the adversarial loss is involved in the loss function.

C. Generated Images and Fourier Spectra after Training

Since all three discussed ways of fine-tuning the StyleGAN generator lead to a decreasing fake-detection accuracy, we compare generated images and their spectra at a fake-detection accuracy level of 70 % for all three variants. Figure 5 shows images associated with the same latent code embeddings as in Figure 2, for training with either only the Fourier loss, a weighted combination of Fourier loss and adversarial loss, or only the adversarial loss being back-propagated.

Amongst the three training configurations, the images generated after training with a mixture of Fourier loss and adversarial loss seem visually closest to the corresponding real images, with a change towards a yellow tone. After fine-tuning only with the Fourier loss, the red components seem to be unnaturally emphasized, while with the adversarial loss exclusively, the generated faces appear to have dark eyes and blurred yellow skin.

The Fourier spectra of the first image turn out to be visually more different from the original real image with fine-tuning than without fine-tuning. The spectra of the second image are visually almost indistinguishable, while the granular high-frequency peaks in the spectrum of the third generated image without fine-tuning are slightly reduced after fine-tuning with the Fourier loss only. The spectra after fine-tuning exclusively with the adversarial loss show the most pronounced differences compared to the spectra of the real images.

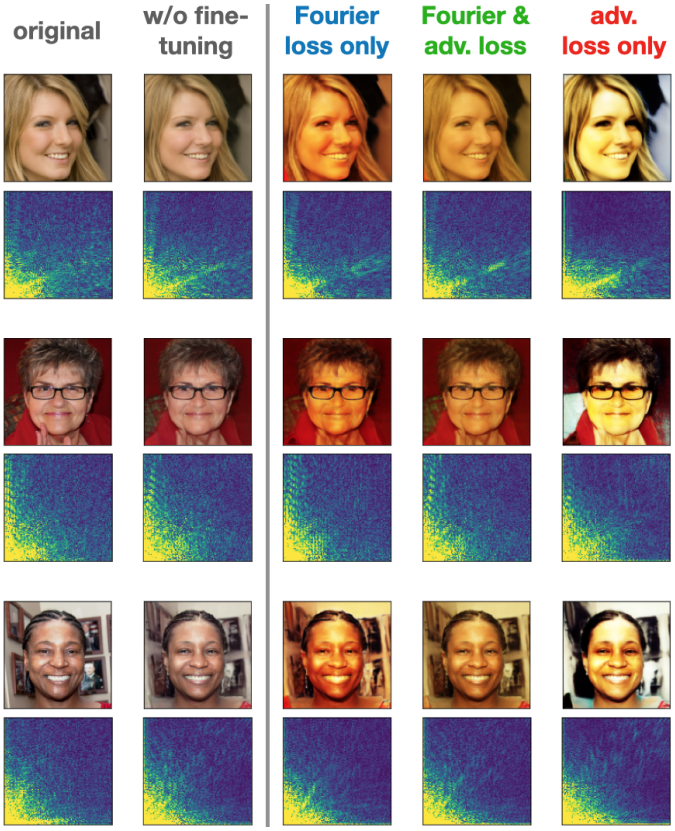


Fig. 5: Real and generated images with their corresponding Fourier spectra before and after fine-tuning. Low frequencies are at the bottom left and high frequencies are at the top right of the spectra. For reference, the first two columns show the real images and their generated counterparts without fine-tuning, respectively. The three columns on the right correspond to fine-tuning by back-propagating either only the Fourier loss, a weighted combination of Fourier loss and adversarial loss, or making gradient steps only with regard to the adversarial loss. They correspond to fine-tuned generators with a detection accuracy of 70 % after 17, 10, and 5 training epochs, respectively.

IV. Discussion

As shown in Section III, the introduced framework for artifact regularization provides a training strategy which makes it harder for detection algorithms to identify synthesized images as such. Although it is possible to improve the detection-evading performance of the StyleGAN version used in [11] without considering Fourier representations but only the adversarial loss, this leads to unrealistic images in terms of color and blurriness.

However, Figure 5 reveals that the additional training conducted by our framework results in more realistic images. Nevertheless, they can still be classified by visual inspection as being artificially generated, especially when compared to the original ones. This is mainly due to the discoloration of the image. On the one hand, one can draw the conclusion that the detection network classifies images as real or fake in a way which is very different from the classification by humans based on visual perception. On the other hand, our approach might not target the artifacts as precisely as desired. We suggest that, in the future, our experiments are conducted using higher resolution images, as the frequency resolution of the Fourier representation of images is proportional to their pixel resolution. Hence, the usage of a resolution higher than 256 by 256 pixels may lead to a frequency resolution which is high enough to isolate the artifacts.

In order to test the suitability of our approach, some simplifications and assumptions were made. For instance, we computed triplets of real images, their latent codes using the in-domain inversion encoder, and the corresponding reconstructed images. This computation was done offline to improve computational efficiency during the training. Nonetheless, it would be interesting to execute the training using online optimization as done in [11]. Further, since the goal was to fine-tune only the generator, fixed weights were used for the discriminator. We propose to expand our framework such that not only the generator is adjusted but the entire GAN. Both mentioned alternatives might result in generators which produce visually more appealing images after many epochs of training.

