IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received May 5, 2022, accepted May 21, 2022, date of publication May 30, 2022, date of current version June 10, 2022.

Digital Object Identifier 10.1109/ACCESS.2022.3179116

Forensic Analysis of Synthetically Generated Western Blot Images

SARA MANDELLI[1], DAVIDE COZZOLINO[2], (Member, IEEE), EDOARDO D. CANNAS[1], (Member, IEEE), JOÃO P. CARDENUTO[3], DANIEL MOREIRA[4], PAOLO BESTAGINI[1], (Member, IEEE), WALTER J. SCHEIRER[4], (Senior Member, IEEE), ANDERSON ROCHA[3], (Senior Member, IEEE), LUISA VERDOLIVA[5], (Fellow, IEEE), STEFANO TUBARO[1], (Senior Member, IEEE), AND EDWARD J. DELP[6], (Life Fellow, IEEE)

[1]Dipartimento di Elettronica, Informazione e Bioingegneria, Politecnico di Milano, 20133 Milan, Italy
[2]Dipartimento di Ingegneria Elettrica e Tecnologie dell'Informazione, University Federico II of Naples, 80125 Naples, Italy
[3]Artificial Intelligence Laboratory, RECOD.ai, Institute of Computing, University of Campinas, Campinas 13083-970, Brazil
[4]Department of Computer Science and Engineering, University of Notre Dame, Notre Dame, IN 46556, USA
[5]Dipartimento di Ingegneria Industriale, University Federico II of Naples, 80125 Naples, Italy
[6]School of Electrical and Computer Engineering, Purdue University, West Lafayette, IN 47907, USA

Corresponding author: Sara Mandelli (sara.mandelli@polimi.it)

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. Hardware support was generously provided by NVIDIA corporation.

ABSTRACT The widespread diffusion of synthetically generated content is a serious threat that needs urgent countermeasures. As a matter of fact, the generation of synthetic content is not restricted to multimedia data like videos, photographs or audio sequences, but covers a significantly vast area that can include biological images as well, such as western blot and microscopic images. In this paper, we focus on the detection of synthetically generated western blot images. These images are largely explored in the biomedical literature and it has been already shown they can be easily counterfeited with few hopes to spot manipulations by visual inspection or by using standard forensics detectors. To overcome the absence of publicly available data for this task, we create a new dataset comprising more than 14K original western blot images and 24K synthetic western blot images, generated using four different state-of-the-art generation methods. We investigate different strategies to detect synthetic western blots, exploring binary classification methods as well as one-class detectors. In both scenarios, we never exploit synthetic western blot images at training stage. The achieved results show that synthetically generated western blot images can be spot with good accuracy, even though the exploited detectors are not optimized over synthetic versions of these scientific images. We also test the robustness of the developed detectors against post-processing operations commonly performed on scientific images, showing that we can be robust to JPEG compression and that some generative models are easily recognizable, despite the application of editing might alter the artifacts they leave.

INDEX TERMS Western blots, GANs, denoising diffusion probabilistic models, synthetically generated images, image forensics.

I. INTRODUCTION

Synthetically generated multimedia content has been flooding the web lately, catching people's attention mainly thanks to the entertainment and the artistic possibilities than can arise from these new technological advancements. State-of-the-art methods for synthetic content generation allow one to synthesize incredibly realistic images and audio sequences [1]–[5]. It is also possible to transfer the identity of a person [6], or even the body movements [7], from one video to another one. The majority of these innovative tools owe their birth to Generative Adversarial Networks (GANs) and probabilistic generative models, which are leading technologies for synthesizing multimedia data. All these tools usually present

The associate editor coordinating the review of this manuscript and approving it for publication was Turgay Celik.

easy-to-use free interfaces, such that any amateur without particular experience in digital arts can use them.

In spite of these evident new artistic opportunities, the vast production of synthetic content inevitably introduces serious threats related to data trustworthiness and integrity. Novel technologies can be maliciously exploited for data counterfeiting. This phenomenon is not only limited to digital multimedia content but it has been spreading worldwide over a significantly larger area, potentially including images reported in scientific publications [8]–[10].

In particular, western blot images are widely used in the biomedical literature concerning molecular biology and immunogenetics. They concern the analysis of proteins at a high sensitivity and precision level [11]. The scientific community started arguing about their authenticity since 2016, when the authors of [12] began to scan images from more than 20K scientific papers, eventually discovering an incredibly high manipulation rate (around 4%) with several duplicated or tampered with images.

Nowadays, the most common procedure to spot manipulations on western blot images is visual inspection. As a matter of fact, forensics techniques aiming at spotting local image tampering have a hard time in detecting manipulations applied to scientific images. This is often due to their reduced pixel resolution and the numerous amount of processing operations applied to create realistic forgeries [13].

The visual observation by an expert is still the most widespread approach, although requiring one important hypothesis: the investigated images are supposed not to be synthetic. The manipulated region is supposed to be derived from an already existing image, aptly processed to hinder tampering traces. If the western blot image under analysis has been synthetically generated, either in its entirety or locally only in specific pixel regions, there would be essentially no hope to spot such traces by visual inspection [10]. Indeed, during some preliminary experiments, the authors of [10] verified that standard image generation techniques based on GANs [14], [15] can synthesize almost indistinguishable western blots with respect to the real ones, even at the experts' eyes.

In this paper, we tackle the detection of western blot images which have been synthetically generated through GANs and probabilistic generative models. Our goal is to explore forensics methodologies to automatically classify synthetic and real western blots. We investigate how different forensics strategies developed for natural images perform over scientific images. In doing so, we simulate the realistic scenario in which synthetic versions of western blot images are not available to the analyst, who therefore cannot develop a forensic detector specifically tailored to them. We experiment with two main approaches:

1) a binary classification approach, borrowed from a recently proposed method for detecting synthetic versions of natural images [16]. This method relies upon a Convolutional Neural Network (CNN) purposely designed to tell real and synthetic images apart. In particular, we never train the detector on western blot images, thus testing its robustness on images of diverse nature such as western blots;
2) a one-class classification approach, in which we train a detector only on original western blots, looking for any anomalies or inconsistencies appearing in the synthetic images.

To compensate for the absence of a publicly available dataset of real and synthetic western blots, we create a new one comprising 14K real and 24K synthetic images, generated by means of three different GANs and one probabilistic generative model. To the best of our knowledge, the detection of synthetic images generated through probabilistic generative models has not been faced yet in the literature. Since these models have recently proved to synthesize images with high fidelity and diversity in a comparable manner to GANs, we propose a first insight on their detectability.

We extensively evaluate the proposed techniques, comparing various binary detectors and one-class detectors over the generated dataset. The achieved results demonstrate that the currently available strategies developed for natural images can be a valid option for identifying synthetic western blots.

Moreover, we test the detector robustness to common post-processing operations like image compression and resizing. We show that the proposed one-class classification approach can be robust to JPEG compression and can detect the synthetic images generated through some generative models almost independently from the post-processing applied.

To summarize, the main contributions of this paper are:

- We create a new dataset of western blot images including 14K real images and 24K synthetic images, generated by means of four different generative models including probabilistic models as well, whose detectability has never been investigated in the state-of-the-art.
- We investigate forensics strategies for the detection of synthetically generated western blots, proposing both binary and one class detection approaches that never exploit synthetic western blots at training stage.
- We explore the robustness of the proposed approaches in case the scientific images are post-processed with common editing operations.
- Results demonstrate the validity and generalization of the proposed methods, although additional research is needed to enhance robustness against standard processing operations and unseen generative models.

The rest of the paper is organized as follows. In Section II, we describe the generation process of synthetic western blots and present the created dataset. In Section III, we provide details on the proposed detection methods to distinguish real from synthetic western blots. In Section IV, we describe the experimental setup and discuss the obtained results. Eventually, in Section V, we draw our conclusions.

II. SYNTHETIC WESTERN BLOT GENERATION

In this section, we provide details about the generation process of synthetic western blot images. We start with a brief

description of the methods used for synthetic image generation, then we illustrate the original images employed as reference for the creation of synthetic samples. Eventually, we present the synthetic generation process and the generated dataset, providing some examples and highlighting the differences among the generation strategies.

A. ARCHITECTURES

To generate synthetic western blot images, we adopt well-known CNN architectures from the literature of natural images generation.

Three of the proposed CNNs belong to the family of GANs, which have been extensively used to generate synthetic images of human faces, animals and various objects. We first illustrate GANs dealing with the image-to-image translation problem. Among the various methods presented in the literature, we focus on Pix2pix [14] and CycleGAN [17] models, being two of the best performing and most widespread generation methods. We also consider style-based GANs [1], [2], [18], [19]; in particular, we employ the StyleGAN2 with Adaptive Discriminator Augmentation (StyleGAN2-ADA), one of the newest and most promising models [1] for the task.

