mathematics

MDPI

Article

Fortify the Guardian, Not the Treasure: Resilient Adversarial Detectors

Raz Lapid [1,2], Almog Dubin [2] and Moshe Sipper [1,*]

[1] Department of Computer Science, Ben-Gurion University, Beer-Sheva 8410501, Israel; razla@post.bgu.ac.il
[2] DeepKeep, Tel-Aviv 6701203, Israel; almog@deepkeep.ai
* Correspondence: sipper@bgu.ac.il

Abstract: Adaptive adversarial attacks, where adversaries tailor their strategies with full knowledge of defense mechanisms, pose significant challenges to the robustness of adversarial detectors. In this paper, we introduce RADAR (Robust Adversarial Detection via Adversarial Retraining), an approach designed to fortify adversarial detectors against such adaptive attacks while preserving the classifier's accuracy. RADAR employs adversarial training by incorporating adversarial examples—crafted to deceive both the classifier and the detector—into the training process. This dual optimization enables the detector to learn and adapt to sophisticated attack scenarios. Comprehensive experiments on CIFAR-10, SVHN, and ImageNet datasets demonstrate that RADAR substantially enhances the detector's ability to accurately identify adaptive adversarial attacks without degrading classifier performance.

Keywords: robustness; adversarial attacks; adaptive adversarial attacks; deep learning

MSC: 68T07

check for updates

Citation: Lapid, R.; Dubin, A.; Sipper, M. Fortify the Guardian, Not the Treasure: Resilient Adversarial Detectors. *Mathematics* **2024**, *12*, 3451. https://doi.org/10.3390/math12223451

Academic Editors: Jing Zhang and Chenyang Bu

Received: 9 October 2024
Revised: 30 October 2024
Accepted: 1 November 2024
Published: 5 November 2024

Copyright: © 2024 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (https://creativecommons.org/licenses/by/4.0/).

1. Introduction

Rapid advances in deep learning systems and their applications in various fields, such as finance, healthcare, and transportation, have greatly enhanced human capabilities [1–5]. However, these systems continue to be vulnerable to adversarial attacks, where unintentional or intentionally designed inputs—known as *adversarial examples*—can mislead the decision-making process [6–16]. These malicious inputs, known as adversarial examples, are typically generated by applying subtle perturbations to legitimate data samples—perturbations that are often imperceptible to human observers but can significantly alter the output of the model. Such adversarial attacks pose severe threats to the reliability and trustworthiness of deep learning technologies, particularly in critical applications where erroneous decisions can have serious consequences [17,18].

Extensive research efforts have been dedicated to developing robust machine learning classifiers that are resilient to adversarial attacks. One popular and effective way relies on adversarial training, which involves integrating adversarial examples within the training process, to improve the classifier's resistance to attack [19,20].

The development of adversarial detectors has become a crucial area of research [21–25]. Adversarial detectors are specialized mechanisms or algorithms designed to identify and filter out adversarial examples before they can adversely affect the decision-making process of machine learning models. By analyzing input data for signs of tampering or abnormal patterns indicative of adversarial manipulation, these detectors aim to enhance the security and robustness of AI systems, thereby maintaining their integrity in the face of malicious attacks.

This paper presents a novel approach to adversarially train detectors, combining the strengths of both classifier robustness and detector sensitivity to adversarial examples.

Adversarial detectors (Figure 1), which aim to distinguish adversarial examples from benign ones, have gained momentum recently, but their robustness is unclear. The purpose of adversarial detection is to enhance the robustness of machine learning systems by identifying and mitigating adversarial attacks—thereby preserving the reliability and integrity of the classifiers in critical applications.

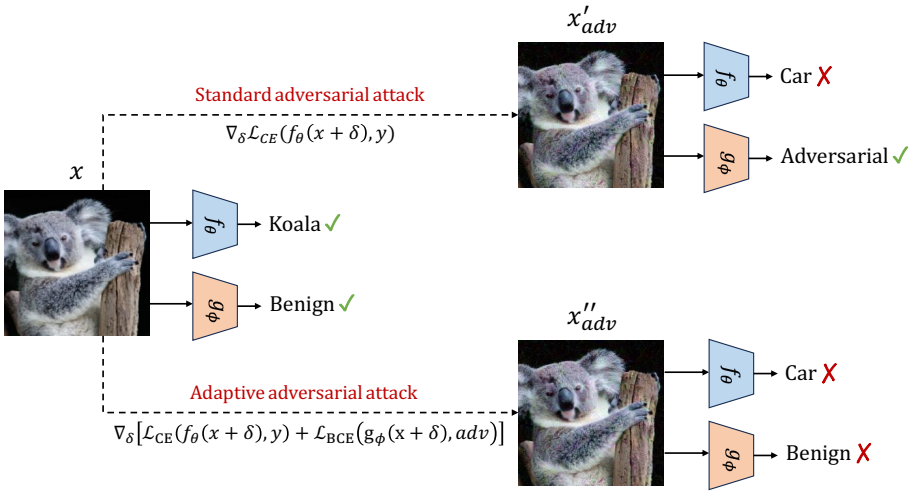


Figure 1. General scheme of adversarial attacks. x: original image. x'_{adv}: standard adversarial attack. x''_{adv}: adaptive adversarial attack, targeting both f_θ (classifier) and g_ϕ (detector). The attacker's goal is to fool the classifier into misclassifying the image and simultaneously deceive the detector into reporting the attack as benign (i.e., failing to detect the adversarial manipulation). The classifier f_θ and the detector g_ϕ share the same input but operate independently, with separate parameters and architectures. The classifier is trained to perform standard classification, while the detector is explicitly trained to identify adversarial instances.

Several studies have found that these detectors—even when combined with robust classifiers that have undergone adversarial training—can be fooled by designing tailored attacks; this engenders a cat-and-mouse cycle of attacking and defending [26,27]. It is therefore critical to thoroughly assess the robustness of adversarial detectors and ensure that they are able to withstand adaptive adversarial attacks.

The attacker's goal is to fool the classifier into misclassifying the image *and*, at the same time, fool the the detector into reporting the attack as benign (i.e., fail to recognize an attack). The attacker, in essence, carries out a two-pronged attack, targeting both the "treasure" and the treasure's "guardian".

An *adaptive* adversarial attack is *aware of the defenses*—and attempts to bypass them. To our knowledge, research addressing such attacks is lacking.

We thus face a crucial problem: How can adversarial detectors be made robust? This question underscores the need for a comprehensive understanding of the interplay between adversarial training strategies and the unique characteristics of adversarial detectors.

We hypothesize that adversarially training adversarial detectors will lead to their improvement in terms of both robustness and accuracy, thus creating a more resilient defense mechanism against sophisticated adversarial attacks. This supposition forms the basis of our investigation and serves as a guiding principle for the experiments and analyses presented in this paper.

There are two major motivations for our work here:

1. Contemporary adversarial detectors are predominantly evaluated within constrained threat models, assuming attackers lack information about the adversarial detector itself. However, this approach likely does not align with real-world scenarios, where sophisticated adversaries possess partial knowledge of the defense mechanisms.

We believe it is necessary to introduce a more realistic evaluation paradigm that considers the potential information available to an attacker. Such a paradigm will improve—perhaps significantly—the reliability of adversarial detection mechanisms.

2. Our fundamental assumption is that adversarial training of the *adversarial detector*—in isolation—has the potential to render the resultant system significantly more challenging for adversaries to thwart. Unlike conventional approaches, where increasing the robustness of the *classifier* often leads to decreased accuracy, our approach focuses solely on enhancing the capabilities of the detector, thereby avoiding such trade-offs. By bolstering the detector through adversarial training, we introduce a distinct tier of defense that adversaries must contend with, amplifying the overall adversarial resilience of the system.

The main innovation in this paper is the introduction of a novel approach to adversarial robustness by focusing on strengthening adversarial detectors rather than the classifiers themselves. Traditional methods enhance classifier robustness through adversarial training, which often leads to a compromise between robustness and clean accuracy. Our method maintains the classifier's original performance on clean data by leaving it unaltered and instead fortifies the adversarial detector against adaptive attacks.

This approach is particularly advantageous in scenarios where preserving the classifier's clean accuracy is critical. By not retraining the classifier, we avoid the typical trade-offs associated with adversarial training. Our fortified detectors demonstrate improved resilience against sophisticated adaptive attacks, offering a robust defense without sacrificing the classifier's performance on benign inputs. **We suggest a paradigm shift: instead of defending the classifier, strengthen the adversarial detector**. Thus, we ask the following:

(Q) *How can adversarial detectors be made robust?*

To our knowledge, problem **(Q)** remains open in the literature. Our contributions are the following:

- Paradigm shift: Unlike existing approaches that focus on robustifying classifiers, we introduce a novel paradigm of robustifying adversarial detectors, eliminating the trade-off between clean and robust classification.
- We introduce **RADAR**, a pioneering adversarial training technique specifically designed to enhance the resilience of adversarial detectors, demonstrating its effectiveness through extensive evaluations on three datasets and across six detection architectures.
- Our study provides comprehensive insights into the generalizability and efficacy of adversarial training strategies across diverse data distributions and model architectures, reinforcing the critical role of robust adversarial detectors in securing machine learning systems.

The next section describes previous work on adversarial training. Section 3 delineates our methodology, followed by the experimental framework in Section 4. We describe our results in Section 5, ending with conclusions in Section 6.

2. Previous Work

The foundation of adversarial training was laid by Madry et al. [19], who demonstrated its effectiveness on the MNIST [28] and CIFAR-10 [29] datasets. Their approach used Projected Gradient Descent (PGD) attacks to generate adversarial examples and incorporate them into the training process alongside clean (unattacked) data.

Subsequent work by Kurakin et al. [30] explored diverse adversarial training methods, highlighting the trade-off between robustness and accuracy. These early pioneering works laid the groundwork for the field of adversarial training and showcased its potential for improving classifier robustness.

Adversarial training has evolved significantly since its inception. Zhang et al. [31] introduced MixUp training, probabilistically blending clean and adversarial samples during training, achieving robustness without significant accuracy loss.

Tramer and Boneh [32] investigated multi-attack adversarial training, aiming for broader robustness against diverse attack types.

A number of works explored formal verification frameworks to mathematically certify classifier robustness against specific attacks, demonstrating provably robust defense mechanisms [33–36].

Zhang et al. [37] provided theoretical insights into the robustness of adversarially trained models, offering a deeper understanding of the underlying mechanisms and limitations. Their work elucidated the trade-offs between robustness, expressiveness, and generalization in adversarial training paradigms, contributing to a more comprehensive theoretical framework for analyzing and designing robust classifiers.

