On Attacks (Dis)Similarities to Test Adversarial Defense: Can We Reduce the Attack Set?*

Tommaso Puccetti[1,*], Tommaso Zoppi[2] and Andrea Ceccarelli[3]

[1] *Department of Mathematics and Informatics, University of Florence, Viale Morgagni 67/a, 50134 Firenze (FI), Italy.*
[2] *Department of Engineering and Information Science, University of Trento, Via Sommarive 9, 38124, Trento, Italy.*

Abstract

Evaluating defensive solutions against adversarial evasion attacks means quantifying the defense's capability to detect or tolerate attacks. Ideally, a defense should be tested against all the possible attacks: however, this is not achievable, and it is necessary to identify a representative attack set for the evaluation. Unfortunately, how to select such an attack set is an open question. Arguably, the selected attacks should apply diverse effects on the original image, in terms of dimension and distribution of the perturbation. We propose to quantify the perturbation through Image Quality metrics in addition to L-norms, such that adversarial attacks can be grouped (and only one representative of the group can be selected to test the defense) if they i) similarly perturb the attacked image, and ii) have similar success rate and detectability rate. Disappointingly, the analysis reveals that attacks with similar image perturbation cannot be related. Substantial evidence discourages grouping attacks and suggests that any reduction of the attack set impacts the validity of the defense evaluation.

Keywords

Evasion attacks, adversarial attacks, attack categories, image quality metrics, distance metrics.

1. Introduction

Deep Neural Networks (DNNs) that perform image classification are vulnerable to adversarial samples, which are deliberately crafted by perturbing legitimate input (e.g., images, texts, tabular data) to mislead the target model [1, 7]. In this paper, we focus on evasion attacks only, aimed at producing altered samples to fail the classification outcome of a trained classifier. The two main approaches to defending against adversarial samples are i) increase the robustness of the image classifier [7], or ii) detect adversarial samples before they are fed to the target classifier [36, 23, 24]. In both cases, a proper evaluation of the defense requires quantifying its ability to protect against adversarial attacks.

Ideally, the attack set used for the evaluation should include all the possible attacks against the target classifier, and the defense should be evaluated against each of these attacks. This is unfeasible because of both computational effort and the potential occurrence of unknown attacks. On top of this, without a rigorous methodology for the evaluation, many of the attacks

ITASEC 2024: The Italian Conference on CyberSecurity, April 08–12, 2024, Salerno, IT
* Corresponding author.
tommaso.puccetti@unifi.it (T. Puccetti); tommaso.zoppi@unitn.it (T. Zoppi); andrea.ceccarelli@unifi.it (A. Ceccarelli)
0000-0002-0297-2108 (T. Puccetti); 0000-0001-9820-6047 (T. Zoppi); 0000-0002-2291-2428 (A. Ceccarelli)

and defenses in the literature have been evaluated with custom approaches that are difficult to compare [1, 2, 3, 4, 5], such that in some cases it may lead to lack of clarity in the efficacy of newly proposed solutions.

The overarching challenge in evaluating defenses lies in the choice of the attack set to be considered: the attack set should deliver a realistic representation of the attack space that allows defining precise boundaries of the defense and, consequently, details the defense's coverage against different attacks. The state of the art agrees that i) there is no individual attack that sufficiently covers the entire attack space, and ii) there is no evidence that a defense able to tolerate certain attacks (i.e., a defined portion of the attack space) will also be able to tolerate others [6]. Unavoidably, the typical approach to select a representative attack set is to include multiple attacks, selected based on i) attack configuration parameters, ii) success rate, and iii) values of the L_0, L_1, L_2, and L_∞ norms (simply termed L-norms in the rest of the paper) [5, 36, 23]. Unfortunately, this approach is still exposed to known pitfalls. Sharing the attack generation parameters guarantees the reproducibility of the experiments but provides limited information about the perturbation. Instead, L-norms quantify the difference between the original image and the attacked counterpart: they are considered a viable proxy to measure the perturbation crafted by the attack and to craft attacks that are not perceived by the human eye. However, the L-norms alone may not suffice in quantifying the diversity of attacks, i.e., a defense may detect attacks of type A and fail to detect attacks of type B, even if both attacks have similar L-norms values. Moreover, the success and detection rates are usually linked with the L-norms, but this relation is weak and varies from attack to attack.

To summarize, different attacks may have similar effects on the target classifier, and similar values of L-norms, but the defense may decide differently. The same applies to attacks that produce different image perturbations. Additionally, grouping attacks based on their mathematical formulation is not effective: small differences in the formulation may have a relevant impact on the perturbation applied to images and on the success rate.

Position statement and approach. We believe that the number of attacks to be used when testing a defense can be reduced only by finding similarities in the perturbation introduced in the adversarial images, together with evidence that such similarities will have similar effects on the classifier and the defense. This paper investigates whether quantifying the perturbation through distance metrics in addition to L-norms allows partitioning adversarial attacks into classes, which could then be used to craft representative attack sets. In other words, we investigate whether measuring perturbations using distance metrics allows for the identification of characteristics of the attacks that permit to discriminate between attack families.

We identify alternative distance metrics to extrapolate different information from the adversarial image. In particular, the Image Quality Assessment (IQA) domain particularly fits our necessities: IQA aims at quantitatively evaluating the quality of images modified by a variety of distortions (e.g., processing, compression, etc.) to exploit the Human Visual System (HVS) model for low-level perception [8, 9]. Unlike the L-norms, image quality metrics account for the position of the pixels inside the image rather than operating on a single-dimension vector. We combine L-norms and image quality metrics to quantify perturbations, a novel approach in this context. We group attacks, where attacks belong to the same group if they have similar values of distance metrics. More precisely, we collocate attacks in the same group if we can show that attacks generate similar perturbations of the target image (measured with

the distance metrics), and consequently have a similar success rate against the target classifier and are detected or mitigated with a similar efficacy by one or more defenses.

We identify groups through an experimental evaluation. First, we generate 222 attack sets from 12 different attacks targeting two state-of-the-art models, namely ConvNet12 [33] and ResNet50 [52]. Then, we apply each of the attacks to the first 100 images of the CIFAR-10 [25] dataset, generating, for each model, a total of 22 200 adversarial images. For each adversarial image, we compute the selected distance metrics. Then, we perform three separate analyses on the images of both models: i) clustering analysis to group the 12 attacks into the same cluster(s) if they have similar distance metric values; ii) regression analysis to predict the success rate of an attack using distance metrics as input features; iii) binary classification analysis to predict the detection of an attack using as input features the distance metrics, using 2 detectors from the state of the art.

The results indicate that employing distance metrics to quantify perturbations exhibits some discriminatory capabilities, as it effectively clusters a small number of the selected attacks with highly similar mathematical formulations. However, these results, while insightful, do not demonstrate a high level of generalizability. Some attacks with shared mathematical formulations are correctly grouped, while others with similar formulations are not consistently recognized as a distinct group. Our experiments reveal that numerous attacks remain ungrouped, lacking observable signatures or defining characteristics in their perturbations. Consequently, we argue that reducing the number of attacks for testing a defense compromises the validity of the results.

