applied sciences

MDPI

Article

Novel Exploit Feature-Map-Based Detection of Adversarial Attacks

Ali Saeed Almuflih [1,*], Dhairya Vyas [2], Viral V. Kapdia [2], Mohamed Rafik Noor Mohamed Qureshi [1], Karishma Mohamed Rafik Qureshi [3] and Elaf Abdullah Makkawi [4]

[1] Industrial Engineering Department, King Khalid University, Abha 62529, Saudi Arabia; mrnoor@kku.edu.sa
[2] Computer Science and Engineering Department, The Maharaja Sayajirao University of Baroda, Vadodara 390002, India; dhairya.vyas-cse@msubaroda.ac.in (D.V.); viral.kapadia-cse@msubaroda.ac.in (V.V.K.)
[3] Department of Mechanical Engineering, Parul University, Waghodia 391760, India; 200305208008@paruluniversity.ac.in
[4] Industrial Engineering and Management System, University of Central Florida, Orlando, FL 32816, USA; emakkawi@knights.ucf.edu
* Correspondence: asalmuflih@kku.edu.sa

Featured Application: This system is applicable to the detection of adversarial attacks on different machine learning (ML) models.

Abstract: In machine learning (ML), adversarial attack (targeted or untargeted) in the presence of noise disturbs the model prediction. This research suggests that adversarial perturbations on pictures lead to noise in the features constructed by any networks. As a result, adversarial assaults against image categorization systems may present obstacles and possibilities for studying convolutional neural networks (CNNs). According to this research, adversarial perturbations on pictures cause noise in the features created by neural networks. Motivated by adversarial perturbation on image pixel attacks observation, we developed a novel exploit feature map that describes adversarial attacks by performing individual object feature-map visual description. Specifically, a novel detection algorithm calculates each object's class activation map weight and makes a combined activation map. When checked with different networks like VGGNet19 and ResNet50, in both white-box and black-box attack situations, the unique exploit feature-map significantly improves the state-of-the-art in adversarial resilience. Further, it will clearly exploit attacks on ImageNet under various algorithms like Fast Gradient Sign Method (FGSM), DeepFool, Projected Gradient Descent (PGD), and Backward Pass Differentiable Approximation (BPDA).

Keywords: adversarial attack; convolutional neural networks; feature-map; VGGNet19; ResNet50; white box; black box

Citation: Almuflih, A.S.; Vyas, D.; Kapdia, V.V.; Qureshi, M.R.N.M.; Qureshi, K.M.R.; Makkawi, E.A. Novel Exploit Feature-Map-Based Detection of Adversarial Attacks. *Appl. Sci.* **2022**, *12*, 5161. https://doi.org/10.3390/app12105161

Academic Editors: Andrea Prati, Luis Javier and Vincent A. Cicirello

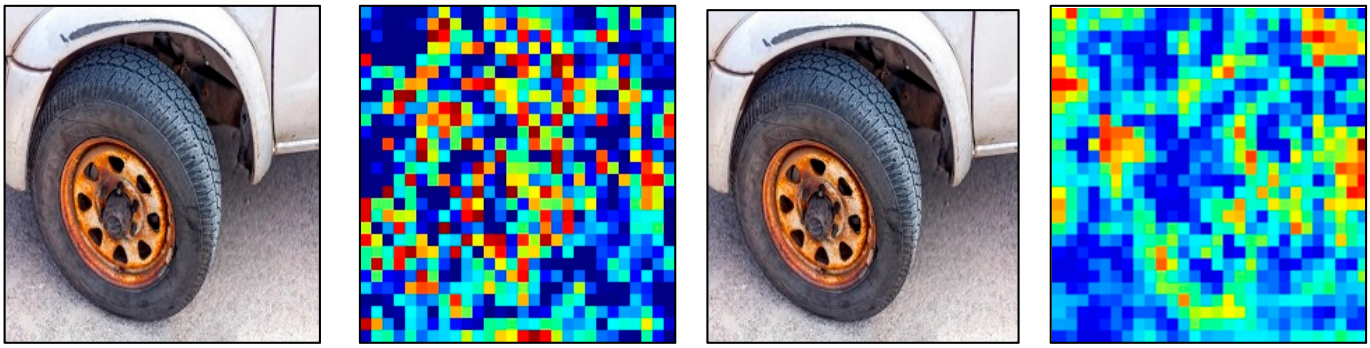
Received: 25 April 2022
Accepted: 18 May 2022
Published: 20 May 2022

1. Introduction

Due to clear advantages in terms of the prediction accuracy of deep learning (DL) models that are presently being applied in a broad variety of disciplines. VGGNet and AlexNet, two of the most popular DL models for computer vision tasks, have a lot of hype because they promise things such as superhuman autonomous driving or health diagnostics [1]. Simultaneously, a drawback of DL is becoming more generally recognized as the opaque nature of its decision-making process. Pre-trained models, commonly called black boxes, make it impossible to understand which components of the input contributed to the output and in what way [2]. The end user may desire an exploit that is simple enough to be easily communicated, but the CNN may be needed to conduct a highly intricate chain of operations in order to accomplish the task. As a consequence, there is a trade-off between how realistic the description is of the internal dynamics of the network and how explainable it is [3].

Image classification systems are prone to adversarial attacks [4], which add tiny alterations to photos, leading these algorithms to make erroneous predictions. These

attacks are extremely powerful against even the most successful convolutional network-based systems [5,6], despite the perturbations being typically unnoticed or regarded as little "noise" in the image. The success of adversarial attacks puts real-world convolutional network applications at risk of security breaches. The network performs calculations that are very different from those made by human brains. As provided below, the Figures depict original and attack images with a feature-map and also its attacked feature-map as shown in Figure 1.



(**a**) (**b**) (**c**) (**d**)

Figure 1. (**a**) Original image (**b**) feature-map (**c**) adversarial attack (**d**) attack feature-map.

This research describes adversarial attack detection on image classification or prediction algorithms such as AlextNet, VGGNet, and ResNet. It further provides different feature-map detection methods and proposes to exploit a feature-map that works well on individual objects in an image possible to use.

To summarize, the following are the major contributions: (1) it presents an adversarial attack using the "Wherefore? What? and How?" concepts. Researchers can quickly and effectively establish adversarial attack awareness using the proposed method; (2) it discusses the strengths and flaws of the proposed methods to explore each for a specific application; and (3) it also describes different types of attacks on image pixel and its detection methods.

2. Related Work

Agarwal et al. [7] proposed a non-DL approach which involves searching for features using a set of well-known image transforms, such as the Discrete Wavelet Transform, the Discrete Sine Transform, and classifying the features using a support-vector-machine-based classifier. The detection algorithm yielded at least 84.2% and 80.1% detection accuracy under seen and unseen database test settings, respectively. Additionally, they also emphasized how the impact of the adversarial perturbation can be neutralized using a wavelet-decomposition-based filtering method of denoising.

