On the Exploitation of Deepfake Model Recognition

Luca Guarnera [1] Oliver Giudice [2] Matthias Nießner [3] Sebastiano Battiato [1]

[1] Department of Mathematics and Computer Science, University of Catania, Italy
[2] Banca d'Italia, Applied Research Team, IT Dept., Rome, Italy
[3] Technical University of Munich, Germany

luca.guarnera@unict.it giudice@dmi.unict.it niessner@tum.de battiato@dmi.unict.it

Abstract

Despite recent advances in Generative Adversarial Networks (GANs), with special focus to the Deepfake phenomenon there is no a clear understanding neither in terms of explainability nor of recognition of the involved models. In particular, the recognition of a specific GAN model that generated the deepfake image compared to many other possible models created by the same generative architecture (e.g. StyleGAN) is a task not yet completely addressed in the state-of-the-art. In this work, a robust processing pipeline to evaluate the possibility to point-out analytic fingerprints for Deepfake model recognition is presented. After exploiting the latent space of 50 slightly different models through an in-depth analysis on the generated images, a proper encoder was trained to discriminate among these models obtaining a classification accuracy of over 96%. Once demonstrated the possibility to discriminate extremely similar images, a dedicated metric exploiting the insights discovered in the latent space was introduced. By achieving a final accuracy of more than 94% for the Model Recognition task on images generated by models not employed in the training phase, this study takes an important step in countering the Deepfake phenomenon introducing a sort of signature in some sense similar to those employed in the multimedia forensics field (e.g. for camera source identification task, image ballistics task, etc).

1. Introduction

The phenomenon of the so-called Deepfake has been observed and treated with growing concerns and interests in recent years by the research community. In summary, there is quite agreement on the nature and disruptive potential of this technology: based on a particular application of Generative Adversarial Networks (GAN) [14] they synthesize extremely realistic multimedia content.

Legal literature in recent years has dealt with the subject from the perspective of national security and safeguarding of the democratic order and the protection of privacy, as well as the reputational damage of natural and legal persons, as well as the crime related to non-consensual dissemination of sexually explicit images. For this reason, the computer science community has addressed the problem by focusing on proving the authenticity of an image [17, 27] or by developing more and more sophisticated Deepfake Detection techniques [15, 25, 26, 32, 33, 35].

Deepfake images are so realistic that even specialists can find it difficult to detect them [19]. On the other hand, it seems that state-of-the-art (SOTA) techniques are surprisingly efficient to spot them and even to recognize the architecture. Thus, a new and more difficult challenge could be to recognize the model (defined as the set of weights/parameters) which is an instance of a specific architecture. Having a Model Recognition solution could enable the possibility of attributing an image to a specific model owner (i.e. for Intellectual Property).

Detection and model recognition are definitely required to approach and counter the Deepfake phenomenon. To prove the ownership or authenticity of an image generated by a given model of a specific architecture, novel strategies with proper "metrics" are needed. This is of utmost importance in the media forensics field, to reconstruct the origin of an image [10]. New definitions can be given for the image ballistics investigation when Deepfake images are involved: SOTA Deepfake Detectors and Architecture Classifiers could be associated with the camera model identification task whilst the camera source identification task could be equivalent to a Deepfake Model Recognition task, given a specific architecture. This last task has not yet been investigated in literature.

A fundamental element of all GAN architectures is their latent space. In summary, the N-dimensional latent space of a GAN represents all the patterns learned by the "Generator" in the training phase related to the training data. Essentially, it maps the space to specific features of the training data. This is fundamentally used to re-create synthetic multimedia data with features similar to those of the

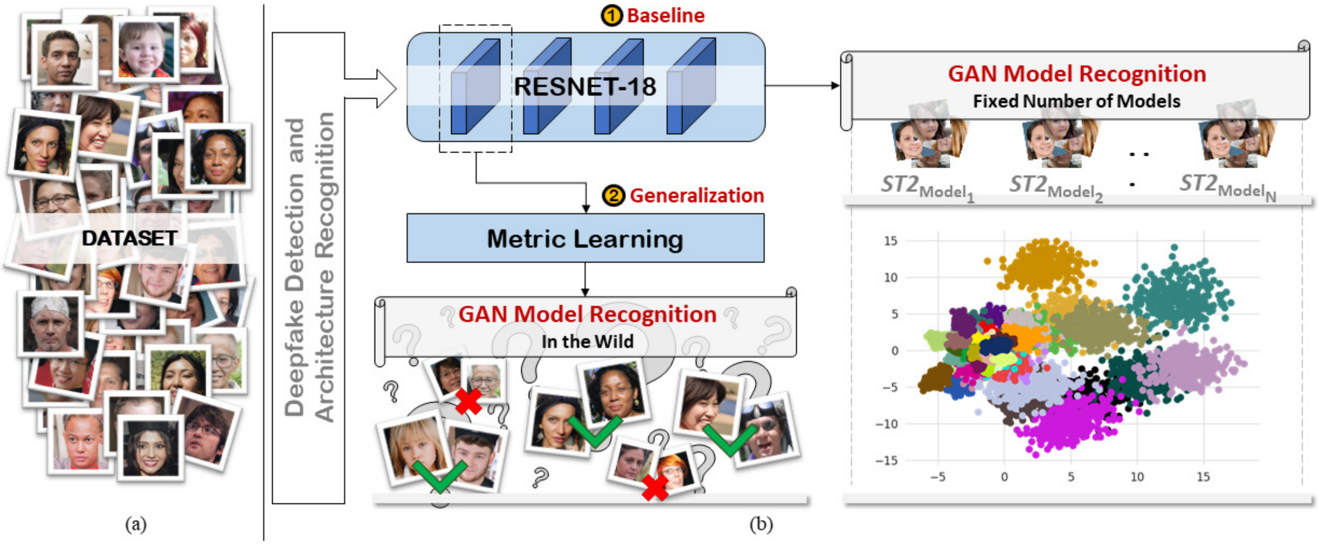


Figure 1. Schematic representation of the robust processing pipeline proposed for Deepfake Model Recognition. At first, a dataset has to be created (a) by fine-tuning different models and thus generating images. In general, an efficient Deepfake detector/classifier could be employed at first from those available in literature. Then, an encoder (b) is trained for the recognition task on the a-priori known models. The obtained feature vectors on a two-dimensional space shows the discriminative power of features encoded by the baseline encoder. Finally, the first layer of the trained encoder is exploited in a metric learning approach.

dataset used during the training phase. Each time a GAN model is trained the latent space will be different. Several attempts to introduce GAN explainability have been proposed by exploiting the visual concept vocabulary [29], analyzing/basing on node collapse [5] , and by interpreting the latent semantic space [31]. StyleGAN [1], among others, is an example of generative architecture that introduced a specific latent space to incorporate semantic information for face editing.

