Adversarial Pixel and Patch Detection Using Attribution Analysis

Chase Walker, Dominic Simon
Department of ECE
University of Central Florida
Orlando, FL, USA
{chase.walker, dominic.simon}@ucf.edu

Sumit Kumar Jha
Knights Foundation School of Computing and Information Sciences
Florida International University
jha@cs.fiu.edu

Rickard Ewetz
Department of ECE
University of Central Florida
Orlando, FL, USA
rickard.ewetz@ucf.edu

***Abstract*—Next-generation warfighters will use sensors and deep learning for advanced scene recognition and situational awareness. Adversarial pixel and patch attacks can severely degrade the performance of deep neural networks (DNNs). This poses a critical security threat future combat systems. Detecting adversarial attacks has been attempted before, but recent advances in explainable artificial intelligence (XAI) have opened the door to better detection methods using attribution analysis. In particular, we observe that benign and attacked images display different characteristics in their attribution maps. Benign images tend to have dense attributions due to the network focusing on the main object of the image, while attacked images tend to have more sparse attributions due to the widespread perturbations applied. Using this intuition, we propose a framework for adversarial attack detection in the form of a binary classifier. Using three methods: Integrated Gradients (IG), Guided Backpropagation (GBP), and Integrated Decision Gradients (IDG), we propose the training of a binary classifier that can analyze an attribution map to detect attacked input data. We evaluate the detection framework for three state-of-the-art attacks with the three attribution analysis methods. We find that IDG achieves state-of-the-art pixel attack detection performance with up to 99% accuracy, and GBP manages state-of-the-art patch detection performance achieving up to 88% accuracy.**

***Index Terms*—Adversarial Robustness, Explainable Artificial Intelligence**

I. Introduction

Similar to other facets of the modern world, military technology has come to rely on deep learning and deep neural networks (DNNs). The OODA loop (observe, orient, decide, act) is now expected to be performed not by humans operating in seconds, but at the speed of light by DNNs. Next-generation wars will therefore be fought with an ever-increasing reliance on technology supported by DNNs. However, research has shown that DNNs used in computer vision applications are susceptible to a number of attacks, including pixel adversarial attacks [1], [2] and adversarial patches [3], [4]. Pixel adversarial attacks manipulate DNN predictions by applying imperceptible perturbations across an entire image. Adversarial patches also manipulate DNN predictions, but the perturbation size is unbounded and localized to a contiguous region on the image. These attacks have shown to be effective in both the digital and physical domains in applications such as copyright [5] and person detection [6].

One approach of detecting these attacks is through attributions [7]. Attributions are part of a wide set of algorithms, known as explainable artificial intelligence, which attempt to make DNNs more interpretable for humans. Attribution analysis determines the contribution of input features to a model decision, often presenting this information in the form of a heat map. Attribution analysis broadly falls into two categories: backpropagation-based [8]–[10] and perturbation-based [11] methods. Due to their speed and quality, backpropagation-based methods are the most popular methods [12].

Previous attribution-based detection approaches separate benign and adversarial attributions through the analysis of attribution pixels. In [13], benign and attacked attributions are shown to be distinct in the magnitude distribution of the pixels. In [14], masking the top attribution pixels of a benign image is shown to have little effect on classification, while it often causes a change in predicted label for attacked images.

Recent advances in attribution analysis have allowed us to observe clear differences in the attribution maps of benign and attacked images. Benign images often have dense, focused attributions on the main object in an image. However, attacked images tend to have sparse, unfocused attributions due to the confusion caused by attacks. Leveraging this insight, we present a framework which trains a binary classifier on attributions to detect adversarial examples from input data. The main contributions of this paper are:

- We observe the attributions of benign and adversarial images have different characteristics.
- We design an attack detection framework which can successfully detect pixel and patch adversarial examples via attribution analysis using a binary classifier.
- We evaluate the proposed framework with three state-of-the-art attack methods and measure up to 88% and 99% patch and pixel attack detection, respectively.

The paper is organized as follows: related work is explored in Section II, an attribution analysis case study is performed in Section III, the classifier is presented in Section IV, evaluation is performed in Section V, the paper is concluded and future work is discussed in Section VI.