In the pursuit of improving robustness against adversarial examples, Xie et al. [38] proposed a novel approach using feature denoising. Their work identified noise within the extracted features as a vulnerability during adversarial training. To address this, they incorporated denoising blocks within the network architecture of a convolutional neural network (CNN). These blocks leverage techniques such as non-local means filters to remove the adversarial noise, leading to cleaner and more robust features. Their approach achieved significant improvements in adversarial robustness on benchmark datasets, when compared with baseline adversarially trained models. This work highlights feature denoising as a promising defense mechanism that complements existing adversarial training techniques. Pinhasov et al. [39] introduced a novel method to combat adversarial attacks on deepfake detectors. The authors leveraged eXplainable Artificial Intelligence (XAI) [40] to generate interpretability maps, revealing the decision-making process of the deepfake detector. By analyzing these maps, their system could identify vulnerabilities exploited by adversarial attacks.

Building on the limitations of standard adversarial training, requiring a large amount of labeled data, Carmon et al. [41] explored leveraging unlabeled data for improved robustness. Their work demonstrates that incorporating unlabeled data through semi-supervised learning techniques, such as self-training, can significantly enhance adversarial robustness. This approach bridges the gap in sample complexity, achieving high robust accuracy with the same amount of labeled data needed for standard accuracy. Their findings were validated empirically on datasets such as CIFAR-10, achieving state-of-the-art robustness against various adversarial attacks. This research highlights the potential of semi-supervised learning as a cost-effective strategy to improve adversarial training and achieve robust models.

Despite its successes, adversarial training faces several challenges. Transferability of adversarial examples remains a concern [42,43], because attacks often lack effectiveness across different classifiers and architectures. The computational cost of generating and training on adversarial examples can be significant, especially for large models and datasets. Moreover, adversarial training can lead to a reduction in clean data accuracy, requiring effective optimization strategies to mitigate this trade-off [44]. Additionally, carefully crafted evasive attacks can sometimes bypass adversarially trained classifiers, highlighting the need for further research and diversification of defense mechanisms.

There are but a few examples of works that compare adversarial attacks that take into account full knowledge of the defense, i.e., adaptive attacks. He et al. [45] showed promising results, at the cost of multiple model inferences coupled with multiple augmented neighborhood images. Klingner et al. [46] demonstrated a detection method based on edge extraction of features, such as depth estimation and semantic segmentation, and comparison with natural image edges based on the SSIM metric. We believe the underlying assumption of adversarial examples containing abnormal edges might not hold in un-natural settings.

Another notable work is that of Yang et al. [47], which introduces an adversarial detection method utilizing a conditional generative adversarial network (CGAN) and an

encoder to reconstruct input images. By comparing the discrepancies between the original and reconstructed images, their ContraNet detects adversarial examples based on reconstruction errors. While this approach demonstrates competitive performance, it introduces significant computational overhead due to the use of the CGAN and encoder components, potentially limiting its applicability in real-time or resource-constrained scenarios.

Previous research has explored using sentiment analysis for adversarial example detection [48], focusing on the impact of adversarial perturbations on deep neural networks' hidden-layer feature maps under attack.

All of the above compared their work with some adaptive power like PGD, yet did not show state-of-the-art adaptive attacks, which optimize under the constraint imposed by the classifier decision boundary, like SPGD and OPGD [49]. Only a few methods were compared under these powerful optimizers. Yang et al. [47] presented a novel encoder-generator architecture that compared the generated image label with the original image label. We believe this was a step in the right direction, yet with a computationally expensive generator that had difficulties differentiating semantically close classes. We compared our results with theirs and others [50–52], which already have some comparison with SPGD or OPGD.

3. Methodology

Before delineating the methodology, we note that the overall framework comprises three distinct components:

1. **Classifier**, which is not modified (i.e., no learning);
2. **Detector**, with a "standard" loss function (to be discussed);
3. **Attacker**, with an adaptive loss function (to be discussed).

3.1. Threat Model

We assume a strong adversary, with full access to both the classifier f and the adversarial detector g. The attacker possesses comprehensive knowledge of the internal workings, parameters, and architecture of both models.

The attacker has two goals:

1. Manipulate the classifier's prediction such that it outputs a class label different from the correct label.
2. Deceive the detector into predicting that the adversarial image is benign.

Goal 1: Manipulate the classifier's prediction such that it outputs a class different from the correct class label y. Specifically, the attacker seeks a perturbation δ that causes misclassification, i.e., $f(x+\delta) \neq y$, while keeping the perturbation within an acceptable limit defined by the l_p norm constraint, $\|\delta\|_p \leq \epsilon$. Here, f is the classifier function, x is the original input, and ϵ is the maximum allowable perturbation magnitude. This problem can be formulated as

$$\underset{k=0,1,\ldots K-1}{\arg\max}\ f_k(x+\delta) \neq y,\ (x,y) \in \mathcal{D},\ \text{ s.t. } \|\delta\|_p \leq \epsilon, \tag{1}$$

where K is the total number of classes in the classification task, $f_k(x+\delta)$ denotes the classifier's output (e.g., probability or logit) for class k when input $x+\delta$ is provided, and $\mathcal{D}$ represents the dataset of input-label pairs.

We can solve this problem using the following optimization objective:

$$\underset{\delta:\|\delta\|_p \leq \epsilon}{\arg\max}\, \mathcal{L}_{\text{CE}}(f_\theta(x+\delta), y), \tag{2}$$

where $\mathcal{L}_{\text{CE}}$ is the cross-entropy loss.

Goal 2: The attacker's second goal is to deceive the detector g into predicting that the adversarial image is benign. The attacker aims to find a perturbation δ such that $g(x+\delta) \neq \texttt{adv}$, indicating that the input is not detected as adversarial, while still satisfying

the perturbation constraint, $\|\delta\|_p \leq \epsilon$. The detector g outputs a prediction indicating whether an input is benign (ben) or adversarial (adv). This objective can be formalized as

$$\arg\max_{k=\texttt{ben},\texttt{adv}} g_k(x+\delta) \neq \texttt{adv},\ x \in \mathcal{D},\ \text{s.t.}\ \|\delta\|_p \leq \epsilon, \tag{3}$$

where $g_k(x+\delta)$ represents the detector's output for label k, and k can be either ben or adv. All other variables were as previously defined.

We can solve this problem using the following optimization objective:

$$\arg\max_{\delta:\|\delta\|_p \leq \epsilon} \mathcal{L}_{\text{BCE}}(g_\phi(x+\delta), \texttt{adv}), \tag{4}$$

where $\mathcal{L}_{\text{BCE}}$ is the binary cross-entropy loss and ϵ is the allowed perturbation norm.

Combining the two goals, the attacker's overall goal is defined as

$$\arg\max_{\delta:\|\delta\|_p \leq \epsilon} \mathcal{L}_{\text{CE}}(f_\theta(x+\delta), y) + \mathcal{L}_{\text{BCE}}(g_\phi(x+\delta), \texttt{adv}). \tag{5}$$

By adjusting δ within the perturbation constraint, $\|\delta\|_p \leq \epsilon$, the attacker aims to achieve the following two objectives:

1. **Misclassification by the classifier**: By maximizing the cross-entropy loss $\mathcal{L}_{\text{CE}}(f\theta(x+\delta), y)$, the attacker increases the discrepancy between the classifier's prediction $f_\theta(x+\delta)$ and the true label y, causing the classifier to misclassify the perturbed input $x+\delta$.
2. **Evade detection**: By maximizing the binary cross-entropy loss $\mathcal{L}_{\text{BCE}}(g\phi(x+\delta), \texttt{adv})$ with respect to the label adv, the attacker forces the detector's output $g_\phi(x+\delta)$ away from correctly identifying the input as adversarial. This results in the detector misclassifying the adversarial input $x+\delta$ as benign.

Through careful adjustment of δ, the attacker simultaneously degrades the classifier's accuracy and circumvents the detector's ability to recognize adversarial inputs.

3.2. Problem Definition

We now formalize the problem of enhancing the robustness of an adversarial detector (g) through adversarial training. The setup involves a classifier (f) and an adversarial detector (g), both initially trained on a clean dataset $\mathcal{D}$ containing pairs (x, y), where x is an input data point and y is the ground-truth label. We denote with θ and ϕ the weight parameters of the classifier and the detector, respectively. Note that $\mathcal{D}$ is the clean dataset—adversarial examples are represented by $x+\delta$.

Adversarial training is formulated as a minimax game, as follows:

$$\min_\theta \left\{ \mathbb{E}_{(x,y)\sim\mathcal{D}} \left[\max_{\delta:\|\delta\|_p \leq \epsilon} \mathcal{L}_{\text{CE}}(f_\theta(x+\delta), y) \right] \right\}. \tag{6}$$

This minimax game pits the classifier against an adversary: the adversary maximizes the classifier's loss on perturbed inputs, while the classifier minimizes that loss across all such perturbations, ultimately becoming robust to attacks.

Our main objective is to improve the robustness of detector g using adversarial training by iteratively updating ϕ based on adversarial examples. Formally, the optimization objective for the adversarial training of the detector is

$$\min_\phi \left\{ \mathbb{E}_{(x,y)\sim\mathcal{D}} \left[\max_{\delta:\|\delta\|_p \leq \epsilon} \mathcal{L}_{\text{BCE}}(g_\phi(x+\delta), \texttt{adv}) \right] \right\}. \tag{7}$$

With regard to the classifier, we have a minimax game for the detector.

However, the above equation does not take into account the classifier, f. Therefore, we added the classifier outputs to our (almost) final objective, as follows:

$$\min_{\phi} \left\{ \mathbb{E}_{(x,y)\sim\mathcal{D}} \left[\max_{\delta:\|\delta\|_p \leq \epsilon} (\mathcal{L}_{\text{CE}}(f_\theta(x+\delta), y) + \mathcal{L}_{\text{BCE}}(g_\phi(x+\delta), \texttt{adv})) \right] \right\}. \tag{8}$$

Here, the objective incorporates both the classifier and the detector in a unified adversarial training framework. The goal is to simultaneously minimize the classification error *and* enhance the detector's robustness.

However, a potential conflict arises with this combined objective. The cross-entropy loss ($\mathcal{L}_{\text{CE}}$) for the classifier and the binary cross-entropy loss ($\mathcal{L}_{\text{BCE}}$) for the detector inherently contradict each other. The attacker's objective is to simultaneously induce misclassification in the classifier and deceive the detector into labeling adversarial examples as benign. This dual objective creates opposing forces in the optimization process. Specifically, the attacker's optimization of $\mathcal{L}_{\text{CE}}$ aims to increase the classification error by adjusting δ so that the predicted class probabilities do not align with the true labels. Conversely, the optimization of $\mathcal{L}_{\text{BCE}}$ seeks to reduce the detector's ability to differentiate between clean and adversarial examples, driving g_ϕ to misclassify adversarial inputs as benign. This opposition can lead to conflicting gradient directions, where optimizing for one objective may detract from progress on the other, thereby complicating the optimization process and potentially preventing either objective from being fully achieved.