In this paper, the latent spaces of 50 slightly different StyleGAN2 [22] models were explored in depth. To this end, to reduce any correlation to high-level semantic content, only images of faces were taken into account. First, 50 different generative models were trained and the generated images were analyzed considering standard visual metrics and distances with respect to the latent space of each model. The discovered insights, confirmed the existence of traces linked to the StyleGAN2 architecture, thus explaining why SOTA detection solutions are able to obtain astonishing results. As a matter of fact, the Deepfake Detection task is simpler than recognizing a model which generated an image from a given architecture. To propose a novel processing pipeline and methodology for this new and more challenging task, a Resnet-18 [18] encoder was trained with the objective of classifying images among 50 a-priori known slightly different models of the StyleGAN2 architecture. An overall accuracy of 96.24% demonstrated the possibility of discriminating extremely similar images for the Deepfake Model Recognition task. Thus, a dedicated metric exploiting the encoded feature was introduced. Experimental results demonstrated the effectiveness of the proposed metric, for making the first milestone in countering the Deepfake phenomenon by introducing a method equivalent to the camera source identification task which is well known to image forensics experts. Finally, the processing pipeline and metric were experimentally tested to understand their limitations and generalizing ability. The main contributions of this paper are summarized as follows:

1. An in-depth exploration of the latent space of the slightly different StyleGAN2 models, and a description of the insights discovered;

2. A new pipeline for the Deepfake Model Recognition task, given a specific architecture. Figure 1 schematically shows the overall framework;

3. A methodology to encode a robust descriptor and a respective metric for Deepfake Model Recognition.

4. Datasets and code for Deepfake Model creation and recognition, publicly available.

The remainder of this paper is organized as follows: Section 2 briefly overviews related works from literature. Section 3 introduces the notation and a detailed description of the methodology used to train different models and generate images. Then, Section 4 presents the pipeline for Deefake Model Recognition which is finally tested and discussed in Section 5. Finally, a summarized discussion is reported in Section 6 that concludes the paper with hints for future works.

2. Related Works

Deepfake images are generally created by techniques based on GANs, originally introduced by Goodfellow et al. [14]. Since 2014, many architectures and variants have been proposed [6]; one of the most recent and powerful methods regarding the entire-face synthesis is the Style Generative Adversarial Network architecture or commonly called StyleGAN [21]. It is capable not only of generating impressive photo-realistic and high-quality faces, but also offers some kind of control in terms of the overall style of the generated image at different levels of detail. Later, Karras et al. made some improvements to the technique proposing the StyleGAN2 [22] architecture. Images generated by StyleGAN2 are extremely realistic and many techniques have already been proposed to detect their fakeness. On this topic, several overviews on Media forensics with a particular focus on Deepfakes have been recently published [11, 26, 33, 35, 44].

To analyze the authenticity of a digital media (image, audio, video) and reconstruct its history (provenance and phylogeny), SOTA techniques could be divided into two major categories: active and passive [4, 28].

Active techniques embed an hidden signature on images by varying the image creation process (generative process when dealing with Deepfakes). This signature can be detected at any time in the life-cycle of an image. Active techniques related to Deepfake images have been proposed recently: watermarking techniques [36, 43], which embed information on pixels and Neural Network Watermarking techniques [34,40,42], which embed information on the network parameters. While being highly efficient, if a watermark is not embedded preliminarly on images only passive techniques could be employed. Moreover, if the embedding process is known, watermark canceling and extraction could be performed.

However, passive techniques attempt to extract hidden signatures accidentally left by the creation process of images. These techniques are less efficient than active ones but they could be employed in any scenario. In general, they can only deal with easier tasks, such as: Deepfake Detection, which is a binary classification task between real and fake images; and Deepfake Attribution (or GAN architecture recognition), which is a multi-class classification task. Indeed, Deepfake Detection techniques in SOTA seem to solve an extremely easy task [38]. This fact was demonstrated to be related to a sort of fingerprint, a convolutional trace, left on the images by the GAN generative process [16, 24]. The underlying trace, invisible to the human eye, can be easily spotted in the frequency domain [8]: Discrete Cosine Transform (DCT) analysis, for example, could lead to the development of extremely powerful, explainable and efficient detectors [9, 12]. Many techniques are available in literature that focus on different architectures and scenarios [2, 7, 13, 16, 37, 39].

All the aforementioned passive approaches are able to achieve high accuracy scores, even in the GAN Architecture Classification task. Creation, detection, and architecture recognition have been the main topics addressed by the scientific community A first study on the recognition task was recently proposed by Asnani et al. [3]: an "agnostic" prediction solution of the architecture that generated a given image. For instance, given an image, they are able to infer the structure (with an overall hyper-parameter setting) of the GAN architecture which generated that image. In particular, they make use of a specific single instance of a model for each considered architecture.

None of the above-mentioned SOTA approaches (Deepfake Detection, GAN Architecture Classification, Attribute Recognition) are sufficient to recognize if an image was generated by someone employing not only a given architecture but a specific model of that given architecture, defined by its set of unique (hyper) parameters [41] (among different instances). This last task could be defined as Deepfake Model Recognition and it has not yet been completely addressed in literature. This paper deals with this lack in literature by proposing a preliminary metric-based solution and methodology.