This work was partly supported by the DARPA cooperative agreement #HR00112020002. The views, opinions and/or findings expressed are those of the authors and should not be interpreted as representing the official views or policies of the Department of Defense or the U.S. Government.

II. Related Work

In this section, we cover the attacks and attribution analysis methods used in the design of the adversarial attack detector binary classifier. We additionally explore previous attack detection approaches.

A. Attack Types

To evaluate our detection method, we use Projected Gradient Descent (PGD) [2] for the pixel attacks and Local and Adversarial Noise (LaVAN) [3] for the patch attacks.

a) Projected Gradient Descent: The PGD pixel attack is completed over multiple steps based on the popular Fast Gradient Sign Method [15]. At each iterative step, a move of size α is made in the gradient direction of a model's cost function J. Following this, the resulting perturbation is clipped to a maximum allowable change of ϵ in the positive or negative direction as follows:

$$x_{n+1}^{adv} = \text{Clip}_{-\epsilon}^{+\epsilon}(x_n^{adv} + \alpha \times sign(\nabla_x J(\theta, x_n^{adv}, y))) \quad (1)$$

where $x_0^{adv} = x$ the original input, y is the label, and θ is the model parameters [2]. We employ both the L_2 and L_∞ versions of the attack [2].

b) Local and Visible Adversarial Noise: LaVAN is an untargeted per-image patch. Per-image patches are generated with respect to their target image, making them nontransferable, but significantly stronger than universal patches, which attack an entire dataset [16]. A LaVAN patch $\hat{p}$ can be generated through the following optimization process:

$$\hat{p} = \text{argmin}\mathbb{E}[logP(y|n, x, l)] \quad (2)$$

where noise n is added to the image x at location l so that the log probability of the image's class y is minimized [3].

B. Attribution Analysis Methods

We will discuss the characteristics of attributions generated by two foundational methods: guided backpropagation (GBP) and integrated gradients (IG), and then we explore integrated decision gradients (IDG).

a) Guided Backpropagation: One of the first attribution analysis methods is called saliency or the plain backpropagation method [7]. By feeding an input to a model and performing one backpropagation pass, model gradients at the output are captured, and these gradients are displayed as a heat map for the attribution visualization [7]. However, the nonlinear activations used in DNNs result in the attribution to be noisy. GBP modifies the original approach, allowing only nonnegative gradients to propagate which results in a substantial reduction in attribution noise [10].

b) Integrated Gradients: IG is also an expansion on the saliency method. It performs a backpropagation pass to capture gradients for multiple images which come from the interpolation from a black image to the input image [8]. This process results in attributions with reduced noise when compared to saliency. However, due to the characteristic of many of the individually captured gradients in IG having high noise, the output retains a large amount of noise around the image subject.

c) Integrated Decision Gradients: IDG, like IG, gathers gradients from the interpolated images via the backpropagation operation, but it eliminates the noisy gradients that IG traditionally captures by detecting which gradients do not correspond to growth in model confidence [9]. By removing these noisy gradients before averaging, IDG generates sharp attributions that are highly focused on the subject and provide dense groupings on the features which are important for the model decision [9].

We compare the three attribution analysis methods for an input from the ImageNet "Warplane" class [17] in Figure 1. It is clear that the noise in the IG attribution is substantial, and the objects of focus are lost in the result. When comparing GBP and IDG, we see in each method that the focus is mainly on the "Warplane" objects, but IDG retains less noise in its attribution. The lower noise of an attribution method creates room for a clear difference to be seen between attributions of benign and attacked images. We explore this in Section III.

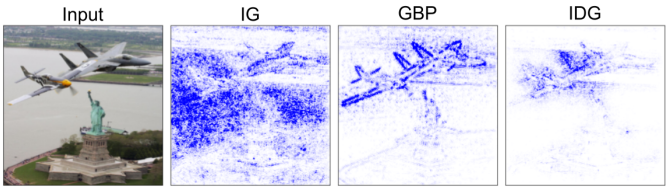

Fig. 1. A comparison of integrated gradients [8], guided backpropagation [10], and integrated decision gradients [9]. It is clear that IDG has the lowest amount of noise in the benign image attribution. This makes IDG a strong candidate for the detection framework, as the IDG attributions for benign and attacked images are more likely to have a greater difference than with other methods since the initial noise is very low.