This dual optimization can result in non-convergent behavior for several reasons. First, the gradients derived from $\mathcal{L}_{\text{CE}}$ and $\mathcal{L}_{\text{BCE}}$ often point in opposing directions. For example, a gradient step that reduces classification accuracy might enhance the detector's ability to identify adversarial examples. This phenomenon occurs because reducing classification accuracy involves adding perturbations to the input data, which in turn makes these perturbations more detectable by the adversarial detector. Consequently, the detector's task of distinguishing between clean and perturbed inputs becomes easier. This tug-of-war can impede the optimization process, preventing the simultaneous realization of both objectives. Lastly, the optimization landscapes of $\mathcal{L}_{\text{CE}}$ and $\mathcal{L}_{\text{BCE}}$ can dynamically shift as δ is updated.

To address this challenge, we employ the selective and orthogonal approaches proposed by Bryniarski et al. [49].

Selective Projected Gradient Descent (SPGD) [49] optimizes only with respect to a constraint that has not been satisfied yet—while not optimizing against a constraint that has been optimized. The loss function used by SPGD is

$$\mathcal{L}_{\text{CE}}(f_\theta(x+\delta), y) \cdot \mathbb{1}[f_\theta(x) = y] + \mathcal{L}_{\text{BCE}}(g_\phi(x+\delta), \texttt{adv}) \cdot \mathbb{1}[g_\phi(x+\delta) = \texttt{adv}]. \tag{9}$$

Rather than minimizing a convex combination of the two loss functions, the idea is to optimize either the left-hand side of the equation, if the classifier's prediction is still y, meaning the optimization has not been completed, or optimize the right-hand side of the equation, if the detector's prediction is still `adv`.

This approach ensures that updates consistently enhance the loss on either the classifier or the adversarial detector, with the attacker attempting to "push" the classifier away from the correct class y and also "push" the detector away from raising the adversarial "flag".

Orthogonal Projected Gradient Descent (OPGD) [49] focuses on modifying the update direction to ensure it remains orthogonal to the constraints imposed by previously satisfied objectives. This orthogonal projection ensures that the update steers the input towards the adversarial objective without violating the constraints imposed by the classifier's decision boundary, allowing for the creation of adversarial examples that bypass the detector. Thus, the update rule is slightly different, as follows:

$$\mathcal{L}_{\text{update}}(x, y) = \begin{cases} \nabla\mathcal{L}_{\text{CE}}(f_\theta(x+\delta), y) - \text{proj}_{\nabla\mathcal{L}_{\text{BCE}}(g_\phi(x+\delta), \texttt{adv})} \nabla\mathcal{L}_{\text{CE}}(f_\theta(x+\delta), y) & \text{if} \quad f_\theta(x) = y \\ \nabla\mathcal{L}_{\text{BCE}}(g_\phi(x+\delta), \texttt{adv}) - \text{proj}_{\nabla\mathcal{L}_{\text{CE}}(f_\theta(x+\delta), y)} \nabla\mathcal{L}_{\text{BCE}}(g_\phi(x+\delta), \texttt{adv}) & \text{if} \quad g_\phi(x) = \texttt{adv}. \end{cases} \tag{10}$$

In essence, we have set up a minimax game: the attacker wants to maximize loss with respect to the image, while the defender wants to minimize loss with respect to the detector's parameters.

Since optimization is customarily viewed as minimizing a loss function, we will consider that the attacker uses minimization instead of maximization. A failed attack would thus be observed though a large loss value (which we will indeed observe in Section 5).

All aforementioned objectives are intractable and thus approximated using iterative gradient-based attacks, such as PGD. Algorithm 1 shows the pseudocode of our method for adversarially training an adversarial detector. Figure 2 illustrates **RADAR**.

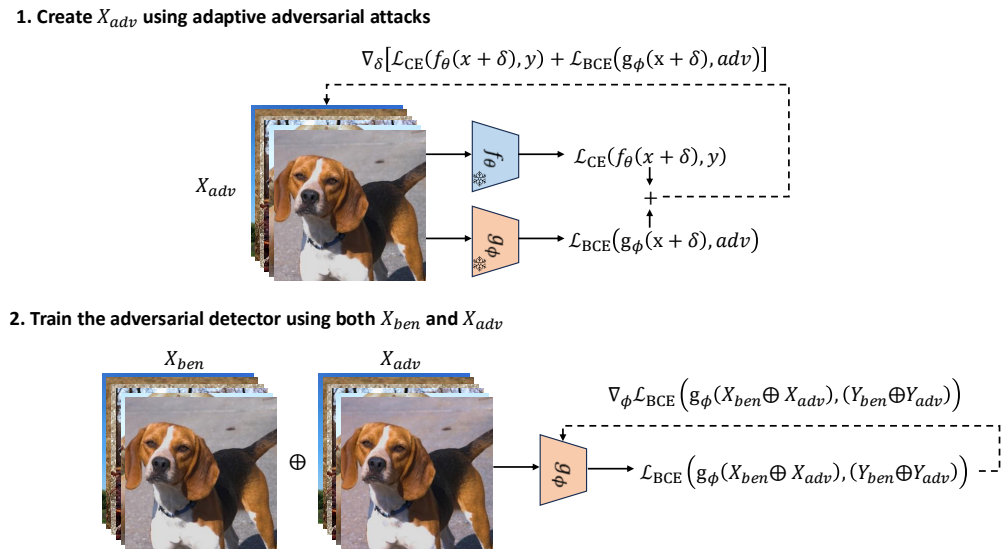


Figure 2. Overview of **RADAR**. (1) The process begins with the generation of adaptive adversarial instances, X_{adv}. (2) After completing the batch attack, we train the detector g_ϕ using both benign instances X_{ben} and adversarial instances X_{adv}. The ❄ symbol refers to the models being frozen.

Algorithm 1: RADAR

Input: dataset $\mathcal{D}$, classifier f_θ, detector, g_ϕ, detector's loss function $\mathcal{L}_{\mathrm{BCE}}$, classifier's loss function $\mathcal{L}_{\mathrm{CE}}$, batch size B, step size α, epsilon ϵ, number of steps I, number of epochs E, learning rate η

Output: adversarially trained g_ϕ

Set f to eval mode

for $e = 1, ..., E$ **do**

  for $X_{ben} \subseteq \mathcal{D}$, *s.t.* $|X_{ben}| = B$ **do**

    Set g to eval mode

    Compute adversarial examples X_{adv}:

$$X_{\mathrm{adv}} \leftarrow \mathrm{OPGD/SPGD}(f_\theta, g_\phi, X_{\mathrm{ben}}, \epsilon, \alpha, I, \mathcal{L}_{\mathrm{BCE}}, \mathcal{L}_{\mathrm{CE}})$$

    Form a new batch:

$$X_{\mathrm{mixed}} \leftarrow X_{\mathrm{ben}} \oplus X_{\mathrm{adv}}$$
$$Y_{\mathrm{mixed}} \leftarrow \{\mathtt{ben}\}_{i=1}^{|B|} \oplus \{\mathtt{adv}\}_{i=1}^{|B|}$$

    Set g to train mode

    Compute average loss gradient for X_{mixed}:

$$\nabla_{\mathrm{mixed}} \leftarrow \frac{1}{2|B|} \sum_{j=1}^{2|B|} \nabla_\phi \mathcal{L}_{\mathrm{BCE}}(x_{\mathrm{mixed}}, y_{\mathrm{mixed}}; \phi)$$

    Update parameters:

$$\phi \leftarrow \phi - \eta \nabla_{\mathrm{mixed}}$$

  end

end

4. Experimental Framework

Datasets and models. We used the VGG-{11, 13, 16} and ResNet-{18, 34, 50} architectures both for classification and for adversarial detection. Specifically, we employed these architectures in dual roles: as classifiers to perform the primary task of image classification and as adversarial detectors to identify adversarial examples. By leveraging the same architectures for both tasks, we ensured consistency in our experimental setup and enabled a comprehensive evaluation of their robustness and effectiveness in the context of adversarial training. We modified the classification head of the detector, $C \in \mathbb{R}^{K \times e}$, where K is the number of classes and e is the embedding space size, to $C \in \mathbb{R}^{2 \times e}$, for binary classification (benign/adversarial), and added a sigmoid transformation at the end.

We utilized the CIFAR-10 [29], the SVHN [53], and a subset of the ImageNet [54] dataset to evaluate our proposed approach. For ImageNet, we randomly selected 50 classes from the original dataset, which contains over 14 million images and 1000 classes. This subset was used to maintain manageability and to focus our evaluation on a representative sample of the larger dataset, while still leveraging the diversity and complexity that ImageNet provides for benchmarking in image classification tasks. It is important to highlight that adversarial training for adversarial detectors incurs substantially higher computational costs compared with conventional adversarial training. As a result, this process is considerably more time-consuming when applied to large datasets. To ensure the feasibility of our experiments, we employed a reduced number of classes from ImageNet.

We used the definition of attack success rate presented by Bryniarski et al. [49] as part of their evaluation methodology. Attack efficacy is measured through Attack Success Rate at N, SR@N: proportion of deliberate attacks achieving their objectives subject to the condition that the defense's false-positive rate is configured to N%. The underlying motivation is that, in real-life scenarios, we must strike a delicate balance between security and precision. A 5% false positive is acceptable, while extreme cases might even use 50%. Our results proved excellent, so we set N to a low 5%, i.e., SR@5.

Throughout the training, the allowed perturbation was constrained by an ℓ_∞ norm, denoted as $\|\cdot\|_\infty$, with the maximum magnitude set to $\epsilon = \frac{16}{255}$. We employed an adversarial attack strategy, specifically utilizing 100 iterations of PGD with a step size parameter α set to 0.03. This approach was employed to generate adversarial instances and assess the resilience of the classifier under scrutiny. We then evaluated the classifiers on the attacked test set—the results are delineated in Table 1.

Table 1. Performance (accuracy percentage) of original classifiers on both clean and adversarial PGD-perturbed test sets.

Dataset	VGG-11		VGG-13		VGG-16		ResNet-18		ResNet-34		ResNet-50	
	Ben	Adv	Ben	Adv	Ben	Adv	Ben	Adv	Ben	Adv	Ben	Adv
CIFAR-10	92.40	0.35	94.21	0.19	94.00	5.95	92.60	0.14	93.03	0.21	93.43	0.26
SVHN	93.96	0.00	94.85	0.01	94.95	0.48	94.88	0.02	94.84	0.11	94.33	0.10
ImageNet	70.08	0.00	70.44	0.00	71.96	0.00	70.24	0.00	72.24	0.00	74.96	0.00

Afterwards, we split the training data into training (70%) and validation sets (30%). We trained 3 VGG-based and 3 ResNet-based adversarial detectors for 20 epochs using the Adam optimizer (with default β_1 and β_2 values), with a learning rate of 1×10^{-4} , a `CosineAnnealing` learning rate scheduler with $T_{\max} = 10$, and a batch size of 32. We tested the adversarial detectors on the test set that was composed of one-half clean images and one-half attacked images. They all performed almost perfectly.