Guesmi et al. [8] carried out defensive approximation classification with the same level of accuracy without the application of retraining. The convolution neural network's resources and energy consumption were also decreased by using an approximation computing model. If a strong transferability-based attack on a LeNet-5 or Alexnet, the proposed implementation makes them more resistant to 99% and 87%, respectively, as the approximate logic implementation is easier to use.

Higashi et al. [9] used the sensitivities of image classifiers to offer a unique technique for identifying adversarial pictures that is fast and accurate. Due to the fact that adversarial pictures are formed by adding noise, they concentrate on the behavior of the outputs of image classifiers for images that have been filtered differently. When the strength of an image classifier is increased, the output of a SoftMax function in the classifier is substantially altered in the case of adversarial images, but the output is rather steady in

the case of normal images. They also explored the operation-oriented properties of several noise reduction techniques.

Tsingenopoulos, et al. [10] proposed an auto attacker revolutionary reinforcement learning system in which agents learn how to operate around a black-box model by questioning it, extracting the underlying decision behavior efficiently, and successfully undermining it. The auto attacker is a first-of-its-kind reinforcement learning system that makes no assumptions about the differentiability or structure of the underlying function. As a result, it is resistant to standard defenses such as gradient obfuscation and adversarial training. The auto-attacker revolutionary reinforcement learning method works well even if the output is less descriptive or missing.

Xie et al. [11] developed innovative network topologies that promote adversary robustness by conducting feature denoising in order to improve adversarial resilience. They have specifically designed blocks in their networks that denoise the features by using non-local techniques or other filters. The complete network is trained from beginning to finish whenever new blocks come into the network. The designed system can be used in both white-box and black-box assault contexts. The ImageNet technique achieves 55.7% accuracy under 10-iteration on projected gradient descent white-box attacks, while the previous algorithm achieves 27.9% accuracy. While under severe 2000-iteration on projected gradient descent white-box attacks, their method achieved 42.6% accuracy.

Miller et al. [5] discussed an overview of the attack anomaly detection system by considering a successful, targeted test-time evasion attack example that was developed with a "clean" example x from a parent class cs and perturbing it until the perturbed example's judgement is identical to the source class cs. They also carried out work on test-time evasion (TTE), data poisoning (DP), backdoor DP, and reverse engineering (RE) attacks and particularly defenses against the anomaly detection.

Vargas and Su [12] considered an assault in which a pixel is selected at random to be perturbed by the same amount that may be used to cause an attack. Propagation Maps and Locality Analysis are two probable tools for understanding a one-pixel attack. In the CIFAR dataset, the average of PMean over 318 successful assaults was calculated on ResNet out of 1000 trials. Every time an attack is successful, it is carried out again at the same spot on the picture. This process is repeated 5000 times on the CIFAR-10 dataset to achieve a statistically significant result.

Ye et al. [13] improved model interpretability, by introducing the saliency map approach which is analogous to bringing the process of attention to the model in order to grasp the progress of object recognition by deep networks. Then, to identify hostile cases, provide a new saliency map that is integrated with additional sounds and makes use of the inconsistent strategy. Using several example adversarial attacks on common data sets, such as ImageNet and popular models, they discovered that their technique was capable of detecting all of the assaults with a high detection success rate successfully.

Jia and Gong [2] used adversarial scenarios, to address the potential and problems of defending against ML-equipped inference attacks. The existing adversarial example construction approaches fall short because they do not take into account the specific problems and needs for producing noise when fighting against inference attacks. They used the example of defending against inferential attacks on online social networks to show the potential and obstacles that might be encountered.

Momeny et al. [4] suggested an approach that may be used to test the resilience of CNNs to several forms of noise at the same time. For the restoration of noisy pictures, it does not need any preprocessing. The suggested Nested-Residual Guided CNN for the classification of noisy pictures has a reduced time complexity when compared to other methods. Further, it is better at classifying images compared to other methods because of its higher accuracy and efficiency.

Akhtar and Mia [14] presented the first thorough overview of adversarial attacks on DL in computer vision, which includes both theoretical and practical considerations. They carried out the design of adversarial assaults, investigated the presence of such attacks,

and proposed countermeasures. Their contribution further explored the feasibility of adversarial attacks in real-world situations.

Martins et al. [15] provided the analysis of adversarial attacks, which is referred to as adversarial ML. They are widely researched in the areas of picture classification and spam detection. They concluded that adversarial attack approaches were successful in both virus and intrusion situations, with a large variety of tactics having been examined. They also determined adversarial defensive strategies that have not yet been properly investigated.

Harder et al. [16] demonstrated how Fourier analysis of input photos and feature-maps may be utilized to identify benign test samples from adversarial images by comparing the Fourier transforms. They suggested two innovative detection methods. First, they present a technique for detecting acoustic signals that is based on acoustic signals; which makes use of the magnitude spectrum of the input pictures. The second technique expands on the first by extracting the phase of Fourier coefficients of feature-maps at various levels. They were able to increase adversarial detection rates as a result of this enhancement when compared to current best-practices detectors.

Panda et al. [6] constructed the dimensionality of inputs or parameters in a network, on the surface, seems to restrict the "space" in which adversarial instances are found. They show that, guided by their intuitive understanding, discretization significantly increases the resilience of DL networks against adversarial attacks. Their work is more specific, discretizing the input space significantly increases the adversarial resilience of the DL network over a wide variety of perturbations while causing only a small loss inaccuracy.

Sutanto and Lee [17] offered an approach for detecting adversarial noise that does not require prior knowledge of the kind of adversarial noise used by the adversary. In order to do this, they offer a blurring network that is trained solely with normal pictures. They also recommended that it be used as the starting condition of the deep image prior network. The usage of adversarial noisy pictures for training the neural network is not required in other neural network-based detection approaches, which need the use of a large number of adversarial noisy photos.

Izmailov et al. [18] investigated the following two issues. How can an adversary take advantage of the geometric and statistical aspects of data distribution when the assault is of a certain scale and scope? When constructing a decision rule, what countermeasures may be utilized to prevent it from malicious distribution shift within the specified size of the attack? Even though we do not supply a complete answer to the problem, we do gather and interpret the observations in a way that can be used to make better decisions about the design of ML algorithms.

Raju and Lipasti [19] studied and proposed BlurNet as a protection against the Robust Physical Perturbation (RP-2) attacks. First, they provided evidence to support their case by carrying out a frequency analysis of the first layer feature-maps of the network using the LISA dataset. The RP-2 technique introduces high-frequency noise into the input picture during the training process. They conducted a black-box transfer attack in order to demonstrate that low-pass filtering the feature-maps is more advantageous than filtering the input data. Several regularization strategies were discussed for incorporating this low-pass filtering characteristic into the network's training regime, as well as white-box assaults on the network. In the end, they carried out an adaptive assault assessment and described that when total variation regularization, one of the recommended countermeasures, is used, the success rate of the attack drops from 90% to 20%.