The last considered technique is based on probabilistic generative models, which have been recently proposed as an alternative to GANs for creating synthetic data with high-fidelity [20]–[23]. In particular, we select the Denoising Diffusion Probabilistic Model (DDPM) proposed in [23].

1) IMAGE-TO-IMAGE TRANSLATION GANs

Image-to-image translation GANs cover the vast area of generative networks which learn a mapping between two image categories and translate one category into the other one. To perform image-to-image translation, we need to train GAN architectures with multiple images selected from the two distinct groups.

a: Pix2pix

Pix2pix [14] is an image-to-image translation GAN inspired by conditional adversarial networks. It follows the typical paradigm of image-to-image translation models as it requires a training set of aligned image pairs in which it exists a correspondence between two images of distinct categories. For instance, an aligned image pair could be composed by a color image and its grayscale version, or by an edge-map and the corresponding photograph. Specifically, Pix2pix exploits a conditional GAN which conditions on an input image and generates an output translated image [14].

b: CycleGAN

CycleGAN [17] is a particular class of image-to-image translation GANs defined as unpaired image-to-image translation model. Since finding paired training data is not always possible and it can be difficult and expensive [17], CycleGAN is trained to translate between images of distinct domains without exploiting aligned image pairs.

The main feature of CycleGAN is its ''cycle-consistency'' property which translates an input image to an output meaningful synthetic image belonging to a different category [17].

2) STYLE-BASED GANs

Style-based GANs were born in 2019 as an alternative to traditional generation models [24]. The generator of StyleGAN [18] introduces a mapping of the latent code into an intermediate latent code, which is transformed to different ''styles'' that control the layers of the synthesis network. The proposed architecture has been further improved with the StyleGAN2 [19], StyleGAN2-ADA [1] and StyleGAN3 [2] models, which remove undesired blob artifacts and enable achieving outstanding synthesis quality by training only on few samples. The main difference between image-to-image translation GANs and style-based ones lies on the input data to be provided for training and for synthesizing new images. If the former needs pairs of input images selected from two distinct categories for training and one image category for synthesizing, the latter requires images of a single category for training and synthesizes new images with the same style starting from the latent code provided to the generator.

3) PROBABILISTIC GENERATIVE MODELS

Probabilistic generative models are a class of generative models which sequentially disturb training data with slowly increasing noise, and then learn to reverse this corruption in order to build a generative model of the original clean data [22]. Among them, DDPMs [20], [21], [23], [25] were introduced in 2015 by [25]. They model the ''noising'' data process as a forward diffusion process which gradually converts any complex data distribution into a simple and tractable noise distribution [25]. Then, they learn the backward process (i.e., to pass from noise distribution to data distribution) which allows to generate new synthetic data. In the last few years, DDPMs have proved to generate data with high fidelity and diversity, often being comparable or outperforming state-of-the-art GANs.

B. ORIGINAL IMAGES

We collected almost 300 original RGB images of different resolutions depicting multiple western blots. Every western blot image may contain different bands, which can have multiple shapes. The final shape depends on the operations done by the biologist who processed the protein, on the protein itself and on the properties of the processing apparatus [11]. The images usually present irregularities like spots, scratches and bubbles. All these ingredients make every western blot image almost unique, and also the single bands contained inside can have small variations among them [11].

Our dataset includes 284 original images downloaded from the web or selected from scientific publications. Since all images present small size (i.e., usually less than 256 pixels on the smallest dimension), we resize them, keeping the aspect ratio of the initial image, such that the minimum dimension

is always equal to 256 pixels. A few examples of the original western blot images are depicted in Figure 1.

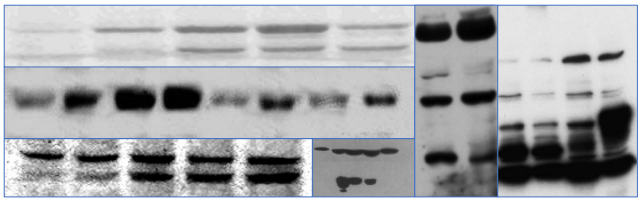

FIGURE 1. Examples of original western blot images selected from the collected dataset.

C. SYNTHETIC IMAGES

We start showing how to generate synthetic images with GANs, then we present probabilistic generative models. Eventually, we illustrate the final generated dataset that is used in our experiments.

1) IMAGE-TO-IMAGE TRANSLATION MODELS

We propose to generate synthetic western blots by feeding image-to-image translational GANs with images selected from the following two categories:

- original western blot images;
- images containing information on the position of western blot bands inside the original images.

In particular, samples belonging to this last category have the same resolution of the original images they refer to, but consist of binary values being 0 in pixels corresponding to a detected blot band and 1 elsewhere. We refer to these images as blot-masks. Given an original western blot image **I**, it is related in one-to-one correspondence to its blot-mask **M**. For example, Figure 2 depicts the blot-masks corresponding to the original images shown in Figure 1. We build the blot-masks through a semi-automatic segmentation process. For each image, we exploit Otsu's image thresholding [26] and Watershed segmentation [27] algorithms to automatically obtain possible blot-masks associated with the image, then we pick the best mask by visual inspection.

FIGURE 2. Examples of blot-masks corresponding to the original western blot images shown in Figure 1.

a: Pix2pix

We generate synthetic western blots by training Pix2pix with images belonging to the previously reported two classes. Pix2pix requires these images to be aligned one with respect to the other. In other words, each original image and the related blot-mask should be included within the same input pair. The network is trained to learn the mapping between the position of western blots (i.e., information carried by blot-masks) and their related representation (i.e., information carried by original images). As Pix2pix requires squared input images, we randomly extract 50 squared patches with size 256×256 from each original image and an equal amount of squared patches from the related blot-mask. For illustration's sake, Figure 3 draws a sketch of the training setup required by Pix2pix.

In the generation phase, we provide as input only binary masks according to the desired western blot location. Pix2pix generates new synthetic images containing western blot bands in these positions.

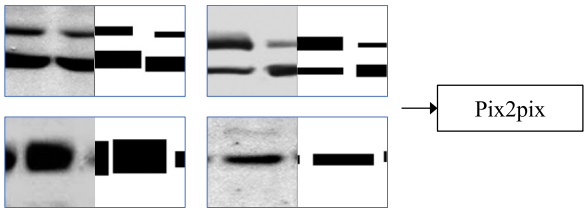

FIGURE 3. Sketch of the training setup required by Pix2pix model. To train the network, we need paired input images of squared size, i.e., one original image and the related blot-mask.

b: CycleGAN

To generate synthetic western blots with CycleGAN, we propose to feed it with the same images exploited for training the Pix2pix model. However, we can remove the alignment constraint and train with unpaired images. As Pix2pix, CycleGAN also requires squared input images, therefore we use the same squared patches extracted for training Pix2pix. Figure 4 depicts a sketch of the training setup required by CycleGAN. Notice the relaxation of alignment constraint with respect to the Pix2pix model reported in Figure 3.

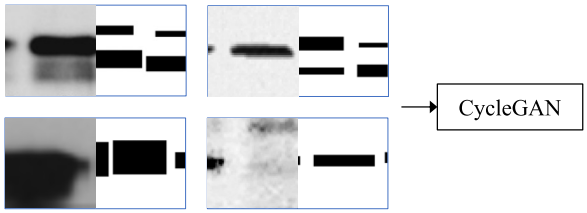

FIGURE 4. Sketch of the training setup required by CycleGAN model. To train the network, we need unpaired input images of squared size, i.e., one original image and one blot-mask.

2) STYLE-BASED GENERATION MODELS

Among the style-based generative models, we exploit StyleGAN2-ADA, which has proved to generate highly realistic images and needs less samples to be trained with respect to StyleGAN2 and StyleGAN3. Differently from image-to-image translation models, we can feed the network with

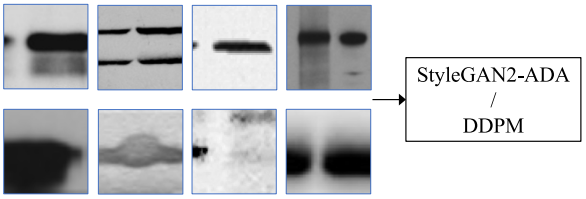

FIGURE 5. Sketch of the training setup required by StyleGAN2-ADA and DDPM models. To train the networks, we need input original images of squared size.

single input squared patches. During training, the network learns to generate new images with the same style of the training dataset. Figure 5 depicts a sketch of the training setup required by StyleGAN2-ADA.

In the generation phase, we can provide different seeds to the synthesis network, each one corresponding to a new synthetic western blot image.

In the generation phase, we provide as input only binary masks according to the desired western blot location. Pix2pix generates new synthetic images containing western blot bands in these positions.