3. A Dataset for Model Recognition

StyleGAN2 is the architecture considered as the first example of delving into the Deepfake Model Recognition task. Specifically, StyleGAN2-ADA [20] was the employed implementation given its fine-tuning training easiness which, with few training examples, is able to obtain almost completely artifact-free Deepfake face images. Moreover, the StyleGAN2-ADA implementation easily allows to freeze the weights/parameters of a user-defined number of layers in the "Discriminator" during the training operations. In order to have this freezing ability also available for the layers of the "Generator", the original implementation of StyleGAN2-ADA was customized. This was of key importance for the task of training different models with slightly different parameters (aka weights) from which generating images extremely similar to each other and almost "identical" to human eyes. To facilitate the reading of this paper, given the high number of slightly different models and the complexity of the created dataset, a mathematical formalism will be introduced in the following.

3.1. The Models

Let M_{ST2} be the well-known pre-trained StyleGAN2 model [1], several fine-tune trainings were carried out by

[1] https://github.com/NVlabs/stylegan2

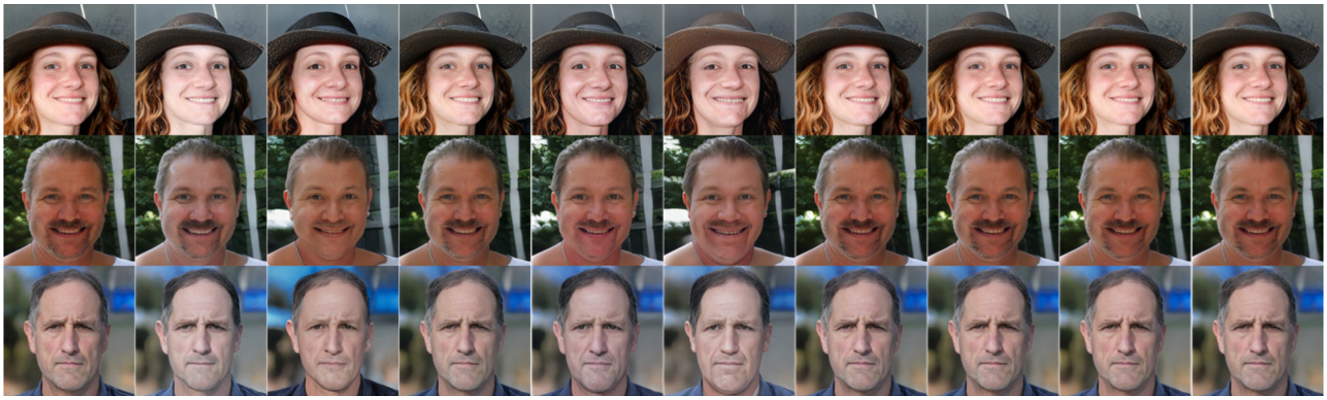


Figure 2. Generated image examples from the slightly different models employed in the experiments.

Generator	Discriminator
synthesis.b1024.conv0	b4.mbstd
synthesis.b1024.conv1	b4.conv
synthesis.b1024.torgb	b4.fc
synthesis.b1024:0	b4.out
synthesis.b1024:1	

Table 1. StyleGAN2 Generator G and Discriminator D structure. The first column describes layer $b1024$ in the synthesis block in G; the second column represents the structure of layer $b4$ in D. These blocks represent the layers that were re-trained during the fine-tuning procedure

freezing all the layers except the latter in the Generator (G) and Discriminator (D) considering the FFHQ as the training dataset (employed in the same way as described in the official StyleGAN2-ADA Github repository [2]). Training operations were carried out by varying only the following hyper-parameters: *k-img* $\in \{1, 2, ..., 10\}$ and $p \in [0, 1]$, which describe training-set augmentation probability. Note that these hyper-parameters affect the training process only, while all the other hyper-parameters were left as StyleGAN2-ADA defaults. All possible combinations of pairs $(\bar{k}, \bar{p})$ with $\bar{k} \in$ *k-img* and $\bar{p} \in p$ were considered (examples (1, 0.1), (1, 0.2), ... (10, 0.9), (10, 1.0)) to obtain a total of 100 StyleGAN2-ADA fine-tuned models ($M_{ST2-ADA}$). For each $M_{ST2-ADA}$, $1,000$ images were generated for a total of $100,000$ images.

Figure 2 shows samples of the generated images: the first column shows images created by the pre-trained StyleGAN2 model (M_{ST2}); the remaining columns represent samples of the images generated by the fine-tuned StyleGAN2-ADA models, with proper combinations of ($\bar{k}$, $\bar{p}$) parameters.

Let $M_i(W)$ be the *i-th* model of $M_{ST2-ADA}$ uniquely defined by its set of parameters (weights) $W = (W_G, W_D)$ with $W_G = \{w_{G1}, w_{G2}, ..., w_{Gn}\}$ and $W_D = \{w_{D1}, w_{D2}, ..., w_{Dm}\}$; n and m are the total number of layers in G and D respectively (e.g., w_{G1} represents all weights of layer1 in G, while w_{G2} all weights of layer2 in G, etc.). It is worth to highlight that each model $M_i(W)$ is uniquely defined by the values of its parameters W and thus creates a unique latent space. The W of all $M_{ST2-ADA}$-created models will differ only by $w_{Gn} \in W_G$ and $w_{Dm} \in W_D$ which correspond to weights belonging only to the last layer of G and D. Specifically, as far as the "Generator" is concerned, only the layer named $b1024$ was not frozen. This layer, represents the last image synthesis operation and directly affects the image creation process. Similarly for the "Discriminator" only the layer named $b4$ was not frozen. A detailed description of the composition of the mentioned layers, for both the Generator and Discriminator, is reported in Table 1. The described training process produced slight differences in each model parameter set W. This is demonstrated in Figure 3 where a plot obtained from a 2-dimensional PCA reduction of the weights extracted from the last layer of the Generator and the Discriminator of each $M_i(W)$ is reported showing an initial exploration of the corresponding latent space.