De Silva et al. [1] have suggested a solution to protect ML algorithms against test data fabrication with a common assumption that feature entries of test data are equally susceptible to falsification. The researchers describe an adversarial learning approach that takes into consideration the susceptibility features of test data entries while developing an attack-resilient classifier.

Chen et al. [20] reviewed literature primarily and conducted adversarial research on specific application scenarios. They generated adversarial examples by adding perturbations to the information carrier in order to realize the adversarial attack on reinforcement

learning systems. They gave a detailed overview of the literature on adversarial attacks in several fields of reinforcement learning applications, along with the best ways to defend against them.

Chai and Velipasalar [21] offered an approach to identify the adversarial instances created by several adversarial assaults, including the dispersion reduction technique, projected gradient descent technique, varied inputs method, and momentum iterative fast gradient sign technique. Their method, which uses one dimensional Gabor filter responses, is very good at figuring out adversarial samples made from a wide range of surrogate neural network models and datasets.

Jang et al. [22] proposed that students learn how to construct hostile instances by using the generator. They offer a recursive and stochastic generator that, in contrast to previous systems that generate a one-shot perturbation via a deterministic generator, creates significantly stronger and more diversified perturbations that thoroughly show the susceptibility of the target classifier. Tests on the MNIST and CIFAR-10 datasets confirmed that the classifier adversarial trained with their method outperforms the classifier trained with a recursive and stochastic generator method under a variety of white-box and black-box attacks.

Akhtar and Dasgupta [23] presented a quick review of diverse hostile and security measures. Frameworks in the first category are aimed at increasing the resilience of DNNs in order to appropriately classify AEs. For example, adversarial training, which entails training the ML approaches with both clean and malicious samples. A single step update along the sign of the gradient of a loss function necessary to the sample is used to construct the attack elements. Natural generative adversarial networks (NGANs) are generative adversarial networks (GANs) that use generative adversarial networks (GANs) to produce AAs by minimizing the distance between the inner representations.

In summary, it can be said that previous researches [1,5,13] were computationally costly because they were based on the pre-generation of adversarial samples. The limited-memory Broyden–Fletcher–Goldfarb–Shanno algorithm [8,13], the rapid gradient sign technique [9,11], projected gradient descent [9,11], distributional adversarial assault [24], and DeepFool [25] were all employed in the frameworks of these threat models. Adversarial training, which aims to increase the resilience of the DL model by integrating adversarial samples into the training stage, is currently the most effective heuristic defense. This research came up with an exploit feature-map method that uses picture weight data to figure out the object weight feature-map to get attack parts.

3. Adversarial Machine Learning (ML)

When using adversarial ML, an attempt is made to trick the model by giving it erroneous information. The most frequent reason is to induce a malfunction in an ML model [15], which is the most prevalent kind of malfunction. Alternatively, it is the process of optimizing the ML model by recognizing what it is meant to do and how it may be attacked while executing its job, and then coming up with solutions to minimize those attacks. There are two ways in which attacks can be classified.

3.1. Black-Box Attack

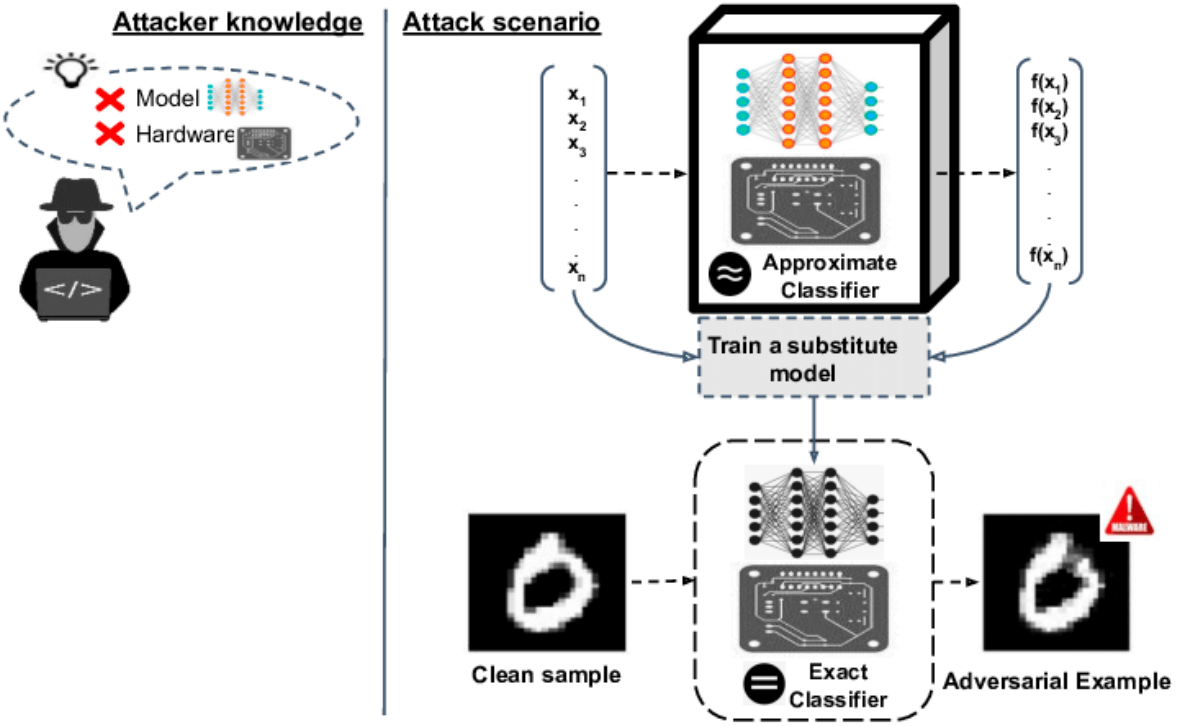
During a blind attack, a person does not know the model or how it works. They do not have access to its gradients or parameters, which is depicted in Figure 2a.

3.2. White-Box Attack

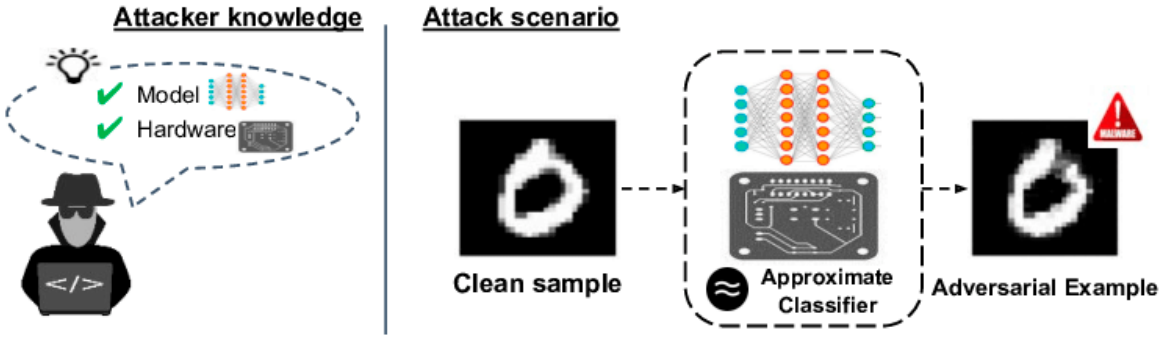
The opposite of this scenario is one where the attacker has full access to the model's parameters and its gradients, which are shown in Figure 2b.