Following the initial clean training phase, we implemented our proposed **RADAR** approach, which involves adversarial fine-tuning on the adversarial detectors. This process entailed attacking the detectors to an adaptive adversarial attack, specifically the OPGD attack. The adversarial fine-tuning was conducted over the course of 20 epochs, utilizing the same optimizer and the `ReduceLROnPlateau` learning rate scheduler, with the patience

parameter set to 3 and a batch size of 32. Subsequently, the performance was evaluated on a test set consisting of an equal distribution of clean images and images subjected to OPGD attacks. Table 2 details the hyperparameters employed throughout the experiments.

Table 2. Summary of hyperparameters and implementation details used in the experiments.

Parameter	Value
Datasets	CIFAR-10, SVHN, ImageNet (50 classes)
Model architectures	VGG-11/13/16, ResNet-18/34/50
Optimizer	Adam ($\beta_1 = 0.9$, $\beta_2 = 0.999$)
Initial learning rate	1×10^{-4}
Learning rate scheduler (clean training)	Cosine annealing ($T_{\max} = 10$ epochs)
Learning rate scheduler (fine-tuning)	ReduceLROnPlateau (patience = 3, factor = 0.1)
Batch size	32
Number of epochs	20
Adversarial attack (PGD)	$\epsilon = 16/255$, $\alpha = 0.03$, 100 iterations
Adversarial attack (OPGD)	$\epsilon = 16/255$, 100 iterations

5. Results

Initially, we conducted an assessment of classifier performance following the generation of adversarial perturbations employing the PGD method.

The outcomes of this evaluation, presented in Table 1, highlight classifier performance on both clean and perturbed test sets. The results reveal significant vulnerability to adversarial manipulations. For instance, on the CIFAR-10 dataset, the VGG-11 classifier drops from 92.40% accuracy on clean data to 0.35% on adversarial examples. Other classifiers on CIFAR-10 show similar trends, with clean accuracy values between 92.40% and 94.21% and adversarial accuracy values below 6%. A similar pattern is observed on the SVHN dataset, where the VGG-11 classifier's accuracy falls from 93.96% on clean data to 0.00% on adversarial attacks. Other SVHN classifiers also exhibit significant performance drops, with adversarial accuracy values near 0%. On the ImageNet dataset, all classifiers reach 0% accuracy when attacked. These findings illustrate the stark vulnerability of standard classifiers to adversarial perturbations, underscoring the need for more robust defense mechanisms.

After standard training involving the use of PGD-generated adversarial images, Table 3 shows the performance outcomes of the adversarial detectors, as assessed through the ROC-AUC metric, before the deployment of **RADAR**. The table summarizes the average ROC-AUC ($AUC_{Avg.}$) and individual ROC-AUC scores for PGD (AUC_{PGD}), OPGD (AUC_{OPGD}), and SPGD (AUC_{SPGD}) attacks. For the CIFAR-10 dataset, the VGG-16 detector achieved the highest average ROC-AUC score of 0.46, which is slightly worse than tossing a coin. All detectors performed perfectly against PGD attacks, but their performance dropped significantly against OPGD and SPGD, with most detectors showing a score close to 0.00. The SVHN and ImageNet datasets displayed a similar trend. All models achieved perfect detection against PGD attacks, but their effectiveness plummeted for OPGD and SPGD attacks, with only ResNet-50 showing minimal detection capability.

Table 3. Without RADAR: Performance of detectors on several classifiers and datasets. In this and subsequent tables, boldface marks best performance. Note: for ROC-AUC, higher is better.

Detector	$AUC_{Avg.}$	CIFAR-10			SVHN			ImageNet		
		AUC_{PGD}	AUC_{OPGD}	AUC_{SPGD}	AUC_{PGD}	AUC_{OPGD}	AUC_{SPGD}	AUC_{PGD}	AUC_{OPGD}	AUC_{SPGD}
VGG-11	0.33	1.00	0.00	0.00	1.00	0.00	0.00	1.00	0.00	0.00
VGG-13	0.33	1.00	0.00	0.00	1.00	0.00	0.00	1.00	0.00	0.00
VGG-16	**0.46**	1.00	0.61	0.59	1.00	0.00	0.00	1.00	0.00	0.00
ResNet-18	0.33	1.00	0.00	0.00	1.00	0.00	0.00	1.00	0.00	0.00
ResNet-34	0.33	1.00	0.00	0.00	1.00	0.00	0.00	1.00	0.00	0.00
ResNet-50	0.37	1.00	0.15	0.14	1.00	0.03	0.05	1.00	0.00	0.00
Avg.	0.36	1.00	0.13	0.12	1.00	0.00	0.00	1.00	0.00	0.00

Table 4 presents results regarding the adversarial detectors' performance, based on the SR@5 metric, prior to using **RADAR**. Notably, the VGG-16 detector exhibits the best performance with a significantly lower $SR@5_{Avg.}$ value of 0.78. It performs particularly well against OPGD and SPGD attacks on CIFAR-10, achieving SR@5 values of 0.37 and 0.40, respectively. This indicates a higher robustness compared with other detectors. In contrast, detectors such as VGG-11, VGG-13, ResNet-18, and ResNet-34 show higher $SR@5_{Avg.}$ values of around 0.97 for both datasets. ResNet-50 performs better than most with an average SR@5 of 0.88 on CIFAR-10 but still falls short of VGG-16. On the ImageNet dataset, the detectors also fail to withstand the attacks, achieving SR@5 values of 0.98 and 0.99 for OPGD and SPGD, respectively.

Table 4. Without RADAR: Performance of detectors on several classifiers and datasets. VGG-16 is the most robust detector in terms of SR@5. Note: for SR@5, lower is better.

Detector	$SR@5_{Avg.}$	CIFAR-10		SVHN		ImageNet	
		$SR@5_{OPGD}$	$SR@5_{SPGD}$	$SR@5_{OPGD}$	$SR@5_{SPGD}$	$SR@5_{OPGD}$	$SR@5_{SPGD}$
VGG-11	0.98	0.97	0.97	0.98	0.99	0.99	1.00
VGG-13	0.99	0.97	0.99	0.99	0.99	1.00	1.00
VGG-16	0.78	0.37	0.40	0.99	0.99	0.97	0.99
ResNet-18	0.98	0.96	0.97	0.99	0.99	0.99	1.00
ResNet-34	0.98	0.96	0.97	0.99	0.99	0.99	1.00
ResNet-50	0.93	0.81	0.83	0.96	0.94	0.97	0.99
Avg.	0.90	0.84	0.85	0.98	0.98	0.98	0.99

We then deployed **RADAR**. The outcomes on the efficacy of adversarial detectors, measured by ROC-AUC and SR@5 metrics after integrating **RADAR**, are presented in Tables 5 and 6. Table 7 shows the accuracy percentages of all the classifiers on clean and adversarially perturbed test sets across the CIFAR-10, SVHN, and ImageNet datasets after applying **RADAR**. Notably, we constructed an equal distribution of clean and adversarially perturbed test sets, effectively doubling the original test set size to ensure a balanced evaluation. Accuracy was computed as follows: If the detector identified the input as adversarial, the classifier's prediction was disregarded; otherwise, the classifier's prediction was considered. The observed accuracy drop in Table 7, particularly noticeable in smaller models, can be attributed to their limited number of parameters, making adaptation to both adaptive and standard adversarial attacks more challenging. Note that the benign accuracy for all classifiers across CIFAR-10, SVHN, and ImageNet remained unchanged.

Table 5. With RADAR: Performance of employing **RADAR** with OPGD on several classifiers and datasets.

Detector	$AUC_{Avg.}$	CIFAR-10			SVHN			ImageNet		
		AUC_{PGD}	AUC_{OPGD}	AUC_{SPGD}	AUC_{PGD}	AUC_{OPGD}	AUC_{SPGD}	AUC_{PGD}	AUC_{OPGD}	AUC_{SPGD}
VGG-11	0.98	0.99	0.95	0.94	1.00	1.00	1.00	1.00	1.00	1.00
VGG-13	0.99	0.99	0.99	0.99	1.00	1.00	1.00	1.00	0.99	0.99
VGG-16	0.99	1.00	0.99	0.99	1.00	1.00	1.00	1.00	0.99	0.99
ResNet-18	0.98	0.99	0.96	0.96	1.00	0.99	1.00	0.95	1.00	1.00
ResNet-34	0.99	0.99	0.99	1.00	1.00	1.00	1.00	0.98	1.00	1.00
ResNet-50	0.99	0.99	0.99	0.99	1.00	1.00	0.99	0.99	1.00	1.00
Avg.	0.99	0.99	0.98	0.98	1.00	0.99	0.99	0.98	0.99	0.99

For all datasets, **RADAR** maintains high accuracy on both PGD and OPGD samples. VGG-11 on CIFAR-10 retains an accuracy of 92.40% across all scenarios, while ResNet-50 on SVHN shows only a slight reduction from 94.33% to 94.32% on OPGD samples. In contrast, the ImageNet dataset reveals more significant accuracy drops under adversarial conditions, particularly for ResNet-18, which falls from 70.24% on benign samples to 61.05% on PGD samples.

Following **RADAR** integration, we observed significant improvements in adversarial detection performance across various classifiers and datasets, as shown in Table 5. Notably,

on CIFAR-10, models such as VGG-13, VGG-16, ResNet-34, and ResNet-50 achieved high average ROC-AUC scores of 0.99, with ResNet-18 slightly lower at 0.98. For SVHN, all models achieved ROC-AUC scores of 1.00, except for ResNet-18 and ResNet-50, which scored 0.99 under specific attack conditions. Similarly, for the ImageNet dataset, models exhibited robust performance, with most achieving ROC-AUC scores of 1.00. Specifically, ResNet-50 scored consistently high at 1.00 or 0.99 across different attack types. We observed that ResNet-18 on ImageNet exhibits a lower AUC_{PGD} score (0.95) compared with other models. This drop is likely due to ResNet-18's shallower architecture, which may not capture the complex feature representations necessary for effective adversarial detection on high-resolution datasets like ImageNet. These results underscore **RADAR**'s ability to fortify adversarial detectors against various attack methods.

Table 6. With RADAR: Performance of detectors on several classifiers and datasets.

Detector	$SR@5_{Avg.}$	CIFAR-10		SVHN		ImageNet	
		$SR@5_{OPGD}$	$SR@5_{SPGD}$	$SR@5_{OPGD}$	$SR@5_{SPGD}$	$SR@5_{OPGD}$	$SR@5_{SPGD}$
VGG-11	0.02	0.04	0.06	0.00	0.00	0.00	0.00
VGG-13	0.00	0.00	0.00	0.00	0.00	0.00	0.00
VGG-16	0.00	0.00	0.00	0.00	0.00	0.00	0.00
ResNet-18	0.02	0.05	0.05	0.00	0.00	0.00	0.00
ResNet-34	0.00	0.00	0.00	0.00	0.00	0.00	0.00
ResNet-50	0.00	0.00	0.00	0.00	0.00	0.00	0.00
Avg.	0.00	0.01	0.02	0.00	0.00	0.00	0.00

Table 7. With RADAR: The accuracy percentage of the original classifiers is assessed on both clean and adversarially perturbed test sets, using both PGD and OPGD, subsequent to the application of **RADAR**. In instances where the adversarial detector indicates an adversarial sample (`adv`), the classifier's prediction is disregarded.