3) PROBABILISTIC GENERATIVE MODELS

We select the DDPM proposed in [23], which recently improved the generation performance of diffusion models both in terms of data fidelity and diversity. As performed for StyleGAN2-ADA, we directly feed the generative model with squared input patches from the training data (see Figure 5). We use these images to implement a diffusion-based noising process and then learn how to reverse it. In generation phase, samples from the noisy distribution can be randomly selected. Starting from them, DDPM is able to gradually remove the noise and return new synthetic western blots.

4) FINAL DATASET

The final dataset that we use to evaluate our experimental setup consists of original and synthetic squared images with a common size of 256×256 pixels.

The original samples are derived from the data described in SectionII-B. Specifically, we randomly extract 50 squared patches per original western blot image. We end up with 14, 200 real images with size 256×256 pixels.

The synthetic samples include:

- 6, 000 squared images with size 256×256 generated by the Pix2pix model, providing as input to the generator the same blot-masks seen in training phase;
- 6, 000 squared images with size 256×256 generated by the CycleGAN model, providing as input to the generator the same blot-masks seen in training phase;
- 6, 000 squared images with size 256×256 generated by the StyleGAN2-ADA model, providing as input to the generator different seeds for each new image to be synthesized.
- 6, 000 squared images with size 256×256 generated by DDPM, providing as input to the generator different noisy samples, corresponding to an equal number of new images to be synthesized.

Figures 6-9 depict a few examples of synthetic western blot images generated by the four proposed models. If we provide the same blot-mask to Pix2pix and CycleGAN, the generated western blot varies according to the generation model. Nonetheless, in both situations, the synthetic images are plausible and realistic. In case of StyleGAN2-ADA and DDPM, the generated samples present high quality and photo-realism. The complete dataset is available at https://www.dropbox.com/sh/nl3txxfovy97b1k/AABqbgkGBEfjS6pjke3a-d7a?dl=0.

III. SYNTHETIC WESTERN BLOT DETECTION

In this section, we present the investigated methods for synthetic western blot detection. Given a query image, we investigate two kinds of classification setups: (i) a binary setup, in which we train a binary classifier on both original and synthetic natural images; (ii) a one-class setup, in which we train a one-class classifier only on the original western blot dataset. We always consider the challenging scenario in which the synthetic dataset of western blots is never seen during the detectors' training phase. In the binary classification framework, the training dataset does not even include the original western blot images, but only natural images. In the one-class detector configuration, we only see a reduced subset of the original western blot images during training.

A. BINARY DETECTION

We investigate the challenging scenario in which we never see western blots, pristine nor synthetic, during the training phase. We consider the realistic situation in which we have available some binary classifiers trained to distinguish original from synthetic images, which however do not belong to the western blot image category. For instance, we may have available binary classifiers trained to detect original and synthetic versions of human faces, animals or objects.

To this purpose, we borrow some of the GAN-image detectors recently proposed in [16], which performs a critical state-of-the-art analysis of the GAN-image detection task. The backbone architecture is a ResNet50, modified to avoid the down-sampling in the first network layer as suggested by [28]. In [16], this architecture modification proves to be robust to compression and resizing operations performed on the testing image dataset.

At deployment stage, each classification is associated with a positive score for the images belonging to the synthetic category and a negative score for the original category.

B. ONE-CLASS DETECTION

In this scenario, we remove the possibility to train the detector over synthetic images of any category, i.e., we consider training only on original images. We propose to train a one-class

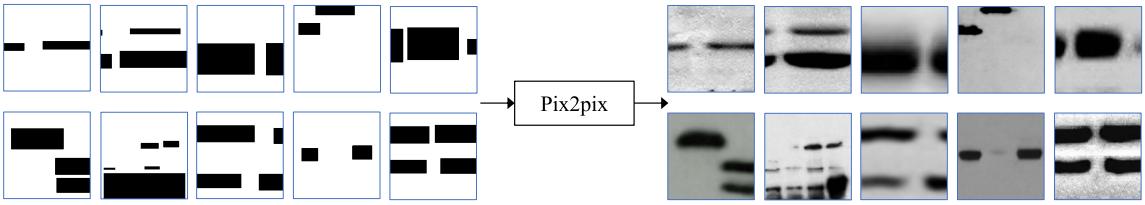

FIGURE 6. Sketch of the synthesis setup required by Pix2pix model. To generate new synthetic western blots, we provide the desired blot-masks to the generator.

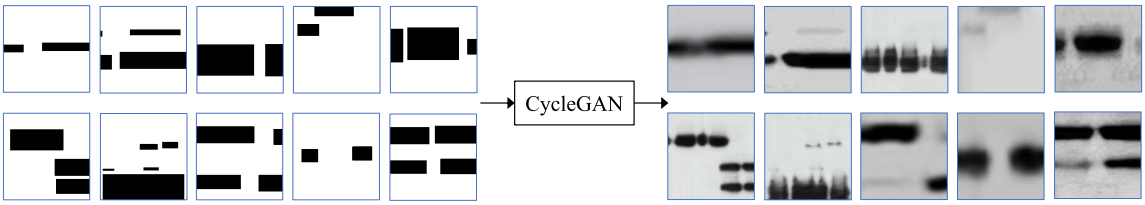

FIGURE 7. Sketch of the synthesis setup required by CycleGAN model. To generate new synthetic western blots, we provide the desired blot-masks to the generator.

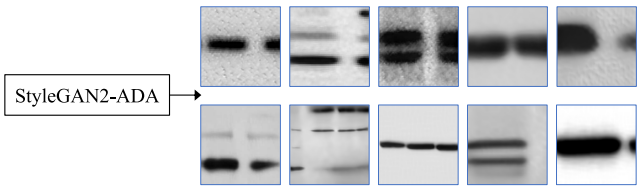

FIGURE 8. Sketch of the synthesis setup required by StyleGAN2-ADA model. To generate new synthetic western blots, we provide different seeds to the generator.

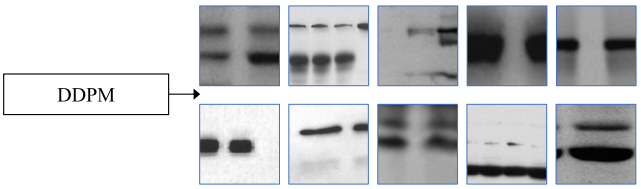

FIGURE 9. Sketch of the synthesis setup required by DDPM. To generate new synthetic western blots, we can randomly select noisy samples and gradually remove the noise from them to return new synthetic western blots.

classifier over a reduced set of the original western blot images.

To describe the texture characteristics of the training images, we propose to extract some features that will be fed to the classifier. Following a common state-of-the-art procedure [29], [30], we convert each color image in grayscale and apply high-pass filtering by subtracting a low-pass version of the grayscale image to itself. The low-pass filter is a 3×3 spatial kernel defined as

$$\mathbf{H} = \frac{1}{4}\begin{bmatrix} 0 & 1 & 0 \\ 1 & 0 & 1 \\ 0 & 1 & 0 \end{bmatrix}. \tag{1}$$

We report some examples of high-pass filtered images in Figure 10. At visual inspection, there are not significant traces to tell real (top row) and synthetic images (bottom row) apart.

Then, we convert the pixels' values to 8-bit unsigned integers and we compute the gray level co-occurrence matrix, a 2D matrix which reports a histogram of co-occurring grayscale values at a given offset over the input image. Indeed, the co-occurrence matrix has been widely exploited in the forensics literature both for binary and one-class detection tasks. For instance, co-occurrences have been used for spotting subtle differences (usually not visible at human inspection) in the textural features of real versus manipulated natural images (through splicing) [30] and of real versus synthetic natural images [31], [32].

We can define the co-occurrence matrix as $\mathbf{C}$, with size 256×256. Every element $[\mathbf{C}]_{ij}$, with $i, j \in [0, \ldots, 255]$, corresponds to the number of times the gray-level j occurs at a certain distance from the gray-level i, along a certain direction. To compute the co-occurrence matrix $\mathbf{C}$, we investigate four different distances d for the gray-levels' comparison, testing $d = \{4, 8, 16, 32\}$, along both the horizontal and vertical directions. We normalize each co-occurrence matrix $\mathbf{C}$ by the sum of its elements, defining $\bar{\mathbf{C}}$ as

$$\bar{\mathbf{C}} = \frac{\mathbf{C}}{\sum_{i,j=0}^{255} \mathbf{C}_{ij}}. \tag{2}$$

To motivate our choice, we randomly select 25 original western blots and 25 synthetic ones and investigate the behavior of co-occurrences on this reduced dataset. We apply gray-scale conversion and high-pass filtering for each image and compute the co-occurrence matrix corresponding to any possible gray-level distance and direction. Then, we aggregate the results through pixel-wise arithmetic mean, ending with an average co-occurrence matrix per image, with a size 256×256. To focus on the main differences between original

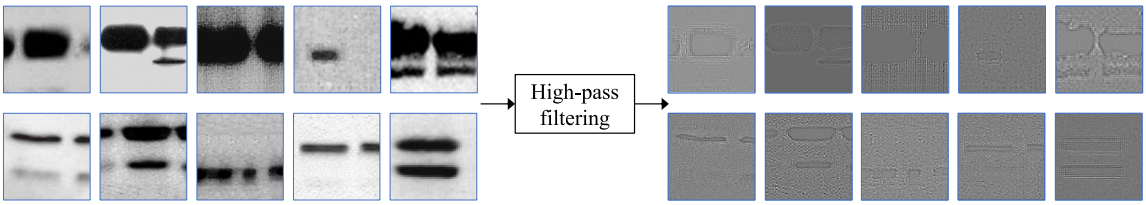

FIGURE 10. High pass filtered versions of the original western blot images (top row) and the synthetic western blot images (bottom row).

and synthetic samples, we propose to compute the Principal Component Analysis (PCA) of their average co-occurrences. We can extract compact descriptors of original and synthetic samples to evaluate their differences visually.