The use of adversarial assaults on image classification systems introduces minor perturbations to pictures, causing the algorithms to make inaccurate predictions in the future. Adversarial perturbations, although minor in the pixel space, cause a significant amount of "noise" in the network's feature-maps, which may be extremely difficult to

detect. The deep neural network (DNN) will be readily fooled when it comes to the prediction stage. In order to apply defensive measures to it, it is necessary to analyze these feature-maps and detect any tiny perturbations.



(**a**)



(**b**)

Figure 2. Attacks scenario [8]. (**a**) Black-Box Attack Scenario. (**b**) White-Box Scenario.

4. Methodology

4.1. ImageNet Dataset

ImageNet is an image database structured according to the WordNet hierarchy, in which each node of the hierarchy is represented by hundreds of thousands of photos. As of right now, each node has an average of around five hundred photographs on it. This site will be very useful for researchers, teachers, students, and others. (https://www.kaggle.com/c/imagenet-object-localization-challenge/overview, (accessed on 17 May 2022)).

4.2. Guided Propagation Model

ImageNet is an image database structured according to the WordNet hierarchy. GuidedBP [9] is a variation on raw gradient backpropagation in that it propagates gradients

back to inputs and uses the received gradients as the saliency values of the inputs. The main difference between the two techniques is how they deal with ReLU layers.

In GuidedBP,

$$G^l = G^{l+1} * 1_{G^{l+1}>0 \text{ and } x^l>0} \quad (1)$$

G^l denotes the gradients of the lth layer, x^l denotes the activations of the layers preceding the ReLU layer, and **1** denotes the indicator function. It is possible that certain input features may get zero gradients because the indicator function filters out portions of the gradients; this is referred to as the filtering effect (FE). The filtering effect of a simple moving average (SMA) is defined technically as follows:

$$s^m * 1_{s^m>0} \quad (2)$$

where s^m is saliency map GuidedBP is discussed in detail in [12], which is a theoretical study. They demonstrated that the filtering effects of the SMA of various classes are comparable, indicating that GuidedBP is not discriminatory based on class membership. Specifically, they demonstrate that the SMA generated by GuidedBP includes class-discriminative information and provide a straightforward method for enhancing the discriminative information in the accompanying saliency maps. It is common practice to use the pre-SoftMax scores (logits) as output scores when creating SMA. Following the previous attribution approaches, it can be shown that distinct classes of scores may be assigned to the same pixels. They provide an explanation of how the scores are generated. The technique reveals where the difference between logits originates from, and this is the precise reason why the network predicts a greater probability for a certain class rather than another one, as explained in the previous section. The loss of the neural network is enhanced throughout the optimization of the process of constructing adversarial pictures, resulting in a change in the rank of logits. Their method can identify the evidence for the difference in scores, i.e., the rank of logits, between two groups of scores. Misclassifications are caused by a change in the rank of the individual. As a result, the upgraded GuidedBP may provide a more thorough explanation of the classification judgments made on adversarial photos.

4.3. Smooth Grad

Smoothgrad has two hyper-parameters: ∂, the noise level or standard deviation of the Gaussian perturbations and n, the number of samples to average over. The smoothed gradient $M_c(x)$, over random samples in a neighborhood of an input x is,

$$M_c(x) = \frac{1}{n} \sum_i^n M_c\left(x + N(0, \partial^2)\right) \quad (3)$$

where n, is the number of samples, and $N\left(0, \partial^2\right)$ represents Gaussian noise with standard deviation ∂. In this study, the influence of noise level was seen for numerous sample photos from ImageNet [21]. They find that adding 10 percent to 20 percent noise (middle columns) seems to balance the sharpness of the sensitivity map while maintaining the structure of the original picture, according to their findings. Moreover, they point out that, although this range of noise produces generally favorable results for inception, the optimal noise level is dependent on the input signal. The significance of the sample size, n. In accordance with expectations, the estimated gradient grows smoother as the number of samples, n, rises. They discovered experimentally that for $n > 50$, there was little apparent change in the visualizations, indicating a declining return.

4.4. Guided-CAM Mapping

A map of the location of an image categorization system uses a special kind of architecture in which global average pooling convolutional feature-maps are sent straight into SoftMax rather than via a pipeline. Allow the penultimate layer to generate K feature-maps to be more specific to

$$A^k \epsilon \, \mathbb{R}^{uxv} \quad (4)$$

where each element indexed by i, j. So, A_{ij}^{k} refers to the activation at location (i, j) of the feature-map A^{k}. These feature-maps are then geographically pooled using Global Average Pooling (GAP) and linearly processed to give a score, Y^{c}, for each class c, which is subsequently used to determine the location of the feature-maps.

$$Y^{c} = \sum_{k} w_{k}^{c} \frac{1}{z} \sum_{i} \sum_{j} A_{ij}^{k} \tag{5}$$

It is possible to swap the order of summations in the SCAM algorithm to yield SCAM, which can then be used to generate the localization map for customized image classification systems. Let us define F^{k} to be the global average pooled output

$$F^{k} = \frac{1}{z} \sum_{i} \sum_{j} A_{ij}^{k} w_{k}^{c} \tag{6}$$

Keep in mind that this change in architecture necessitated retraining since not all designs contain w_{k}^{c} weights linking the features maps to the outputs. When Grad-CAM is applied to these structures, the result is, $A_{ij}^{k} = w_{k}^{c}$ which makes Grad-CAM a strict generalization of the CAM algorithm. A further application of the previously mentioned generalization is the generation of visual explanations using CNN-based models that cascade convolutional layers with far more intricate interconnections. A Convolutional Neural Network (ConvNet/CNN) is a DL algorithm which can take in an input image, assign importance (learnable weights and biases) to various aspects or objects in the image, and be able to differentiate one from the other. Indeed, we use Grad-CAM to tasks that go beyond classification, such as picture captioning and visual recognition models that make use of CNNs.