2. Background

2.1. Adversarial Evasion Attacks Exercised in Our Study

Adversarial attacks consist of deliberately manipulating the input to a DNN to cause wrong predictions [27]. In this paper, we focus on evasion attacks against image classifiers, because i) they have large applications in the real world [30], and ii) image classification is the typical application domain for such attacks [28, 29]. Evasion attacks can be broadly classified based on the knowledge of the attacker about the target classifier: white box attacks imply some knowledge of the classifier architecture, while black box attacks approximate such information. Evasion attacks can be grouped into finer categories, based on the specific implementation details needed to run the attack: i) *gradient-based*, ii) *score-based*, and iii) *decision-based*. Gradient-based attack methods exploit the gradient of the loss of the target classifier during the input processing. The score-based attacks solely on the output score of the target classifier. Decision-based attacks are the most indicated for a black box setting as they need only prediction labels [31]. We select white box and black box attacks from the three categories, to create a heterogeneous set of 12 attacks.

Gradient-Based Attacks. One of the most effective gradient-based attacks is Carlini and Wagner (CW2, [37]). It is formulated as the following optimization problem:

$$x' = argmin_x\{||x - x_0||^2 + cg(x)\} \qquad (1)$$

The first term enforces perturbation on the original image, the second term is the loss function of the model, and g(x) is g(x)=$max\{f(x)_{y_0} - max_{i \neq y_0} f(x)_i, 0\}$ where y_0 is the

ground truth label of the input x_0, is the score computed on the input x for the y_0 label, and $f(x)_i$ is the score returned predicting x with label i.

Instead, Elastic Net (ELA, [38, 39]) formulates the generation of adversarial attacks as an elastic-net regularized optimization problem:

$$x' = \ argmin_x \left\{ ||x - x_0||_2^2 + \beta |x - x_0|_1 \ + cg(x) \right\} \quad (2)$$

where $g(x)$ is the same as the CW2 attack.

Projected Gradient Descent (PGD, [41]) solves the optimization problem using the projected gradient descent. Practically, it finds adversarial images by adding or subtracting a small error to each dimension of the input based on the gradient sign. Fast Gradient Sign Method (FGM, [40]) can be viewed as a one-step PGD attack, while the Basic Iterative Method (BIM, [42]) is the iterative form of the attack. Deep Fool (DEEP, [43]), performs an iterative linearization of the classifier to calculate the closest decision boundary for a genuine image x. Jacobian-based Saliency Map (JSMA, [44]), generates an adversarial saliency map, which helps identify the input features to be included in the perturbation. Newton Fool Attack (NEW, [45]) operates by executing a gradient descent, which reduces the likelihood of the initial class.

Score-Based Attacks. We select the Zeroth Order Optimization-Based Attack (ZOO, [46]). The attack is a black box, and it uses a finite difference method to estimate the gradient sign of the loss function w.r.t the input image. The gradient estimation is used to run the CW2 attack and craft the final adversarial image.

Decision-Based Attack. We select from this category the Boundary Attack (BOUND, [47]) based on the random walk on the decision boundary of the input image. The method starts with a sample categorized in the target class and searches for a minimum amount of perturbation to keep the image adversarial. Also, we select the HopSkipJump Attack (HOP, [48]), which can be formulated as solving a zeroth order optimization problem [49].

2.2. Distance Metrics Selected in Our Study

Modifying the template — including but not limited to adjusting margins, typeface sizes, line spacing, paragraph and list definitions — is not allowed. At the state of the art, the amount of perturbation introduced by an attacker is quantified using the pixel L_p norms, for any

$$p>0: |v|_p = \left(\sum_{i=1}^{N} |v_i|^p \right)^{\frac{1}{p}} \quad (3)$$

where $v = ||x-x'||$ is the perturbation introduced in an image x to obtain its adversarial counterpart x', and $v_i=||x_i-x_i'||$ is the difference pixel by pixel of the two images. Each L-norm has a different mathematical meaning and, therefore, captures different characteristics of the perturbation [50]. L_0 distance measures the number of coordinates i such that $x_i \neq x_i'$. The L_0 distance is equal to the number of pixels that have been altered in an image. L_1 distance, also known as Manhattan Distance, is the sum of the absolute difference between pixels of two images. L_2 distance, also known as Euclidean distance, is the squared root of the sum of the squared absolute difference between pixels of two images. The L_2 distance can remain small when there are many small changes to many pixels. L_∞ distance is the largest absolute difference between pixels of two images. It measures the maximum change to any of the coordinates: $||x - x'||_\infty = \max(|x_i-x_i'|, \ldots, |x_n-x_n'|)$. For images, we can think there is a maximum budget, and each

pixel is allowed to be changed up to this limit, with no limit on the number of pixels that are modified.

We also use distance metrics from the image quality domain. L-norms work very well to quantify the wideness and intensity of the perturbation but may not be sufficient to capture the finer characteristics of the perturbation. We select the following metrics.

Mean Squared Error (MSE) calculates the cumulative squared error between the original image and the distorted image:

$$MSE = \frac{1}{MN} \sum_{y=0}^{M-1} \sum_{x=0}^{N-1} [E(x,y)]^2 \qquad (4)$$

where x and y provide the pixel position, M and N are the image width and height, and E(x, y) = I_o(x, y) − I_p(x, y) is the difference pixel by pixel of the original image I_o and the perturbed image I_p [8]. Although it does not correlate well with the perceived image quality [12, 13], we select the MSE because it is used as the basis for many of the HVS-based metrics.

Block Sensitive - Peak Signal-to-Noise Ratio (PSNR-B, [13]) can be seen as an advanced version of the Peak Signal-to-Noise Ratio (PSNR [8]) that includes a blocking effect factor (BEF) specifically used for measuring the quality of images that present blocking artifacts. As reported in [14], BEF is calculated by considering horizontal and vertical neighboring pixel pairs that do not lie across block boundaries. The blocking effect factor specifically measures the amount of blocking artifacts of the image. The mean square error including the blocking effects MSE-B for a reference image x and test image y is defined as MSE-B(x, y)=MSE(x, y)+BEF_{Tot}(y), where $BEF_{Tot}(y)$ is the BEF computed over all block sizes. At last, PSNR-B is obtained as PNSR-B(x, y)=10 $\log_{10}$ (255^2 /MSE-B(x, y)).

Universal Quality Image Index (UQI, [15]) calculates the amount of transformation of relevant data from the reference image into the perturbed image. UQI is defined as:

$$\mathrm{UQI} = \frac{4\sigma_{xx'} \quad \bar{x}\bar{x}'}{(\sigma_x^2 + \sigma_{x'}^2 \quad)[(\bar{x})^2 + (\bar{x}')^2]} \qquad (5)$$

where $\bar{x}$ and $\bar{x}'$ are respectively the mean values of the original and perturbed images, σ_x^2 and $\sigma_{x'}^2$ are the variances and $\sigma_{xx'}$ is the covariance. The range of this metric is -1 to 1, where 1 indicates that the reference and perturbed images are similar [8].

