SAD: SALIENCY ADVERSARIAL DEFENSE WITHOUT ADVERSARIAL TRAINING

Anonymous authors
Paper under double-blind review

ABSTRACT

Adversarial training is one of the most effective methods for defending adversarial attacks, but it is computationally costly. In this paper, we propose Saliency Adversarial Defense (SAD), an efficient defense algorithm that avoids adversarial training. The saliency map is added to the input with a hybridization ratio to enhance those pixels that are important for making decisions. This process causes a distribution shift to the original data. Interestingly, we find that this shift can be effectively fixed by updating the statistics of batch normalization with the processed data without further training. We justify the algorithm with a linear model that the added saliency maps pull data away from its closest decision boundary. Updating BN effectively evolves the decision boundary to fit the new data. As a result, the distance between the decision boundary and the original inputs are increased such that the model is able to defend stronger attacks and thus improve robustness. Then we show in experiments that the results still hold for complex models and datasets. Our results demonstrate that SAD is superior in defending various attacks, including both white-box and black-box ones.

1 INTRODUCTION

Learning a model that is accurate on natural data and robust on adversarially perturbed data is crucial in safety and security-critical applications. Adversarial robust accuracy, defined as the ratio of adversarial examples correctly recognized by models, is commonly used to measure model robustness. Although it is easy to calculate, it shows no insight into adversarial examples. Recently, Yin et al. (2019) proposed Rademacher complexity and Bastani et al. (2016), Etmann et al. (2019) and Jordan et al. (2019) proposed average distance of data to its closest decision boundaries as alternates. The latter inspires us to understand adversarial training from a distance between data and decision boundaries perspective. We shall name this distance as *robust radius*.

Naturally trained models have little robustness accuracy on adversarial samples, implying that each sample is close to a decision boundary (Tanay & Griffin, 2016). Tanay & Griffin (2016) has proposed a boundary tilting hypothesis, which is related to the explanation given by Szegedy et al. (2014), to explain this phenomenon. The hypothesis states that though the learned decision boundaries well separate the training data, it is "tilted" from the training manifold. When the tilted angle is small, every sample lies close to the decision boundary, and adversarial examples can be generated by slightly lifting samples out of the manifold they live in to across the decision boundary easily. Such results show that it is difficult to obtain a robust model with natural training if there is a direction (or dimension) where the data is separable but with low variance. Adversarial training pushing the decision boundaries away from the original data by augmenting those dimensions as we shown in our method.

The saliency map, defined as the gradient of logit with respective to input, is proposed to explain which pixels are important for deep neural networks to make the decision (Simonyan et al., 2014; Ribeiro et al., 2016). Recent studies have shown that models with adversarial training has more understandable saliency map for human than their naturally trained counterparts Etmann et al. (2019). We show that the saliency maps care only which information are important for the prediction of input, while adversarial attacks aggregate information from all classes simultaneously to fool the models. An example is shown in Fig. 1. The saliency map of the original image x_0, classified as a dog, focuses on the dog while that of x_0's adversarial counterpart, x_0', attends on both cat and dog.

We process x_0 and x_0' by adding their corresponding saliency maps with a ratio. The resulting images' saliency map only attends on the model's predicted class. That is, saliency map of processed x_0 attends on dog only while that of processed x_0' attends on cat. This procedure is animated in the middle-top of Fig. 1. One can observe that data are pulled away from the decision boundaries, no matter for original images or their adversarial counterparts. With the robust radius perspective discussed, we believe that processing data with saliency maps can improve robustness.

However, the processing with saliency maps shifts the original data distribution, resulting in models failing to recognize the processed data. It is straightforward to finetune or retrain the model but it is more computational costly. Li et al. (2016) proposed AdaBN, which updates BN statistics with the distribution shifted data to adapt existing models to that new distribution. We find that it is sufficient to update BN's statistics with the processed training data to adapt the decision boundaries to fit the process data. This simple adaptation make the algorithm much more efficient.

In summary, we propose an efficient algorithm SAD, which achieves adversarial robustness by preprocessing inputs with their saliency map, called Saliency Adversarial Defense (SAD), without adversarial training. Experimentally, we demonstrate that this simple and efficient solution outperforms state-of-the-art methods in both adversarial robustness and vanilla accuracy.