Dataset	VGG-11			VGG-13			VGG-16			ResNet-18			ResNet-34			ResNet-50		
	Ben	PGD	OPGD	Ben	PGD	OPGD	Ben	PGD	OPGD	Ben	PGD	OPGD	Ben	PGD	OPGD	Ben	PGD	OPGD
CIFAR-10	92.40	92.40	92.37	94.21	94.17	94.08	94.00	93.96	93.91	92.60	92.41	91.84	93.03	93.03	93.03	93.43	93.43	93.14
SVHN	93.96	93.96	93.95	94.85	94.85	94.85	94.95	94.95	94.94	94.88	94.88	94.86	94.84	94.83	94.82	94.33	94.33	94.32
ImageNet	70.08	68.89	68.73	70.44	69.87	69.83	71.96	71.96	71.96	70.24	61.05	68.63	72.24	68.18	72.18	74.96	74.63	74.96

The SR@5 metric further confirms the robustness of **RADAR**-enhanced detectors, as detailed in Table 6. Models like VGG-13, VGG-16, ResNet-34, and ResNet-50 achieved perfect SR@5 scores of 0.00 on all datasets, indicating successful detection of adversarial attacks. VGG-11 and ResNet-18 demonstrated slightly lower performance on CIFAR-10. **RADAR** significantly improves the robustness of adversarial detectors, as evidenced by high ROC-AUC scores and low SR@5 values across different models and datasets, including the challenging ImageNet dataset. This comprehensive performance across diverse datasets reinforces the efficacy of our proposed approach in enhancing adversarial detector resilience.

Figure 3 show the generalization performance of **RADAR**-trained detectors on classifiers they were not trained on. This table shows the ROC-AUC values of **RADAR**-trained detectors when evaluated on the different classifier models. Each cell in the table corresponds to the ROC-AUC value achieved by the detector–classifier pair. For example, the top-left cell in the top-right table indicates the performance of a VGG-11 detector model trained on a VGG-11 classifier model using the ImageNet dataset and subsequently evaluated on attacks utilizing a ResNet-50 classifier model. The results indicate consistently high generalization across different classifiers, with most detector–classifier pairs achieving ROC-AUC values of 0.99 or higher. This demonstrates the effectiveness of **RADAR**-trained detectors in maintaining high adversarial detection performance across diverse datasets and classifier architectures. These findings underscore the resilience of adversarial detectors trained with **RADAR**, showcasing their ability to maintain robust detection capabilities even when confronted with unseen classifiers. The high ROC-AUC values indicate that our

approach effectively fortifies the detectors themselves, rather than relying solely on robust classifier training, thereby enhancing the overall security and reliability of the adversarial detection system.

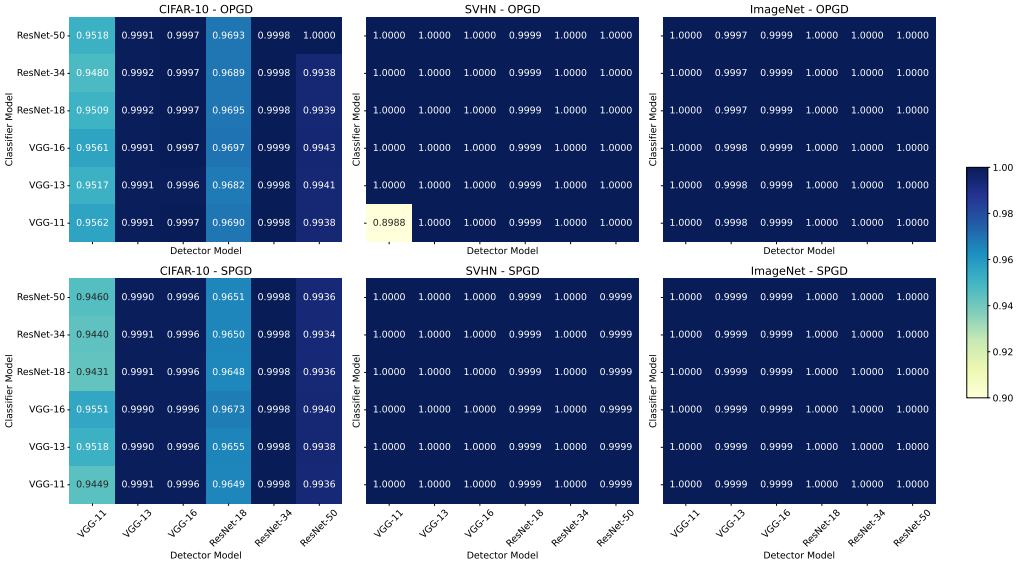


Figure 3. Generalization performance of adversarially trained detectors trained on CIFAR-10, SVHN, and ImageNet. Each adversarial detector was trained using each corresponding classifier; e.g., ResNet-50 adversarial detector was trained using ResNet-50 image classifier. This table shows the generalization of each detector to other classifiers, which it did not train with. A value represents the ROC-AUC of the respective detector–classifier pair for OPGD (top row) and SPGD (bottom row) with $\epsilon = \frac{16}{255}$.

A notable enhancement is observed across all detectors with respect to ROC-AUC and SR@5. Moreover, our findings suggest that adversarial training did not optimize solely for adaptability to specific adversarial techniques, but also demonstrated efficacy against conventional PGD attacks, which are different in nature, as illustrated in Figure 4.

Impact of adversarial training on optimization dynamics. Before the incorporation of adversarial training, the optimization process exhibited a trend of rapid convergence towards zero loss within a few iterations, as can be seen in the top rows of Figures 5–7.

Once we integrated adversarial training into the detector, we observed a distinct shift in the optimization behavior. Upon deploying **RADAR**, the optimization process displayed a tendency to plateau after a small number of iterations, with loss values typically higher by orders of magnitude, as compared with those observed prior to deployment. This plateau phase persisted across diverse experimental settings and datasets, indicating a fundamental change in the optimization landscape, induced by **RADAR**. This behavior can be seen as a robustness-enhancing effect, because the detector appears to resist rapid convergence towards trivial solutions, thereby enhancing its generalization capabilities. This showcases the efficacy of our method in bolstering the defenses of detection systems against adversarial threats—without sacrificing clean accuracy.

Robustness analysis with various epsilon values. The results of our experiments, depicted in Figures 8 and 9, provide a comprehensive evaluation of the robustness of the adversarial detectors against OPGD and SPGD attacks with different ϵ values.

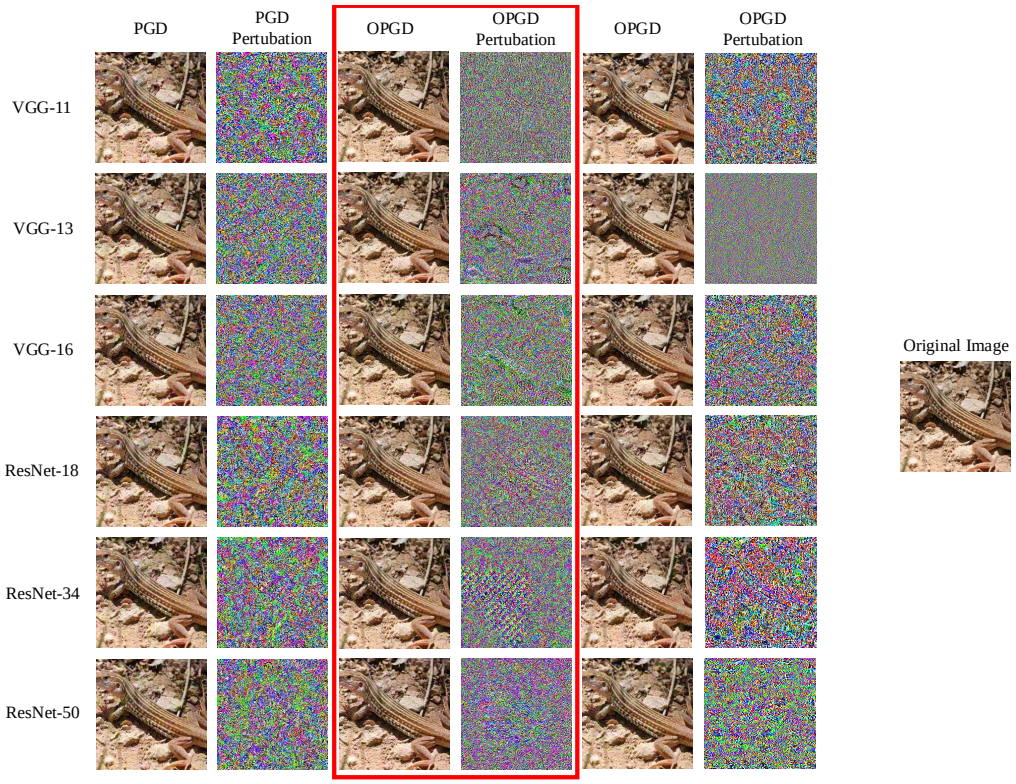


Figure 4. Comparison of perturbations generated by PGD and OPGD across adversarial detectors. The red box contains the manipulated images and the corresponding perturbation over the adversarially trained detectors. Each row represents one model architecture, while the columns show the adversarial image generated by PGD, the corresponding perturbation from PGD, the adversarial image generated by OPGD, and the corresponding perturbation from OPGD. The original image is shown on the right for reference. Note: PGD only attacks a classifier, while OPGD attacks both a classifier and a detector.

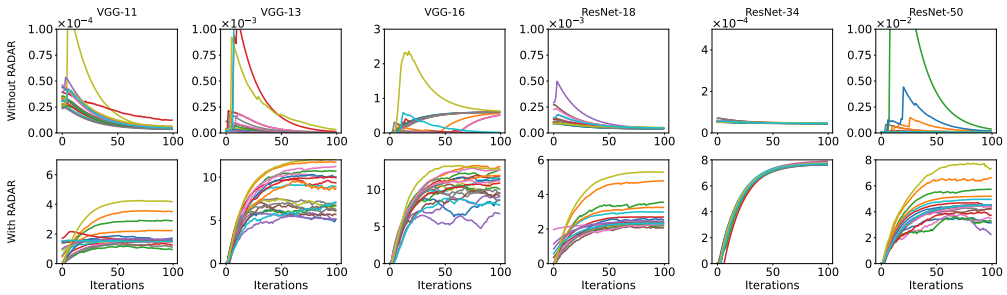


Figure 5. CIFAR-10. Binary cross-entropy loss metrics, from the point of view of an attacker, are presented here in the context of crafting an adversarial instance from the test set. These plots illustrate the progression of loss over 20 different images of orthogonal projected gradient descent (OPGD) with the main goal being to minimize the loss. The progression for each image is represented in a distinct color. Top: Prior to adversarial training, the loss converges to zero after a small number of iterations. Bottom: After adversarial training, the incurred losses are significantly higher by orders of magnitude (note the difference in scales) compared with those observed in their standard counterparts. This shows that the detector is now resilient, i.e., far harder to fool.