C. Attack Detection Frameworks

Defenses that attempt to detect attacks through input image analysis use a variety of approaches. Adversary Detection Networks [18] add layers to existing networks which give a binary prediction on whether an input is adversarial. Spectral Defense [19] determined that pixel adversarial attacks cause easily detectable perturbations in the Fourier domain, so they use a binary classifier to detect attacked images from their Fourier domain transformations. Scalecert [20] determined that adversarial patches must cause large activation values in the beginning layers of DNNs for their effects to propagate to the network output, so the detector analyzes the top layers of a network to determine whether an input is adversarial.

Prior detection approaches have also relied on attributions [13], [14]. In ML-LOO [13] the attributions of benign and pixel attacked images are analyzed, and it is shown the two attributions have distinct and detectable distributions. However, these approaches are constrained by noisy attribution analysis methods that are unable to accurately pinpoint the most important pixels.

With the development of better attribution analysis methods, can we develop a binary classifier to detect an attacked input from its attribution?

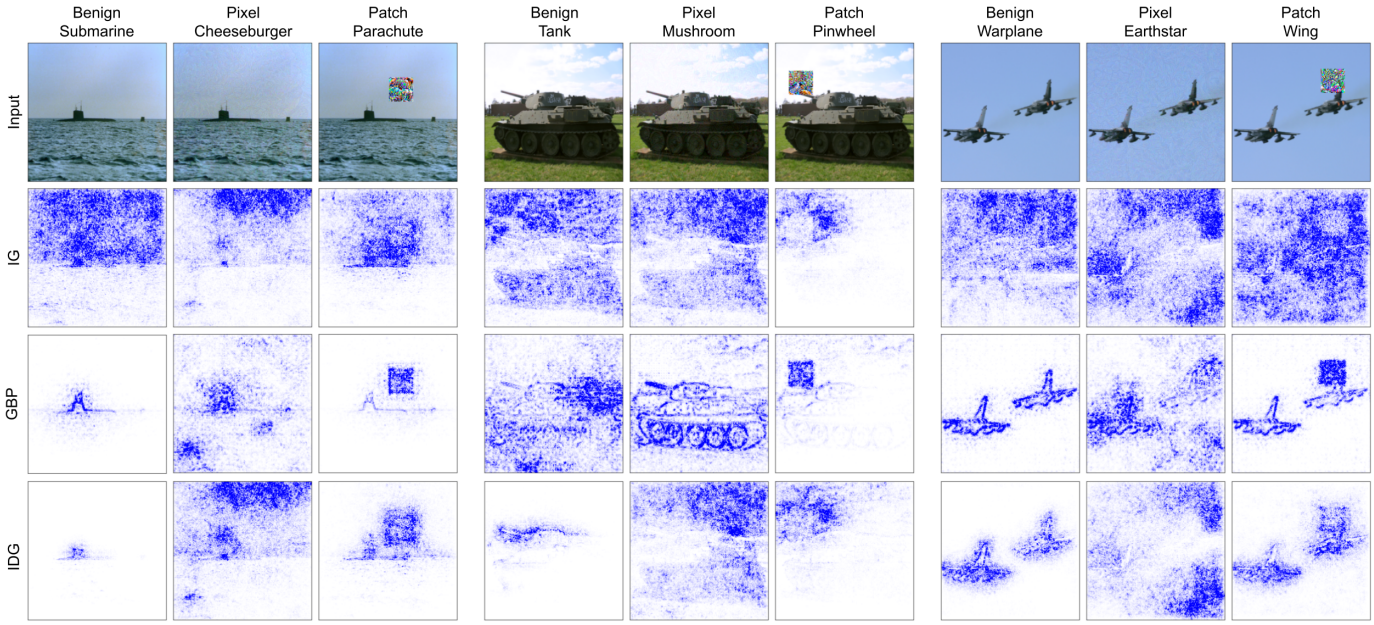

Fig. 2. A case study of the attributions methods and attacks. We compare the attributions from IG [8], GBP [10], and IDG [9] for the pixel (PGD L_∞) [2] and patch (LaVAN) [3] attacks. We examine each given method by the difference between the benign and attacked attribution outputs. It is clear that the IG method does not generate clearly distinguishable attributions between a benign and attacked image, presenting a potential challenge for classification. Benign and pixel attacked IDG attributions show a clear difference in attribution distribution and noise. GBP attributions for patch attacks clearly highlight the square patch in each sample. IDG and GBP will therefore be strong choices for pixel and patch attack detection, respectively.

III. Case Study

The integrated decision gradients method has proven to reliably generate sharp attributions on the subject of an image. This created promise for the use of IDG to help determine if an image is attacked. In Figure 2 we perform a case study of the output behavior of three attribution analysis methods: IG, GBP, and IDG when they are given benign and adversarial images as input with reference to a pixel and patch attack. In the figure, the pixel and patch attacks are PGD L_∞ [2] and LaVAN patch [3] attacks, respectively. The top row presents the benign image, pixel, and patch attacks for the three sample images. Additionally, the original or new class is listed under the image type. The following three rows of attributions show the output of the IG, GBP, and IDG attribution analysis methods respectively.