4.5. Inverted Image Representations

Introducing approach for computing an approximate inverse of an image representation, which is discussed in detail in this section. This is expressed as the challenge of identifying a picture whose representation is the most similar to the one provided. To put it another way, given a representation function: $\Phi = \mathbb{R}^{HXWXC} \rightarrow \mathbb{R}^{d}$ and an inverted representation $\Phi_{0} = \Phi(x_{0})$, reconstruction finds the image $x \in \mathbb{R}^{HXWXC}$ that reduces the goal to the smallest possible value.

$$X^{*} = \underbrace{argmin}_{X \; E \; \mathbb{R}^{HXWXC}} \; l(\Phi(\mathrm{x}), \Phi_{0}) + \beth * R(x) \tag{7}$$

In which the loss l compares the image representation $\Phi(x)$, to the target representation Φ_{0} and $R: \mathbb{R}^{HXWXC} \rightarrow \mathbb{R}$ is a regularize that captures a *natural image prior*. From the perspective of the representation, minimization (equation) results in a picture X^{*} that "resembles" the image x_{0} produced by minimization. There may not be a single answer to this question but sampling the space of alternative reconstructions can be used to figure out the space of pictures that the representation thinks are the same, revealing the invariances of its decision.

5. Novel Exploit Feature-Map Approach

The novel exploit feature-map which is divided into two parts: the first part will detect each and every object map on the input picture, as shown in Figure 3a. The second part will combine all of the parts of the feature-map to achieve the final detection on the input image, as shown in Figure 3b.

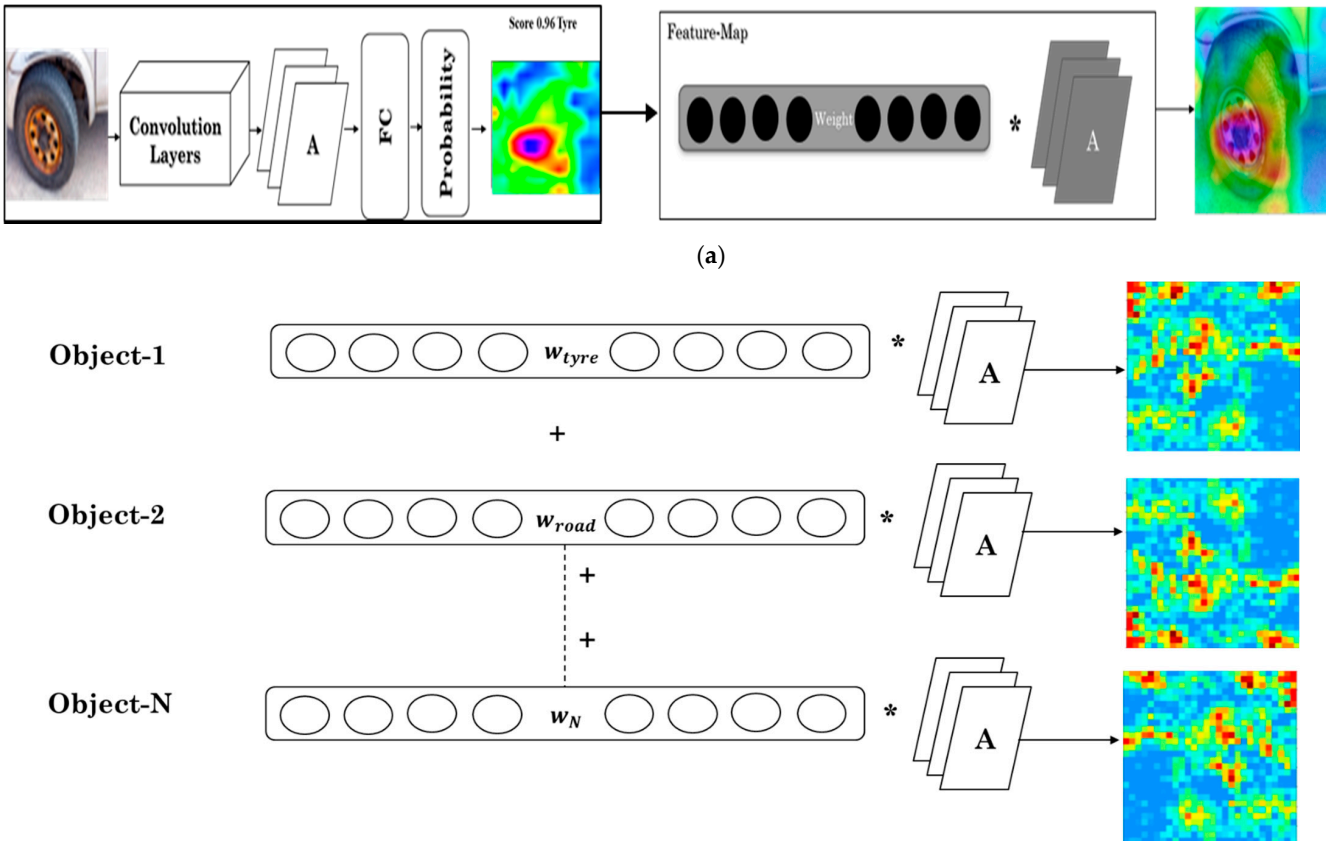


Figure 3. (**a**) Exploit feature-map basic idea and (**b**) exploit map into objects. * Activation map.

In order to define an explanatory rule for a black box *f(x)*, one must start by specifying which variations of the input x will be used to study f. The aim of saliency is to identify which regions of an image x_0 are used by the black box to produce the output value $f(x_0)$. We can do so by observing how the value of $f(x)$ changes as x is obtained "deleting" different regions R of x_0. For example, if $f(x_0) = +1$ denotes a robin image, we expect that $f(x) = +1$ as well unless the choice of R deletes the robin from the image. Given that x is a perturbation of x_0, this is a local explanation, and we expect the explanation to characterize the relationship between f and x_0.

While conceptually simple, there are several problems with this idea. The first one is to specify what it means by "delete" information. We are generally interested in simulating naturalistic or plausible imaging effect, leading to more meaningful perturbations and hence explanations. Since we do not have access to the image generation process, we consider three obvious proxies: replacing the region R with a constant value, injecting noise, and blurring the image.

Formally, let $m : \Lambda \rightarrow [0,1]$ be a *mask*, associating each pixel $u \in \Lambda$ with a scalar value $m(u)$. Then the perturbation operator is defined as

$$[\Phi(X_0; m)](u) = \begin{cases} m(u)x_0(u) + (1 - m(u))\mu_0, \text{ Constant} \\ m(u)x_0(u) + (1 - m(u))n(u), \text{ Noise} \\ \int g_{\sigma_0} m(u)\ (v - u)x_0(v)dv, \text{ Blur} \end{cases} \quad (8)$$

where μ_0 represents the average color, $n(u)$ represents the number of i.i.d. Gaussian noise samples for each pixel, and σ_0 represents the maximum isotropic standard deviation of the Gaussian blur kernel g_σ (we choose $\sigma_0 = 10$, which results in a highly blurred picture). One feature of the proposed approach is that the produced visualizations are obviously exploiting adversarial assaults, which is a significant advantage. When one plays the

deletion game, a minimum mask is made that stops the item from being recognized by the network.