Besides, we initially considered replacing the final layers of the generator to relax the connectivity constraints of convolutional layers, i.e. allowing for more complex mappings in the last few layers of the generator. This convolutional upsampling is known to cause the emergence of artifacts during the synthesis of images [5]. Yet, since this is not the only source of artifacts, we primarily focused on penalizing the artifacts in general instead of tackling a specific error source individually. Since the penalization of artifacts led to the desired result, this validates our approach which was the scope of this project. We leave the idea to relax the connectivity constraints for future investigations.

V. Summary

We established an extensive training framework to reduce generation artifacts in synthesized images which are typical for GANs and easily detectable for sophisticated detection algorithms. In particular, we introduced a layer which quantifies the Fourier dissimilarity between a real input image and its counterpart which was inverted and generated from the latent code beforehand. This dissimilarity is used as a loss term in addition to the adversarial loss in order to target the artifacts in the Fourier representation of the output. Related research focuses on removing the artifacts in the GAN's output after the generation, rather than adjusting the generator itself. Our approach manifests that it is indeed possible to take the generator of a GAN and fine-tune it without changing the architecture to reduce the chance of being detected by a dedicated classifier.

REFERENCES

[1] I. Goodfellow, J. Pouget-Abadie, M. Mirza,B. Xu,D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio, "Generative adversarial nets," in *Advances in neural information processing systems*, vol. 27, pp. 2672–2680, 2014.

[2] H. Alqahtani, M. Kavakli-Thorne, and G. Kumar, "Applications of generative adversarial networks (GANs): An updated review," *Arch. Comput. Methods Eng.*, 2019.

[3] T. Karras, S. Laine, and T. Aila, "A style-based generator architecture for generative adversarial networks," in *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019.

[4] S.-Y. Wang, O. Wang, R. Zhang, A. Owens, and A. 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.

[5] X. Zhang, S. Karaman, and S.-F. Chang, "Detecting and simulating artifacts in GAN fake images," in *2019 IEEE International Workshop on Information Forensics and Security (WIFS)*, 2019.

[6] T. Dzanic, K. Shah, and F. Witherden, "Fourier spectrum discrepancies in deep network generated images," arXiv [eess.IV], 2019.

[7] J. C. Neves, R. Tolosana, R. Vera-Rodriguez, V. Lopes, H. Proenca, and J. Fierrez, "GANprintR: Improved fakes and evaluation of the state of the art in face manipulation detection," *IEEE J. Sel. Top. Signal Process.*, vol. 14, no. 5, pp. 1038–1048, 2020.

[8] Y. Huang et al., "FakePolisher: Making DeepFakes more detection-evasive by shallow reconstruction," in *Proceedings of the 28th ACM International Conference on Multimedia*, 2020.

[9] Y. Choi, M. Choi, M. Kim, J.-W. Ha, S. Kim, and J. Choo, "StarGAN: Unified generative adversarial networks for multi-domain image-to-image translation," in *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2018.

[10] J.-Y. Zhu, T. Park, P. Isola, and A. A. Efros, "Unpaired image-to-image translation using cycle-consistent adversarial networks," in *2017 IEEE International Conference on Computer Vision (ICCV)*, 2017.

[11] J. Zhu, Y. Shen, D. Zhao, and B. Zhou, "In-domain GAN inversion for real image editing," in *Computer Vision – ECCV 2020*, Cham: Springer International Publishing, 2020, pp. 592–608.

[12] C. Han et al., "GAN-based synthetic brain MR image generation," in *2018 IEEE 15th International Symposium on Biomedical Imaging (ISBI 2018)*, 2018.

[13] D. Nie et al., "Medical image synthesis with context-aware generative adversarial networks," *Med. Image Comput. Comput. Assist. Interv.*, vol. 10435, pp. 417–425, 2017.

[14] S. McCloskey and M. Albright, "Detecting GAN-generated Imagery using Color Cues," arXiv [cs.CV], 2018.

[15] L. Nataraj et al., "Detecting GAN generated Fake Images using Co-occurrence Matrices," *IS&T Int. Symp. Electron. Imaging*, vol. 2019, no. 5, pp. 532-1-532–7, 2019.

[16] F. Marra, C. Saltori, G. Boato, and L. Verdoliva, "Incremental learning for the detection and classification of GAN-generated images," in *2019 IEEE International Workshop on Information Forensics and Security (WIFS)*, 2019.

[17] F. Marra, D. Gragnaniello, D. Cozzolino, and L. Verdoliva, "Detection of GAN-generated fake images over social networks," in *2018 IEEE Conference on Multimedia Information Processing and Retrieval (MIPR)*, 2018.

[18] H. Zhang, T. Xu, and H. Li, "StackGAN: Text to photo-realistic image synthesis with stacked generative adversarial networks," in *2017 IEEE International Conference on Computer Vision (ICCV)*, 2017.

[19] Y. Huang et al., "FakeRetouch: Evading DeepFakes detection via the guidance of deliberate noise," arXiv [cs.CV], 2020.

[20] N. Carlini and H. Farid, "Evading deepfake-image detectors with white- and black-box attacks," in *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, 2020.