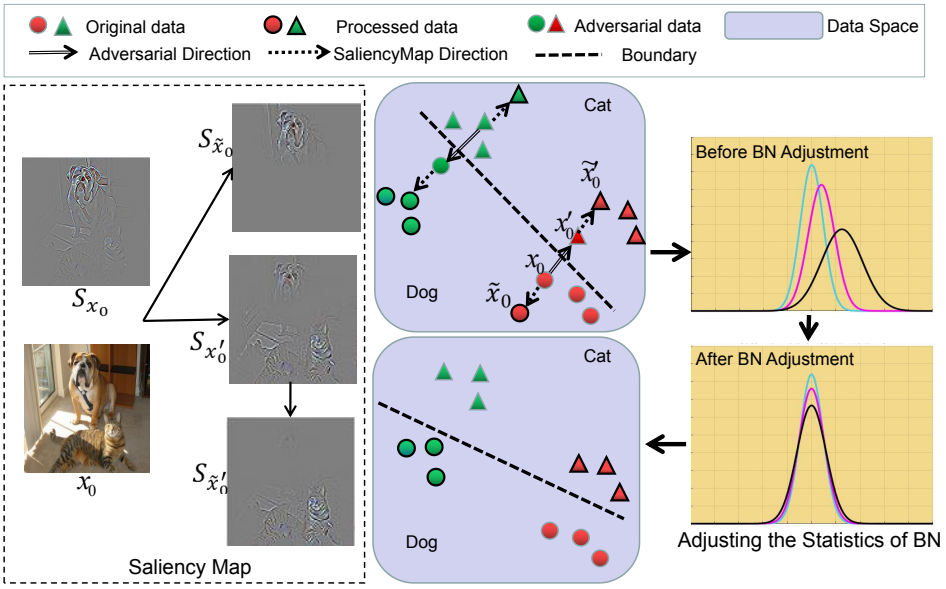

Figure 1: Left: Saliency maps for different version of original image, x_0. Its adversarial example denotes as x_0'. $\tilde{x}$ are image after pressing with saliency map. S_x denotes saliency map of image x. Right: An overview of SAD. Colors and shapes represent the true label and prediction respectively. x_0 (red circle) is attacked to classified as cat (red triangle). Processed images are pulled away from the decision boundary. Statistics of BN are updated with the processed images, resulting a new decision boundary to correctly classify the processed images.

The main contributions of our paper are summarized below:

- We propose a defense algorithm SAD based on input transformations that achieves better defense performance than adversarial training under various attacks, including white-box and black-box.
- SAD is much more efficient than adversarial training. More interestingly, we demonstrate that SAD can further improve the robustness of adversarial training.
- We show that the SAD model has surprisingly understandable saliency maps, providing evidence that better interpretability is not just a side-effect of adversarial training but a shared property of robust models Etmann et al. (2019).

2 RELATED WORK

2.1 DEFENSES BASED ON ADVERSARIAL TRAINING

Adversarial attacks can make the model give wrong prediction by adding negligible perturbations for humans to input data (Szegedy et al., 2013). Adversarial training first proposed in (Goodfellow et al., 2015) can effectively defense such attacks by training on adversarial examples. Madry et al. (2017a) formulates adversarial training as a bi-level min-max optimization problem and trains models exclusively on adversarial images rather than both clean and adversarial images. Although it effectively improves the adversarial robustness, expensive computational cost, and performance degradation on clean images are the two fatal shortcomings of it. Many works try to reduce the computation cost to the natural training of a model (Shafahi et al., 2019; Wong et al., 2020; Zhang et al., 2019b;a). These works still need adversarial training, while our methods can obtain robustness throught adjusting the decision boundary without adversarial training.

2.2 DEFENSES BASED ON INPUT TRANSFORMATIONS

The input transformation based methods attempt to reduce the amount of perturbation that may exist in the sample to be predicted by various transformation methods, and then directly input the converted sample into the original model for prediction (Guo et al., 2017; Xie et al., 2017; Song et al., 2017; Buckman et al., 2018). The advantage of input conversion defense is that the models remained unchanged and adversarial training is not necessary. Li et al. (2020b) encourages a larger gradient component in the tangent space of data manifold, suppressing the gradient leaking phenomenon, which is similar to a data dimension reduction approach. However, experiments shown that the existing defense methods are not robust enough compared to adversarial training. Our method partially belong to this class as we add saliency map to the original data. But our algorithm surpass the performance of adversarial training.