6. Results and Discussion

This section will discuss different methods of feature-map visualization with and without adversarial attacks. In the end, all the methods are compared using time and error rate.

6.1. VGGNet-19 Model with ImageNet Dataset

Figure 4a–e show VGG-19 pre-trained network for ImageNet dataset results for car tire images. Each and every model did not have an indication on the feature-map. While novel exploit feature-map techniques give different attack image features.

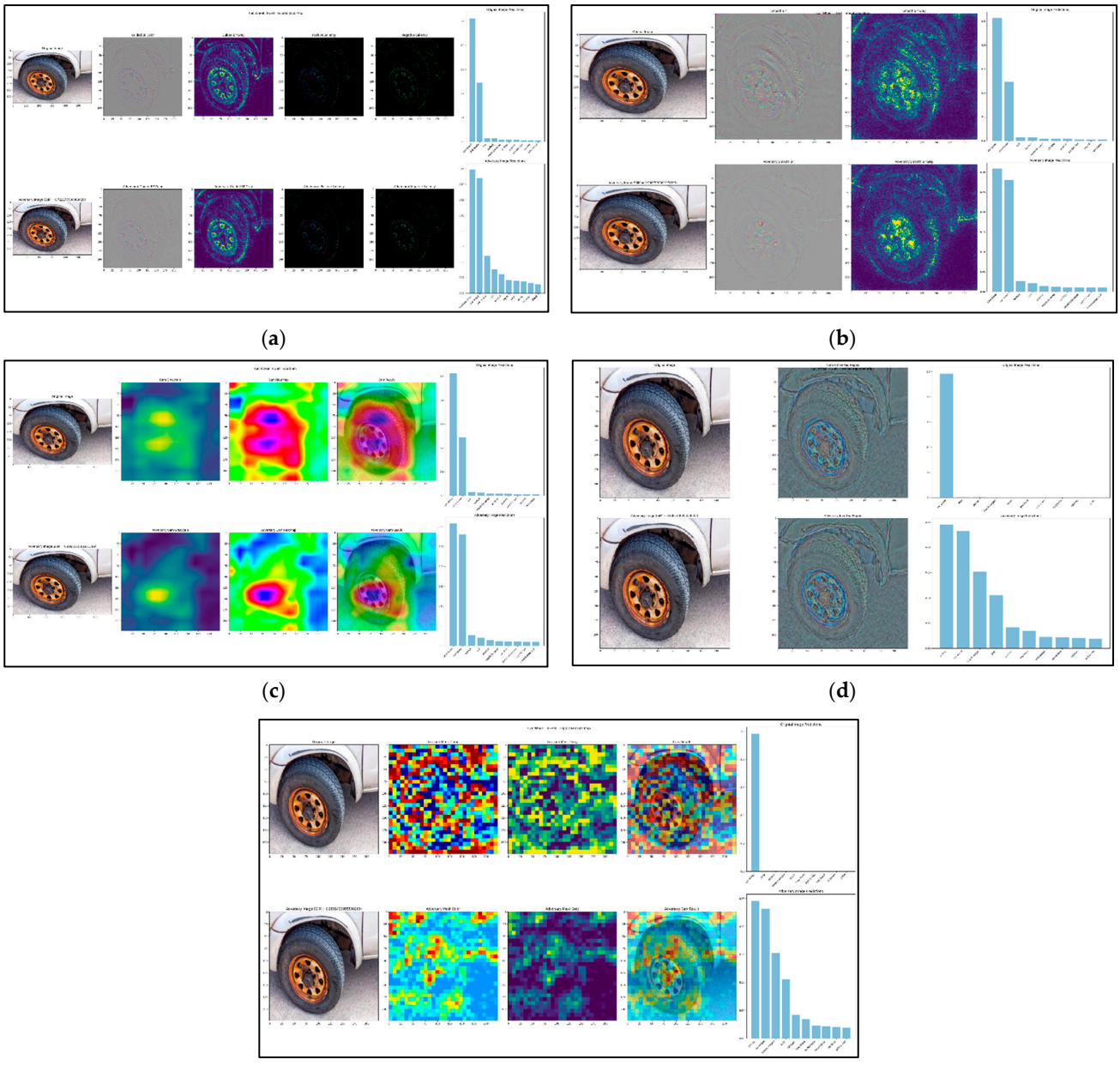


Figure 4. (**a**) Guided propagation model; (**b**) smooth grad; (**c**) guided class activated mapping; (**d**) inverted gradient; and (**e**) novel exploit feature-map.

6.2. ResNet-50 Model with ImageNet Dataset

Figure 5a–e show ResNet-50 pre-trained network for ImageNet dataset results for car tire image. Each and every model did not have an indication on the feature-map, while novel exploit feature-map techniques give different attack image features.