In more details, we extract the PCA of the co-occurrence matrices of the original images. Then, we compute the projection on the extracted principal components for both original and synthetic samples. Figure 11 reports the PCA projections by considering 15, 10, 5, 3, 2 and 1 principal components. The 25 top rows correspond to the original samples, while the last 25 rows correspond to the synthetic samples. The synthetic images exhibit quite different projections concerning original images, which motivates the use of co-occurrences as good descriptors to separate the two classes. Considering that we are extracting a high-pass filtered version of the western blots, Figure 11 confirms the findings reported in [33], [34], i.e., that original natural images show a more homogeneous behavior in the high-frequency components concerning synthetically generated ones.

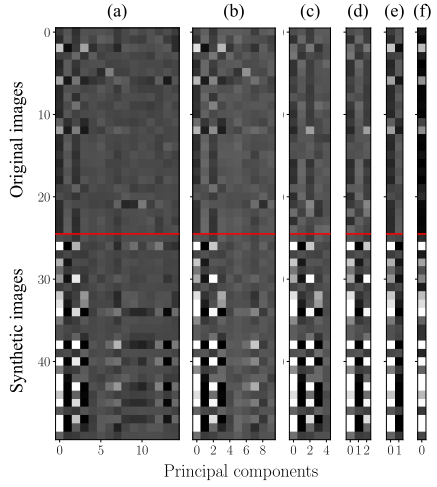

FIGURE 11. PCA projections of the co-occurrences extracted from 25 original and 25 synthetic western blot images. The number of extracted components equals to: (a) 15; (b) 10; (c) 5; (d) 3; (e) 2; (f) 1.

Given these premises, we process the co-occurrence matrices to extract several texture properties that enable to distinguish real from synthetic samples. In particular, we explore 5 different processing methods that extract one scalar feature from every image:

- Contrast-weighted feature: given $\bar{\mathbf{C}}$, we weight each element by the squared difference of its coordinates, and we sum over all the matrix elements. We define the contrast-weighted feature as

$$f_c = \sum_{i,j=0}^{255} \bar{\mathbf{C}}_{ij} \cdot (i-j)^2. \tag{3}$$

- Homogeneity-weighted feature: given $\bar{\mathbf{C}}$, we divide each element by the squared difference of its coordinates shifted by 1, and we sum over all the matrix elements. We define the homogeneity-weighted feature as

$$f_h = \sum_{i,j=0}^{255} \frac{\bar{\mathbf{C}}_{ij}}{1+(i-j)^2}. \tag{4}$$

- Dissimilarity-weighted feature: given $\bar{\mathbf{C}}$, we weight each element by the absolute difference of its coordinates, and we sum over all the matrix elements. We define the dissimilarity-weighted feature as

$$f_d = \sum_{i,j=0}^{255} \bar{\mathbf{C}}_{ij} \cdot |i-j|. \tag{5}$$

- Energy feature: given $\bar{\mathbf{C}}$, we compute the square root of its energy, defining the energy-related processing feature as

$$f_e = \sqrt{\sum_{i,j=0}^{255} \bar{\mathbf{C}}_{ij}^2}. \tag{6}$$

- Correlation-weighted feature: given $\bar{\mathbf{C}}$, we weight each element by a cross-correlation measure of its coordinates, and we sum over all the matrix elements. Precisely, we compute the correlation-weighted feature as

$$f_\rho = \sum_{i,j=0}^{255} \bar{\mathbf{C}}_{ij} \cdot \mathbf{R}_{ij}, \tag{7}$$

where $\mathbf{R}$ is a square matrix with the same size of $\bar{\mathbf{C}}$, which emulates a normalized cross-correlation between row and column coordinates, weighted by the matrix $\bar{\mathbf{C}}$. For the sake of clarity, we define $\mathbf{R}$ as

$$[\mathbf{R}]_{ij} = \frac{(i-\mu_i)(j-\mu_j)}{\sigma_i \cdot \sigma_j}, \tag{8}$$

where

$$\mu_i = \sum_{i,j=0}^{255} \bar{\mathbf{C}}_{ij} \cdot i, \quad \mu_j = \sum_{i,j=0}^{255} \bar{\mathbf{C}}_{ij} \cdot j \tag{9}$$

and

$$\sigma_i^2 = \sum_{i,j=0}^{255} \bar{\mathbf{C}}_{ij} \cdot (i-\mu_i)^2, \quad \sigma_j^2 = \sum_{i,j=0}^{255} \bar{\mathbf{C}}_{ij} \cdot (j-\mu_j)^2. \tag{10}$$

Considering that we extract 5 textural features (i.e., f_c, f_h, f_d, f_e and f_ρ) for each co-occurrence matrix version, we finally end up with 40 different features per query image.

We propose to feed every single feature to a one-class classifier, investigating both the well-known and widely exploited One Class Support Vector Machine (OCSVM) [35] that we consider as a baseline reference and the more recent Isolation Forest (IF) [36], which has been proposed as efficient strategy for anomaly detection. Both algorithms are trained for detecting outlier samples which are not distributed as the original training data.

At deployment stage, each classification is associated with a positive score for the images belonging to the training category (i.e., original images) and a negative score for outlier images (i.e., synthetically generated images).

IV. RESULTS

In this section, we report the experimental setup and the achieved results in the detection of synthetically generated western blot images. First, we report the performance of binary detection methods, then we show the results achieved by the one-class detector approach.

A. EXPERIMENTAL SETUP

1) BINARY DETECTION

We train three binary detectors by following the findings reported in [16]. In the first detector, the modified ResNet50 is trained over the training image dataset provided in [37], comprising 362K real images extracted from the LSUN dataset [38] and 362K generated images obtained by 20 ProGAN [24] models, each trained on a different LSUN object category. In the second detector, the modified ResNet50 is trained using 720K StyleGAN2 images and 552K real images selected from different public datasets, i.e., the LSUN [38], the AFHQ [39], the AnimalWeb [40], the BreCaHAD [41], the FFHQ [42] and the MetFaces [43]. The synthetic images were generated by training StyleGAN2 with real images selected from these datasets. In the third detector, we explore the situation in which the modified ResNet50 is trained to distinguish all the considered real images versus both ProGAN and StyleGAN2 synthetic images. It is worth noticing that none of the considered detectors exploits western blot images, real nor synthetic, during training.

2) ONE-CLASS DETECTION

The IF detector is trained by setting the number of samples to train each IF embedded estimator equal to the maximum possible one, i.e., the total number of training images. The remaining detectors' parameters are those suggested in [44].

We train the two proposed one-class detectors over the features extracted from half of the available real images depicting western blots. To avoid possible bias in evaluating the results, we split the pristine dataset of patches according to the original western blot images they have been extracted from, as described in Section II-B. In doing so, all the patches extracted from the same original image belong to the same dataset split. We end up with 142 training western blot images, corresponding to 7100 original patches with resolution 256×256 pixels.

Since the one-class approach is trained over original western blots, this inevitably reduces the number of original images to include in the test set. Therefore, to have a fair comparison with the binary detection results, evaluated over the final dataset and thus including all the original western blot patches, we apply a 2-fold cross-validation approach: we exploit each of the two dataset splits once as training set and once as testing set. Then, we average the achieved results.

B. BINARY DETECTION RESULTS

Binary classification results are shown in Tables 1 and 2. We always report results by keeping separated the images synthesized through the four investigated generation methods, thus the real images are compared four times against a different synthetic dataset. Table 1 depicts the achieved AUC of the Receiver Operating Characteristic (ROC) curve built for the binary classification task, while Table 2 reports the achieved balanced accuracy in correctly classifying real and synthetic images. Notice that we are including also the

TABLE 1. Area Under the Curve (AUC) achieved by different binary classifiers. The synthetic dataset used to train each detector is shown in between square brackets. Best results in bold.