a) Pixel Attack Attributions: We first analyze the benign and pixel attacked attributions shown in the first and second columns of the three groups of images. First observing IG, we see that the benign and pixel attacked attributions appear to be very similar, with no clear distinction in the attribution distribution, making classification potentially challenging. When viewing GBP, there is more contrast between a given benign and attacked attribution than IG, as a sharp attribution becomes noisy after attack. However, in comparison to IDG, the benign GBP attribution is more noisy, and the attacked GBP attribution is less noisy than those from IDG. Clearly, benign and attacked attributions generated from IDG have the largest difference out of the three methods. This indicates that IDG will be the strongest performer for pixel attack detection.

b) Patch Attack Attributions: We now analyze patch attack attributions. Once, again IG does not show a consistent difference between benign and patch attributions, which insinuates a class boundary will be hard to determine. When viewing IDG, we make two observations. First, the focus of the attribution consistently shifts towards the patch location. Second, the resulting attribution is amorphous and relatively small. Together, this leads to patch attack attributions appearing with a similar form as benign IDG attributions. This is unlike GBP. For GBP, in all presented examples, the patch is highlighted in the attributions as a sharp, dense, square. This creates an east-to-extract feature for a classifier. Due to this, GBP should provide the best detection rate out of the three attribution analysis methods for patch attacks.

We recognize these observations may not always hold true and the distribution of attacked and benign attributions could be the opposite, which is a topic of further investigation. However, we operate under the assumption that the observed distributions hold true and design pixel and GBP patch attack detectors using IDG and GBP respectively.

IV. Methodology

In this section we present the framework for implementing the proposed classifier. We present the classifier architecture, detail the data generation process, and provide the training process shown in Figure 3.

A. Proposed Detector Architecture

We treat the objective of detecting adversarial examples as an image classification task. Analyzing the attributions of