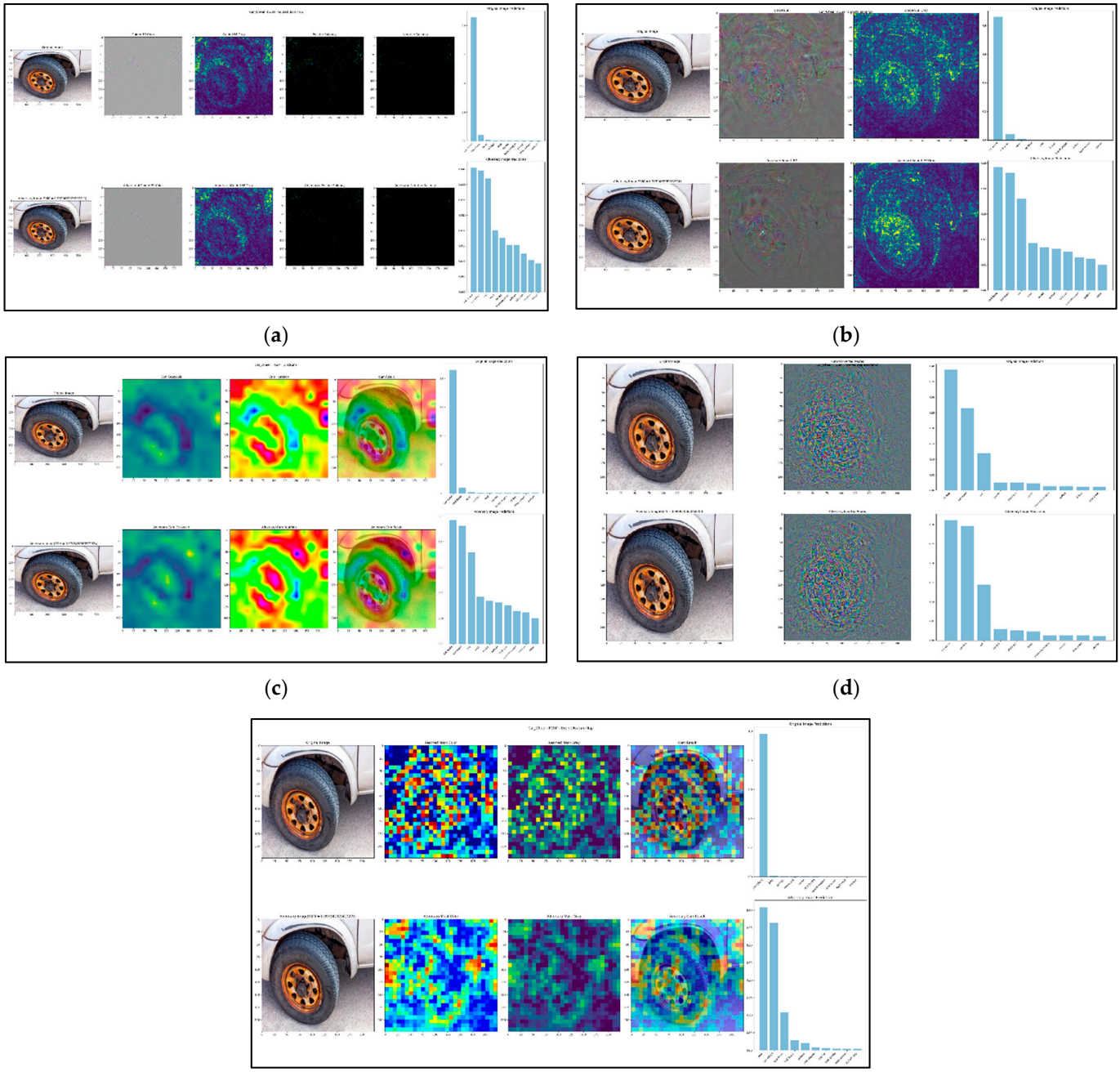


Figure 5. (**a**) Guided propagation model; (**b**) smooth grad; (**c**) guided class activated mapping; (**d**) inverted gradient; and (**e**) novel exploit feature-map.

The main difficulty faced was the weight selection of different layers in class activation-map, epoch and image-size conversation. While applying the novel exploit feature-map technique, the graphical processing unit (GPU) is required with a minimum configuration of 12 GB of RAM and 500 GB of hard disk. It is preferred to use the free Google Colab Cloud services.

6.3. Analysis

- Time: It specifies the maximum amount of time that a job may use the processor. In other words, it specifies the maximum CPU usage time allowed for the job to execute. It is an optional parameter. It can be coded at the job level and step level too.
- Error rate: The error rate is expressed as a ratio and is calculated by dividing the total number of true images classified by the total number of errors made.

$$\partial = \left| \frac{V_A - V_E}{V_E} \right| * 100\ \% \tag{9}$$

where ∂: % Error, V_A: Actual value observation and V_E : Expected error.

Table 1 depicts parameter analysis and provides a comparative study of existing algorithms, i.e., image transformation [3], discretization [14], spectral defense [16], smooth suided [6], Guided-CAM [19], BlurNet [1], inverted representation [21], guided back propagation [23] with novel exploit feature-map using time and error rate. Among them, the novel exploit feature-map works efficiently for white box and black box attacks in less time.

Table 1. Parameter analysis.

No	Method	Model Supported	Attacks Supported	Execution Time Average(s)	Error Rate (%) 50 Samples
1	**Image Trasformation** [3]	CNN	White Box	≈540 s	≈36.6%
2	**Discretization** [14]	CNN	White Box	≈630 s	≈38.9%
3	**Spectral Defence** [16]	CNN	White Box	≈720 s	≈26.2%
4	**Smooth Guided** [6]	AlexNet	White Box	≈840 s	≈66.4%
5	**Guided-CAM** [19]	ResNet50	White Box, Black Box	≈960 s	≈48%
6	**BlurNet** [1]	ResNet50	White Box, Black Box	≈870 s	≈36%
7	**Inverted Representation** [21]	AlexNet	White Box	≈660 s	≈76%
8	**Guided Back Propagation** [23]	ResNet50	White Box	≈720 s	≈46.8%
9	**Novel Exploit Feature-Map**	VGG19, AlexNet, ResNet50	White Box, Black Box	≈620 s	≈28.42%

7. Conclusions

According to this study, a thorough, formal framework for learning explanations as a meta-predictor has been developed. In addition, a unique image exploit feature-map paradigm is introduced here, which teaches an algorithm where to look by evaluating which parts of an image have the most influence on its result when it is disturbed. The suggested approach, in contrast to many mapping methods, makes explicit alterations to the picture, making it more exploitable and testable. When compared to other approaches, novel exploit feature-map works in less than 8–9 min, indicating that it has a modest temporal complexity. Whereas the error rate for existing techniques is not less than 40%, our method has the best performance of 50 samples, with an error rate of 28.42%, while the error rate for existing approaches is 40%. As a result, the defensive filtering technique will be used in the future, which will denoise the assault pixel. A method that can be used for any ML model should also be developed to provide protection against both black-box and white-box threats. It is revealed that there is currently no defensive mechanism exist that is. efficient and effective when dealing with hostile samples. Adversarial training, which is the most effective defensive mechanism, is too computationally costly for practical deployment, and several efficient heuristic defenses have been shown to be susceptible to adaptive white-box adversaries.

In the future it is possible to improve the interpretability of deep networks. For instance, how can the algorithm be used to guard against assault in gray-box situations? We anticipate that future research will address these concerns in depth.

Author Contributions: Conceptualization, A.S.A., D.V. and V.V.K.; methodology, A.S.A., D.V. and V.V.K.; software, D.V. and V.V.K.; validation, D.V., M.R.N.M.Q. and V.V.K.; formal analysis, D.V. and V.V.K.; investigation, D.V. and V.V.K.; resources, A.S.A.; data curation, A.S.A., K.M.R.Q., M.R.N.M.Q. and E.A.M.; writing—D.V., V.V.K. and E.A.M., K.M.R.Q.; writing—review and editing, M.R.N.M.Q.; visualization D.V. and V.V.K.; supervision, A.S.A., D.V. and V.V.K. project administration, A.S.A. and E.A.M.; funding acquisition, A.S.A. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by the Deanship of Scientific Research, King Khalid University, Kingdom of Saudi Arabia, and the grant number is R.G.P.2/178/43.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Not applicable.