Erreur Relative Globale Adimensionnelle Synthèse (ERGAS, [16]) measures the global radiometric distortion between two images; it calculates the average error of each band of the perturbed image for the reference one. As reported in [17], high values of ERGAS indicate low quality of the perturbed image, while lower values indicate good quality. The ERGAS is given as:

$$\mathrm{ERGAS} = 100 \frac{h}{w} \sqrt{\frac{1}{n} \sum_{k=1}^{n} \left(\frac{RMSE(B_k)^2}{mean(k)^2} \right)} \qquad (6)$$

where h and w are the height and width of the image, the $RMSE(B_k)$ is the Root Mean Squared Error for k-band computed between the original and altered image, and $mean(k)^2$ denotes the mean k-band of the original image. Relative Average Spectral Error (RASE, [21]) determines the difference in spectral information between each band of the merged image and the original image. Given M the mean radiance of the N spectral bands B_i of the original image, and the root mean square error RMSE, RASE is computed as:

$$\text{RASE} = \frac{1}{M}\sqrt{\frac{1}{N}\sum_{i=1}^{N} RMSE^2(B_i)} \quad (7)$$

Spectral Angle Mapper (SAM, [22]) computes the spectral angle between the pixel vector of the reference image and of the perturbed image. It is performed on a pixel-by-pixel basis. A value of SAM equal to zero denotes the absence of spectral distortion. In the following expression, x is the original image, and x' is the perturbed image:

$$\text{SAM} = arccos\left(\frac{\langle x, x' \rangle}{||x||_2 \cdot ||x'||_2}\right) \quad (8)$$

Spatial Correlation Coefficient (SCC, [19]) represents the correlation between two visual signals of images in a cortical visual space. The SCC is expressed as:

$$\text{SCC} = \frac{\sum\sum (x_i - \mu_x)^2 (x'_i - \mu_{x'})^2}{\sqrt{\sum (x_i - \mu_x)^2 \sum (x'_i - \mu_{x'})^2}} \quad (9)$$

where x' is the modified image, x is the original image, μ_x is the mean of the original image, and $\mu_{x'}$ is the mean of the modified image. The value of SCC ranges from -1 to 1 [20].

Visual Information Fidelity (VIF, [18]) quantifies the Shannon information present in a processed image. As summarized in [8], VIF uses a natural scene model based on a Gaussian scale mixture model in the wavelet domain. The visual distortion is modeled as a stationary, zero-mean, additive white Gaussian noise process in the wavelet domain $e = c + n$, and $f = d + n$, where e, n, and f are the random coefficient vectors for the same wavelet subband in the perceived original and perceived distorted image. We model c, a collection of M neighboring wavelet coefficients from a local patch in a subband, as $c = \sqrt{z}u$, where u is a zero-mean Gaussian vector and $\sqrt{z}$ is an independent scalar random variable. Random vectors c and d are from the same location in the same subband for the original and distorted image, and n denotes the independent white Gaussian noise with the covariance matrix $C_n = \sigma^2_n$. The VIF is defined as:

$$\text{VIF} = \frac{I(C;F\,|z)}{I(C;E\,|z)} = \frac{\sum_{i=1} I(c_i; f_i\,|z_i)}{\sum_{i=1} I(c_i; e_i\,|e_i)} \quad (10)$$

where i is the index of local coefficient patches, including all subbands.

2.3. Detector Exercised in our Study

We select two state-of-the-art attack detectors as adversarial defenses, namely MagNet [23] and Feature Squeezing [24]. MagNet [23] exercises an autoencoder trained on the normal images to reconstruct an input image before it is fed to the classifier. Once the image is reconstructed, the detector computes the reconstruction error between the input x and a reformed input x', and applies a threshold learned during the training to target a specific false positive rate (FPR). If the image is normal, the reconstruction error lies below the target threshold; otherwise, the input x is marked as an adversarial attack. Feature Squeezing (Squeezer, [24]) compares the prediction of the classifier on the inputs with the predictions obtained using pre-processed inputs. The Squeezer detector computes a score that is the maximum distance among these predictions and then applies a threshold that is learned in an unsupervised fashion to target a specific FPR on the training set.

3. Installing the Libertinus fonts

We set up an experimental campaign to group adversarial attacks. Our experimental methodology is in Section 3.1, and details on attacks, image classifiers, and detectors are in Section 3.2. We execute the experiments on a Dell Precision 5820 Tower with an Intel I9- 9920X, GPU NVIDIA Quadro RTX6000 with 24GB VRAM, 128GB RAM, Ubuntu 18.04 with kernel 5.4.0, and runtime CUDA 11.0.

3.1. Methodology

We first generate adversarial images targeting two state-of-the-art models. We apply the 12 attacks in Section 2.1 with multiple configurations on the first 100 images of the CIFAR-10 dataset [25]. Out of the existing benchmark datasets for image classification, we choose CIFAR-10 [25] because it is composed of RGB images that have a reasonable size to make experiments feasible. We detail the selected image classifiers and the attack configurations in Section 3.2. Then, we compute distance metrics (both L-norms and image quality metrics) between each of the adversarial and original (clean) images. We feed the adversarial images to both detectors MagNet [23] and Feature Squeezing [24], logging their answer: 0 if the attack was not detected, 1 otherwise. These actions allow building a tabular dataset as in Table 1, which contains the following data for each adversarial image: i) the values of the 12 distance metrics, ii) the success outcome (i.e., does the attack trigger a misclassification?), iii) binary flags that indicate if either MagNet [23] or Feature Squeezing [24] can detect that the image was counterfeited, and iv) the attack configuration (not in the paper for brevity). This data will allow conducting the following three analyses detailed below

Distance metric clusters. We analyze the values of distance metrics to investigate if grouping adversarial images according to the perturbation they apply is meaningful. Intuitively, distance metrics quantify different perturbation aspects: we want to investigate if they capture differences between the way attack images are constructed. For this purpose, we run several clustering algorithms and check if images created using (the same or) different attacks are grouped.

Correlation between distance metrics and the success rate of the target classifier. We investigate if distance metrics are informative enough to predict the success of an attack against the target classifier. This allows discovering if distance metrics relate to the success of a given attack.

Table 1
Six sample rows from the tabular dataset obtained with our experiments.

	Distance metrics														
attacks	L2	L1	Linf	L0	MSE	UQI	ERGAS	SAM	SCC	VIF	RASE	PSNR-B	Success	MagNet	Squeezer
JSMA	0.43	3.0	0.18	3072	0.00	1.00	604	0.01	0.96	0.99	87	43.6	1	0	1
PGD	2.25	116.5	0.05	3072	0.02	0.99	4842	0.07	0.77	0.97	698	27.7	1	1	1
HSP	0.21	8.3	0.02	3014	0.00	0.99	1042	0.01	0.96	0.99	149	21.9	1	0	0
FGM	1.66	91.9	0.03	3072	0.01	0.99	2648	0.05	0.80	0.98	382	30.5	1	0	0
PGD	4.08	205.2	0.10	3072	0.05	0.96	1336	0.17	0.53	0.90	192	22.6	1	1	1
UNI	1.44	55.8	0.16	3069	0.01	0.99	3246	0.04	0.94	0.99	468	32.0	0	0	0

Correlation between distance metrics and the detector outputs. We also want to understand if distance metrics are informative enough to predict if the attack will be detected by MagNet [23] or Feature Squeezing [24] detectors.

3.2. Target Target Attacks, Image Classifier, and Detectors

We selected two trained state-of-the-art classifiers, namely ConvNet12 from [33] and ResNet50 [52]. The ConvNet12 model is composed of 6 convolutional layers, 1 dense layer, 3 pooling layers, and 3 dropout layers with 2 923 050 trainable parameters, and an accuracy of 0.85 on the CIFAR-10 test set. The ResNet50 architecture is the one provided by Keras and is composed of 4 convolutional layers with 26 162 698 trainable parameters.

We configure MagNet [23] and Feature Squeezing [24] to defend our models. We train MagNet using 5 000 normal images from the CIFAR-10 training set, with a False Positive Rate (FPR) of ≈ 0.02. We set the threshold for Feature Squeezing manually to have the same FPR on the test set.