	[37] [ProGAN]	[16] [ProGAN]	[16] [StyleGAN2]	[16] [ProGAN, StyleGAN2]
Pix2pix	0.633	0.901	0.948	**0.959**
CycleGAN	0.504	0.782	0.760	**0.921**
StyleGAN2-ADA	0.779	0.872	0.899	**0.915**
DDPM	0.809	0.909	**0.961**	0.949

TABLE 2. Balanced accuracy achieved by different binary classifiers. The synthetic dataset used to train each detector is shown in between square brackets. Best results in bold.

	[37] [ProGAN]	[16] [ProGAN]	[16] [StyleGAN2]	[16] [ProGAN, StyleGAN2]
Pix2pix	71.24%	81.38%	85.92%	**92.58%**
CycleGAN	66.50%	72.43%	72.58%	**84.55%**
StyleGAN2-ADA	79.42%	84.16%	86.15%	**88.93%**
DDPM	78.86%	82.80%	**94.69%**	93.91%

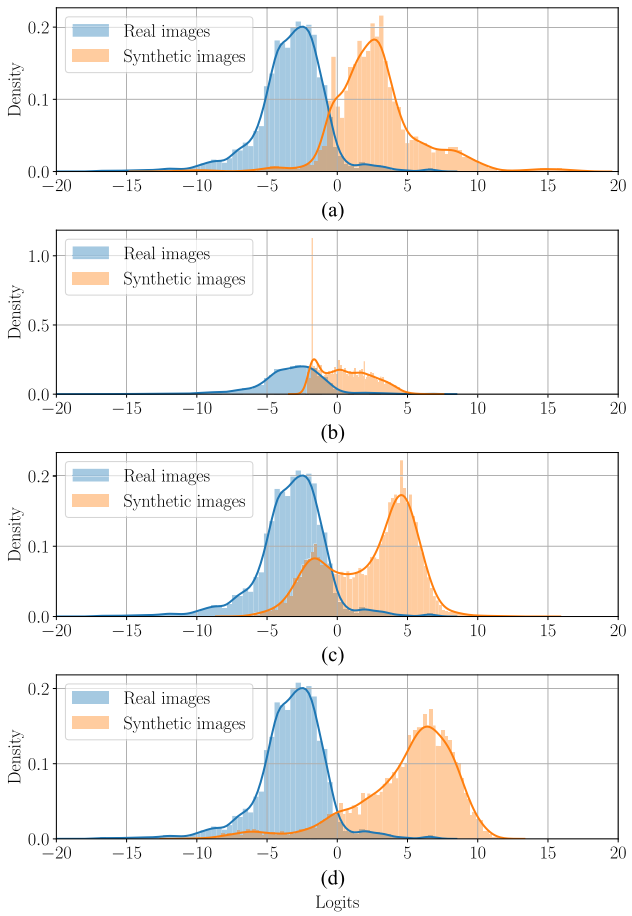

FIGURE 12. Distribution of the logit scores produced by the binary detector proposed in [16] when trained on ProGAN and StyleGAN2 images. In particular, (a) corresponds to real versus Pix2pix synthetic western blot images; (b) to real versus CycleGAN synthetic western blot images; (c) to real versus StyleGAN2-ADA western blot synthetic images; (d) to real versus DDPM western blot synthetic images.

classification results achieved by the state-of-the-art GAN detector proposed in [37].

The best detector almost always consists in the one proposed by [16] in the last configuration, i.e., trained on both ProGAN and StyleGAN2 synthetic images. This result confirms the experiments performed in the original paper [16]: the bigger the training dataset, the better the generalization capability of the detector. Overall, Pix2pix and DDPM synthetic images are the most detectable ones. For Pix2pix, this might be expected, being the Pix2pix generation method the oldest of the four and reasonably introducing generation artifacts that might be easier to be spot. DDPM images, despite their recentness and their high-quality realism, still present more generation artifacts than current state-of-the-art GANs. Evaluations on the CycleGan and StyleGAN2-ADA datasets achieve similar AUCs, however the results on CycleGAN samples report an accuracy of more than 4 percentage points below the one reached on StyleGAN2-ADA western blot images.

TABLE 3. Best AUC achieved by OCSVM classifier when trained on the 5 proposed features extracted from the real images. Best results in bold.

	f_c	f_h	f_d	f_e	f_ρ
Pix2pix	0.543	0.776	0.639	**0.886**	0.829
CycleGAN	0.586	0.770	0.637	0.816	**0.955**
StyleGAN2-ADA	0.731	0.578	0.772	0.677	**0.894**
DDPM	0.788	**0.976**	0.853	0.955	0.743

TABLE 4. Best AUC achieved by IF classifier when trained on the 5 proposed features extracted from the real images. Best results in bold.

	f_c	f_h	f_d	f_e	f_ρ
Pix2pix	0.626	0.805	0.620	**0.902**	0.835
CycleGAN	0.577	0.781	0.645	0.829	**0.947**
StyleGAN2-ADA	0.734	0.578	0.761	0.704	**0.885**
DDPM	0.781	**0.998**	0.810	0.993	0.799

We investigate this behaviour in Figure 12, which depicts the distribution of the logit scores achieved by the best detector in case of synthetic images generated through Pix2pix, CycleGAN, StyleGAN2-ADA and DDPM, respectively. A significant amount of CycleGAN images is associated with a negative logit score, especially for the scores ≈ -1.8. This phenomenon is much more reduced for Pix2pix, StyleGAN2-ADA and DDPM synthetic images. When computing the AUC, the negative CycleGAN scores do not cause a strong impact on the performances, as the ROC curve is built by considering all the possible thresholds related to the binary decision problem. The balanced accuracy, instead, is computed by thresholding the logit scores with a fixed threshold equal to 0. This fixed thresholding inevitably assigns the wrong label to a great amount of synthetic images, thus lowering the detection performances.

C. ONE-CLASS DETECTION RESULTS

1) SINGLE FEATURE ANALYSIS

As reported in Section IV-A, we investigate 40 different features per query image, which correspond to an equal number of classification scores per image for each detector. For brevity's sake, we report only the best classification results for each of the 5 proposed processing features, i.e., $f_c, f_h, f_d, f_e, f_\rho$. In reporting results, we follow the same approach employed for binary classification, that is, we separately evaluate our performances on the four datasets of synthetic western blot images. Tables 3 and 4 show the best achieved AUC on each selected feature by exploiting OCSVM and IF, respectively.

The features related to the contrast and dissimilarity never report the best results. For image-to-image translational models, the energy and correlation features are often the most discriminative ones, while on the StyleGAN2-ADA generated samples the AUCs approaches 0.9 only for the correlation feature. Over DDPM images, we achieve excellent results with f_h and f_e.

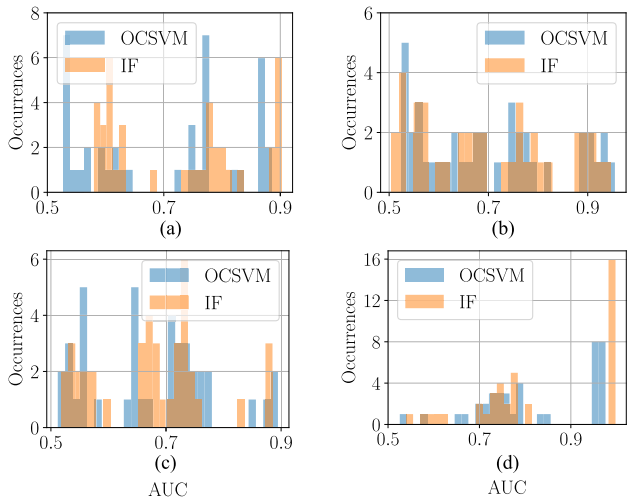

FIGURE 13. Histogram of the achieved AUCs associated with the extracted 40 feature. In particular, (a) real images versus Pix2pix; (b) real images versus CycleGAN; (c) real images versus StyleGAN2-ADA; (d) real images versus DDPM.