Acknowledgments: We would like to express our gratitude to the Deanship of Scientific Research, King Khalid University, Kingdom of Saudi Arabia for funding this work, as well as family, friends, and colleagues for their constant inspiration and encouragement.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. De Silva, S.; Kim, J.; Raich, R. Cost Aware Adversarial Learning Shashini De Silva, Jinsub Kim, and Raviv Raich School of EECS, Oregon State University, Corvallis, Oregon, US 97331. In Proceedings of the (ICASSP 2020) 2020 IEEE International Conference on Acoustics, Speech, and Signal Processing, Virtual Conference, 4–9 May 2020; pp. 3587–3591.
2. Jia, J.; Gong, N.Z. Defending Against Machine Learning Based Inference Attacks via Adversarial Examples: Opportunities and Challenges. *Adapt. Auton. Secur. Cyber Syst.* **2020**, 23–40. [CrossRef]
3. Xue, M.; Yuan, C.; Wu, H.; Zhang, Y.; Liu, W. Machine Learning Security: Threats, Countermeasures, and Evaluations. *IEEE Access* **2020**, *8*, 74720–74742. [CrossRef]
4. Momeny, M.; Latif, A.M.; Agha Sarram, M.; Sheikhpour, R.; Zhang, Y.D. A noise robust convolutional neural network for image classification. *Results Eng.* **2021**, *10*, 100225. [CrossRef]
5. Miller, D.J.; Xiang, Z.; Kesidis, G. Adversarial Learning Targeting Deep Neural Network Classification: A Comprehensive Review of Defenses against Attacks. *Proc. IEEE* **2020**, *108*, 402–433. [CrossRef]
6. Panda, P.; Chakraborty, I.; Roy, K. Discretization Based Solutions for Secure Machine Learning Against Adversarial Attacks. *IEEE Access* **2019**, *7*, 70157–70168. [CrossRef]
7. Agarwal, A.; Singh, R.; Vatsa, M.; Ratha, N.K. Image Transformation based Defense Against Adversarial Perturbation on Deep Learning Models. *IEEE Trans. Dependable Secur. Comput.* **2020**, *5971*, 2106–2121. [CrossRef]
8. Guesmi, A.; Alouani, I.; Khasawneh, K.N.; Baklouti, M.; Frikha, T.; Abid, M.; Abu-Ghazaleh, N. *Defensive Approximation: Securing CNNs Using Approximate Computing*; Association for Computing Machinery: New York, NY, USA, 2021; Volume 1, ISBN 9781450383172.
9. Higashi, A.; Kuribayashi, M.; Funabiki, N.; Nguyen, H.H.; Echizen, I. Detection of Adversarial Examples Based on Sensitivities to Noise Removal Filter. In Proceedings of the 2020 Asia-Pacific Signal and Information Processing Association Annual Summit and Conference, Virtual Conference, 1 July 2020; pp. 1386–1391.
10. Tsingenopoulos, I.; Preuveneers, D.; Joosen, W. AutoAttacker: A reinforcement learning approach for black-box adversarial attacks. In Proceedings of the 2019 IEEE European Symposium on Security and Privacy Workshops (EuroS&PW), Stockholm, Sweden, 17–19 June 2019; pp. 229–237. [CrossRef]
11. Xie, C.; Wu, Y.; van der Maaten, L.; Yuille, A.L.; He, K. Feature denoising for improving adversarial robustness. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Long Beach, CA, USA, 16–20 June 2019; pp. 501–509. [CrossRef]
12. Vargas, D.V.; Su, J. Understanding the one-pixel attack: Propagation maps and locality analysis. In Proceedings of the 2020 Workshop on Artificial Intelligence Safety, AISafety 2020, Yokohama, Japan, 11–12 July 2020; Volume 2640.
13. Ye, D.; Chen, C.; Liu, C.; Wang, H.; Jiang, S. Detection defense against adversarial attacks with saliency map. *Int. J. Intell. Syst.* **2021**. [CrossRef]
14. Akhtar, N.; Mian, A. Threat of Adversarial Attacks on Deep Learning in Computer Vision: A Survey. *IEEE Access* **2018**, *6*, 14410–14430. [CrossRef]
15. Martins, N.; Cruz, J.M.; Cruz, T.; Henriques Abreu, P. Adversarial Machine Learning Applied to Intrusion and Malware Scenarios: A Systematic Review. *IEEE Access* **2020**, *8*, 35403–35419. [CrossRef]
16. Harder, P.; Pfreundt, F.J.; Keuper, M.; Keuper, J. SpectralDefense: Detecting Adversarial Attacks on CNNs in the Fourier Domain. In Proceedings of the 2021 International Joint Conference on Neural Networks (IJCNN), Shenzhen, China, 18–22 July 2021. [CrossRef]

17. Sutanto, R.E.; Lee, S. Real-time adversarial attack detection with deep image prior initialized as a high-level representation based blurring network. *Electronics* **2021**, *10*, 52. [CrossRef]
18. Izmailov, R.; Sugrim, S.; Chadha, R.; McDaniel, P.; Swami, A. Enablers of Adversarial Attacks in Machine Learning. In Proceedings of the 2018 IEEE Military Communications Conference (MILCOM), Los Angeles, CA, USA, 29–31 October 2018; pp. 425–430. [CrossRef]
19. Raju, R.S.; Lipasti, M. BlurNet: Defense by Filtering the Feature Maps. In Proceedings of the 2020 50th Annual IEEE/IFIP International Conference on Dependable Systems and Networks Workshops (DSN-W), Valencia, Spain, 29 June–2 July 2020; pp. 38–46. [CrossRef]
20. Chen, T.; Liu, J.; Xiang, Y.; Niu, W.; Tong, E.; Han, Z. Adversarial attack and defense in reinforcement learning-from AI security view. *Cybersecurity* **2019**, *2*, 11. [CrossRef]
21. Chai, W.; Velipasalar, S. Detecting Adversarial Images via Texture Analysis. In Proceedings of the 2020 54th Asilomar Conference on Signals, Systems, and Computers, Pacific Grove, CA, USA, 1–4 November 2020; pp. 215–219. [CrossRef]
22. Jang, Y.; Zhao, T.; Hong, S.; Lee, H. Adversarial defense via learning to generate diverse attacks. In Proceedings of the 2019 IEEE/CVF International Conference on Computer Vision (ICCV), Seoul, Korea, 27 October–2 November 2019; pp. 2740–2749. [CrossRef]
23. Akhtar, Z.; Dasgupta, D. *A Brief Survey of Adversarial Machine Learning and Defense Strategies*; Technical Report No. CS-19-002; The University of Memphis: Memphis, TN, USA, 2019; p. 11.
24. Quiring, E.; Rieck, K. Adversarial machine learning against digital watermarking. In Proceedings of the 2018 26th European Signal Processing Conference (EUSIPCO), Rome, Italy, 3–7 September 2018; pp. 519–523. [CrossRef]
25. Sadeghi, K.; Banerjee, A.; Gupta, S.K.S. A System-driven taxonomy of attacks and defenses in adversarial machine learning. *IEEE Trans. Emerg. Top. Comput. Intell.* **2020**, *4*, 450–467. [CrossRef] [PubMed]