

DETECTING GAN-GENERATED IMAGES BY ORTHOGONAL TRAINING OF MULTIPLE CNNs

Sara Mandelli, Nicolò Bonettini, Paolo Bestagini, Stefano Tubaro

Dipartimento di Elettronica, Informazione e Bioingegneria - Politecnico di Milano - Milan, Italy

ABSTRACT

In the last few years, we have witnessed the rise of a series of deep learning methods to generate synthetic images that look extremely realistic. These techniques prove useful in the movie industry and for artistic purposes. However, they also prove dangerous if used to spread fake news or to generate fake online accounts. For this reason, detecting if an image is an actual photograph or has been synthetically generated is becoming an urgent necessity. This paper proposes a detector of synthetic images based on an ensemble of Convolutional Neural Networks (CNNs). We consider the problem of detecting images generated with techniques not available at training time. This is a common scenario, given that new image generators are published more and more frequently. To solve this issue, we leverage two main ideas: (i) CNNs should provide “orthogonal” results to better contribute to the ensemble; (ii) original images are better defined than synthetic ones, thus they should be better trusted at testing time. Experiments show that pursuing these two ideas improves the detector accuracy on NVIDIA’s newly generated StyleGAN3 images, never used in training.

Index Terms— Image forensics, synthetic images, GAN, CNN

1. INTRODUCTION

Over the last few years, we assisted in an escalation of methods for the production of increasingly more realistic synthetically generated images [1–4]. The first architectures produced blurry and low-resolution images with a general lack of details. Recently, giant steps have been made to raise the bar and overcome those issues. This is evident by the recent release of a new Generative Adversarial Network (GAN) architecture by NVIDIA, namely StyleGAN3 [5], which produces high-quality images that can easily fool human eyes.

On the one hand, the authors of image generators put much effort into generating very realistic pictures. On the other hand, they are aware of the variety of problems an overly realistic architecture can create. Generated images can be used over social media for many malicious intents, from scams to identity stealing, and the general public is not ready to face this menace. A recent work [6] shows how is it difficult for humans to tell real and generated faces apart when they have just a few seconds to make the decision. The interviewed people rated StyleGAN2 [2] images as real in the 68% of

the cases, whereas real images were rated as real only in the 52% of the cases. Similarly, the study conducted in [7] shows how synthetic faces prove even more trustworthy at human inspection. Notice that these studies do not even consider the more recent StyleGAN3 yet.

Given these premises, it is evident that being able to detect if an image is a natural photograph or it has been synthetically generated is becoming a task of utmost importance. For this reason, the multimedia forensics community is developing a series of techniques to solve the synthetically generated image detection problem. Some methods are based on hand-crafted features fed to classifiers [8–11]. A wide variety of solutions prefer a purely data-driven approach based on training an end-to-end detector [12, 13]. However, many of these techniques tend to suffer in classifying images that deviate from the characteristics of their training set. Unfortunately, it is nowadays impractical to assume that the characteristic of any synthetically generated image can be perfectly known at training stage, as new image generation techniques are developed continuously. For this reason, the latest research trend is to develop methods that can generalize well, detecting images generated with unseen techniques [11, 14].

In this paper, we tackle the problem of GAN-generated image detection. This is, given an image under analysis, to detect if it is a real photograph or it has been synthetically generated by a GAN. We consider the realistic and challenging scenario in which test images may come from generators that were unknown to the analyst at training time. To solve the GAN-generated image detection problem, we propose an ensemble of Convolutional Neural Networks (CNNs).

The proposed method leverages two main ideas to increase the robustness of unseen generated images. First, CNNs contributing to the ensemble should be as much “orthogonal” as possible. We propose a training strategy that increases the diversity among the different learners for this purpose. This prevents the CNNs from overfitting the image generators used in training, thus enabling the ensemble to take a better decision on newly generated images. Second, the detection problem is better defined over real images than synthetic ones. Indeed, it is safer to assume that the analyst can train on a broad set of real photographs that better represent the real-image class. On the contrary, it is hard to assume that the analyst can train on synthetic images generated with all the possible existing techniques, as these change and get updated too frequently in time. Therefore, we propose a score aggregation strategy that better favour decisions towards the real-image class.