We craft attacks using the ART Toolbox [35]. In Table 2 in the appendix A, we report the number of configurations applied for each attack and the configuration values.

4. Results and Discussion

4.1. Distance Metrics Cluster

We run the clustering algorithms K-Means, DBSCAN, and variants of the Expectation-Maximization and Self-Organizing Maps that suit clustering analyses and are available in the WEKA toolkit [51]. However, we found that K-Means with $k = \{2, 3, 5, 6, 8, 9, 12\}$ provided us with enough information to search for groups of attacks based on the values of the collected distance metrics. We start with k=2 without observing distinct groups of attacks. Then we increase the number of possible group k to investigate if we can observe a well-defined partitioning. In *distance metric cluster* analysis and Table 3 in the appendix, we report the detailed analysis when $k = 12$ for both the ConvNet12 and the ResNet50 models. In the analysis, we observed some distinct groupings of adversarial attacks based on their similarities. Notably, attacks like BIM, PGD, and FGM clustered together, which was expected as BIM and PGD can be considered iterative versions of FGM. BOUND HOP ELA are grouped by the two analyses with no clear explanation based on mathematical formulation, indicating a unique relationship discerned solely through distance metrics. JSMA and NEW form isolated clusters suggesting that they are different from any other attacks.

4.2. Correlation between Distance Metrics and Behavior of the Classifier

We look for a correlation between the success rate of the attack and the values of the selected metrics, to the extent that we can predict the success rate with minimal error. We measure our prediction capability by computing the cosine distance CD between the actual and the predicted success rate of the adversarial images on the target classifier. More specifically, we train an XGBoost regressor with the distance metrics of the adversarial images and the attack outcome to predict the success of each adversarial image. Given the distance metrics computed from an adversarial image as input, the trained regressor outputs a good estimation of the success probability, i.e., the CD is generally low. We repeat the study by training the regressor multiple

times using adversarial images generated from the same attack and testing on adversarial images generated using the other attacks. This shows the mutual predictability of the success rate between attacks, which we can use to understand if two attacks are similar. According to the distance metric cluster analysis in Section 4.1, both models grouped BIM and FGM. However, unlike the cluster analysis, PGD was excluded from this grouping, indicating a weaker similarity compared to the other two attacks. We report in *correlation between distance metrics and the behavior of the target classifier* and Table 4 in the appendix the detailed discussion and the related data, respectively.

4.3. Correlation between Distance Metrics and the Detector Outputs

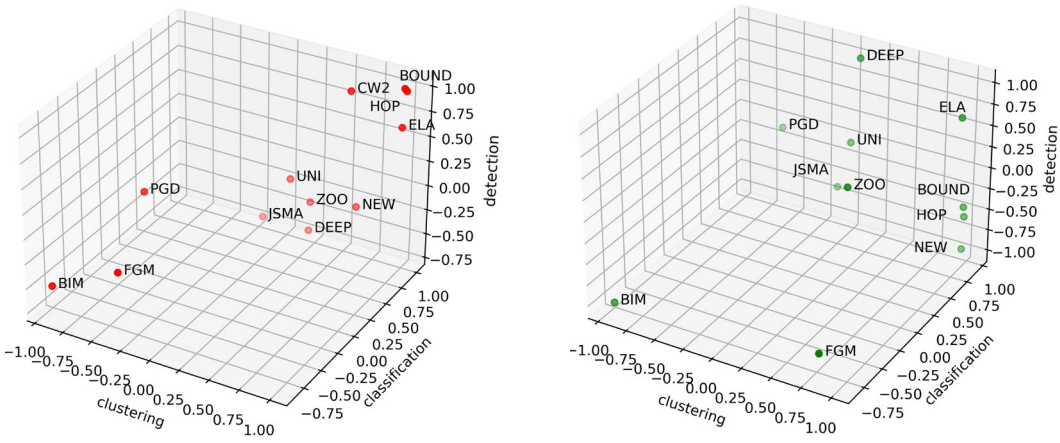
We quantify the correlation between the detectability of an adversarial image and the value of the distance metrics. In this case, we train an XGBoost classifier by using distance metrics as input features, to predict the binary output (detected/undetected) of a detector, either Squeezer or MagNet. Similarly to the previous experiment, the attacks in the columns of Table 5.a and Table 5.b are used for training, and the attacks in the rows are used for testing. Each cell of the table reports the mean Matthew Correlation Coefficient (MCC) obtained by XGBoost when predicting the outputs of MagNet and Squeezer detectors. Notably, BOUND and HOP exhibited strong mutual predictability, with MCC scores indicating their ability to predict each other's detectability. Additionally, CW attacks showed some predictive capability for BOUND and HOP. However, PGD, UNI, and ZOO attacks showed no mutual predictability with any other attacks.

On the ResNet50 model, PGD, UNI, and ZOO attacks displayed poor predictability with other attacks, with MCC scores near zero. Conversely, BIM, BOUND, DEEP, ELA, FGM, HOP, JSMA, and NEW attacks showed some degree of predictability with each other, suggesting relationships between these attacks. Notably, HOP and BOUND exhibited the strongest mutual predictability, consistent with previous findings. Furthermore, while BIM, FGM, and DEEP attacks demonstrated predictive capability for each other on ResNet50, this relationship was not observed in the ConvNet12 results.

4.4. Concluding Discussion

We aggregate together the results of each of the three analyses for both models, respectively, in Figure 1.a and 1.b. The x-axis refers to the clustering analysis in Section 4.1 for clusters k=2, the y-axis to the CD scores in Section 4.2, and the z-axis to the MCC scores in Section 4.3.

From the ConvNet12 analysis in Figure 1.a, two groups can be identified easily. BOUND, HOP, and ELA are very close (BOUND and HOP are almost overlapping), while PGD is a bit far from BIM and FGM: since PGD has success rate 1, its scores in the analysis in Section 4.2 are not meaningful and are represented as 0 in the plot. Interestingly, CW2 is overall close to BOUND, HOP, and ELA, even if not so evidently as those three attacks. Other attacks are far from anyone else such as NEW or JSMA. DEEP, UNI and ZOO. Note that they are visually close in Figure 1.a, but according to the clustering analysis, they belong to neither group when k=2, and the other two analyses report contradicting results. This may suggest that those three attacks could represent a new group. However, the results of the other analyses do not confirm this result.



a) Aggregated results on ConvNet12.

b) Aggregated results on ResNet50.

Figure 1: The results of the 3 analyses aggregated: more similar attacks are spatially closer in the images.

From the ResNet50 results in Figure 1.b, we can identify one group composed of BOUND, HOP, and NEW. BOUND and HOP are very close to each other, validating the outcome of the ConvNet12 analysis. Differently, ELA is not included in the group as the detection axis has opposite scores (BOUND and HOP have a score near -0.5 while ELA is 0.5). In Figure 1.b, ZOO and JSMA are very close suggesting a possible group, but this is just due to the view angle of the plot: the two attacks show different scores on each of the 3 dimensions. Differently, JSMA and ZOO are closer to each other but with different scores from the detectability analysis: JSMA is likely part of a cluster, while UNI does not belong to any of them. DEEP is the farthest from all the others, forming a group by itself. According to the results of detectability and classification analyses, BIM and FGM are likely to be in the same group. However, they are very far from each other as it results from the clustering analysis. This contrasts with what we observed in Figure 1.a, in which they are grouped strongly.