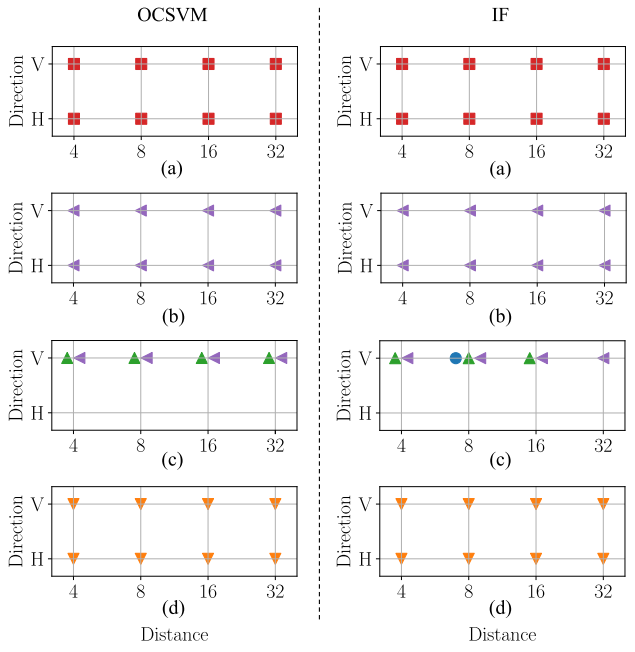

FIGURE 14. Parameters related to the 8 best features for each one-class detector. On the left, the parameters related to OCSVM, on the right those related to IF. In particular, (a) corresponds to real versus Pix2pix synthetic western blot images; (b) to real versus CycleGAN synthetic western blot images; (c) to real versus StyleGAN2-ADA synthetic western blot images; (d) to real versus DDPM synthetic western blot images. The blue ● corresponds to the feature f_c, the orange ▼ to f_h, the green ▲ to f_d, the red ■ to f_e, the purple ◀ to f_ρ.

We further investigate how performances vary according to the exploited features in Figure 13, where we report the histogram of the achieved AUCs considering all the 40 investigated features. It is noticeable that, for any GAN-based generation method, there are few features which allow to achieve high AUCs. For Pix2pix and CycleGAN images, only 8 features achieve AUCs greater than 0.85, while for StyleGAN2-ADA samples only 4 features exceed 0.8 of AUC. DDPM images seem the easiest to be detected, and count 16 features that allow to reach AUCs values above 0.9.

To provide insight on the nature of these features, Figure 14 investigates which are the parameters characterizing the best 8 features for each detector, i.e., which are the selected gray-level distance (i.e., 4, 8, 16 or 32), the direction of computation (horizontal (H) or vertical (V)) and the kind of textural metrics used (i.e., f_c, f_h, f_d, f_e or f_ρ) providing the best performances. The best features characterizing the OCSVM detector are the same of IF, except for the images generated through StyleGAN2-ADA, in which OCSVM and IF differ for only one feature. From Figure 14(a)-(b)-(d) we can notice that Pix2pix, CycleGAN and DDPM images can be differentiated by one single metric, f_e, f_h and f_ρ, respectively, and any combination of gray-level distance and direction achieves acceptable results. StyleGAN2-ADA images (see Figure 14(c)) present stronger artifacts along the vertical direction: none of the 8 best AUCs is found over the horizontal direction. Moreover, both f_d and f_ρ report acceptable results, even though f_ρ demonstrates to be more accurate, as reported in Tables 3 and 4.

TABLE 5. Best AUC achieved by OCSVM classifier when trained on one single feature, on combinations of the 8 best features and on the features proposed in [29]. Best results in bold.

	1-feat	Comb-2	Comb-3	Comb-4	[29]
Pix2pix	**0.886**	0.885	0.884	0.883	0.748
CycleGAN	0.955	**0.961**	**0.961**	0.958	0.848
StyleGAN2-ADA	0.894	**0.904**	**0.904**	**0.904**	0.533
DDPM	**0.976**	0.973	0.972	0.971	0.944

TABLE 6. Best AUC achieved by IF classifier when trained on one single feature, on combinations of the 8 best features and on the features proposed in [29]. Best results in bold.

	1-feat	Comb-2	Comb-3	Comb-4	[29]
Pix2pix	0.903	0.917	0.922	**0.923**	0.711
CycleGAN	0.947	0.953	**0.954**	0.952	0.706
StyleGAN2-ADA	0.891	**0.902**	0.900	0.901	0.601
DDPM	**0.999**	0.998	0.998	0.998	0.923

2) COMBINED FEATURE ANALYSIS

We also explore the scenario in which the proposed one-class detectors are trained not over a single feature per image but over a combination of multiple features. At deployment stage, we extract the feature combination from the query image, and we feed them to the detectors. For the sake of brevity, for each detector and for each generative method, we investigate only the combinations among the features returning the best 8 AUC values, i.e., the features described in Figure 14. Thus, we explore 3 different scenarios: (i) training on the combina-

TABLE 7. Best balanced accuracy achieved by OCSVM classifier when trained on one single feature, on combinations of the 8 best features and on the features proposed in [29]. Best results in bold.

	1-feat	Comb-2	Comb-3	Comb-4	[29]
Pix2pix	71.56%	**73.78%**	73.74%	73.53%	71.87%
CycleGAN	74.45%	**75.43%**	74.78%	74.56%	71.06%
StyleGAN2-ADA	70.24%	71.93%	73.10%	**73.29%**	52.61%
DDPM	73.97%	76.24%	**76.30%**	75.71%	72.07%

TABLE 8. Best balanced accuracy achieved by IF classifier when trained on one single feature, on combinations of the 8 best features and on the features proposed in [29]. Best results in bold.

	1-feat	Comb-2	Comb-3	Comb-4	[29]
Pix2pix	86.35%	88.26%	88.80%	**88.93%**	50.85%
CycleGAN	**88.99%**	88.19%	87.80%	87.31%	55.24%
StyleGAN2-ADA	**84.69%**	82.74%	81.50%	80.25%	50.04%
DDPM	90.37%	91.92%	92.36%	**92.65%**	91.67%

tion of two features; (ii) training on the combination of three features; (iii) training on the combination of four features. We investigate all the 28 possible combinations for the first scenario; all the 56 combinations for the second one and all the 70 for the third one. We depict the best achieved AUC by OCSVM and IF in Tables 5 and 6, respectively. In this scenario, we also show the best achieved balanced accuracy by OCSVM and IF in Tables 7 and 8, respectively.

It is worth noticing that selecting combinations of multiple features may improve the results, but does not bring a significant boost to the performances. Indeed, combining more features may lead to worse results. In more details, the performance change in exploiting more than one feature does not always represent an improvement and, whenever results are improved, the gain is reduced to a maximum of $+2.22\%$ in the AUC (see Table 6, first row) and to a maximum of $+4.34\%$ in the balanced accuracy (see Table 7, third row). Moreover, in the worst scenarios, exploiting four features can lead to -0.51% of performance loss in the AUC (see Table 5, last row) and to -5.24% in the balanced accuracy (see Table 8, third row). On average, exploiting more than one feature returns an AUC gain of 0.5% and a balanced accuracy gain of 1.2%. Thus, the choice to train the classifiers on more than one features might not be the preferred option because it is computationally and temporarily more expensive than the single feature scenario.

For a further comparison with a standard feature extraction procedure followed in the literature [30]–[32], we also extract the co-occurrence based local features proposed in [29]. We train the one-class detectors on these features extracted from the training images; in testing phase, for each query image, we feed these features to the detectors. In order to provide a clear comparison with the proposed methodology, we report the achieved AUC in Tables 5 and 6, while we report the achieved balanced accuracy in Tables 7 and 8. In none of the considered scenarios the features of [29] outperform the proposed methodology.

TABLE 9. Best results achieved by the binary and one-class classifiers, in terms of AUC and balanced accuracy. Best detector in bold.

	Binary AUC	One-class AUC	Binary Accuracy	One-class Accuracy
Pix2pix	**0.959**	0.923	**92.58%**	88.93%
CycleGAN	0.921	**0.961**	84.55%	**88.99%**
StyleGAN2-ADA	**0.915**	0.904	**88.93%**	84.69%
DDPM	0.961	**0.999**	**94.69%**	92.65%

3) ONE-CLASS DETECTOR COMPARISON

In general, we achieve the best results by means of the IF classifier. When comparing AUCs of the two detectors (see Tables 5 and 6), OCSVM reports accurate and comparable performances with respect to IF. On the contrary, the achieved accuracy by OCSVM is significantly lower than IF's (see Tables 7 and 8). This discrepancy in the reported AUC and accuracy can be explained with the same considerations done in Section IV-B. The IF detector demonstrates to be more stable and less prone to errors when exploiting a fixed thresholding strategy, i.e., selecting a threshold equal to 0 to discriminate images when solving the binary decision problem.

D. BINARY VS ONE-CLASS RESULTS

For clarity's sake, we summarize the best results of the binary and the one-class classification approaches in Table 9. Interestingly, the one-class approach outperforms the binary one on the CycleGAN and DDPM datasets, even if on DDPM only in terms of achieved AUC. In this scenario, learning textural properties of real western blot images brings a significant improvement with respect to a binary classifier trained on real and synthetic natural images not depicting western blots. As a matter of fact, in all the considered situations, the one-class classifier reports valid and comparable results to those achieved by the binary one, considering that it is trained only on original western blot images, never looking at synthetic data.