2.3 EXPLANATION METHODS

The core idea of the interpretation methods based on backpropagation is to propagate significant signals that influence decisions from the output layer to the input layer by layer to deduce the important pixel of the input sample (Simonyan et al., 2014; Springenberg et al., 2015; Zhou et al., 2015; Selvaraju et al., 2016; Chattopadhyay et al., 2017). Tsipras et al. (2019) shows that saliency maps of robustified classifiers tend to be well interpretable and describe this as an "unexpected benefit" of adversarial robustness. Saliency maps of robustified classifiers tend to be far more interpretable, for that structures in the input image also emerge in the corresponding saliency maps (Etmann et al., 2019). Mangla et al. (2020) shows that saliency maps can be used as adversarial perturbation in adversarial training. Our method shows more interpretable saliency maps as shown in Fig. 3.

3 METHOD

3.1 MOTIVATION OF SAD

Adversarial training is the most effective way to get adversarial robustness. Nevertheless the understanding of adversarial training is limited. We give a closer look at what adversarial training does with binary linear classification and how it motivates our method. We consider the adversarial optimization problem for a linear classifier:

$$\min_{\mathbf{w}} \mathbb{E}_{(\mathbf{x},y)\sim\mathcal{D}} \left[\max_{\delta\in\mathbb{B}_\epsilon^p} \mathcal{L}(\mathbf{w}^\top(\mathbf{x}+\delta), y) \right], \tag{1}$$

where input $(\mathbf{x}, y)$ sampled from $\mathcal{D} \subseteq \mathbb{R}^d \times \{-1, 1\}$, loss $\mathcal{L}(\mathbf{w}^\top\mathbf{x}, y) = \ell(y\mathbf{w}^\top\mathbf{x})$, $\ell : \mathbb{R} \to [0, 1]$ is monotonically nonincreasing, $\mathbb{B}_\epsilon^p = \{\delta \in \mathbb{R}^d : \|\delta\|_p \leq \epsilon\}$ is the ℓ_p ball of radius ϵ. If consider the linear binary classification problem of adversarial training, we get the close form perturbation as in following lemma (Moosavi-Dezfooli et al., 2016; Chen et al., 2020; Dobriban et al., 2020; Yin et al., 2019; Awasthi et al., 2020):

Lemma 1 *If* $\delta^*_{\epsilon,p} = \arg\max_{\delta\in\mathbb{B}^p_\epsilon} \ell(y\mathbf{w}^\top\mathbf{x})$, *then* $\delta^*_{\epsilon,p} = -\epsilon y \frac{|\mathbf{w}|^{p^*-1}}{\|\mathbf{w}\|^{p^*-1}_{p^*}} \odot sgn(\mathbf{w})$. *In particular,* $\delta^*_{\epsilon,\infty} = -\epsilon y sgn(\mathbf{w})$.

This lemma shows that the adversarial example of ℓ_∞ attack move towards the linear decision boundary in the direction $-ysgn(\mathbf{w})$ for all the input data, see Fig. 2(a). Also, adversarial training pushes the linear decision boundary from l_0 to l_1 to get good performance on the attacked data. In this way, original data become much further from the new linear classifier so we can get adversarial robustness up to magnitude ϵ. Adversarial training is very expensive and shows the trade-off between accuracy and robustness. There are two stages in adversarial training, first to get new adversarial examples towards the decision boundary, then adjust the decision boundary to fit the adversarial examples. The main benefit of adversarial training is to enlarge the robust radius in light of the binary linear classification. This motivates our method, which moves the data away from the decision boundary.

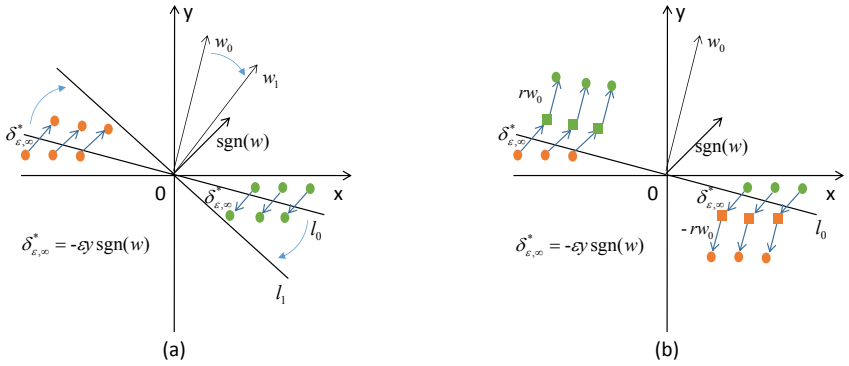

Figure 2: (a) Adversarial training for linear classifier. (b) SAD defense for linear classifier. $\delta^*_{\epsilon,p}$ indicates the adversarial direction. x_0 indicates the saliency map direction. *r* indicates the strength of the added saliency map.