We proposed an experimental methodology to identify the most suitable attack set when testing defenses against evasion attacks. The overall idea is to apply distance metrics to group attacks that introduce a similar perturbation on the image and have a similar effect on the target classifier. While the analysis performed on the two models shows some discriminative power, it does not provide enough evidence to drive guidelines that can reduce the number or types of attacks to be used when testing a defense. Results on ConvNet12 and ResNet50 agree in grouping BOUND and HOP attacks, which, share the same approach to craft attacks, according to the mathematical formulation. In weaker form, BIM and FGM are also grouped, with experiments on ConvNet12 giving a solid grouping and experiments in ResNet50 partially confirming the result. This is somehow expected as BIM is the iterative form of the FGM attack. However, this result alone applies only to those specific attacks, and no relevant relations are found in all the other cases.

In conclusion, our preliminary assessment suggests that measuring the adversarial perturbations does not provide a sensitive proxy to select diverse attack classes, meaning that different attack methods, with distinct mathematical formulations, do not introduce distinguishable (measurable) perturbations in the image. However, the methodology should be

further enriched to provide additional verification of these results. This can be achieved by applying a broader range of models and attack methods, as well as identifying different metrics. However, the preliminary results we obtained are quite worrying. When evaluating a defense, it is risky to reduce the diversity of attacks, because the exhaustiveness of the evaluation can be unexpectedly compromised by essentially any reduction attempt.

Acknowledgments

This paper was partially supported by the MUR PRIN 2022 project FLEGREA - Federated Learning for Generative Emulation of Advanced Persistent Threats and the PRIN PNRR 2022 project BREADCRUMBS- Building up Robust and Efficient Routing Algorithms for Drones by integrating Connectivity and Risk awareness in an Urban air Mobility Bvlos Scenario.

References

[1] Nicholas Carlini. Is AMI (attacks meet interpretability) robust to adversarial examples? arXiv preprint arXiv:1902.02322, (2019).

[2] Nicholas Carlini and David Wagner. Defensive distillation is not robust to adversarial examples. arXiv preprint arXiv:1607.04311, (2016).

[3] Nicholas Carlini and David Wagner. Adversarial examples are not easily detected: Bypassing ten detection methods. 10th ACM Workshop on Artificial Intelligence and Security, pp. 3–14. ACM, (2017).

[4] Nicholas Carlini and David Wagner. Magnet and "efficient defenses against adversarial attacks" are not robust to adversarial examples. arXiv preprint arXiv:1711.08478, (2017).

[5] Nicholas Carlini and David Wagner. Evaluating the robustness of neural networks. In 2017 IEEE Symposium on Security and Privacy (SP), pp. 39–57. IEEE, (2017).

[6] Lukas Schott, Jonas Rauber, Matthias Bethge, and Wieland Brendel. Towards the first adversarially robust neural network model on mnist. International Conference for Learning Representations, (2019).

[7] I. Goodfellow, J. Shlens, and C. Szegedy, "Explaining and harnessing adversarial examples," arXiv:1412.6572, 2014.

[8] Pedersen, Marius, and Jon Yngve Hardeberg. "Full-reference image quality metrics: Classification and evaluation." Foundations and Trends® in Computer Graphics and Vision 7.1 (2012): 1-80.

[9] Samajdar, Tina, and Md Iqbal Quraishi. "Analysis and evaluation of image quality metrics." Information Systems Design and Intelligent Applications: Proceedings of Second International Conference INDIA 2015, Volume 2. Springer India, 2015.

[10] Hore, Alain, and Djemel Ziou. "Image quality metrics: PSNR vs. SSIM." 2010 20th international conference on pattern recognition. IEEE, 2010.

[11] Wang, Zhou, and Alan C. Bovik. Modern Image Quality Assessment. Springer Nature, 2022.

[12] D. M. Chandler and S. S. Hemami, “VSNR: A wavelet-based visual signalto-noise ratio for natural images,” IEEE Transactions on Image Processing, vol. 16, no. 9, pp. 2284– 2298, September 2007.

[13] Silpa, K., and S. Aruna Mastani. "NEW APPROACH OF ESTIMATING PSNR-B FOR DEBLOCKED IMAGES." International Journal of Advances in Engineering & Technology 7.1 (2014): 183.

[14] Yim, Changhoon, and Alan Conrad Bovik. "Quality assessment of deblocked images." IEEE Transactions on Image Processing 20.1 (2010): 88-98.
[15] Wang, Zhou, and Alan C. Bovik. "A universal image quality index." IEEE signal processing letters 9.3 (2002): 81-84.
[16] Wald, Lucien. "Quality of high-resolution synthesised images: Is there a simple criterion?" SEE/URISCA, 2000.
[17] Renza, Diego, Estibaliz Martinez, and Agueda Arquero. "A new approach to change detection in multispectral images by means of ERGAS index." IEEE Geoscience and Remote Sensing Letters 10.1 (2012): 76-80.
[18] Sheikh, Hamid R., and Alan C. Bovik. "Image information and visual quality." IEEE Transactions on image processing 15.2 (2006).
[19] Zhou, Jie, Daniel L. Civco, and J. A. Silander. "A wavelet trans-form method to merge Landsat TM and SPOT panchromatic data." International journal of remote sensing 19.4 (1998).
[20] Pushparaj, Jagalingam, and Arkal Vittal Hegde. "Evaluation of pan-sharpening methods for spatial and spectral quality." Applied Geomatics 9 (2017): 1-12.
[21] González-Audícana, María, et al. "Fusion of multispectral and pan-chromatic images using improved IHS and PCA mergers based on wavelet decomposition." IEEE Transactions on Geoscience and Remote sensing 42.6 (2004): 1291-1299.
[22] Yuhas, Roberta H., Alexander FH Goetz, and Joe W. Boardman. "Discrimination among semi-arid landscape endmembers using the spectral angle mapper (SAM) algorithm." Summaries of the 3rd Annual JPL Airborne Geoscience Workshop. Volume 1: AVIRIS Workshop. 1992.
[23] D. Meng, and H. Chen, "Magnet: a two-pronged defense against adversarial examples," Proceedings of the 2017 ACM SIGSAC conference on computer and communications security. 2017.
[24] W. Xu, D. Evans, and Y. Qi, "Feature Squeezing: Detecting Adversarial Examples in Deep Neural Networks," Network and Distributed System Security Symposium, 2018.
[25] A. Krizhevsky, and G. Hinton, "Learning multiple layers of features from tiny images," Technical Report, 2009.
[26] D. Warde-Farley, and I. Goodfellow, "Adversarial perturbations of deep neural networks," Perturbation, Optimization, and Statistics (editors: T. Hazan, G. Papandreou, D. Tarlow), 2016.
[27] M. Xue, et al., "Machine learning security: Threats, countermeasures, and evaluations," IEEE Access 8: 74720-74742, 2020.
[28] S. Bulusu et al., "Anomalous instance detection in deep learning a survey, " In: IEEE Symposium on Security and Privacy, 2020.
[29] D. J. Miller, Z. Xiang, and G. Kesidis, "Adversarial learning targeting deep neural network classification: A comprehensive review of defenses against attacks" Proceedings of the IEEE 108.3 (2020): 402-433.
[30] Y. Deng, et al., "An analysis of adversarial attacks and defenses on autonomous driving models," IEEE Int. Conf. on Pervasive Computing and Communications (PerCom), 2020.
[31] Li, Yao, et al. "A review of adversarial attack and defense for classification methods." The American Statistician 76.4 (2022): 329-345.