3.2. The images

Let $I_{M_i(W)}$ be the set of images generated by the model $M_i(W)$, the $I_{M_{ST2-ADA}(W)}$ images will turn out to be very similar to each other in terms of visual appearance. This is due to the slight differences in the weights W among the different models $M_i(W)$.

In Figure 2 are reported several images of $I_{M_i(W)}$: some may show visible artifacts, whereas others may present totally imperceptible variations, not only w.r.t. the basic StyleGAN2 model M_{ST2} (Figure 2 first column) but also w.r.t. all the other models $M_{ST2-ADA}$ (Figure 2 from column 2 onwards). In order to understand these impercep-

[2] https://github.com/NVlabs/stylegan2-ada-pytorch

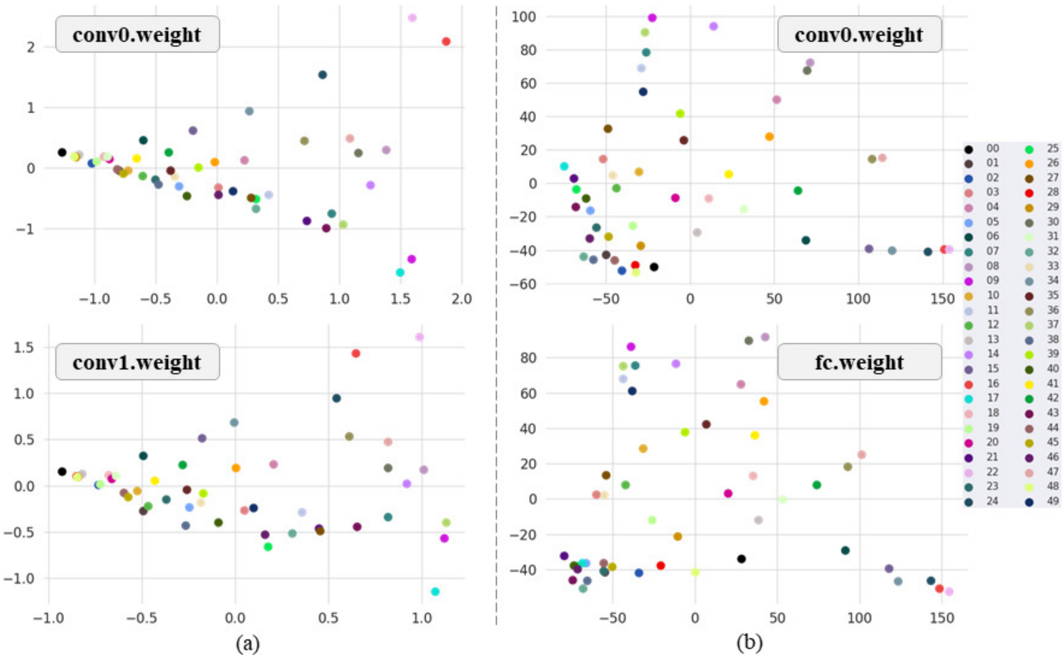


Figure 3. 2D PCA of the parameters extracted from the last layer of the Generator (a) and the Discriminator (b) of each of the 50 models.

tible differences, the *Structural Similarity Index Measure* (SSIM) was employed between the image samples generated by M_{ST2} vs. all those generated by the $M_{ST2-ADA}$ models. Figure 4 shows an example of the results obtained in form of a matrix comparison. It is possible to note that in almost all cases the variations between each pair of images are very small (Figure 4(a)) consequently the Model Recognition could be a very complicated task.

3.3. Training and Test data definition

In the previous Sections, the creation of $M_i(W)$ with $i = 1, \dots, 100$ different models were described. Only 50 (M_{1-50}) of the 100 $M_i(W)$ models were randomly selected for the training procedures of the algorithms presented in the following Sections. Specifically, for each of the selected model, $1,000$ images were generated. The "baseline" and the "generalization" phases of the pipeline, as shown in Figure 1, were trained on this set of images. Testing was carried out on different sets of images: the "baseline" phase was tested on 300 other images generated by M_{1-50} models and the "generalization" phase was tested to compute proper scores on 300 images generated by each M_{51-100} models (for a total of $30,000$ images of the two sets).

4. Deepfake Model Recognition

The image dataset and models described in Section 3 only show the faces of people as content. This creates an extremely challenging scenario in which the characteristics can be summarized as follows: (i) there are images representing almost the same faces among the slightly different employed models; (ii) each face has very small variations depending on the model. These properties could be of great importance in obtaining invariance to high-level semantic features related to the image content. Thus, it is possible to build a robust processing pipeline that is able to capture only the discriminating parameters in the latent space of the $M_{ST2-ADA}$ models, making it not related to a specific face feature.

The objective of a Deepfake Model Recognition task is to assess whether two images were generated by the same model by defining a metric L. Formally,

$$L(F(I_a^i), F(I_b^j)) \in [-1, 1] \qquad (1)$$

where $I_a^i \in I_{M_i(W)}$ and $I_b^j \in I_{M_j(W)}$ are two images generated respectively by $M_i(W)$ and $M_j(W)$ models ($M_i \in M_{ST2-ADA}$ and $M_j \in M_{ST2-ADA}$), and $F(\cdot)$ is a function for fingerprint extraction from a given input image. The more L is near to "1", the higher is the probability that the two images were generated using the same model. On the other hand, L near to "-1", means that images were generated by different models.