3.2 Saliency map and adversarial attack

To enlarge the distance of original data from the decision boundary, we first generate adversarial examples and then push them further by their saliency maps. Consider a trained network $\Phi = (\Phi_1, ..., \Phi_K) : \mathbb{R}^D \to \mathbb{R}^K$, the predicted label of a sample $(\mathbf{x}, y_j)$ is defined as $i^* = \arg\max_i \Phi_i(\mathbf{x})$. Define the probability by softmax as $p_k = \frac{\exp(\Phi_k)}{\sum_{i=1}^K \exp(\Phi_i)}$, and the cross-entropy loss of $(\mathbf{x}, y_j)$ is $\ell(\mathbf{x}, y_j)) = -\log p_j = -\Phi_j(\mathbf{x}) + \log(\sum_{i=1}^K \exp(\Phi_i(\mathbf{x}))$. The *saliency map* is

$$\mathcal{S}(\mathbf{x}) = \frac{\partial \Phi_{i^*}}{\partial \mathbf{x}} \tag{2}$$

and the attack direction is $\delta(\mathbf{x}, y) = \frac{\partial \ell(\mathbf{x},y)}{\partial \mathbf{x}}$. More precisely,

$$\delta(\mathbf{x}, y_j) = -\frac{\partial \Phi_j}{\partial \mathbf{x}} + \sum_{i=1}^K p_i \frac{\partial \Phi_i}{\partial \mathbf{x}} = -(1-p_j)\frac{\partial \Phi_j}{\partial \mathbf{x}} + \sum_{i\neq j} p_i \frac{\partial \Phi_i}{\partial \mathbf{x}} \tag{3}$$

This implies that adversarial attacks reduce the information of the right class and induce information of other classes to fool the model, see Fig. 1. Saliency map, on the contrary, collection important pixels for a particular class only. Adding the saliency map to the data enhance those important pixels. After adding their saliency map on adversarial examples, we adjust the decision boundary to fit the new data distribution. In this way, the decision boundary will be far away from the original data, which implies adversarial robustness. In the linear binary classification setting above, the saliency map is $y\mathbf{w}_0$, which moves data away from the original decision boundary, as shown in Fig. 2(b). For general classification problems, we use the saliency map as the direction moving away from the original decision boundary.

Algorithm 1 Saliency map defense

Model Update:
Input: Training set $\mathcal{D}_{train}$, naturally trained model $\Phi^0 = \Phi^0(\mathbf{W}, \{\gamma_l, \beta_l, \mu_l^0, \sigma_l^0\}_{l=1}^L)$ with BN parameters $\{\gamma_l, \beta_l\}_{l=1}^L$, and BN statistics $\{\mu_l^0, \sigma_l^0\}_{l=1}^L$.
for each batch $\mathcal{B}$ in $\mathcal{D}_{train}$ **do**
compute saliency map $\mathcal{S}^0$ and adversarial perturbation δ^0 under Φ^0 by Eq. (2) and (3);
compute $\mathcal{D}_{adv}^{\mathcal{B}} = \{\mathbf{x} + \epsilon\delta^0(\mathbf{x}, y) : (\mathbf{x}, y) \in \mathcal{B}\}$ and $\mathcal{D}_{sadv}^{\mathcal{B}} = \{\mathbf{x} + \eta\mathcal{S}^0(\mathbf{x}) : (\mathbf{x}, y) \in \mathcal{D}_{adv}^{\mathcal{B}}\}$;
update only BN statistics by passing $\mathcal{D}_{sadv}^{\mathcal{B}}$ through network with training mode;
end for
Output: SAD model $\Phi^1 = \Phi^1(\mathbf{W}, \{\gamma_l, \beta_l, \mu_l^1, \sigma_l^1\}_{l=1}^L)$ with new BN statistics $\{\mu_l^1, \sigma_l^1\}_{l=1}^L$.