[32] N. Carlini, and D. Wagner, "Towards evaluating the robustness of neural networks," IEEE symposium on security and privacy (SP), pp. 39-57, IEEE, 2017.
[33] Ma, Xingjun, et al. "Characterizing adversarial subspaces using lo-cal intrinsic dimensionality." arXiv:1801.02613 (2018).
[34] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. 2016. Deep residual learning for image recognition. In Proceedings of the IEEE conference on computer vision and pattern recognition. 770–778.
[35] M. I. Nicolae, et al., "Adversarial Robustness Toolbox v1.0.0," arXiv:1807.01069v4, 2019.
[36] Xu, Han, et al. "Adversarial attacks and defenses in images, graphs and text: A review." International Journal of Automation and Computing 17 (2020): 151-178.
[37] Carlini, N., and Wagner, D., (2017a), "Adversarial Examples Are Not Easily Detected: Bypassing Ten Detection Methods," in Proceedings of the 10th ACM Workshop on Artificial Intelligence and Security, 3–14.G. Eason, B. Noble, and I. N. Sneddon, "On certain integrals of Lipschitz-Hankel type involving products of Bessel functions," Phil. Trans. Roy. Soc. London, vol. A247, pp. 529–551, April 1955. (references)
[38] Zou, H., and Hastie, T. (2005), "Regularization and Variable Selection via the Elastic Net," Journal of the Royal Statistical Society: Series B, 67, 301–320. DOI: 10.1111/j.1467-9868.2005.00503.x.
[39] Chen, P.-Y., Sharma, Y., Zhang, H., Yi, J., and Hsieh, C.-J. (2018) "EAD: Elastic-Net Attacks to Deep Neural Networks via Adversarial Examples," in Thirty-Second AAAI Conference on Artificial Intelligence.
[40] Goodfellow, I., Shlens, J., and Szegedy, C. (2015), "Explaining and Harnessing Adversarial Examples," in International Conference on Learning Representations.
[41] Madry, A., Makelov, A., Schmidt, L., Tsipras, D., and Vladu, A. (2018), "Towards Deep Learning Models Resistant to Adversarial Attacks," in International Conference on Learning Representations.
[42] Kurakin, A., Goodfellow, I., and Bengio, S. (2016), "Adversarial Examples in the Physical World."
[43] Moosavi-Dezfooli, S.-M., Fawzi, A., and Frossard, P. (2016), "Deepfool: A Simple and Accurate Method to Fool Deep Neural Networks," in Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2574–2582.
[44] Papernot, N., McDaniel, P., Jha, S., Fredrikson, M., Celik, Z. B., and Swami, A. (2016), "The Limitations of Deep Learning in Adversarial Settings," in Security and Privacy (EuroS&P), 2016 IEEE European Symposium on, 372–387.
[45] Jang, Uyeong, Xi Wu, and Somesh Jha. "Objective metrics and gradient descent algorithms for adversarial examples in machine learning." Proceedings of the 33rd Annual Computer Security Applications Conference. 2017.
[46] Chen, P.-Y., Zhang, H., Sharma, Y., Yi, J., and Hsieh, C.-J. (2017), "Zoo: Zeroth Order Optimization Based Black-Box Attacks to Deep Neural Networks Without Training Substitute Models," in Proceedings of the 10th ACM Workshop on Artificial Intelligence and Security
[47] Brendel, W., Rauber, J., and Bethge, M. (2018), "Decision-Based Adversarial Attacks: Reliable Attacks Against Black-Box Machine Learning Models," International Conference on Learning Representations.

[48] Hen, J., Jordan, M. I., and Wainwright, M. J. (2020), "Hopskipjumpattack: A QueryEfficient Decision-Based Attack." In 2020 IEEE Symposium on Security and Privacy (SP), 1277–1294. DOI: 10.1109/SP40000.2020.00045.
[49] Cheng, S., Dong, Y., Pang, T., Su, H., and Zhu, J. (2019b), "Improving Black-Box Adversarial Attacks with a Transfer-Based Prior," in Advances in Neural Information Processing Systems, 10932–10942. 50.
[50] Carlini, Nicholas, et al. "On evaluating adversarial robustness." arXiv:1902.06705 (2019).
[51] Sharma, N., Bajpai, A., & Litoriya, M. R. (2012). Comparison the various clustering algorithms of weka tools. Facilities, 4(7), 78-80.
[52] Puccetti, T., Github Repository with source code https://github.com/TommasoPuccetti/adv_perturb/
[53] ResNet50 implementation, https://github.com/keras-team/keras/blob/v2.14.0/keras/applications/resnet.py#L499-L533.

A. Appendix

Attack configuration parameters. In Table 2 we report the configuration parameters used to craft the attacks in the experimental campaign. How these values are combined, and all configuration details can be found at [52] along with the code used for the generation. All the images generated and the code to reproduce our experiments are at [52].

Distance metrics cluster analysis. First, we observe in Table 3.a that cluster 5 and cluster 8 group all the attacks except, again, BIM and PGD. On the contrary, some clusters indicate small groups of attacks. Cluster 2 and 11 strengthen the BIM, FGM, and PGD grouping. We also identify a group of attacks composed of HOP, BOUND, and ELA, in clusters 3, 5, and 8. Another cluster group is composed of ZOO and DEEP, which mostly fall in clusters 5, 8, 9, and 12. Interestingly, the ZOO attack does not share its mathematical formulation with DEEP but generates a perturbation that distance metrics quantify very similar to the perturbation created by DEEP. Lastly, we observe that the JSMA attack is different from any other attack since it is

Table 2
Generated attack, generation parameters, success rate, and attack detectability.

Attack	Generation Parameters	Success Rate	#adv_img	# config	MagNet Det Rate	Squeezer Det Rate
BIM	ε = {0.001-0.1}, ε-step = {0.001-0.1}	0.88	1951	17	0.630	0.65
BOUND	ε={0.001-0.1}, δ={0.001-0.1}, max_iter={500-5000}	0.98	3325	26	0.212	0.065
CW2	confidence = {1-50}, max_iter = {10-1000}	0.99	1435	10	0.725	0.776
DEEP	ε = {0-10}, max_iter = {10-300}	0.97	1239	15	0.031	0.038
ELA	confidence = {0-50}, max_iter = {10-90}	1.00	3315	19	0.100	0.023
FGM	ε = {0.001-0.01}	0.70	596	32	0.057	0.040
HOP	max_iter = {50-300}, max_eval = {10-20000}, init_eval = {20, 200}, init_size = {20-200}	0.99	2209	26	0.230	0.120
JSMA	Θ = {0.01-5}, γ = {0.2-1}	0.96	3359	19	0.211	0.305
NEW	max_iter = {10-300}, η = {0.01-10}	1.00	1615	26	0.082	0.183
PGD	max_iter = {10-300}, ε = {0.01-10}	1.00	1615	39	0.99	1.0
UNI	ε = {0.1-20}, δ = {0.1-10}	0.44	1200	10	0.128	0.083
ZOO	confidence = {1-10}, max_iter = {100-200}	0.90	762	40	0.056	0.033

Table 3
Groups of attacks are identified based on the distance metrics. The most evident groups are in dark colors.