Our experimental campaign is designed to test the proposed training and aggregation strategies on top of a baseline CNN detector based on EfficientNet [15]. We show that our technique is able to: (i) better draw the separation line between real and synthetic images by separating the score distributions of the two classes; (ii) accurately detect StyleGAN3 images as GAN-generated, even though they have never been used in training.

This material is based on research sponsored by the Defense Advanced Research Projects Agency (DARPA) and the Air Force Research Laboratory (AFRL) under agreement number FA8750-20-2-1004. The U.S. Government is authorized to reproduce and distribute reprints for Governmental purposes notwithstanding any copyright notation thereon. The views and conclusions contained herein are those of the authors and should not be interpreted as necessarily representing the official policies or endorsements, either expressed or implied, of DARPA and AFRL or the U.S. Government. This work was supported by the PREMIER project, funded by the Italian Ministry of Education, University, and Research within the PRIN 2017 program.

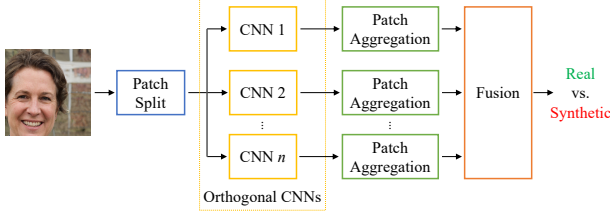


Fig. 1: Proposed method for GAN image detection. Given an image, we split it into patches which are fed to multiple orthogonal CNNs. We aggregate the patch scores to obtain a single image score per CNN. Eventually, we fuse the scores into the final image score to classify the image as real or synthetic.

2. PROPOSED METHOD

We can summarize the main objectives of the GAN-generated image detection problem into three primary tasks: (i) generalize very well to new GANs unseen during training phase; (ii) be robust against post-processing operations applied on images; (iii) achieve a missed detection rate (i.e., the number of synthetic images detected as real) as low as possible. To improve GAN generalization and robustness to editing operations, we propose a procedure based on orthogonal training of multiple CNNs, all based on the same backbone architecture but each trained on a different training dataset. At the testing stage, we propose a patch selection and aggregation strategy that considerably reduces the missed detection rate.

Fig. 1 reports the sketch of the proposed testing pipeline. In a nutshell, given a query image, we classify it as being real or synthetically generated by selecting several patches from it and passing them through multiple orthogonal CNNs. Then, we aggregate the patch scores and fuse the CNNs results into a single prediction associated with the entire image. We provide more details about the proposed approach in the following lines.

2.1. Orthogonal Training

To improve the generalization of the proposed method against new GANs, we train multiple CNNs over different training datasets, which are “orthogonal” one another (with a slight abuse of terminology). For clarity’s sake, we consider two datasets “orthogonal” if one of the following conditions is met:

- the datasets include images depicting different semantic content (e.g., cats or humans);
- the datasets include images that underwent different post-processing (e.g., uncompressed or compressed images);
- the datasets include images that underwent different compressions (e.g., different JPEG implementations);
- the datasets include images synthesized by different GANs.

The key idea of the proposed training orthogonalization is that every single CNN should capture slightly different traces with respect to the others. The ensemble of many CNNs trained in an orthogonal fashion proves to achieve improvements with respect to training a unique CNN over the whole entirety of data at disposal.

The common backbone used for all the proposed CNNs is the EfficientNet-B4 model [15], well known in the computer vision and multimedia forensics communities due to the outstanding results achieved in many tasks although requiring few network parameters [10]. Each CNN works at patch level, always considering squared RGB patches of $N \times N$ pixels as input and providing a single score per patch. Considering an ensemble of C CNNs analyzing

P patches per image, we define as \hat{y}_p^c the score estimated by the c -th CNN for the p -th patch, with $c \in [1, C]$ and $p \in [1, P]$.

To improve the robustness against various post-processing operations, we apply strong data augmentations as suggested in many state-of-the-art works for synthetic image detection [13, 14]. The list of possible augmentations emulates common editing operations that can be applied by amateur users when retouching their photographs. Moreover, malicious users could also apply the same operations to hide the traces left by the synthetic generation process. We consider horizontal and vertical flip, random 90-degree rotation, histogram equalization, random blur, random changes in brightness, contrast, color and saturation, random downscale and upscale, and finally JPEG Compression with quality factors randomly selected from 30 to 100. Each augmentation is applied with probability 50%, except for JPEG compression, which is applied with probability 70%. The parameters are those defined in [16].