Model inference:
for any $\hat{\mathbf{x}}$ in $\mathcal{D}_{test}$ and adversarial data generated from $\mathcal{D}_{test}$ **do**
compute $\mathcal{S}^1(\hat{\mathbf{x}}) = \frac{\partial \Phi^1_{max}}{\partial \mathbf{x}}$, $\hat{y} = \arg\max_j \Phi_j^1(\hat{\mathbf{x}} + \mathcal{S}^1(\hat{\mathbf{x}}))$.
end for

3.3 Batch Normalization helps to push the decision boundary

For more complicated datasets, we will use ResNet which contains BN layers, and many works have found that BN layers are sensitive to distribution shift (Li et al. (2016); Xie et al. (2019); Li et al. (2020a)). The data distribution will change after we move adversarial data away from the decision boundary by adding saliency maps of the naturally trained model. Therefore we update the statistics of BN layers to obtain suitable model for processed data. This makes the decision boundary move away from the original data. AdaBN (Li et al. (2016)) modifies the statistics of BN layers by data from the new domain in order to get better generalization in that domain. This update can be simply implemented with Pytorch (Paszke et al., 2019) by feeding the data forward to the network under training mode. In the inference stage, we first add the saliency map to clean or adversarial data, then use the updated model to test. Our algorithm is outlined in Algorithm 1.

4 Experimental Results

4.1 Implementation

To demonstrate SAD's effectiveness, we chose three typical white-box attacks and three black-box attacks and run experiments on CIFAR-10 and CIFAR-100 (Krizhevsky et al., 2009). To compare with the state-of-the-art methods, we train TRADES (1/λ=6) (Zhang et al., 2019b), Fast (Wong et al., 2020) and Free (m=8) (Shafahi et al., 2019). These results are produced with their open-source code. The hybridization ratio of SAD is set to *ratio*=0.8. Besides, we combine adversarial training (AT) and SAD to show that the SAD improves the performance of AT. TRADES is used in the experiment for AT.

4.2 White-Box Attacks

Project Gradient Descent (PGD) (Madry et al., 2017b) is a typical first-order attack (directly using gradient attack). Defending against PGD also means that it can resist other first-order attacks. We define WeakPGD: k=20, ϵ=4/255; StrongPGD: k=20, ϵ=8/255. C&W attack (Carlini & Wagner, 2016) is generally considered one of the most powerful white-box attack algorithms and is an optimization-based adversarial sample generation algorithm, which allows it to attack many defenses based on gradient obfuscation successfully. We set the hyperparameter c of C&W to 0.2. APGD (Liu et al., 2018) is a combination of Expectation over Transformation (EoT) (Athalye et al., 2018) and PGD. Physical confrontation samples produced based on EoT can effectively attack various applications of image recognition or object detection. We set the APGD attack strength as k=12, ϵ=12/255, sampling=32.

We apply the SAD algorithm on various CIFAR-10 models, includes ResNet-18 and WRN-32-10 (WiderResNet), and compare the results with other methods. As shown in Table 1, SAD outperforms Fast and Free in all attacks while it is comparable to TRADES. In particular, with the WRN-

Table 1: Results of CIFAR-10 compared to other methods. Results of TRADES, Fast and Free are produced with their open source code. Scores of (AT+) SAD that are better than other methods are highlighted.

Model	Method	Nat. Images	WeakPGD	StrongPGD	C&W	APGD
ResNet-18	TRADES	82.77%	69.46%	52.14%	78%	37.66%
	Fast	83.81%	61.1%	46.06%	72.86%	34.82%
	Free	85.96%	62.73%	46.33%	71.74%	35.12%
	SAD (ours)	83.86%	65.33%	**52.61**%	77.2%	**38.88**%
	AT+SAD (ours)	82.73%	**71.83**%	**53.65**%	**79.04**%	**39.11**%
WRN-32-10	TRADES	85.55%	70.24%	52.00%	77.04%	37.57%
	Fast	86.02%	66.64%	48.19%	73.16%	35.42%
	Free	85.96%	65.5%	46.13%	72.61%	37.22%
	SAD (ours)	**86.96**%	69.27%	**53.47**%	**79.97**%	**38.73**%
	AT+SAD (ours)	84.61%	**73.33**%	**54.81**%	**79.2**%	**39.68**%

Table 2: Results of CIFAR-100 compared to baseline methods. Scores of (AT+) SAD that are better than other methods are highlighted.