a) Results with k=12 clusters, using the ConvNet12 model

K=12	cluster1	cluster2	cluster3	cluster4	cluster5	cluster6	cluster7	cluster8	cluster9	cluster10	cluster11	cluster12
BIM	0.000	0.383	0.029	0.000	0.056	0.000	0.000	0.142	0.091	0.000	0.297	0.001
BOUND	0.000	0.000	0.415	0.000	0.225	0.000	0.000	0.355	0.004	0.000	0.000	0.001
CW2	0.007	0.003	0.039	0.020	0.227	0.000	0.003	0.518	0.035	0.000	0.006	0.141
DEEP	0.000	0.020	0.061	0.010	0.157	0.000	0.000	0.280	0.155	0.000	0.002	0.314
ELA	0.000	0.000	0.304	0.000	0.212	0.000	0.000	0.472	0.000	0.012	0.000	0.000
FGM	0.000	0.172	0.076	0.000	0.145	0.000	0.000	0.361	0.128	0.000	0.118	0.000
HOP	0.000	0.000	0.376	0.000	0.227	0.000	0.000	0.390	0.006	0.000	0.000	0.000
JSMA	0.006	0.000	0.001	0.359	0.001	0.016	0.433	0.017	0.001	0.004	0.000	0.161
NEW	0.000	0.000	0.047	0.000	0.259	0.000	0.000	0.694	0.000	0.000	0.000	0.000
PGD	0.000	0.380	0.000	0.000	0.017	0.263	0.000	0.000	0.080	0.000	0.259	0.000
UNI	0.000	0.037	0.000	0.000	0.004	0.000	0.000	0.000	0.343	0.000	0.012	0.604
ZOO	0.000	0.025	0.006	0.000	0.291	0.000	0.000	0.267	0.247	0.044	0.002	0.119

b) Results with k=12 clusters, using the ResNet50 model

K=12	cluster1	cluster2	cluster3	cluster4	cluster5	cluster6	cluster7	cluster8	cluster9	cluster10	cluster11	cluster12
BIM	0.000	0.000	0.000	0.450	0.288	0.000	0.000	0.000	0.259	0.000	0.003	0.000
BOUND	0.000	0.000	0.000	0.000	0.002	0.000	0.000	0.000	0.998	0.000	0.000	0.000
DEEP	0.038	0.028	0.041	0.000	0.047	0.000	0.459	0.011	0.202	0.100	0.001	0.073
ELA	0.006	0.025	0.032	0.002	0.075	0.000	0.011	0.012	0.768	0.019	0.033	0.016
FGM	0.000	0.000	0.000	0.202	0.200	0.000	0000	0.000	0.589	0.000	0.009	0.000
HOP	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	1.000	0.000	0.000	0.000
JSMA	0.237	0.208	0.030	0.000	0.004	0.000	0.059	0.010	0.099	0.108	0.004	0.240
NEW	0.005	0.065	0.039	0.005	0.092	0.000	0.007	0.001	0.747	0.022	0.011	0.005
PGD	0.000	0.000	0.000	0.383	0.314	0.000	0.061	0.000	0.040	0.202	0.000	0.000
UNI	0.006	0.016	0.323	0.112	0.015	0.000	0.000	0.444	0.000	0.032	0.000	0.052
ZOO	0.000	0.000	0.000	0.007	0.022	0.576	0.000	0.000	0.391	0.000	0.004	0.000

the only attack that gets assigned to clusters 4 and 7. The analysis shows similar results concerning the ResNet50 model (Table 3.b). The BIM, FGM, and PGD group seems confirmed since the three attacks are the only ones with high scores in clusters 4 and 5. This is somehow expected as the Basic Iterative Method (BIM) and Projected Gradient Descent Attack (PGD) can be considered iterative versions of FGM [31]. As observed in ConvNet12 results, JSMA is isolated in clusters 1, 2, and 12 suggesting no similarities for the other attacks. Differently, cluster 9 evidences a big group of attacks. However, BOUND and HOP show a score (respectively, 1 and 0.998) that is significantly higher than the others, suggesting a stronger

similarity. Further, ELA and NEW provide scores that are high (respectively, 0.768 and 0.747) but separated from the rest of the scores. This trend agrees partially with the analysis on ConvNet12 that groups together BOUND, HOP, and ELA, while NEW is excluded. The similarity between HOP and BOUND with the ELA and NEW cannot be explained or confirmed by checking their mathematical formulation, and it is a relation observable solely thanks to the distance metrics. Differently from the ConvNet12 case, DEEP and ZOO are not similar and form isolated groups (clusters 7 and 8 of Table 3.b). Lastly, UNI is different from any other attack according to the analysis of both models.

Correlation between distance metrics and the behavior of the target classifier. We show the results of, respectively, the ConvNet12 and the ResNet50 models in Table 4.b and Table 4.b. The columns report the attacks used to train the regressors, while the rows enlist the attacks used to test the regressors. Each cell *i,j* of the tables reports the CD when predicting the success rate

Table 4
Cosine distance is measured when predicting the success rate using the distance metrics. The lower the cosine distance, the higher the correlation.

a) Results obtained on the ConvNet12 model.

	Trained using distance metrics from the adversarial images of attacks:											
Tested on:	BIM	BOUND	CW2	DEEP	ELA	FGM	HOP	JSMA	NEW	PGD	UNI	ZOO
BIM		0.029	0.030	0.024	0.026	0.002	0.026	0.020	0.026	0.026	0.029	0.024
BOUND	0.108		0.000	0.002	0.000	0.131	0.000	0.001	0.000	0.000	0.000	0.002
CW2	0.035	0.000		0.001	0.000	0.038	0.000	0.002	0.000	0.000	0.001	0.001
DEEP	0.038	0.001	0.001		0.001	0.039	0.001	0.002	0.001	0.001	0.004	0.002
ELA	0.071	0.000	0.000	0.001		0.084	0.000	0.001	0.000	0.000	0.000	0.001
FGM	0.002	0.060	0.064	0.056	0.059		0.059	0.050	0.059	0.059	0.061	0.063
HOP	0.103	0.000	0.000	0.001	0.000	0.125		0.001	0.000	0.000	0.000	0.002
JSMA	0.009	0.003	0.002	0.003	0.002	0.009	0.002		0.002	0.002	0.003	0.003
NEW	0.034	0.000	0.000	0.001	0.000	0.047	0.000	0.002		0.000	0.000	0.001
PGD	0.001	0.000	0.000	0.000	0.000	0.001	0.000	0.002	0.000		0.001	0.001
UNI	0.010	0.011	0.011	0.011	0.011	0.011	0.011	0.014	0.011	0.011		0.011
ZOO	0.033	0.002	0.002	0.002	0.002	0.036	0.002	0.003	0.002	0.002	0.004	

b) Results obtained on the ResNet50 model.