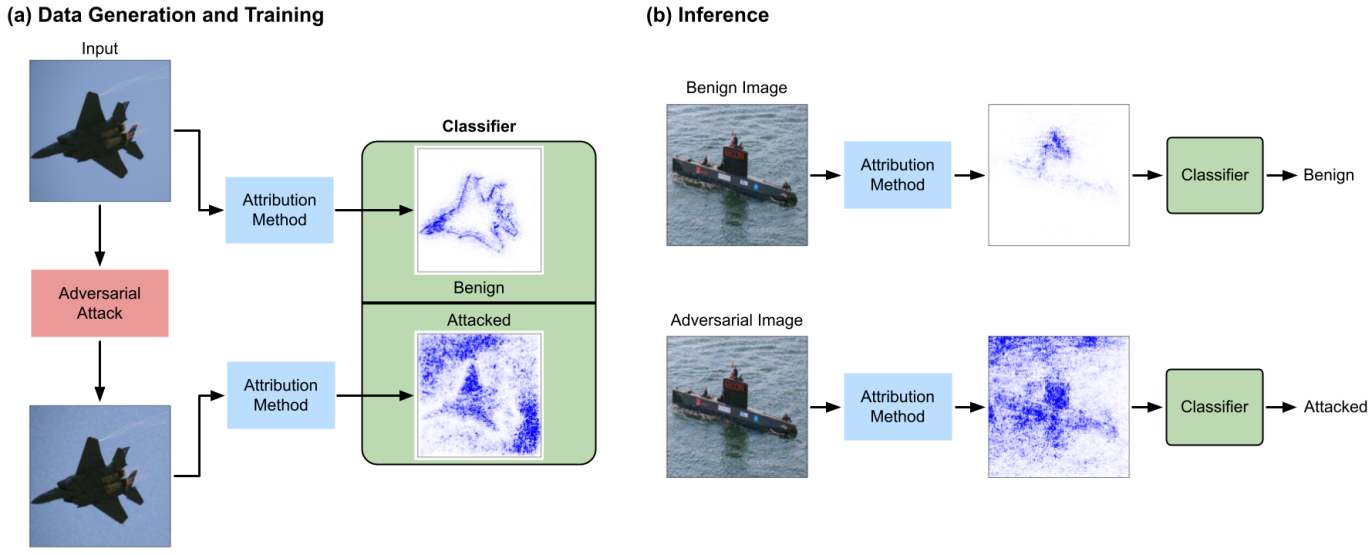

Fig. 3. An overview of the full detector training and inference process. There are two steps to the process shown in (a): data generation and training. During data generation, for every training input image selected, a given adversarial attack is applied (PGD [2] L_∞ or L_2 or LaVAN patch [3]. This results in benign and attacked images, both of which are given to an attribution analysis method (IG [8], GBP [10], or IDG [9]) and a benign and attacked attribution are created. Once this is done for all training images, the binary classifier is trained on the two classes of attributions. For inference in (b), an image will be classified as benign or attacked when an attribution of the image is given to the respective classifier (e.g. a classifier trained on IDG attributions of benign and PGD L_∞ images) which can predict if the image is attacked or benign.

benign and attacked images shows the existence of a clear distinction in noise levels and noise distribution between the two classes of attributions. Therefore aiming to distinguish attributions from these two classes, an image classification CNN will provide a strong foundation.

The classifier design is built from a ResNet101 [21] backbone. ResNet101 is a powerful and well-founded CNN architecture which is suited for the task of classifying high-information images. As we generate the attributions from the ImageNet dataset [17], this strong backbone for the classifier is necessary for the high-information attributions. To train the ResNet101 classifier to provide binary classification of attributions we start without weights, and modify the final fully-connected layer of the model to an output size of 2. We present the training details of this classifier in Section IV-C.

B. Data Generation

Before training a classifier we generate training and evaluation sets of benign and adversarial attributions. We source input images from the 2012 ImageNet training images dataset for training data generation.

We used the following criteria for image selection: the image must be RGB, correctly classified by ResNet101, and the subject must cover no more than 50% of the image area. This image area limitation is in place to give all the attribution analysis methods the best possibility of generating a sharp benign attribution. In general, the smaller the subject, the more focused the attribution and the sharper the benign attribution, the more separable the attacked attributions will be.

All selected images are attacked and attributions for the benign and attacked images are generated. This results in nine datasets of benign attributions and attacked attributions for each pairing of an attack and attribution analysis method, e.g. IDG attributions of benign and patch attacked images. If 100 images were chosen this would result in 200 attributions (half benign, half attacked) for each dataset.

As the attribution appearance is greatly decided by the visualization method, a standard method which presents highly visible features was chosen. As seen so far in the paper, attributions are visualized with the Captum [22] visualization library. The training data are generated by this library using the default blue heat map and the absolute value of the attribution so that all attribution data is presented equivalently. Once all attributions are generated, we train the classifiers.