Model	Method	Nat. Images	WeakPGD	StrongPGD	C&W	APGD
ResNet-50	TRADES	56.96%	41.95%	29.15%	48.25%	19.84%
	Fast	62.43%	30.05%	20.08%	41.19%	16.77%
	Free	62.36%	30.14%	21.42%	41.22%	17.14%
	SAD (ours)	56.27%	**42.91**%	**33.15**%	**52.05**%	**19.88**%
	AT+SAD (ours)	56.37%	**44.82**%	**35.21**%	**54.01**%	**20.34**%
WRN-32-10	TRADES	55.5%	40.05%	26.54%	42.39%	17.46%
	Fast	59.94%	36.91%	22.67%	37.2%	17.23%
	Free	65.28%	35.24%	20.64%	36.15%	17.1%
	SAD (ours)	63.86%	39.65%	**28.55**%	37.76%	**29.88**%
	AT+SAD (ours)	57.81%	**41.7**%	**30.69**%	**42.66**%	**21.06**%

32-10 model, SAD outperforms all other models in either natural accuracy or adversarial accuracy except TRADES on WeakPGD. These results demonstrate that SAD not only improves the robustness against adversarial attack but also potentially improves accuracy on the clean image. When combined with AT, in most of the cases, AT+SAD performs the best in defending various attacks. To demonstrate our method can generalize to more complicated datasets, we run experiments on CIFAR-100 with WRN-32-10 and ResNet-50. As shown in Table 2, SAD shows superior performance compared with other methods. With ResNet-50, SAD, and AT+SAD models outperform all other methods on all attacks. Especially, SAD algorithms show superior performance on defending StrongPGD and C&W attack, exceeding TRADES for more than 4%. Besides, SAD has far better performance in defending against APGD attacks on WRN-32-10 with a 12% improvement. When defending against C&W attacks with the WRN-32-10 model, SAD is slightly inferior to TRADES but with higher clean accuracy. To this end, we demonstrate that SAD show excellent performance in various models (ResNet-18, ResNet50, WRN-32-10) and datasets (CIFAR-10, CIFAR-100).

4.3 Black-Box Attacks

In practice, attackers often cannot obtain detailed information about the model. Therefore, black-box attacks are more common. We choose the One Pixel Attack based on differential evolution (Su et al., 2019), ZOO (Chen et al., 2017), and adversarial samples generated from the WideResNet-32-10 model for the black box defense test. To compare with the state-of-the-art method, we show the comparison between TRADES and SAD in Table 3. SAD also has satisfactory performance in the black-box defense. The performance of using the SAD algorithm in model ResNet-50 (CIFAR100) is worse than that of TRADES, but there is no need to be nervous. The effort of SAD is related to the hyperparameter ratio and the performance of the used natural model. We can adjust the two factors to optimize the result. Here we also show the performance of an adversarially trained model with SAD (AT+SAD). The impact of *ratio* on the algorithm will be showed in subsequent experiments.

Table 3: Results of Black-Box Attacks. OnePixel means one pixel attack, ZOO means ZOO attack, and WRN means the adversarial samples are generated by WideResNet-32-10.

Model	Method	Nat. Images	OnePixel	ZOO	WRN
ResNet-18, CIFAR10	TRADES	82.77%	76%	82.22%	82.07%
	SAD	**83.86**%	75.75%	**82.5**%	83.14%
	AT+SAD	**82.73**%	**79.33**%	**82.65**%	**82.11**%
ResNet-50, CIFAR100	TRADES	56.96%	47.63%	55.94%	56.65%
	SAD	56.27%	46.41%	55.72%	56.19%
	AT+SAD	56.37%	**47.89**%	**56.08**%	56.23%

4.4 SALIENCY MAP OF SAD

Recent studies (Zhang & Zhu, 2019; Tsipras et al., 2019) have shown that representations learned by adversarial trained convolutional neural networks (ATCNNs) tend to evince more interpretable saliency maps corresponding to their prediction than their non-robust equivalents. Etmann et al. (2019) proves that as robust radius grows, so does the alignment between the input image and the saliency map. Our results are consistent with their conclusion. As shown in Fig. 3, normally trained model exhibits unstructured saliency map, which is difficult to recognize by the human. Surprisingly, with SAD, the saliency map of the normally trained model shows clearly recognizable shape. Meanwhile, there is a strong correlation between the saliency map and the input structure of the robust model obtained by the adversarial training. After the adjustment of SAD, the adversarially trained model can generate clearer and smoother saliency maps. The last row is the processed data we used to infer, which is formed by overlaying the saliency map on the natural data. Surprisingly, we see that the processed images become clearer to human, implying that the relevant features are enhanced.