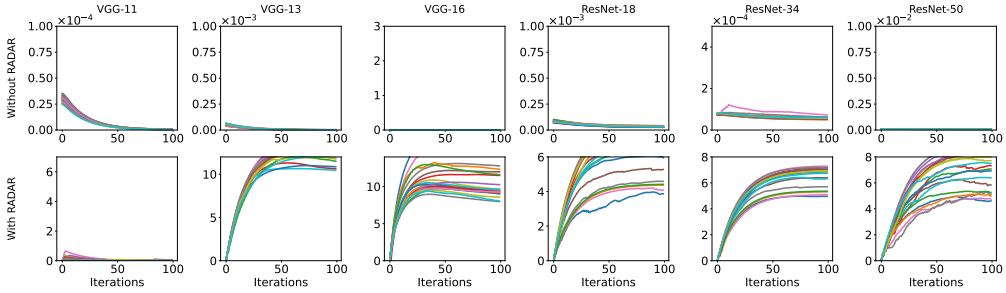


Figure 6. SVHN. Binary cross-entropy loss metrics, from the point of view of an attacker, are presented here in the context of crafting an adversarial instance from the test set. The progression for each image is represented in a distinct color.

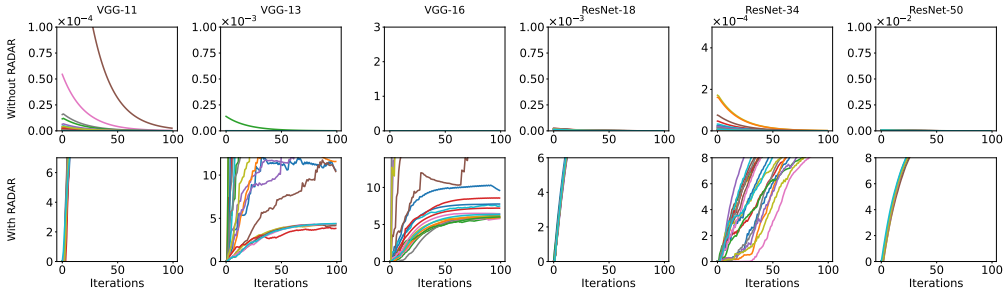


Figure 7. ImageNet. Binary cross-entropy loss metrics, from the point of view of an attacker, are presented here in the context of crafting an adversarial instance from the test set using OPGD. The progression for each image is represented in a distinct color.

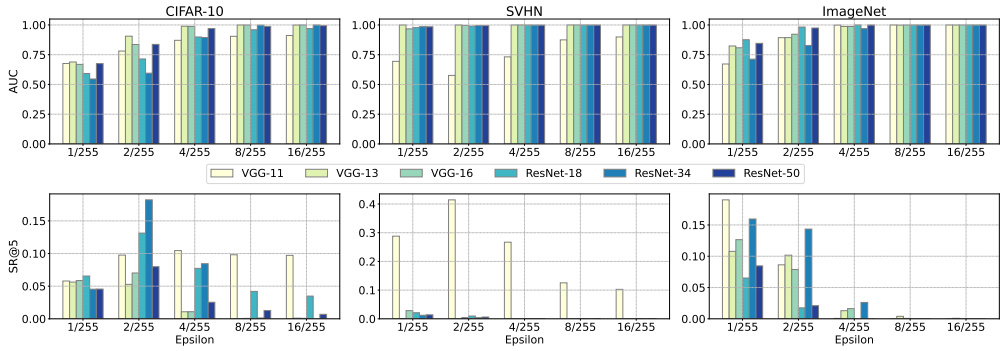


Figure 8. AUC and SR@5 scores across different epsilon values for CIFAR-10, SVHN, and ImageNet datasets using OPGD. The performance of the adversarial detectors is illustrated, highlighting how AUC and SR@5 vary across different perturbation magnitudes.

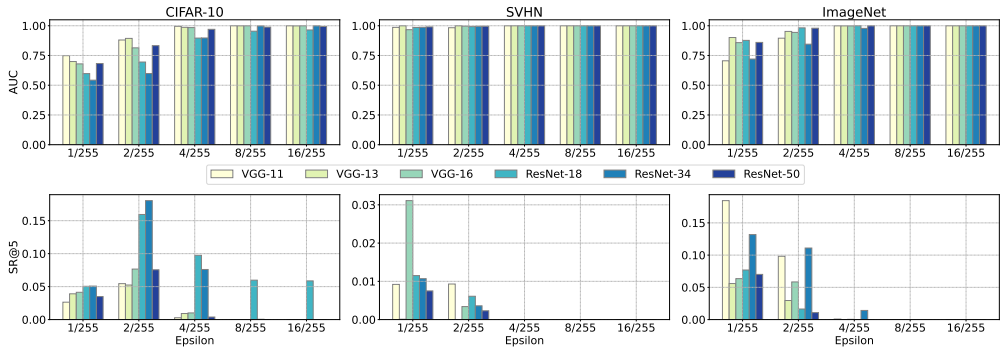


Figure 9. AUC and SR@5 scores across different epsilon values for CIFAR-10, SVHN, and ImageNet datasets using SPGD.

For the CIFAR-10, SVHN, and ImageNet datasets, increasing ϵ values generally results in increasing ROC-AUC scores and decreasing SR@5 rates using both OPGD and SPGD, indicating improved detection capabilities and decreased success of adversarial attacks.

Specifically, VGG-11 shows a significant drop in AUC and a corresponding rise in SR@5 as ϵ decreases, reflecting its susceptibility to lower magnitude perturbations. VGG-13 and VGG-16 exhibit similar patterns, though VGG-16 demonstrates slightly better robustness at lower ϵ values. This trend is consistent across both OPGD and SPGD attack evaluations.

The ResNet models, particularly ResNet-50, demonstrate superior resilience performance. Across various ϵ values, ResNet-50 consistently exhibits higher ROC-AUC scores and lower SR@5 rates compared with other models, signifying its robust ability to detect subtle adversarial examples. Notably, our method maintains consistently low SR@5 values across all datasets, underscoring its resilience. Specifically, experiments on the SVHN dataset consistently achieve an SR@5 of 0%. For CIFAR-10 and ImageNet, SR@5 ranges between 0% and 17% with OPGD and between 0% with 15% with SPGD, affirming the efficacy of our approach in enhancing adversarial detection capability. These results highlight the advantage of applying adversarial training directly to adversarial detectors rather than focusing solely on classifiers themselves.

Comparison with other detectors against adaptive attacks. Table 8 presents the robust accuracy ($\text{Acc}_{\text{robust}}$) of various defenses against OPGD and SPGD adaptive attacks at two perturbation levels, $\epsilon = 0.01$ and $\epsilon = 8/255$, and at two false-positive rates, 5% (FP@5) and 50% (FP@50). The defenses evaluated are ContraNet [55], Trapdoor [50], DLA [51], and SID [52]. **RADAR** consistently achieves the highest robust accuracy across both attack types and perturbation levels, maintaining nearly 100% accuracy in all scenarios. ContraNet also shows high performance, particularly at lower perturbation levels, but its accuracy decreases at higher perturbation levels. Trapdoor, DLA, and SID exhibit varying degrees of robustness, with significant decreases in accuracy at higher perturbation levels. For instance, Trapdoor's accuracy drops to almost 0% under both attack types at $\epsilon = 8/255$.

Table 8. Robust accuracy of different defenses under OPGD and SPGD attacks. The table presents the robust accuracy ($\text{Acc}_{\text{robust}}$) of various defenses when subjected to the adaptive attacks, OPGD and SPGD. The accuracy is evaluated at two perturbation levels, $\epsilon = 0.01$ and $\epsilon = 8/255$, and is reported for two false-positive rates (FP), 5% (FP@5) and 50% (FP@50).

Attack	Defense	$\epsilon = 0.01$		$\epsilon = 8/255$	
		$\text{Acc}_{\text{robust}}FP@5$	$\text{Acc}_{\text{robust}}FP@50$	$\text{Acc}_{\text{robust}}FP@5$	$\text{Acc}_{\text{robust}}FP@50$
OPGD	**RADAR**	99.2%	**99.3%**	**99.8%**	**99.9%**
	ContraNet [55]	93.7%	99.2%	89.8%	94.7%
	Trapdoor [50]	0.0%	7.0%	0.0%	8.0%
	DLA [51]	62.6%	83.7%	0.0%	28.2%
	SID [52]	6.9%	23.4%	0.0%	1.6%
SPGD	**RADAR**	**99.2%**	**99.1%**	**99.8%**	**99.9%**
	ContraNet [55]	93.7%	99.3%	89.7%	95.1%
	Trapdoor [50]	0.2%	49.5%	0.4%	37.2%
	DLA [51]	17.0%	55.9%	0.0%	13.5%
	SID [52]	8.9%	50.9%	0.0%	11.4%

Susceptibility of datasets and models to adversarial attacks. The susceptibility of machine learning models to adversarial attacks is influenced both by the characteristics of the datasets and by the architectures of the models employed. Datasets with higher complexity, such as those containing more classes, higher-resolution images, or greater intra-class variability, can lead to models that are more vulnerable to adversarial perturbations. For example, models trained on datasets like ImageNet may exhibit higher susceptibility compared with those trained on simpler datasets like CIFAR-10 or SVHN due to the increased complexity and diversity of the data, as shown in Table 1. Model architecture also plays a significant role in vulnerability to adversarial attacks. Deep neural networks with complex and highly non-linear decision boundaries can be more easily manipulated by

small perturbations. Models with a larger number of parameters may overfit to the training data, making them less robust to unseen adversarial examples. However, larger and deeper models can also capture more complex patterns and representations, potentially enhancing their ability to generalize and resist certain types of adversarial attacks. This is evidenced in our experimental results (see Tables 5 and 6 and Figure 3), where deeper architectures demonstrate improved performance in detecting adversarial inputs.

Ablation Studies

In this section, we conduct comprehensive ablation studies to evaluate the impact of critical hyperparameters in our proposed loss function on the performance of our adversarial detectors on the validation set. Specifically, we focus on these five key parameters: (1) number of steps, (2) step size (α), (3) batch size, (4) learning rate, and (5) whether the model was pretrained through standard training. Each of these parameters plays a significant role in the training process and potentially influences the robustness and effectiveness of our adversarial detectors. To evaluate the effectiveness and trade-offs associated with these parameters, we conducted a series of experiments using the VGG-11 architecture on the CIFAR-10 dataset. The results are delineated in Figure 10.