The elements of Equation 1 are found in three different (but consequent) moments:

1. To define and to train an encoder (F), able to discriminate 50 different classes (M_{1-50}) and demonstrate its effectiveness by measuring the accuracy even in noisy scenarios;

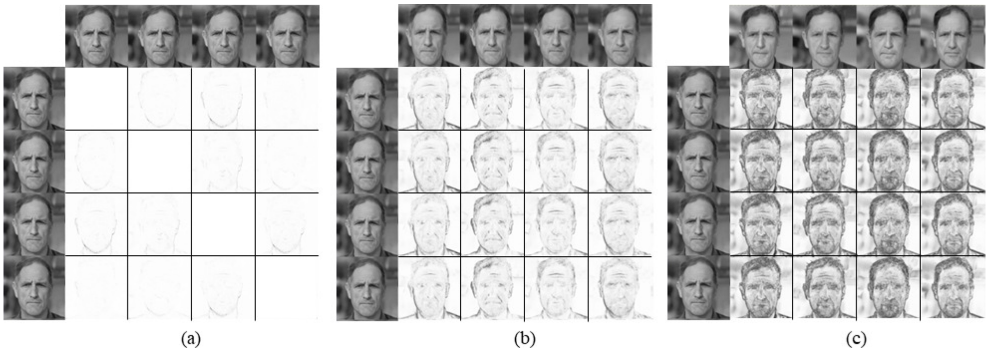


Figure 4. SSIM maps comparisons between gray-scale images. (a) shows samples of extremely similar images; (b) shows samples where the differences are related mainly to the smoothing effects. (c) shows examples where the difference is more evident and related to misalignments.

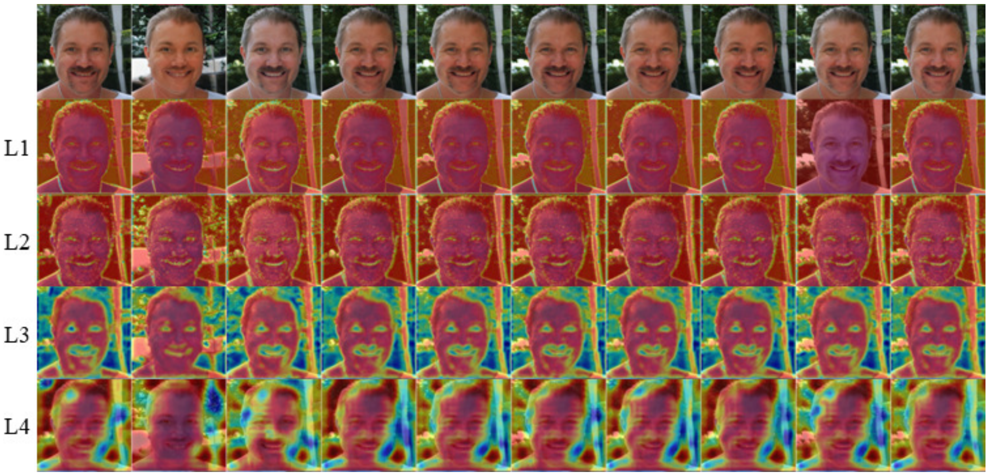


Figure 5. Attention maps of the first four layers (L1, L2, L3, L4) of the Resnet-18 trained for the a-priori known model classification task.

2. To employ the trained classifier, as an embedding method (F) that is able to exploit the latent spaces, even for images generated by models never seen during the training phase (M_{51-100});

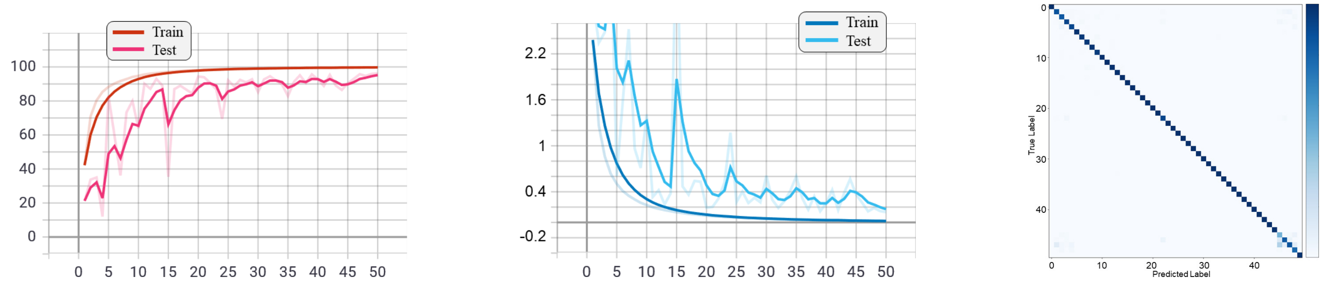
3. To define and to learn a robust metric L to compare two images I and assess whether they were generated by the same model.

Figure 1 graphically schematizes the proposed approach: starting from the dataset generation phase to the final metric definition; it is possible to build a Model Recognition pipeline by exploiting a robust "baseline" classifier and a metric for "generalization", given a Deepfake architecture.

4.1. Building an encoder for classification

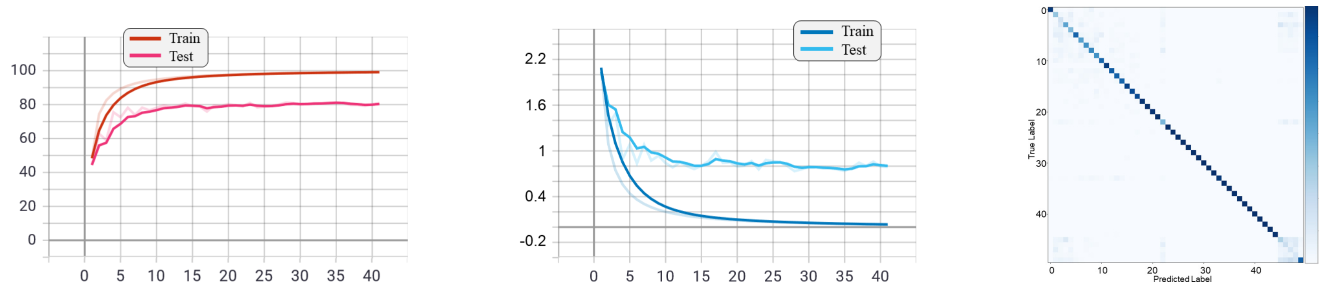
The Resnet-18 encoder was employed for the classification task of the images generated by M_{1-50}. Specifically, the Pytorch implementation of Resnet was used starting from the pre-trained version trained on ImageNet (publicly available in the torchvision module).A fully-connected layer with an output size of 50 followed by a SoftMax were added to the last layer of the Resnet-18 in order to be trained on the specific task. The following settings were employed for training: learning rate = 0.001, batch size = 30, epochs = 50, "cross entropy loss" as loss function and the Stochastic Gradient Descent (with momentum = 0.9). Figure 6 shows the accuracy (a) and loss (b) values of the training phase.