At testing stage, for each CNN, we obtain different scores associated with the patches extracted from the query image. Real and synthetic patches are associated with negative and positive scores, respectively. We fuse these scores by following the aggregation strategy presented in the next section to obtain the final image score.

2.2. Patch Aggregation Strategy

Given a test image, every orthogonal CNN returns many scores associated with the patches extracted from the image. When fusing the patch scores, we aim at reducing the detection errors on the synthetic images. Thus, the missed detection rate is the most critical parameter to maintain as low as possible.

The proposed approach is based on the consideration that, when training a generic GAN detector, it is reasonable to assume that the characteristics of real images are easier to be captured than those of synthetic ones. Indeed, everybody could collect a set of original photographs and assign them the label “real”, whereas collecting a sufficiently vast and various synthetic dataset might be more elaborate and certainly requires a little expertise. Moreover, contrarily to the “real” class, the “synthetic” one is constantly and rapidly evolving, as new proposed methodologies for generating synthetic content emerge every day, not limited to GANs only [4, 17]. Given these premises, we can reasonably assume that many CNN detectors trained over orthogonal datasets might correctly classify a real query image with a high precision level, as long as they are accurately trained. We cannot make the same assumption for synthetic query images, as they might be generated from novel unseen GANs.

Here comes the proposed patch aggregation strategy. When a query image passes from a CNN detector and *all* its extracted patches are classified as real, the CNN classifies the entire image as real. If *at least one* patch among those extracted from the test image is detected as synthetic, the CNN assigns the entire image to the synthetic-image class. In particular, the CNN score associated with the image is the best score achieved among all the patches for what concerns the detected class. Since real and synthetic images are associated with negative and positive scores, respectively, the image is assigned the minimum score among the patches if the detected class is real-image. Otherwise, we assign the maximum score. Formally, the image score by the c -th CNN is defined as

$$\hat{y}^c = \begin{cases} \min_p \hat{y}_p^c & \text{if } \hat{y}_p^c < 0, \forall p \in [1, P] \\ \max_p \hat{y}_p^c & \text{otherwise} \end{cases}. \quad (1)$$

Eventually, we equally weight the orthogonal CNNs to assign the global image score, which is the arithmetic mean among all the image scores returned by the networks, i.e., $\hat{y} = \frac{1}{C} \sum_{c=1}^C \hat{y}^c$.

3. EXPERIMENTAL ANALYSIS

3.1. Dataset

We perform our investigations over the dataset used for the competition recently organized by NVIDIA on StyleGAN3 Synthetic Image Detection [18] within the DARPA’s SemaFor program. The purpose was to simulate an open-world setting in which new unseen GANs (e.g., StyleGAN3 [5]) should be detected.

The real class of the testing data consisted of images selected from three public datasets: the FFHQ [19] (depicting human faces taken from photographs), the Metfaces [20] (depicting human faces taken from works of art), and the AFHQ2 [21] (including photographs of animal faces from three domains of cat, dog, and wildlife). The synthetic images to be tested were all generated through the recently released StyleGAN3 network [5], trained on real images selected from the three previously reported datasets. Every real dataset corresponds to two possible synthetic versions of it, the version r and the version t , according to the specific StyleGAN3 configuration chosen at generation stage. The images from Metfaces and AFHQ2 datasets did not undergo post-processing or compression, while a few synthetic images from the FFHQ dataset underwent compression and resizing.

The competition did not pose any limit on the kind of training data to be used for developing the proposed GAN detector, except for removing from the training data the real images belonging to the testing dataset and every synthetic image generated through StyleGAN3. Given these premises, our testing dataset coincides with the testing dataset of the NVIDIA competition. The training dataset consists of different datasets, purposely built so to implement the orthogonal CNN training described in Section 2.1. Every orthogonal dataset is exploited for training an EfficientNet-B4, working with squared RGB patches of size 128×128 . Following our previous considerations, we build 5 “orthogonal” datasets:

Dataset \mathcal{D}_1 . This dataset includes all the real images from FFHQ, Metfaces and AFHQ2 available for training ($\sim 116K$). The synthetic images ($\sim 200K$) are selected from the synthetic versions of the three datasets, generated through state-of-the-art models for synthetic image generation (i.e., StyleGAN2 [2], StarGAN-v2 [21], Taming Transformers [22], FaceVid2Vid [23] and Score-based models [4]). During training, the images undergo multiple augmentations from the list reported in Section 2.1, JPEG compression included. Then, 1 patch per image is randomly selected and fed to the CNN.