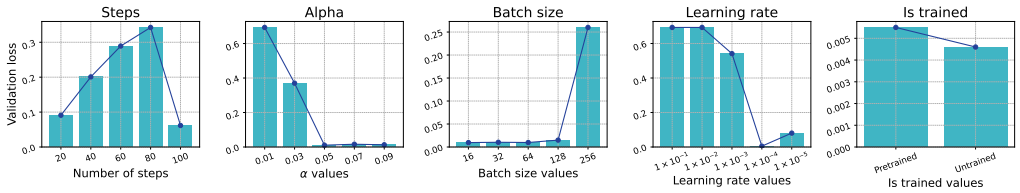


Figure 10. Ablation studies were conducted using VGG-11, varying the (1) number of steps, (2) step size α, (3) batch size, (4) learning rate, and (5) whether the model was pretrained through standard training, presented sequentially from left to right.

The results reveal that the number of steps significantly affects the model's performance. The detector's performance reaches its peak at 100 steps. This suggests that more iterations in the adversarial training process lead to better optimization and enhanced robustness of the adversarial detector. The choice of α also plays a critical role. Our findings show that a smaller α (0.05) yields the best performance, while increasing α increases effectiveness. This indicates that bigger perturbations during training are more effective in strengthening the detector's resilience to adversarial attacks. However, we want to detect attacks that are unrecognizable; thus we chose to use $\alpha = 0.03$.

Regarding batch size, the performance remains relatively stable across smaller batch sizes but shows a notable decline at the largest batch size tested (256).

Learning rate is another crucial factor. Initially, learning rates of 1×10^{-1} and 1×10^{-2} perform worse, but as the learning rate decreases, there is an improvement in performance. However, a learning rate that is too small (1×10^{-5}) results in under-performance, highlighting the need for a balanced learning rate to ensure effective training.

The "Is trained" graph reveals that pretrained models exhibit a slightly higher validation loss than untrained ones, suggesting slightly poorer performance. This may be due to untrained models' overfitting to the new adversarial data, while pretrained models—already informed by standard training—generalize better. Thus, pretraining appears to provide a foundation but may limit adaptability to newly introduced adversarial nuances.

Based on this ablation study, we selected the hyperparameters that yielded the best performance: a batch size of 32, a learning rate of 1×10^{-2}, 100 steps, and an α value of 0.03. These settings were found to optimize the robustness of our adversarial detector, providing a strong defense against adversarial attacks.

As illustrated in Figure 11, we conducted an in-depth analysis of the impact of excluding standard training and relying solely on adversarial training across three distinct datasets: CIFAR-10, SVHN, and ImageNet.

For CIFAR-10, the results demonstrate that standard training significantly boosts performance—particularly for ResNet models—at lower epsilon values (1/255 and 2/255). Accuracy for ResNet-18, for example, shows a clear advantage with standard training, highlighting its role in enhancing model robustness at smaller perturbations. However, as the epsilon values increase, the gap between models trained with and without standard training narrows, suggesting that adversarial training alone is sufficient to maintain robustness at higher perturbations.

In the case of SVHN, the benefits of standard training are evident in ResNet-based models but less pronounced compared with CIFAR-10. Accuracy trends show that while standard training does provide a marginal improvement—particularly noticeable for the ResNet architectures—models trained exclusively with adversarial training still achieve good performance. This suggests that, for simpler datasets, e.g., SVHN, the reliance on standard training might be reduced without significant loss in robustness, especially at mid to high epsilon values.

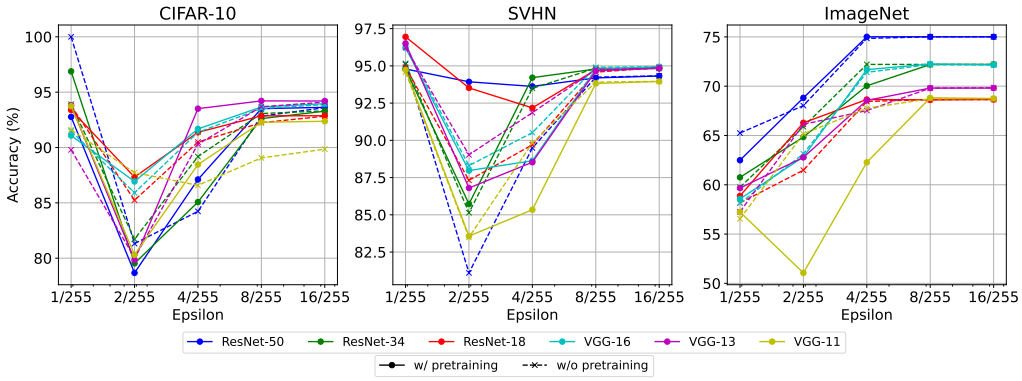


Figure 11. Comparison of classification accuracies of models utilizing adversarial detectors trained with standard pretraining followed by adversarial training (solid lines) versus detectors trained exclusively with adversarial training (dashed lines) across CIFAR-10, SVHN, and ImageNet datasets. The plot demonstrates the impact of increasing ϵ values on the accuracy of the classification models when using the respective adversarial detectors.

For ImageNet, results show a relatively modest impact with standard training compared with the other datasets. Performance improvement when incorporating standard training is not as pronounced, which may be attributed to the fact that all models in this evaluation were pretrained on ImageNet. This pretraining likely provided the models with a strong baseline robustness, diminishing the relative benefit of standard training in this specific case. As with the other datasets, when the epsilon values increase, the performance gap between models with and without standard training narrows further, suggesting that adversarial training alone is effective at higher perturbation strengths.

Overall, the analysis across these datasets indicates that while standard training contributes positively to model robustness, particularly at lower epsilon values, adversarial training alone can achieve comparable performance as the perturbation strength increases. The decision to incorporate standard training may therefore depend on the specific dataset and the computational resources available.

6. Conclusions

We presented a paradigmatic shift in the approach to adversarial training of deep neural networks, transitioning from fortifying classifiers to fortifying networks dedicated to adversarial detection. Our findings illuminate the prospective capacity to endow deep neural networks with resilience against adversarial attacks. Rigorous empirical inquiries substantiate the efficacy of our developed adversarial training methodology.

There is no trade-off between clean and adversarial classification because the classifier is not modified.

Our results bring forth a sense of optimism regarding the attainability of adversarially robust deep learning detectors. Notably, the significant robustness exhibited by our networks across the datasets examined manifested in increased accuracy against a diverse array of potent ℓ_∞-bound adversaries.

However, despite the promising results, our approach has a number of limitations that warrant further investigation. First, our adversarial training methodology primarily focuses on ℓ_∞-norm bounded attacks, which may limit its effectiveness against adversaries employing different norms, such as ℓ_2 or ℓ_1 attacks. This norm-specific robustness suggests that our detectors might not generalize well to all possible adversarial perturbations.

A further limitation lies in the dependency of **RADAR**'s effectiveness on the architecture of the detector: detectors with limited expressive capacity may fail to provide adequate gradient signals, thus hindering the robustness of the detection mechanism.

Additionally, the computational overhead associated with training and deploying the adversarial detector could pose challenges for resource-constrained applications. The necessity for extensive adversarial examples during training increases computational cost and may not be feasible for larger-scale systems.

Future research directions include extending our adversarial training approach to cover a wider range of attack norms and developing norm-agnostic detection mechanisms. Investigating the scalability of our methodology to larger and more complex datasets, as well as its applicability to different neural network architectures, remains an open question for now.

Our study not only contributes valuable insights into enhancing the resilience of deep neural networks against adversarial attacks but also underscores the importance of continued exploration in this domain. Therefore, we encourage researchers to persist in their endeavors to advance the frontier of adversarially robust deep learning detectors.

Author Contributions: Writing—original draft, A.D.; Methodology, R.L.; Software, R.L.; Writing—review & editing, R.L. and M.S.; Supervision, M.S.; Project administration, M.S. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by the Israeli Innovation Authority through the Trust.AI consortium.

Data Availability Statement: Data are contained within the article.

Conflicts of Interest: Authors Raz Lapid and Almog Dubin were employed by the company DeepKeep. The remaining authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

References

1. Huang, J.; Chai, J.; Cho, S. Deep learning in finance and banking: A literature review and classification. *Front. Bus. Res. China* **2020**, *14*, 13. [CrossRef]
2. Miotto, R.; Wang, F.; Wang, S.; Jiang, X.; Dudley, J.T. Deep learning for healthcare: Review, opportunities and challenges. *Brief. Bioinform.* **2018**, *19*, 1236–1246. [CrossRef] [PubMed]
3. Chen, L.; Pelger, M.; Zhu, J. Deep learning in asset pricing. *Manag. Sci.* **2024**, *70*, 714–750. [CrossRef]
4. Shamshirband, S.; Fathi, M.; Dehzangi, A.; Chronopoulos, A.T.; Alinejad-Rokny, H. A review on deep learning approaches in healthcare systems: Taxonomies, challenges, and open issues. *J. Biomed. Inform.* **2021**, *113*, 103627. [CrossRef]
5. Lv, Z.; Zhang, S.; Xiu, W. Solving the security problem of intelligent transportation system with deep learning. *IEEE Trans. Intell. Transp. Syst.* **2020**, *22*, 4281–4290. [CrossRef]
6. Goodfellow, I.J.; Shlens, J.; Szegedy, C. Explaining and harnessing adversarial examples. *arXiv* **2014**, arXiv:1412.6572.
7. Szegedy, C.; Zaremba, W.; Sutskever, I.; Bruna, J.; Erhan, D.; Goodfellow, I.; Fergus, R. Intriguing properties of neural networks. *arXiv* **2013**, arXiv:1312.6199.
8. Alter, T.; Lapid, R.; Sipper, M. On the Robustness of Kolmogorov-Arnold Networks: An Adversarial Perspective. *arXiv* **2024**, arXiv:2408.13809.
9. Tamam, S.V.; Lapid, R.; Sipper, M. Foiling Explanations in Deep Neural Networks. *arXiv* **2023**, arXiv:2211.14860.
10. Chen, Z.; Li, B.; Wu, S.; Jiang, K.; Ding, S.; Zhang, W. Content-based unrestricted adversarial attack. *Adv. Neural Inf. Process. Syst.* **2024**, *36*, 51719–51733 .
11. Lapid, R.; Langberg, R.; Sipper, M. Open Sesame! Universal Black-Box Jailbreaking of Large Language Models. In Proceedings of the ICLR 2024 Workshop on Secure and Trustworthy Large Language Models, Vienna, Austria, 11 May 2024.