In order to define the best $F(\cdot)$ (see Equation 1), an evaluation on images was carried out with respect to attention heat-maps [30] for each layer of the trained Resnet-18. While Layer-1 encodes features as far as possible both from the final task and from the semantics, it is possible to note that Layer-4 concentrates much more on face skin and keypoints than on the overall image parts (Figure 5). Thus, the best $F(\cdot)$ would be the Layer-1. Moreover, Layer-1 is the smallest in terms of dimensions and would be better for computational costs for any kind of further processing. We also note that all parameters of the employed Resnet were trained to solve the proposed task. This implies that even the first layers are able to extract salient features.



(a) (b) (c)

Figure 6. Training accuracy (a) and loss (b) through 50 epochs on raw-images. The confusion matrix in (c) shows classification results obtained by the best classification model, over 50 classes.



(a) (b) (c)

Figure 7. Training accuracy (a) and loss (b) through 41 epochs on noisy images. The confusion matrix in (c) shows classification results obtained by the best classification model, over 50 classes.

4.2. Learning a metric

The best model obtained in the operations described in Section 4.1 was selected as F w.r.t. Equation 1. Specifically, given a generic image I created from any model $M_{ST2-ADA}$, given the feature vector extracted from the output of Layer-1 of the trained Resnet-18, a tensor with dimensions $(1, 64, 256, 256)$, $F(I)$ is the function that extracts this tensor, flattens it and then applies a Singular Value Decomposition (SVD). SVD was fitted on 50% of the training-set samples. The explained variance was employed as a score to find the best reduction (400, with a corresponding explained variance of 74%).

Given F, the definition of L will be carried out by means of a metric learning approach developed specifically for working with high-dimensional data [23]. This approach was chosen for being able to handle a specific kind of input feature vector and for being a weakly supervised metric learning approach, which is the case, given the weak quality of labelling limited only to similar or dissimilar labels.

5. Experimental Results

The solution described in the previous Sections and summarized in Figure 1 was deeply analyzed and tested on images belonging to the data described in Section 3. At first the trained feature extraction method (F) will be demonstrated to be robust and efficient in discriminating the images generated by 50 slightly different models. Moreover, F is tested for robustness on a noisy version of the images on which different attacks were applied (Section 5.1). Finally, a metric L is defined and experimentally tested in its ability to overcome the state-of-the-art and be the first solution for Deepfake Model Recognition (Section 5.2).

5.1. A-priori known models classification results

Given a set of known models of a given architecture (M_{1-50}), and the respective $1,000 \times 50$ images generated from them, a training-validation set split was performed (75%-25%). A Resnet-18 architecture was trained on the training-set and the training/validation loss plots obtained are shown in Figure 6 (a) and (b). Then, the trained model was evaluated in the classification task on the validation set. Results, are shown in Figure 6 in the form of a heat-map confusion matrix (c). An average classification accuracy value of 96.24% was obtained demonstrating the effectiveness of the solution on a-priori known models.

Robustness evaluations were carried out by training a new Resnet-18 model on images where several filters were applied. In particular: (i) *Gaussian Blur* considering different kernel sizes ($3x3$, $9x9$, $15x15$); (ii) *Rotations* with degrees equal to $45, 135, 255, 315$ degrees; (iii) *Mirroring* along one axis (horizontal, vertical or both); (iv) *Scaling* considering -50% and $+50\%$ w.r.t. the original image dimensions; (v) *JPEG compression* with different quality factor values: $QF = \{50, 60, ..., 90\}$ and standard Quantization Tables. Many of these attacks are able to destroy several frequencies in Deepfake images making the extraction of a discriminative signature for each model more complex. Details of the noise-augmented version of the data is available at a dedicated site[3]. In the same way as before, a training-validation set split was performed (75%-25%) and the pre-trained Resnet-18 obtained before, for model classification on RAW images, was fine-tuned on this augmented dataset. The obtained training and validation plots are shown in Figure 7 (a) and (b) and the overall classification results are reported in Figure 7 in the form of a heatmap confusion matrix (c). In this case, the obtained average classification accuracy of 81.37% demonstrated the robustness of the approach for the classification task on a-priori known models.

5.2. Unknown model classification results

Given a set of images "in the wild", the task of Deepfake Model Recognition could not be carried out with techniques available in literature and with the classification model experimentally tested in the previous Section. Ideally, a first assessment could be performed using SOTA Deepfake detectors and architecture classifiers. Now, given a set of unknown models of a given architecture, the Model Recognition task could be carried out by exploiting a metric L.

At first, metric L was trained on a-priori known models (M_{1-50} of the dataset) by means of F (as defined and trained in the previous Section). The learnt metric L was then employed in the tests. Three test sets were employed: (i) further 300×50 images generated by a-priori known models M_{1-50}; (ii) images generated by unknown models never used for any training operation M_{51-100}; (iii) the union of two previous sets and (iv) a noisy augmented version of (iii) with the noises described in Section 4.1.

All tests were carried out considering 20% of each dataset as a reference and the remaining part was employed in a k-NN classification experiment employing the L metric. 5-fold cross-validation was carried out and results, in terms of average accuracy, are reported in Table 2.

A further test was carried out on images belonging to the (i) set against images generated by a different architecture: StyleGAN [21]. As presented in Table 2, an average accuracy of almost 100%, demonstrates that this metric is able to maintain the correct distancing concept even between two different architectures, without any further training. Finally, to introduce a preliminary concept of scalability, the learnt metric L was further tested on images generated by two different pre-trained models of the StyleGAN-1 [21] architecture (30 images for each model for a total of 60).