Dataset \mathcal{D}_2 . This dataset includes the same real and synthetic images exploited for \mathcal{D}_1 , with the difference that here we first randomly extract 1 patch per image, then we apply the same augmentations defined for \mathcal{D}_1 . From the point of view of the post-processing applied, \mathcal{D}_1 is orthogonal to \mathcal{D}_2 , especially for the JPEG compression. Indeed, by construction, all the patches from \mathcal{D}_2 are aligned to the 8×8 pixel grid introduced by JPEG compression, while in \mathcal{D}_1 the patches can have any random alignment. As already shown in [24], taking care of the JPEG grid alignment is of paramount importance for multimedia forensics tasks. The datasets \mathcal{D}_1 and \mathcal{D}_2 allow exploring this issue for the GAN detection problem.

Dataset \mathcal{D}_3 . This dataset includes only the real images from AFHQ2 available for training ($\sim 14K$) and an equal number of their synthetic versions generated through StyleGAN2 and StarGAN-v2. 10 random patches are extracted per image, then undergo all the augmentations, except for JPEG compression. \mathcal{D}_3 focuses only on one semantic category (i.e., the animal faces), on a few GANs and is entirely orthogonal to \mathcal{D}_1 and \mathcal{D}_2 for what regards JPEG compression.

Dataset \mathcal{D}_4 . Ideally, this dataset would include only the real images

from Metfaces available for training ($\sim 2K$) and an equal number of their synthetic versions generated through StyleGAN2. This would guarantee complete semantic orthogonality with respect to \mathcal{D}_3 . Actually, the training process was unstable due to the very limited number of Metfaces images. To augment the dataset dimensions, we decided to include in \mathcal{D}_4 also the AFHQ2-related images. We extract 10 random patches per image and apply augmentations, except for JPEG compression. As well as \mathcal{D}_3 , \mathcal{D}_4 is entirely orthogonal to \mathcal{D}_1 and \mathcal{D}_2 for what concerns JPEG compression.

Dataset \mathcal{D}_5 . This dataset includes only real images from FFHQ available for training ($\sim 100K$) and almost $170K$ synthetic versions of them generated through StyleGAN2, Taming Transformers, FaceVid2Vid and Score-based models. We randomly extract 1 patch per image and pass it through the augmentations, JPEG compression included. \mathcal{D}_5 is entirely orthogonal to \mathcal{D}_3 and \mathcal{D}_4 concerning the semantic content and is partially orthogonal for the GANs used. Moreover, \mathcal{D}_5 is entirely orthogonal to \mathcal{D}_1 about JPEG alignment.

At deployment stage, we extract RGB patches 128×128 from the query image in different ways according to the CNN to be fed. For the CNN trained over \mathcal{D}_1 , we randomly select 200 patches per image. For the remaining CNNs, we always feed them with around 180 patches per image, aligned with the 8×8 pixel grid introduced by JPEG compression. This operation is done to ensure that the potential editing traces undergone by the test patches match with the JPEG training augmentations. Indeed, for building \mathcal{D}_1 , the training patches can be misaligned to the JPEG grid, while the remaining datasets always match the JPEG grid alignment, if JPEG is present.

3.2. Experimental setup

We keep 80% of the training images for training phase, leaving the remaining 20% for the validation. As commonly done in CNN training, we initialize the network weights using those trained on the ImageNet database. Every CNN is trained using cross-entropy loss and Adam optimizer with default parameters for a maximum of 500 epochs. The learning rate is initialized to 0.001 and is decreased by a factor 10 if the loss does not decrease for 10 epochs. Training is stopped if the loss does not improve for more than 20 epochs, then the model providing the best validation loss is selected. The experimental code is available at <https://github.com/polimi-ispl/GAN-image-detection>.