12. Carlini, N.; Wagner, D. Towards evaluating the robustness of neural networks. In Proceedings of the 2017 IEEE Symposium on Security and Privacy (SP), San Jose, CA, USA, 22–26 May 2017; pp. 39–57.
13. Lapid, R.; Sipper, M. I See Dead People: Gray-Box Adversarial Attack on Image-To-Text Models. In Proceedings of the 5th Workshop on Machine Learning for Cybersecurity, part of ECMLPKDD 2023, Turin, Italy, 22 September 2023 .
14. Andriushchenko, M.; Croce, F.; Flammarion, N.; Hein, M. Square attack: A query-efficient black-box adversarial attack via random search. In Proceedings of the European Conference on Computer Vision. Springer, Glasgow, UK, 23–28 August 2020; pp. 484–501.
15. Lapid, R.; Haramaty, Z.; Sipper, M. An evolutionary, gradient-free, query-efficient, black-box algorithm for generating adversarial instances in deep convolutional neural networks. *Algorithms* **2022**, *15*, 407. [CrossRef]
16. Lapid, R.; Sipper, M. Patch of invisibility: Naturalistic black-box adversarial attacks on object detectors. *arXiv* **2023**, arXiv:2303.04238.
17. Finlayson, S.G.; Bowers, J.D.; Ito, J.; Zittrain, J.L.; Beam, A.L.; Kohane, I.S. Adversarial attacks on medical machine learning. *Science* **2019**, *363*, 1287–1289. [CrossRef]
18. Liu, M.; Zhang, Z.; Chen, Y.; Ge, J.; Zhao, N. Adversarial attack and defense on deep learning for air transportation communication jamming. *IEEE Trans. Intell. Transp. Syst.* **2023**, *25*, 973–986. [CrossRef]
19. Madry, A.; Makelov, A.; Schmidt, L.; Tsipras, D.; Vladu, A. Towards deep learning models resistant to adversarial attacks. *arXiv* **2017**, arXiv:1706.06083.
20. Zhao, W.; Alwidian, S.; Mahmoud, Q.H. Adversarial Training Methods for Deep Learning: A Systematic Review. *Algorithms* **2022**, *15*, 283. [CrossRef]
21. Lu, J.; Issaranon, T.; Forsyth, D. Safetynet: Detecting and rejecting adversarial examples robustly. In Proceedings of the IEEE International Conference on Computer Vision, Venice, Italy, 22–29 October 2017; pp. 446–454.
22. Grosse, K.; Manoharan, P.; Papernot, N.; Backes, M.; McDaniel, P. On the (statistical) detection of adversarial examples. *arXiv* **2017**, arXiv:1702.06280.
23. Gong, Z.; Wang, W. Adversarial and clean data are not twins. In Proceedings of the Sixth International Workshop on Exploiting Artificial Intelligence Techniques for Data Management, Seattle, WA, USA, 18 June 2023; pp. 1–5.
24. Lust, J.; Condurache, A.P. Gran: An efficient gradient-norm based detector for adversarial and misclassified examples. *arXiv* **2020**, arXiv:2004.09179.
25. Metzen, J.H.; Genewein, T.; Fischer, V.; Bischoff, B. On Detecting Adversarial Perturbations. In Proceedings of the International Conference on Learning Representations, San Juan, Puerto Rico, 2–4 May 2016.
26. Carlini, N.; Wagner, D. Adversarial examples are not easily detected: Bypassing ten detection methods. In Proceedings of the 10th ACM Workshop on Artificial Intelligence and Security, Dallas, TX, USA, 3 November 2017; pp. 3–14.
27. Grosse, K.; Papernot, N.; Manoharan, P.; Backes, M.; McDaniel, P. Adversarial examples for malware detection. In Proceedings of the Computer Security–ESORICS 2017: 22nd European Symposium on Research in Computer Security, Oslo, Norway, 11–15 September 2017; Proceedings, Part II 22; pp. 62–79.
28. LeCun, Y.; Bottou, L.; Bengio, Y.; Haffner, P. Gradient-based learning applied to document recognition. *Proc. IEEE* **1998**, *86*, 2278–2324. [CrossRef]
29. Krizhevsky, A. Learning Multiple Layers of Features from Tiny Images, 2009. Available online: https://www.cs.toronto.edu/~kriz/learning-features-2009-TR.pdf (accessed on 9 September 2024).
30. Kurakin, A.; Goodfellow, I.J.; Bengio, S. Adversarial Machine Learning at Scale. In Proceedings of the International Conference on Learning Representations, San Juan, Puerto Rico, 2–4 May 2016.
31. Zhang, H.; Cisse, M.; Dauphin, Y.N.; Lopez-Paz, D. mixup: Beyond empirical risk minimization. *arXiv* **2017**, arXiv:1710.09412.
32. Tramer, F.; Boneh, D. Adversarial training and robustness for multiple perturbations. In Proceedings of the 33rd Conference on Neural Information Processing Systems, Vancouver, BC, Canada, 8–14 December 2019; Volume 32 .
33. Cohen, J.; Rosenfeld, E.; Kolter, Z. Certified adversarial robustness via randomized smoothing. In Proceedings of the International Conference on Machine Learning. PMLR, Long Beach, CA, USA, 9–15 June 2019; pp. 1310–1320.
34. Lecuyer, M.; Atlidakis, V.; Geambasu, R.; Hsu, D.; Jana, S. Certified robustness to adversarial examples with differential privacy. In Proceedings of the 2019 IEEE Symposium on Security and Privacy (SP), San Francisco, CA, USA, 19–23 May 2019; pp. 656–672.
35. Li, L.; Xie, T.; Li, B. Sok: Certified robustness for deep neural networks. In Proceedings of the 2023 IEEE Symposium on Security and Privacy (SP), San Francisco, CA, USA, 22–25 May 2023; pp. 1289–1310.
36. Singh, G.; Gehr, T.; Mirman, M.; Püschel, M.; Vechev, M. Fast and effective robustness certification. *Adv. Neural Inf. Process. Syst.* **2018**, *31*, 10825–10836.
37. Zhang, H.; Yu, Y.; Jiao, J.; Xing, E.; El Ghaoui, L.; Jordan, M. Theoretically principled trade-off between robustness and accuracy. In Proceedings of the International Conference on Machine Learning, PMLR, Long Beach, CA, USA, 9–15 June 2019; pp. 7472–7482.
38. Xie, C.; Wu, Y.; Maaten, L.v.d.; Yuille, A.L.; He, K. Feature denoising for improving adversarial robustness. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Long Beach, CA, USA, 15–20 June 2019; pp. 501–509.
39. Pinhasov, B.; Lapid, R.; Ohayon, R.; Sipper, M.; Aperstein, Y. Xai-based detection of adversarial attacks on deepfake detectors. *arXiv* **2024**, arXiv:2403.02955.

40. Došilović, F.K.; Brčić, M.; Hlupić, N. Explainable artificial intelligence: A survey. In Proceedings of the 2018 41st International Convention on Information and Communication Technology, Electronics and Microelectronics (MIPRO), Opatija, Croatia, 21–25 May 2018; pp. 0210–0215.
41. Carmon, Y.; Raghunathan, A.; Schmidt, L.; Duchi, J.C.; Liang, P.S. Unlabeled data improves adversarial robustness. In Proceedings of the Advances in Neural Information Processing Systems, Vancouver, BC, Canada, 8–14 December 2019; Volume 32.
42. Cheng, S.; Dong, Y.; Pang, T.; Su, H.; Zhu, J. Improving black-box adversarial attacks with a transfer-based prior. *Adv. Neural Inf. Process. Syst.* **2019**, *32*.
43. Dong, Y.; Cheng, S.; Pang, T.; Su, H.; Zhu, J. Query-efficient black-box adversarial attacks guided by a transfer-based prior. *IEEE Trans. Pattern Anal. Mach. Intell.* **2021**, *44*, 9536–9548. [CrossRef] [PubMed]
44. Yang, Y.Y.; Rashtchian, C.; Zhang, H.; Salakhutdinov, R.R.; Chaudhuri, K. A closer look at accuracy vs. robustness. *Adv. Neural Inf. Process. Syst.* **2020**, *33*, 8588–8601.
45. He, Z.; Yang, Y.; Chen, P.Y.; Xu, Q.; Ho, T.Y. Be Your Own Neighborhood: Detecting Adversarial Example by the Neighborhood Relations Built on Self-Supervised Learning, 2022. Available online: http://arxiv.org/abs/2209.00005 (accessed on 9 September 2024).
46. Klingner, M.; Kumar, V.R.; Yogamani, S.; Bär, A.; Fingscheidt, T. Detecting Adversarial Perturbations in Multi-Task Perception. In Proceedings of the 2022 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), Kyoto, Japan, 23–27 October 2022. [CrossRef]
47. Yang, Y.; Gao, R.; Li, Y.; Lai, Q.; Xu, Q. What You See is Not What the Network Infers: Detecting Adversarial Examples Based on Semantic Contradiction. In Proceedings of the Proceedings 2022 Network and Distributed System Security Symposium. Internet Society, NDSS 2022, San Diego, CA, USA, 24–28 April 2022. [CrossRef]
48. Wang, Y.; Li, T.; Li, S.; Yuan, X.; Ni, W. New Adversarial Image Detection Based on Sentiment Analysis. *IEEE Trans. Neural Netw. Learn. Syst.* **2023**, *35*, 14060–14074. [CrossRef]
49. Bryniarski, O.; Hingun, N.; Pachuca, P.; Wang, V.; Carlini, N. Evading Adversarial Example Detection Defenses with Orthogonal Projected Gradient Descent. In Proceedings of the International Conference on Learning Representations, Vienna, Austria, 4 May 2021.
50. Shan, S.; Wenger, E.; Wang, B.; Li, B.; Zheng, H.; Zhao, B.Y. Gotta catch'em all: Using honeypots to catch adversarial attacks on neural networks. In Proceedings of the 2020 ACM SIGSAC Conference on Computer and Communications Security, Virtual Event USA, 9–13 November 2020; pp. 67–83.
51. Sperl, P.; Kao, C.Y.; Chen, P.; Lei, X.; Böttinger, K. DLA: Dense-layer-analysis for adversarial example detection. In Proceedings of the 2020 IEEE European Symposium on Security and Privacy (EuroS&P), Genoa, Italy, 7–11 September 2020; pp. 198–215.
52. Tian, J.; Zhou, J.; Li, Y.; Duan, J. Detecting adversarial examples from sensitivity inconsistency of spatial-transform domain. In Proceedings of the AAAI Conference on Artificial Intelligence, Virtual, 2–9 February 2021; Volume 35, pp. 9877–9885.
53. Netzer, Y.; Wang, T.; Coates, A.; Bissacco, A.; Wu, B.; Ng, A.Y. Reading digits in natural images with unsupervised feature learning. In Proceedings of the NIPS Workshop on Deep Learning and Unsupervised Feature Learning, Granada, Spain, 12–14 December 2011; Volume 2011, p. 7.
54. Deng, J.; Dong, W.; Socher, R.; Li, L.J.; Li, K.; Fei-Fei, L. Imagenet: A large-scale hierarchical image database. In Proceedings of the 2009 IEEE Conference on Computer Vision and Pattern Recognition, Miami, FL, USA, 20–25 June 2009; pp. 248–255.
55. Yang, Y.; Gao, R.; Li, Y.; Lai, Q.; Xu, Q. What you see is not what the network infers: Detecting adversarial examples based on semantic contradiction. *arXiv* **2022**, arXiv:2201.09650.

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.