	1-NN	3-NN	5-NN
A-priori known	95.08%	95.74%	95.74%
Unknown models	92.31%	92.55%	94.33%
Known + unknown	93.70%	94.24%	95.06%
Noise-augmented	75.66%	77.00%	79.66%
ST1-M1 vs. StyleGAN2	99.98%	99.98%	99.98%
ST1-M1 vs. ST1-M2	84.50%	88.20%	92.00%

Table 2. Results of the learnt metric L by means of employing it in a k-NN classification test. Values reported are the average accuracies obtained on 5-fold cross validation tests employing 20% of data as reference and the remaining for classification. ST1-M1 stands for the first model of StyleGAN1.

6. Discussion and Conclusion

In this paper, a preliminary solution for Deepfake Model Recognition was presented. The underlying task, not yet completely addressed in the state-of-the-art, was described and attacked with an effective solution. The work proposed by Asnani et al [3] is the most similar work, available in SOTA, compared to the proposed approach. Despite this, it is not possible to perform a comparison with [3] because the proposed solution focuses on a more specific task: discriminating between different models created by the same GAN engine. The images generated by the 100 StyleGAN2 models were visually almost the same (Figure 2) and the main aim was to find features to discriminate them. Every new StyleGAN2 model has its own latent space that will be (even if in a "minimal" way) different from the 100 used in this paper. This happens because the training procedure will inevitably change the parameters of the neural architecture (so the latent space will be different). We are confident that the proposed method is scalable in any domain, because in general any other generative architecture, by its nature, will have a different latent representation since the architecture itself will be different. The ability to compare whether two images are generated by the same model, sidesteps the problem of predicting/classifying a model out of potentially infinite possibilities. The proposed methodology was demonstrated to be generalizable even for different architectures providing useful insights for building a sort of feature dataset for each involved architecture similar to those PRNU-based for camera source identification methods. Finally, the proposed method was demonstrated to be effective in real-case scenarios. As a matter of fact, we widely tested the robustness of the technique in the wild: for instance IM platforms such as Whatsapp compress images with a quality factor equal to 50: in this context, our method performed with an accuracy of over 80%.

[3] https://iplab.dmi.unict.it/mfs/Deepfakes/PaperCVPRW2022/

References

[1] Rameen Abdal, Yipeng Qin, and Peter Wonka. Image2stylegan: How to embed images into the stylegan latent space? In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 4432–4441, 2019. 2

[2] Shruti Agarwal, Hany Farid, Tarek El-Gaaly, and Ser-Nam Lim. Detecting deep-fake videos from appearance and behavior. In *2020 IEEE International Workshop on Information Forensics and Security (WIFS)*, pages 1–6. IEEE, 2020. 3

[3] Vishal Asnani, Xi Yin, Tal Hassner, and Xiaoming Liu. Reverse engineering of generative models: Inferring model hyperparameters from generated images. *arXiv preprint arXiv:2106.07873*, 2021. 3, 8

[4] Sebastiano Battiato, Oliver Giudice, and Antonino Paratore. Multimedia forensics: discovering the history of multimedia contents. In *Proceedings of the 17th International Conference on Computer Systems and Technologies 2016*, pages 5–16, 2016. 3

[5] David Bau, Jun-Yan Zhu, Jonas Wulff, William Peebles, Hendrik Strobelt, Bolei Zhou, and Antonio Torralba. Seeing what a GAN Cannot Generate. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 4502–4511, 2019. 2

[6] Antonia Creswell, Tom White, Vincent Dumoulin, Kai Arulkumaran, Biswa Sengupta, and Anil A Bharath. Generative adversarial networks: An overview. *IEEE Signal Processing Magazine*, 35(1):53–65, 2018. 3

[7] Hao Dang, Feng Liu, Joel Stehouwer, Xiaoming Liu, and Anil K Jain. On the detection of digital face manipulation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5781–5790, 2020. 3

[8] Ricard Durall, Margret Keuper, and Janis Keuper. Watch your up-convolution: CNN based generative deep neural networks are failing to reproduce spectral distributions. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7890–7899, 2020. 3

[9] R. Durall, M. Keuper, F. Pfreundt, and J. Keuper. Unmasking deepfakes with simple features. *arXiv preprint arXiv:1911.00686*, 2019. 3

[10] Hany Farid. Digital image ballistics from JPEG quantization: A followup study. *Department of Computer Science, Dartmouth College, Tech. Rep. TR2008-638*, 2008. 1

[11] Joel Frank, Thorsten Eisenhofer, Lea Schönherr, Asja Fischer, Dorothea Kolossa, and Thorsten Holz. Leveraging frequency analysis for deep fake image recognition. In *International Conference on Machine Learning*, pages 3247–3258. PMLR, 2020. 3

[12] Oliver Giudice, Luca Guarnera, and Sebastiano Battiato. Fighting Deepfakes by Detecting GAN DCT Anomalies. *Journal of Imaging*, 7(8), 2021. 3

[13] Michael Goebel, Lakshmanan Nataraj, Tejaswi Nanjundaswamy, Tajuddin Manhar Mohammed, Shivkumar Chandrasekaran, and BS Manjunath. Detection, attribution and localization of GAN generated images. *Electronic Imaging*, 2021(4):276–1, 2021. 3

[14] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. In *Advances in Neural Information Processing Systems*, pages 2672–2680, 2014. 1, 3

[15] Luca Guarnera, Oliver Giudice, and Sebastiano Battiato. DeepFake Detection by Analyzing Convolutional Traces. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops*, pages 666–667, 2020. 1

[16] Luca Guarnera, Oliver Giudice, and Sebastiano Battiato. Fighting Deepfake by Exposing the Convolutional Traces on Images. *IEEE Access*, 8:165085–165098, 2020. 3

[17] Haya R Hasan and Khaled Salah. Combating deepfake videos using blockchain and smart contracts. *Ieee Access*, 7:41596–41606, 2019. 1

[18] 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*, pages 770–778, 2016. 2

[19] Nils Hulzebosch, Sarah Ibrahimi, and Marcel Worring. Detecting CNN-Generated Facial Images in Real-World Scenarios. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops*, pages 642–643, 2020. 1