3.3. Results

This section reports the results achieved by the performed experimental campaign. First, we show the performance of the proposed patch aggregation strategy, then we evaluate the orthogonal CNN training. Eventually, we compare our results with state-of-the-art.

Patch aggregation. To show the effectiveness of the proposed patch aggregation strategy, we compare our approach with standard patch aggregation methodologies. For brevity’s sake, we show the benefits of our strategy only on the results achieved by one single CNN, as the trend is the same for all the considered networks.

Fig. 2 depicts the achieved image scores’ distributions by aggregating the patch scores returned by the CNN trained on \mathcal{D}_2 . In particular, Fig. 2(a) reports the results of the proposed method: if at least one patch is detected as synthetic, the image is assigned the “best” score among the synthetic ones. Figs. 2(b)-(c)-(d) show the results obtained by modifying this strict condition, letting the number of patches required for assigning the label “synthetic” grow to 5, 10 and 25 patches, respectively. Figs 2(e)-(f) report the results obtained by selecting the arithmetic mean and the median among the patch scores, respectively. For each scenario, we report the corresponding Area Under the Curve (AUC) of the Receiver Operating

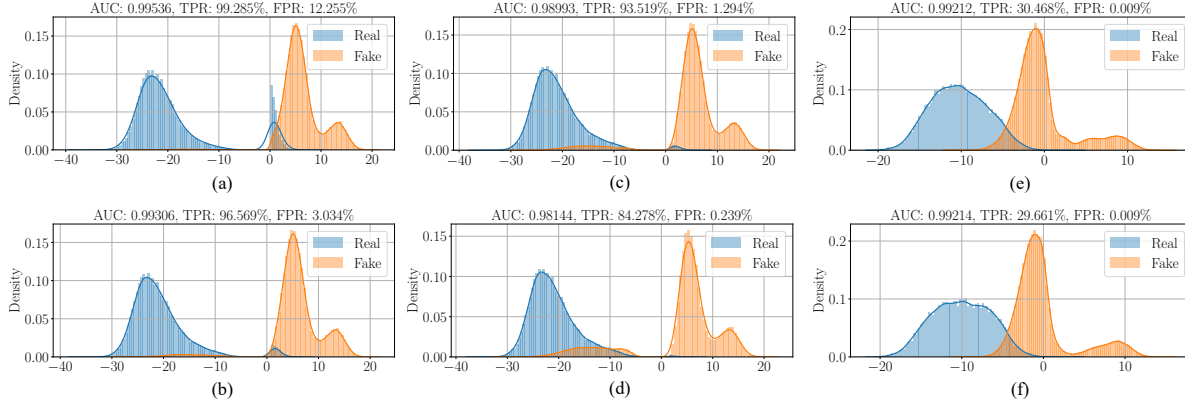


Fig. 2: Histogram of the image scores achieved by the CNN trained on dataset \mathcal{D}_2 : (a) reports the scores obtained by following the proposed patch aggregation strategy; (b), (c) and (d) show the scores obtained by modifying the threshold on the number of “synthetic” patches for assigning the image label to 5, 10 and 25, respectively; (e) and (f) report the results if we aggregate the patch scores by computing their arithmetic mean and median, respectively.

Table 1: AUC achieved in the eight considered testing scenarios and over the global test dataset. In bold, the two best AUCs for each scenario.

	AFHQ2-r	AFHQ2-t	Metfaces-r	Metfaces-t	FFHQ-r	FFHQ-t	FFHQ-r, res-comp	FFHQ-t, res-comp	Global
CNN ₁	0.9971	0.9995	0.9777	0.9896	0.9999	0.9995	0.9932	0.9932	0.9951
CNN ₂	0.9884	0.9954	0.8996	0.9347	0.9999	0.9997	0.9997	0.9997	0.9954
CNN ₃	0.9954	0.9996	0.9909	0.9834	0.9999	0.9994	0.4218	0.4313	0.8092
CNN ₄	0.9755	0.9986	0.9982	0.9991	0.9999	0.9988	0.8432	0.8534	0.9456
CNN ₅	0.6021	0.5858	0.6185	0.7322	0.9998	0.9995	0.9997	0.9997	0.9682
Fusion	0.9991	0.9999	0.9919	0.9964	0.9999	0.9999	0.9995	0.9995	0.9995

Characteristic (ROC) curve and the True Positive Rate (TPR) and False Positive Rate (FPR) achieved in the confusion matrix.