E. ROBUSTNESS TO POST-PROCESSING OPERATIONS

As last experiment, we investigate scenarios where western blots underwent some post-processing operations. In doing so, we simulate realistic situations in which images to be included in a scientific publication might be resized and/or compressed due to limited resources dedicated to the manuscript in terms of maximum number of pages, Byte count, etc. The applied post-processing also simulates operations that might be done by malicious users who tampered with or generated completely synthetic versions of scientific images. Indeed, to create realistic forgeries, it is common to apply various post-processing on the modified images to conceal the tampering traces.

TABLE 10. Best AUC achieved by the binary classifier on post-processed and non post-processed images. The classifier training is performed on non post-processed images.

	No proc.	Upscale 1.25	Upscale 1.5	Down-Upscale 0.5	Down-Upscale 0.75	Down-Upscale 0.9	JPEG-80	JPEG-90	JPEG-100
Pix2pix	0.959	0.684	0.682	0.503	0.565	0.614	0.687	0.717	0.737
CycleGAN	0.921	0.814	0.858	0.613	0.679	0.794	0.512	0.667	0.793
StyleGAN2-ADA	0.915	0.698	0.710	0.640	0.745	0.731	0.722	0.727	0.775
DDPM	0.961	0.528	0.424	0.201	0.383	0.465	0.564	0.589	0.574

TABLE 11. Best AUC achieved by the one-class classifiers when trained on one single feature. The classifier training is performed on post-processed images. For clarity's sake, we also report the best AUC achieved on non post-processed images.

	No proc.	Upscale 1.25	Upscale 1.5	Down-Upscale 0.5	Down-Upscale 0.75	Down-Upscale 0.9	JPEG-80	JPEG-90	JPEG-100
Pix2pix	0.923	0.646	0.638	0.634	0.651	0.630	0.565	0.591	0.874
CycleGAN	0.961	0.749	0.756	0.663	0.705	0.809	0.769	0.841	0.941
StyleGAN2-ADA	0.904	0.625	0.615	0.632	0.621	0.685	0.839	0.879	0.909
DDPM	0.999	0.997	0.996	0.991	0.998	0.998	0.932	0.998	0.998

TABLE 12. Best AUC achieved by the one-class classifiers when trained on one single feature. The classifier training is performed on original non post-processed images. For clarity's sake, we also report the best AUC achieved on non post-processed images.

	No proc.	Upscale 1.25	Upscale 1.5	Down-Upscale 0.5	Down-Upscale 0.75	Down-Upscale 0.9	JPEG-80	JPEG-90	JPEG-100
Pix2pix	0.923	0.710	0.711	0.659	0.684	0.645	0.563	0.580	0.847
CycleGAN	0.961	0.737	0.745	0.759	0.759	0.828	0.778	0.837	0.931
StyleGAN2-ADA	0.904	0.625	0.626	0.719	0.700	0.728	0.839	0.877	0.901
DDPM	0.999	0.966	0.943	0.977	0.993	0.995	0.941	0.977	0.999

In this vein, we investigate three kinds of post-processing that might undermine the performance of our synthetic image detectors:

- an upscaling post-processing, in which the images are enlarged by factors 1.25 and 1.5, and then randomly cropped to fit the 256 × 256 pixel resolution;
- a down-upscaling post-processing, in which images are downscaled by factors 0.5, 0.75 and 0.9, and then upscaled back to fit their original resolution of 256 × 256 pixels;
- a JPEG compression with different quality factors (i.e., 80, 90 and 100) corresponding to increasing image visual quality.

We resort to Albumentation [45] as data augmentation library.

Table 10 reports the achieved AUCs in classifying the post-processed images with the binary detector proposed in [16] trained on ProGAN and StyleGAN2 images. We pick this binary detector as it reports the best results in the experimental analysis on non post-processed images. In training phase, we do not apply any post-processing to the images. Unfortunately, the binary detector reports a consistent performance loss in almost all scenarios, especially on Pix2pix and DDPM images.

For the one-class detection, we investigate two possible training scenarios corresponding to realistic situations that forensics analysts commonly deal with:

- training on the *post-processed* images. In this scenario, all data underwent some known editing operation and we have available a portion of them for the training phase;
- training on the original *non post-processed* images. In this scenario, we miss information about the potential editing operations applied on testing data.

In both scenarios, we follow the same 2-fold cross-validation approach reported in Section IV-A to have fair comparisons with the binary detection approach.

Tables 11 and 12 report the achieved AUCs in the first and second training scenarios, respectively. Notice that the average performances of the two scenarios are similar. By training on the original *non post-processed* images, i.e., simulating an agnostic scenario from the post-processing view point, we achieve almost the same results of the perfect knowledge situation.

Contrarily to the binary detection approach, the one class classifier is significantly more robust against the JPEG-based post-processing. As long as the compression quality factor does not excessively reduce the image quality, the performance loss can be contained around 12%, with the only exception of the Pix2pix synthetic images.

It is also worth noticing that DDPM synthetic images can be spotted with high accuracy level independently of the kind of post-processing applied, almost approaching the results achieved in the experiments executed on the non post-processed dataset.

V. CONCLUSION

In this paper, we performed a forensics analysis of synthetically generated western blot images. Previous works have already shown that western blots can be tampered with or totally synthesized in a relatively easy way, with expert inspectors having a hard time in spotting forgeries.

We were not able to find in the literature a sufficiently vast dataset of original and synthetic western blot images to perform scientific experiments. Therefore, we created a new dataset containing more than 14K original and 24K fully synthetic western blots, generated through four state-of-the-art generation methods based on GANs and DDPMs.

Regarding the detection, we investigated the realistic scenario in which the analyst does not have available any synthetic versions of western blot images. To do so, we explored how forensics detectors purposely developed for binary classification of real versus synthetic natural images perform in distinguishing original and synthetic western blots. We also explored one-class classification approaches, in which we learned textural feature properties of original western blots and looked for any anomalies occurring in the synthetic data.

We extensively evaluated the proposed detectors on the collected dataset. Our results showed that synthetic western blots can be distinguished from real ones with a high accuracy in all the considered experimental scenarios. This is noteworthy, considering that we never exploited synthetic western blot images to optimize the detectors. Up to now, forensics detectors trained only on natural images or on original western blots represent valid solutions to identify fully synthetic versions of them.

Our experiments also highlighted that the one-class approach is robust to JPEG compression applied to the images to be analyzed, even if the training is performed on original non compressed images. Robustness to resizing operations is still a challenging issue for spotting the majority of synthetic images. Nonetheless, the one-class approach can easily detect the synthetic images generated through DDPMs, no matter the resizing applied.

Future work will include additional generative models, particularly adapting the binary detection approach for the attribution of the model used to create synthetic samples and investigating open-set recognition techniques to identify images generated with methods never seen during training. Furthermore, we will focus on improving detectors' robustness to common post-processing operations to make them more suitable for in-the-wild scenarios where an unknown processing chain might hinder traces of the synthetic generation process.

ACKNOWLEDGMENT

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. Hardware support was generously provided by NVIDIA Corporation.

REFERENCES

[1] T. Karras, M. Aittala, J. Hellsten, S. Laine, J. Lehtinen, and T. Aila, "Training generative adversarial networks with limited data," in *Proc. NeurIPS*, 2020, pp. 12104–12114.

[2] T. Karras, M. Aittala, S. Laine, E. Härkönen, J. Hellsten, J. Lehtinen, and T. Aila, "Alias-free generative adversarial networks," in *Proc. NeurIPS*, 2021, pp. 1–12.

[3] C. Donahue, J. McAuley, and M. Puckette, "Adversarial audio synthesis," in *Proc. ICLR*, 2019, pp. 1–16.

[4] J. Shen, R. Pang, R. J. Weiss, M. Schuster, N. Jaitly, Z. Yang, Z. Chen, Y. Zhang, Y. Wang, R. Skerrv-Ryan, R. A. Saurous, Y. Agiomvrgiannakis, and Y. Wu, "Natural TTS synthesis by conditioning wavenet on MEL spectrogram predictions," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process. (ICASSP)*, Apr. 2018, pp. 4779–4783.

[5] K. J. Shih, R. Valle, R. Badlani, A. Lancucki, W. Ping, and B. Catanzaro, "RAD-TTS: Parallel flow-based TTS with robust alignment learning and diverse synthesis," in *Proc. ICML Workshop Invertible Neural Netw., Normalizing Flows, Explicit Likelihood Models*, 2021, pp. 1–8.

[6] J. Thies, M. Zollhöfer, and M. Nießner, "Deferred neural rendering: Image synthesis using neural textures," *ACM Trans. Graph.*, vol. 38, no. 4, pp. 1–12, 2019.

[7] C. Chan, S. Ginosar, T. Zhou, and A. Efros, "Everybody dance now," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2019, pp. 5933–5942.