[20] Tero Karras, Miika Aittala, Janne Hellsten, Samuli Laine, Jaakko Lehtinen, and Timo Aila. Training generative adversarial networks with limited data. In H. Larochelle, M. Ranzato, R. Hadsell, M. F. Balcan, and H. Lin, editors, *Advances in Neural Information Processing Systems*, volume 33, pages 12104–12114. Curran Associates, Inc., 2020. 3

[21] Tero Karras, Samuli Laine, and Timo Aila. A style-based generator architecture for generative adversarial networks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4401–4410, 2019. 3, 8

[22] Tero Karras, Samuli Laine, Miika Aittala, Janne Hellsten, Jaakko Lehtinen, and Timo Aila. Analyzing and improving the image quality of stylegan. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8110–8119, 2020. 2, 3

[23] Eric Yi Liu, Zhishan Guo, Xiang Zhang, Vladimir Jojic, and Wei Wang. Metric learning from relative comparisons by minimizing squared residual. In *2012 IEEE 12th International Conference on Data Mining*, pages 978–983. IEEE, 2012. 7

[24] Francesco Marra, Diego Gragnaniello, Luisa Verdoliva, and Giovanni Poggi. Do GANs leave artificial fingerprints? In *2019 IEEE Conference on Multimedia Information Processing and Retrieval (MIPR)*, pages 506–511. IEEE, 2019. 3

[25] Momina Masood, Marriam Nawaz, Khalid Mahmood Malik, Ali Javed, and Aun Irtaza. Deepfakes generation and detection: State-of-the-art, open challenges, countermeasures, and way forward. *arXiv preprint arXiv:2103.00484*, 2021. 1

[26] Yisroel Mirsky and Wenke Lee. The creation and detection of deepfakes: A survey. *ACM Computing Surveys (CSUR)*, 54(1):1–41, 2021. 1, 3

[27] Ding Sheng Ong, Chee Seng Chan, Kam Woh Ng, Lixin Fan, and Qiang Yang. Protecting intellectual property of generative adversarial networks from ambiguity attacks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3630–3639, 2021. 1

[28] Alessandro Piva. An overview on image forensics. *International Scholarly Research Notices*, 2013, 2013. 3

[29] Sarah Schwettmann, Evan Hernandez, David Bau, Samuel Klein, Jacob Andreas, and Antonio Torralba. Toward a Visual Concept Vocabulary for GAN Latent Space. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 6804–6812, 2021. 2

[30] Ramprasaath R Selvaraju, Michael Cogswell, Abhishek Das, Ramakrishna Vedantam, Devi Parikh, and Dhruv Batra. Grad-cam: Visual explanations from deep networks via gradient-based localization. In *Proceedings of the IEEE international conference on computer vision*, pages 618–626, 2017. 6

[31] Yujun Shen, Jinjin Gu, Xiaoou Tang, and Bolei Zhou. Interpreting the Latent Space of GANs for Semantic Face Editing. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9243–9252, 2020. 2

[32] P Swathi and Saritha Sk. Deepfake creation and detection: A survey. In *2021 Third International Conference on Inventive Research in Computing Applications (ICIRCA)*, pages 584–588. IEEE, 2021. 1

[33] Ruben Tolosana, Ruben Vera-Rodriguez, Julian Fierrez, Aythami Morales, and Javier Ortega-Garcia. Deepfakes and beyond: A survey of face manipulation and fake detection. *arXiv preprint arXiv:2001.00179*, 2020. 1, 3

[34] Yusuke Uchida, Yuki Nagai, Shigeyuki Sakazawa, and Shin'ichi Satoh. Embedding watermarks into deep neural networks. In *Proceedings of the 2017 ACM on International Conference on Multimedia Retrieval*, pages 269–277, 2017. 3

[35] Luisa Verdoliva. Media forensics and deepfakes: an overview. *IEEE Journal of Selected Topics in Signal Processing*, 14(5):910–932, 2020. 1, 3

[36] Vedran Vukotić, Vivien Chappelier, and Teddy Furon. Are deep neural networks good for blind image watermarking? In *2018 IEEE International Workshop on Information Forensics and Security (WIFS)*, pages 1–7. IEEE, 2018. 3

[37] R. Wang, L. Ma, F. Juefei-Xu, X. Xie, J. Wang, and Y. Liu. Fakespotter: A simple baseline for spotting ai-synthesized fake faces. *arXiv preprint arXiv:1909.06122*, 2019. 3

[38] 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/CVF Conference on Computer Vision and Pattern Recognition*, pages 8695–8704, 2020. 3

[39] Ning Yu, Larry S Davis, and Mario Fritz. Attributing fake images to GANs: Learning and analyzing GAN fingerprints. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 7556–7566, 2019. 3

[40] Ning Yu, Vladislav Skripniuk, Dingfan Chen, Larry Davis, and Mario Fritz. Responsible disclosure of generative models using scalable fingerprinting. *arXiv preprint arXiv:2012.08726*, 2020. 3

[41] Baiwu Zhang, Jin Peng Zhou, Ilia Shumailov, and Nicolas Papernot. Not my deepfake: Towards plausible deniability for machine-generated media. *arXiv e-prints*, pages arXiv–2008, 2020. 3

[42] Jialong Zhang, Zhongshu Gu, Jiyong Jang, Hui Wu, Marc Ph Stoecklin, Heqing Huang, and Ian Molloy. Protecting intellectual property of deep neural networks with watermarking. In *Proceedings of the 2018 on Asia Conference on Computer and Communications Security*, pages 159–172, 2018. 3

[43] Ru Zhang, Shiqi Dong, and Jianyi Liu. Invisible steganography via generative adversarial networks. *Multimedia tools and applications*, 78(7):8559–8575, 2019. 3

[44] Xu Zhang, Svebor Karaman, and Shih-Fu Chang. Detecting and simulating artifacts in GAN fake images. In *2019 IEEE International Workshop on Information Forensics and Security (WIFS)*, pages 1–6. IEEE, 2019. 3