Our approach achieves the highest AUC and an extremely high detection accuracy for synthetic images at the cost of a few false alarms. As we increase the number of patches detected as synthetic for assigning the final score to the image (see Figs. 2(b)-(c)-(d)), the number of false alarms reduces, but the missed detections also increase. The arithmetic mean and median of the patch scores are far from being competitive with the proposed method.

Orthogonal CNN training. Table 1 reports the results of every single CNN and of the ensemble. We show the AUC achieved on the global test set, but we also investigate different scenarios in which only the real images of a particular dataset (e.g., FFHQ, Metfaces or AFHQ2) are compared with their synthetic versions generated through StyleGAN3. The considered scenarios are: (i) real AFHQ2 vs. synthetic AFHQ2 generated with r configuration; (ii) real AFHQ2 vs. synthetic AFHQ2 generated with t configuration; (iii) real Metfaces vs. synthetic Metfaces, r -versions; (iv) real Metfaces vs. synthetic Metfaces, t -versions; (v) real FFHQ vs. synthetic FFHQ, r -versions, without resizing and compression; (vi) real FFHQ vs. synthetic FFHQ, t -versions, without resizing and compression; (vii) real FFHQ vs. real FFHQ, r -versions, with resizing and compression; (viii) real FFHQ vs. synthetic FFHQ, r -versions, with resizing and compression.

The CNN ensemble often reports the best results. Regarding the single CNNs, the best methods on average are those trained over \mathcal{D}_1 and \mathcal{D}_2 . This was expected, as these CNNs were trained over a larger and more various amount of data with respect to the last three of them. However, every orthogonal training dataset carries important contributions related to the specific kind of data it is focused on. For instance, CNN₃ and CNN₄ achieve extremely high AUCs on AFHQ2 and Metfaces datasets, respectively. CNN₅ achieves almost

perfect AUCs over FFHQ undergone post-processing operations.

All CNNs report acceptable results in the global test scenario. Nonetheless, due to their specific training implementation, some CNNs might be more prone to detection errors than others in particular test scenarios, whereas their ensemble always maintains robust. Aiming at simulating realistic situations in which test images come from unknown generative models, the ensemble of multiple CNNs proves to be a valid option for synthetic image detection, paving the way towards robust and generalized solutions.

Comparison with state-of-the-art. The proposed GAN detector ranked first in the competition organized by NVIDIA, outperforming the results achieved by many expert teams in the field of multimedia forensics. Indeed, our method achieved the highest AUC over the global test set, as well as the best results in all the eight testing scenarios described previously. We refer the interested reader to [18] for any additional details and for comparing the state-of-the-art results.

4. CONCLUSIONS

In this paper we proposed a synthetic image detector based on an ensemble of CNNs, which are trained to increase the diversity within the ensemble. Our score aggregation strategy takes into account the fact that some image generators can be unknown at training time. Results show that these ideas help improving the detector accuracy on StyleGAN3 images that have never been used for training.

Despite the promising results, the orthogonality among the trained CNNs is only empirically verified at test time by observing the detector accuracy. Future work will be devoted to a deeper study of the CNNs diversity from a more theoretical view point. This will enable the development of ad-hoc training strategies.