	Trained using distance metrics from the adversarial images of attacks:										
Tested on:	BIM	BOUND	DEEP	ELA	FGM	HOP	JSMA	NEW	PGD	UNI	ZOO
BIM		0.02	0.02	0.02	0.00	0.02	0.02	0.02	0.02	0.02	0.01
BOUND	0.08		0.00	0.00	0.10	0.00	0.00	0.00	0.00	0.00	0.03
DEEP	0.02	0.00		0.00	0.03	0.00	0.00	0.00	0.00	0.00	0.01
ELA	0.10	0.00	0.00		0.12	0.00	0.00	0.00	0.00	0.00	0.02
FGM	0.00	0.04	0.04	0.04		0.04	0.04	0.04	0.04	0.03	0.02
HOP	0.07	0.00	0.00	0.00	0.10		0.00	0.00	0.00	0.00	0.03
JSMA	0.01	0.00	0.00	0.00	0.01	0.00		0.00	0.00	0.00	0.01
NEW	0.06	0.00	0.00	0.00	0.05	0.00	0.00		0.00	0.00	0.01
PGD	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00		0.00	0.00
UNI	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00		0.00
ZOO	0.15	0.06	0.07	0.06	0.14	0.06	0.06	0.06	0.06	0.06	

of the attack in the *i*-th row using as a training set the attack in the *j*-th column. The CD is low (i.e., there is mutual predictability) in most cases: therefore, it is more interesting to observe the combinations for which the CD is high. Considering ConvNet12 and Table 4.a, it is the case of FGM against all attacks but BIM and PGD; also, BIM shows low CD only when paired with FGM. UNI shows good predictability when used to train the regressor, but rather poor predictability if it is used as a test set: therefore, we cannot get actionable information about this attack. Similar results can be observed in the ResNet50 analysis (Table 4.b). BIM can predict with CD=0 the success rate of FGM, PGD, and UNI, with a very low CD in detecting JSMA (see the 1st column in Table 4.b). We obtain similar results using FGM as training, showing a CD=0 in predicting BIM, PGD, and UNI. Using both UNI and PGD, the CD is 0 except for BIM and FGM. This seems to define a group composed of BIM and FGM, excluding UNI and PGD.

Correlation between distance metrics and the detector outputs. In Table 5.a, we observe that BOUND and HOP show mutual predictability. BOUND can predict the detectability of HOP with MCC=0.698, whereas, using HOP to predict the detectability of BOUND, we obtain MCC=0.642. CW attacks seem grouped with BOUND and HOP: if used as the training set, it can predict the detectability of HOP and BOUND attacks with an MCC of roughly 0.42. The results are similar if using HOP to predict CW2 (MCC = 0.365) but are lower using BOUND attacks (MCC = 0.234). Using HOP, BOUND, and CW2 as training sets shows some capabilities in detecting the BIM attack, however, the MCC scores are very low (below 0.3). The PGD attacks are not considered because it has a detection rate of 1.0 and as such cannot be efficiently used to train a classifier. Training with JSMA shows MCC of 0.360, 0.350, and 0.381 when tested with BOUND, CW2, and HOP respectively. Despite there being no mutual predictability between JSMA and these attacks (using CW2, BOUND, and HOP in the training, and JSMA as a test shows a low MCC), training on JSMA offers some capability to predict the others. The UNI, ZOO, NEW, and DEEP attacks are the ones with the lowest average MCC if used as training.

We repeat the analysis on the ResNet50 model, considering all the attacks except for CW2 which is computationally too complex to generate with our hardware. We show the results in Table 5.b. PGD, UNI, and ZOO do not have mutual predictability with any other attacks. Each of them shows an MCC score near 0 in predicting the detectability of each other, with a few exceptions showing a low MCC score. PGD, UNI, and ZOO are predicted poorly by other attacks. BIM, BOUND, DEEP, ELA, FGM, HOP, JSMA, and NEW attacks appear somehow related: attacks can be predicted by or can predict others. These results can be easily seen by looking at the rows of Table 5.b: except for PGD, UNI, and ZOO, there is no attack with the majority of MCC score to zero. The only exception is the BOUND attack (in the 2nd column) that shows MCC greater than 0 only in predicting BIM, HOP, and NEW. Also, the MCC score obtained using NEW when training suggests that NEW can be related to BIM, DEEP, ELA, FGM, HOP, and JSMA. The strongest similarity between attacks is observed for HOP and BOUND, with HOP being the only attack that shows MCC substantially greater than 0 in detecting BOUND. This result is consistent with the BOUND and HOP group evidenced by the ConvNet12 results. Differently from ConvNet12 results, BIM can predict FGM and DEEP with, respectively, MCC=0.60 and MCC=0.48. We obtain analogous results using FGM or DEEP in the training. FGM achieves MCC=0.50 in predicting BIM and MCC=0.62 in predicting DEEP. Similarly DEEP can predict BIM and FGM with MCC=0.39 and MCC=0.47.

Table 5
Matthew Coefficient Correlator to predict detection of Magnet and Squeezer using the distance metric as input features.

a) Results obtained using the ConvNet12 model.

	Trained using distance metrics from the adversarial images of attacks:											
Tested on:	BIM	BOUND	CW2	DEEP	ELA	FGM	HOP	JSMA	NEW	PGD	UNI	ZOO
BIM		0.288	0.205	0.166	0.093	0.004	0.223	0.189	0.009	~~0.000~~	0.046	0.043
BOUND	0.220		0.416	0.147	0.162	0.036	0.642	0.360	0.016	~~0.000~~	0.047	0.030
CW2	0.206	0.234		0.026	0.071	0.048	0.365	0.360	0.127	~~0.000~~	0.047	0.041
DEEP	0.054	0.043	0.083		0.028	0.167	0.075	0.032	0.011	~~0.000~~	0.185	0.104
ELA	0.228	0.262	0.226	0.163		0.030	0.241	0.176	0.180	~~0.000~~	0.000	0.000
FGM	0.116	0.071	0.073	0.056	0.036		0.052	0.071	0.043	~~0.000~~	0.034	0.031
HOP	0.188	0.698	0.417	0.135	0.209	0.017		0.381	0.031	~~0.000~~	0.052	0.028
JSMA	0.112	0.143	0.113	0.046	0.097	0.039	0.101		0.038	~~0.000~~	0.020	0.037
NEW	0.131	0.149	0.182	0.124	0.044	0.020	0.109	0.088		~~0.000~~	0.000	0.000
PGD	0.003	0.018	0.000	0.000	0.000	0.001	0.023	0.025	0.000		0.002	0.001
UNI	0.018	0.074	0.000	0.033	0.000	0.009	0.059	0.172	0.053	~~0.000~~		0.044
ZOO	0.161	0.117	0.057	0.020	0.043	0.011	0.067	0.096	0.109	~~0.000~~	0.053	

b) Results obtained using the ResNet50 model.

	Trained using distance metrics from the adversarial images of attacks:										
Tested on:	BIM	BOUND	DEEP	ELA	FGM	HOP	JSMA	NEW	PGD	UNI	ZOO
BIM		0.23	0.39	0.32	0.50	0.38	0.23	0.41	0.02	0.01	0.10
BOUND	0.11		0.01	0.03	0.11	0.41	0.17	0.09	0.00	0.00	0.00
DEEP	0.48	0.00		0.46	0.62	0.25	0.29	0.54	0.11	0.15	0.08
ELA	0.34	0.03	0.38		0.26	0.23	0.28	0.40	0.01	0.02	0.11
FGM	0.60	0.08	0.47	0.30		0.34	0.18	0.51	0.08	0.00	0.19
HOP	0.19	0.23	0.03	0.17	0.10		0.18	0.17	0.01	0.00	0.00
JSMA	0.22	0.08	0.26	0.26	0.22	0.03		0.23	0.02	0.01	0.01
NEW	0.29	0.40	0.37	0.43	0.31	0.14	0.13		0.01	0.00	0.09
PGD	0.01	0.11	0.00	0.05	0.04	0.00	0.00	0.04		0.00	0.00
UNI	0.00	0.00	0.00	0.01	0.00	0.00	0.00	0.01	0.02		0.00
ZOO	0.13	0.00	0.03	0.14	0.12	0.03	0.03	0.11	0.01	0.01	