[8] C. Fu, S. Lee, D. J. Ho, S. Han, P. Salama, K. W. Dunn, and E. J. Delp, "Three dimensional fluorescence microscopy image synthesis and segmentation," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. Workshops (CVPRW)*, Jun. 2018, pp. 2221–2229.

[9] A. Ghorbani, V. Natarajan, D. Coz, and Y. Liu, "Dermgan: Synthetic generation of clinical skin images with pathology," in *Proc. Mach. Learn. Health Workshop*, 2020, pp. 155–170.

[10] C. Qi, J. Zhang, and P. Luo, "Emerging concern of scientific fraud: Deep learning and image manipulation," *bioRxiv*, 2021, doi: 10.1101/2020.11.24.395319.

[11] *Science Integrity Digest: Western Blots*. Accessed: Feb. 1, 2022. [Online]. Available: https://scienceintegritydigest.com/2020/02/21/western-blots/

[12] E. M. Bik, A. Casadevall, and F. C. Fang, "The prevalence of inappropriate image duplication in biomedical research publications," *mBio*, vol. 7, no. 3, Jul. 2016, Art. no. e00809.

[13] E. Sabir, S. Nandi, W. Abdalmageed, and P. Natarajan, "BioFors: A large biomedical image forensics dataset," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2021, pp. 10963–10973.

[14] P. Isola, J.-Y. Zhu, T. Zhou, and A. A. Efros, "Image-to-image translation with conditional adversarial networks," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 1125–1134.

[15] T. Park, M.-Y. Liu, T.-C. Wang, and J.-Y. Zhu, "Semantic image synthesis with spatially-adaptive normalization," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 2337–2346.

[16] 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 *Proc. IEEE Int. Conf. Multimedia Expo (ICME)*, Jul. 2021, pp. 1–6.

[17] J.-Y. Zhu, T. Park, P. Isola, and A. A. Efros, "Unpaired image-to-image translation using cycle-consistent adversarial networks," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Oct. 2017, pp. 2223–2232.

[18] T. Karras, S. Laine, and T. Aila, "A style-based generator architecture for generative adversarial networks," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 4401–4410.

[19] T. Karras, S. Laine, M. Aittala, J. Hellsten, J. Lehtinen, and T. Aila, "Analyzing and improving the image quality of StyleGAN," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2020, pp. 8110–8119.

[20] J. Ho, A. Jain, and P. Abbeel, "Denoising diffusion probabilistic models," in *Advances in Neural Information Processing Systems*, vol. 33, H. Larochelle, M. Ranzato, R. Hadsell, M. F. Balcan, and H. Lin, Eds. Red Hook, NY, USA: Curran Associates, 2020, pp. 6840–6851. [Online]. Available: https://proceedings.neurips.cc/paper/2020/file/4c5bcfec8584af0d967f1ab10179ca4b-Paper.pdf

[21] A. Q. Nichol and P. Dhariwal, "Improved denoising diffusion probabilistic models," in *Proc. 38th Int. Conf. Mach. Learn.*, vol. 139, M. Meila and T. Zhang, Eds., Jul. 2021, pp. 8162–8171. [Online]. Available: https://proceedings.mlr.press/v139/nichol21a.html
[22] Y. Song, J. Sohl-Dickstein, D. P. Kingma, A. Kumar, S. Ermon, and B. Poole, "Score-based generative modeling through stochastic differential equations," in *Proc. Int. Conf. Learn. Represent.*, 2021. [Online]. Available: https://openreview.net/forum?id=PxTIG12RRHS
[23] P. Dhariwal and A. Q. Nichol, "Diffusion models beat GANs on image synthesis," in *Proc. 35th Conf. Neural Inf. Process. Syst. (NeurIPS)*, Sydney, NSW, Australia, 2021, pp. 1–15.
[24] T. Karras, T. Aila, S. Laine, and J. Lehtinen, "Progressive growing of GANs for improved quality, stability, and variation," in *Proc. Int. Conf. Learn. Represent.*, 2018.
[25] J. Sohl-Dickstein, E. Weiss, N. Maheswaranathan, and S. Ganguli, "Deep unsupervised learning using nonequilibrium thermodynamics," in *Proc. Int. Conf. Mach. Learn.*, 2015, pp. 2256–2265.
[26] N. Otsu, "A threshold selection method from gray-level histograms," *IEEE Trans. Syst., Man, Cybern., Syst.*, vol. SMC-9, no. 1, pp. 62–66, Feb. 1979.
[27] L. Vincent and P. Soille, "Watersheds in digital spaces: An efficient algorithm based on immersion simulations," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 13, no. 6, pp. 583–598, Jun. 1991.
[28] M. Boroumand, M. Chen, and J. Fridrich, "Deep residual network for steganalysis of digital images," *IEEE Trans. Inf. Forensics Security*, vol. 14, no. 5, pp. 1181–1193, May 2019.
[29] J. Fridrich and J. Kodovský, "Rich models for steganalysis of digital images," *IEEE Trans. Inf. Forensics Security*, vol. 7, no. 3, pp. 868–882, Jun. 2012.
[30] D. Cozzolino, G. Poggi, and L. Verdoliva, "Splicebuster: A new blind image splicing detector," in *Proc. IEEE Int. Workshop Inf. Forensics Secur. (WIFS)*, Nov. 2015, pp. 1–6.
[31] L. Nataraj, T. M. Mohammed, B. Manjunath, S. Chandrasekaran, A. Flenner, J. H. Bappy, and A. K. Roy-Chowdhury, "Detecting GAN generated fake images using co-occurrence matrices," *Electron. Imag.*, vol. 2019, no. 5, p. 532, 2019.
[32] M. Barni, K. Kallas, E. Nowroozi, and B. Tondi, "CNN detection of GAN-generated face images based on cross-band co-occurrences analysis," in *Proc. IEEE Int. Workshop Inf. Forensics Secur. (WIFS)*, Dec. 2020, pp. 1–6.
[33] R. Durall, M. Keuper, and J. Keuper, "Watch your up-convolution: CNN based generative deep neural networks are failing to reproduce spectral distributions," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2020, pp. 7890–7899.
[34] J. Frank, T. Eisenhofer, L. Schönherr, A. Fischer, D. Kolossa, and T. Holz, "Leveraging frequency analysis for deep fake image recognition," in *Proc. Int. Conf. Mach. Learn.*, 2020, pp. 3247–3258.
[35] B. Schölkopf, J. C. Platt, J. Shawe-Taylor, A. J. Smola, and R. C. Williamson, "Estimating the support of a high-dimensional distribution," *Neural Comput.*, vol. 13, no. 7, pp. 1443–1471, Jul. 2014.
[36] F. T. Liu, K. M. Ting, and Z.-H. Zhou, "Isolation forest," in *Proc. 8th IEEE Int. Conf. Data Mining*, Dec. 2008, pp. 413–422.
[37] S.-Y. Wang, O. Wang, R. Zhang, A. Owens, and A. A. Efros, "CNN-generated images are surprisingly easy to spot... for now," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2020, pp. 8695–8704.
[38] F. Yu, A. Seff, Y. Zhang, S. Song, T. Funkhouser, and J. Xiao, "LSUN: Construction of a large-scale image dataset using deep learning with humans in the loop," 2015, *arXiv:1506.03365*.
[39] Y. Choi, Y. Uh, J. Yoo, and J.-W. Ha, "StarGAN v2: Diverse image synthesis for multiple domains," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2020, pp. 8188–8197.
[40] M. H. Khan, J. McDonagh, S. Khan, M. Shahabuddin, A. Arora, F. S. Khan, L. Shao, and G. Tzimiropoulos, "AnimalWeb: A large-scale hierarchical dataset of annotated animal faces," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2020, pp. 6939–6948.
[41] A. Aksac, D. J. Demetrick, T. Ozyer, and R. Alhajj, "BreCaHAD: A dataset for breast cancer histopathological annotation and diagnosis," *BMC Res. Notes*, vol. 12, no. 1, pp. 1–3, Dec. 2019.
[42] *Flickr-Faces-HQ Dataset (FFHQ)*. Accessed: Feb. 1, 2022. [Online]. Available: https://github.com/NVlabs/ffhq-dataset
[43] *Metfaces Dataset*. Accessed: Feb. 1, 2022. [Online]. Available: https://github.com/NVlabs/metfaces-dataset
[44] *Scikit-Learn: Novelty and Outlier Detection*. Accessed: Feb. 1, 2022. [Online]. Available: https://scikit-learn.org/stable/modules/outlier_detection.html#outlier-detection
[45] A. Buslaev, V. I. Iglovikov, E. Khvedchenya, A. Parinov, M. Druzhinin, and A. A. Kalinin, "Albumentations: Fast and flexible image augmentations," *Information*, vol. 11, no. 2, p. 125, Feb. 2020.