5. REFERENCES

- [1] T. Karras, T. Aila, S. Laine, and J. Lehtinen, "Progressive growing of GANs for improved quality, stability, and variation," in *International Conference on Learning Representations*, 2018.
- [2] T. Karras, S. Laine, M. Aittala, J. Hellsten, J. Lehtinen, and T. Aila, "Analyzing and improving the image quality of StyleGAN," in *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020.
- [3] T. Karras, M. Aittala, J. Hellsten, S. Laine, J. Lehtinen, and T. Aila, "Training Generative Adversarial Networks with Limited Data," in *Conference on Neural Information Processing Systems (NeurIPS)*, 2020.
- [4] Y. Song, J. Sohl-Dickstein, D. P. Kingma, A. Kumar, S. Ermon, and B. Poole, "Score-based generative modeling through stochastic differential equations," in *International Conference on Learning Representations*, 2021.
- [5] T. Karras, M. Aittala, S. Laine, E. Härkönen, J. Hellsten, J. Lehtinen, and T. Aila, "Alias-free generative adversarial networks," in *Conference on Neural Information Processing Systems (NeurIPS)*, 2021.
- [6] F. Lago, C. Pasquini, R. Böhme, H. Dumont, V. Goffaux, and G. Boato, "More real than real: A study on human visual perception of synthetic faces," *IEEE Signal Processing Magazine*, vol. 39, pp. 109–116, 2022.
- [7] S. J. Nightingale and H. Farid, "AI-synthesized faces are indistinguishable from real faces and more trustworthy," *Proceedings of the National Academy of Sciences (PNAS)*, vol. 119, 2022.
- [8] F. Marra, D. Gragnaniello, L. Verdoliva, and G. Poggi, "Do gans leave artificial fingerprints?," in *2019 IEEE Conference on Multimedia Information Processing and Retrieval (MIPR)*, 2019.
- [9] M. Barni, K. Kallas, E. Nowroozi, and B. Tondi, "Cnn detection of gan-generated face images based on cross-band co-occurrences analysis," in *2020 IEEE International Workshop on Information Forensics and Security (WIFS)*, 2020.
- [10] N. Bonettini, E. D. Cannas, S. Mandelli, L. Bondi, P. Bestagini, and S. Tubaro, "Video face manipulation detection through ensemble of CNNs," in *International Conference on Pattern Recognition (ICPR)*, 2021.
- [11] S. Mandelli, D. Cozzolino, E. D. Cannas, J. P. Cardenuto, D. Moreira, P. Bestagini, W. Scheirer, A. Rocha, L. Verdoliva, S. Tubaro, and E. J. Delp, "Forensic analysis of synthetically generated scientific images," *arXiv preprint arXiv:2112.08739*, 2021.
- [12] S.-Y. Wang, O. Wang, R. Zhang, A. Owens, and A. A. Efros, "Cnn-generated images are surprisingly easy to spot... for now," in *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020.
- [13] D. Cozzolino, D. Gragnaniello, G. Poggi, and L. Verdoliva, "Towards universal gan image detection," in *International Conference on Visual Communications and Image Processing (VCIP)*, 2021.
- [14] D. Gragnaniello, D. Cozzolino, F. Marra, G. Poggi, and L. Verdoliva, "Are gan generated images easy to detect? a critical analysis of the state-of-the-art," in *IEEE International Conference on Multimedia and Expo (ICME)*, 2021.
- [15] M. Tan and Q. Le, "Efficientnet: Rethinking model scaling for convolutional neural networks," in *International Conference on Machine Learning (ICML)*, 2019.
- [16] A. Buslaev, V. I. Iglovikov, E. Khvedchenya, A. Parinov, M. Druzhinin, and A. A. Kalinin, "Albumentations: fast and flexible image augmentations," *Information*, vol. 11, pp. 125, 2020.
- [17] P. Dhariwal and A. Q. Nichol, "Diffusion Models Beat GANs on Image Synthesis," in *Conference on Neural Information Processing Systems (NeurIPS)*, 2021.
- [18] "NVIDIA StyleGAN3 synthetic image detection," <https://github.com/NVlabs/stylegan3-detector>, 2021.
- [19] "Flickr-Faces-HQ dataset (FFHQ)," <https://github.com/NVlabs/ffhq-dataset>.
- [20] "Metfaces dataset," <https://github.com/NVlabs/metfaces-dataset>.
- [21] Y. Choi, Y. Uh, J. Yoo, and J.-W. Ha, "StarGANv2: Diverse image synthesis for multiple domains," in *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020.
- [22] P. Esser, R. Rombach, and B. Ommer, "Taming transformers for high-resolution image synthesis," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2021.
- [23] T.-C. Wang, A. Mallya, and M.-Y. Liu, "One-shot free-view neural talking-head synthesis for video conferencing," in *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2021.
- [24] S. Mandelli, N. Bonettini, P. Bestagini, and S. Tubaro, "Training cnns in presence of jpeg compression: Multimedia forensics vs computer vision," in *IEEE International Workshop on Information Forensics and Security (WIFS)*, 2020.