C. Training

We train one classifier for each of the nine datasets using a modification of the PyTorch ImageNet training example [23]. The classifier is a modified ResNet101 classifier without pre-trained weights. For each dataset, the mean and standard deviation is measured before training as the attributions do not share the same distribution as the original ImageNet data. Lastly, the traditional classification accuracy measure is modified to the binary classifier accuracy measure in Eq.(3).

We train for 50 epochs with a $60/40$ train/validation split. The batch size is set to 1024 and shared across four GPUs. We use cross entropy loss and a stochastic gradient descent optimizer with an initial learning rate of 0.1, momentum of 0.9, and weight decay of $1e-4$. The epoch with the highest validation accuracy is saved.

In Figure 3 we show an overview of the data generation and training process as well as the inference process. In (a)

TABLE I
CLASSIFICATION ACCURACY FOR PROPOSED DETECTORS

	Pixel Attack		Patch Attack
Attribution Method	***PGD L_∞***	***PGD L_2***	***3% LaVAN***
IG	93.52	93.44	69.50
GBP	98.80	95.27	**87.98**
IDG	**99.13**	**97.98**	72.04

TABLE II
F1 SCORES FOR PROPOSED DETECTORS

	Pixel Attack		Patch Attack
Attribution Method	***PGD L_∞***	***PGD L_2***	***3% LaVAN***
IG	0.935	0.933	0.674
GBP	0.989	0.953	0.874
IDG	0.991	0.980	0.703

we show the generation of the training data IDG attributions from benign and PGD L_∞ attacked images. The attributions are passed to a modified binary CNN classifier which learns the distribution of the two classes of attributions. In (b) we see the inference process where an attribution is generated for an image and is given to the trained binary classifier which will predict if the original input image is benign or attacked. This process takes less than one second on modern hardware.

V. EVALUATION

In this section we present the evaluation performed for the proposed attack detection method. We first present the experimental setup for the hardware, libraries, and evaluation methods used. Next, we present details on the results of all the trained proposed attack detection models and select the best models for comparison. Finally, we compare the best proposed detectors against existing methods in the field.

A. Hardware and Libraries

We performed all training and quantitative evaluation in PyTorch [23] on a server with four NVIDIA A40 GPUs. We use libraries for the PGD attacks [24], patch attack [25], and GBP attribution analysis method [22]. The IG and IDG methods use personal implementations of the code.

B. Quantitative Evaluation

We use binary classifier accuracy for evaluation of the proposed and comparison methods. This accuracy measure classifies predictions as true positive (TP), false positive (FP), true negative (TN), and false negative (FN) as follows:

$$Accuracy = \frac{TP + TN}{TP + FP + TN + FN} \tag{3}$$

We additionally use the F1 score to measure how reliable the training process for the proposed methods is. A low F1 score indicates the reported accuracy is due to dataset imbalances, whereas a high score indicates each class has equivalent accuracy. The F1 score are defined as follows:

$$F1 = \frac{TP}{TP + \frac{1}{2}(FP + FN)} \tag{4}$$

C. Training Results

We trained the nine potential detection models of the proposed framework on a large dataset to determine the highest performing models. From 11000 input images, we trained the nine models on a 13200 and 8800 train and validation split as explained by the details in Section IV-C. We evaluate the validation accuracy and F1 scores in this section.

In Table I, we present the binary classification accuracy as calculated from Eq.(3). The best detection accuracy for each of the three attacks is in bold. For the pixel attacks, IDG provides a clear lead over GBP and IG in detection accuracy as expected from Section III. GBP also performs well, but not to the level of IDG. Surprisingly, IG manages a high pixel attack detection accuracy which may be explained by the existence of patterns only seen by a classifier. For the patch attack detectors, the results also meet expectations. GBP performs the best as expected from the analysis in Section III. For IG and IDG, accuracy is well above random prediction, but does not meet GBP as expected from the analysis of Figure 2.

In Table II, we report the F1 scores for the nine trained classifiers. We analyze scores in comparison to the accuracy scores of Table I to determine the reliability of the reported accuracy scores. Since we use a class-balanced dataset and the accuracy and F1 scores are nearly equivalent, we confirm the model is performing well on both classes, and the accuracy reported translates to real-world usage.

D. Comparison With Previous Work

In this section we compare the best-performing proposed methods of IDG pixel attack detection and GBP patch attack detection against competitors on ImageNet data.

a) Patch Attack Detection: In Table III, we compare the classification accuracy of the proposed GBP patch attack detection model to the accuracy of the PatchGuard++ [26] and ScaleCert [20] detection frameworks. The PatchGuard++ and ScaleCert evaluation numbers are taken from the ScaleCert [20] paper, as there is no public code available for ScaleCert. To compare, we train the GBP patch detection model on patch sizes of 1, 2, and 3% on a set of 1320 attributions and validated on 880 (equal parts benign and attacked). There we no details on the dataset size of the ScaleCert paper results. Evaluating the proposed model against PatchGuard++ and ScaleCert, shows a large increase in detection accuracy for all patch sizes. With up to a 30% detection accuracy improvement, the GBP method clearly outperforms the previous work.

b) Pixel Attack Detection: In Table III, we compare the classification accuracy of the proposed IDG pixel attack detection model to the accuracy of the ML-LOO [13] attribution-based attack detection framework. Both detection models are trained on the same train and validation split as the patch comparison (1320/880) for each of the PGD L_∞ and L_2 pixel attacks. The ML-LOO results were measured from our personal implementation of their detection framework. In the comparison, ML-LOO receives a near-random accuracy of roughly 50% in each of the tests, while the proposed IDG

TABLE III
IMAGENET PATCH AND PIXEL ATTACK DETECTION ACCURACY

	Patch Attack			Pixel Attack	
Detector	***1% LaVAN***	***2% LaVAN***	***3% LaVAN***	***PGD L_∞***	***PGD L_2***
PatchGuard++ [20]	36.30	33.90	31.10	-	-
ScaleCert [20]	60.40	55.40	52.80	-	-
ML-LOO [13]	-	-	-	49.86	49.57
Proposed	**80.57**	**80.93**	**83.56**	**95.26**	**96.06**

detector manages over a 95% detection accuracy on both attack variations. The larger accuracy reported in the ML-LOO paper [13] may be attributed to using the less complex CIFAR-10.

We recognize this dataset is relatively small and more extensive training and evaluation on larger datasets would be needed to thoroughly validate performance. In particular, we do not measure the detectors resilience against adaptive attacks. In the future, adversarial training or other techniques may be needed to harden the detectors. Additionally, conducting ablation studies to validate classifier architecture choice could reveal significant opportunities for further improvement.

VI. CONCLUSION

Pixel adversarial attacks and adversarial patches are attacks on DNNs that have shown to be effective at causing model confusion in many applications. We have shown that these attacks are easily detected through the analysis of the heat maps produced by different attribution methods. We trained binary classifiers to take attribution maps as inputs and determine whether the image from which the attribution was generated has been attacked. These models achieved up to 99% accuracy on pixel adversarial attacks and 88% accuracy on adversarial patches. We found that analysis of guided backpropagation attributions provides strong patch detection, and analysis of integrated decision gradients attributions provides strong pixel attack detection detection.

Recently proposed defenses do not determine whether an image has been attacked before applying a defense [16], [25]. This results in adverse effects to benign images and increased computational overhead. We plan to apply our detector to existing defenses in order to avoid these adverse effects, improve computational efficiency, and increase recovery accuracy. Additionally, as this paper focused on image classification, we are working to extend the approach to other domains such as object detection and semantic segmentation.

REFERENCES

[1] C. Szegedy *et al.*, "Intriguing properties of neural networks," 2014.

[2] A. Madry, A. Makelov, L. Schmidt, D. Tsipras, and A. Vladu, "Towards deep learning models resistant to adversarial attacks," 2019.

[3] D. Karmon, D. Zoran, and Y. Goldberg, "Lavan: Localized and visible adversarial noise," 2018.

[4] X. Li and S. Ji, "Generative dynamic patch attack," 2021.

[5] P. Saadatpanah, A. Shafahi, and T. Goldstein, "Adversarial attacks on copyright detection systems," 2019.

[6] K. Xu *et al.*, "Adversarial t-shirt! evading person detectors in a physical world," in *Computer Vision – ECCV 2020*, A. Vedaldi, H. Bischof, T. Brox, and J.-M. Frahm, Eds. Cham: Springer International Publishing, 2020, pp. 665–681.

[7] K. Simonyan, A. Vedaldi, and A. Zisserman, "Deep inside convolutional networks: Visualising image classification models and saliency maps," 2014.

[8] M. Sundararajan, A. Taly, and Q. Yan, "Axiomatic attribution for deep networks," in *Proceedings of the 34th International Conference on Machine Learning - Volume 70*, ser. ICML'17. JMLR.org, 2017, p. 3319–3328.

[9] C. Walker, S. Jha, K. Chen, and R. Ewetz, "Integrated decision gradients: Compute your attributions where the model makes its decision," *arXiv preprint arXiv:2305.20052v1*, 2023.

[10] J. T. Springenberg, A. Dosovitskiy, T. Brox, and M. Riedmiller, "Striving for simplicity: The all convolutional net," *arXiv preprint arXiv:1412.6806*, 2014.

[11] M. T. Ribeiro, S. Singh, and C. Guestrin, ""why should i trust you?" explaining the predictions of any classifier," in *Proceedings of the 22nd ACM SIGKDD international conference on knowledge discovery and data mining*, 2016, pp. 1135–1144.

[12] M. Ancona, E. Ceolini, C. Öztireli, and M. Gross, "Towards better understanding of gradient-based attribution methods for deep neural networks," *arXiv preprint arXiv:1711.06104*, 2017.

[13] P. Yang, J. Chen, C.-J. Hsieh, J.-L. Wang, and M. Jordan, "Ml-loo: Detecting adversarial examples with feature attribution," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 34, no. 04, 2020, pp. 6639–6647.

[14] S. Jha *et al.*, "Attribution-driven causal analysis for detection of adversarial examples," *arXiv preprint arXiv:1903.05821*, 2019.

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

[16] J. Liu, A. Levine, C. P. Lau, R. Chellappa, and S. Feizi, "Segment and complete: Defending object detectors against adversarial patch attacks with robust patch detection," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022, pp. 14 973–14 982.

[17] O. Russakovsky *et al.*, "ImageNet Large Scale Visual Recognition Challenge," *International Journal of Computer Vision (IJCV)*, vol. 115, no. 3, pp. 211–252, 2015.

[18] J. H. Metzen, T. Genewein, V. Fischer, and B. Bischoff, "On detecting adversarial perturbations," *arXiv preprint arXiv:1702.04267*, 2017.

[19] P. Harder, F.-J. Pfreundt, M. Keuper, and J. Keuper, "Spectraldefense: Detecting adversarial attacks on cnns in the fourier domain," 2021.

[20] H. Han *et al.*, "Scalecert: Scalable certified defense against adversarial patches with sparse superficial layers," in *Advances in Neural Information Processing Systems*, A. Beygelzimer, Y. Dauphin, P. Liang, and J. W. Vaughan, Eds., 2021.

[21] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 770–778.

[22] N. Kokhlikyan *et al.*, "Captum: A unified and generic model interpretability library for pytorch," *arXiv preprint arXiv:2009.07896*, 2020.

[23] A. Paszke *et al.*, "Pytorch: An imperative style, high-performance deep learning library," in *Proceedings of the 33rd International Conference on Neural Information Processing Systems*. Curran Associates Inc., 2019.

[24] H. Kim, "Torchattacks: A pytorch repository for adversarial attacks," *arXiv preprint arXiv:2010.01950*, 2020.

[25] C. Xiang, A. N. Bhagoji, V. Sehwag, and P. Mittal, "Patchguard: A provably robust defense against adversarial patches via small receptive fields and masking," in *30th USENIX Security Symposium (USENIX Security)*, 2021.

[26] C. Xiang and P. Mittal, "Patchguard++: Efficient provable attack detection against adversarial patches," *arXiv preprint arXiv:2104.12609*, 2021.