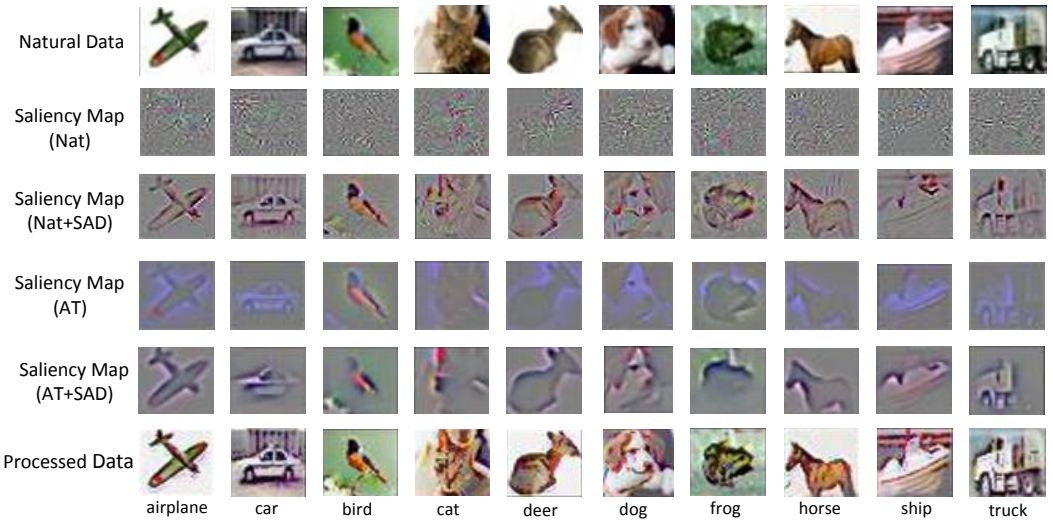

Figure 3: Saliency maps of samples from CIFAR-10. The first row: the natural input images. The second row: the saliency map of the naturally trained model. The third row: the saliency map of the naturally trained model with SAD. The fourth row: the saliency map of the adversarially trained model. The fifth row: the saliency map of the adversarially trained model with SAD. Last row: the processed data, which consists of original data and saliency maps of SAD. Overlaying the saliency map with the original image makes the area of focus more prominent.

4.5 TRADE-OFF BY VARYING HYBRIDIZATION RATIO

Tsipras et al. (2019) shows that the goal of adversarial robustness might be incompatible with that of standard generalization, which means there is a trade-off between robustness and generalization accuracy. The SAD algorithm can make a good trade-off between adversarial robustness and vanilla

Table 4: Effect of hybridization ratio on CIFAR-100 WRN-32-10.

Accuracy \ Ratio / Data	0.9	0.8	0.7	0.6	0.5
Nat.Image	60.36%	62.29%	63.86%	64.79%	65.08%
StrongPGD	30.01%	29.43%	28.55%	27.2%	26.39%

accuracy by adjusting the ratio of saliency map overlaid on input data, as shown in Table 4. We see that as the ratio decreases, the accuracy on clean images increases, and the adversarial robustness decreases. This means that the SAD algorithm can quickly make a trade-off between adversarial robustness and vanilla accuracy, rather than spending a large computation to retrain the model as adversarial training does.

4.6 Ablation Studies

The SAD algorithm involves modifying the statistics of the batch normalization layers of the original model to get the SAD model and adding saliency maps to the testing data. In this section, we conduct ablation experiments on both of these factors.

Firstly, we use a normally trained model to classify the data overlaid with the saliency map directly without updating the statistics of BN (see Table 5). With the increase of ratio, the robust accuracy will be improved slightly, but the natural accuracy decays quickly by contrast. Even with a large ratio, robust accuracy is much lower than adversarial training. This proves that updating the statistics of BN brings great benefits as the statistics are sensitive to the distribution of data.

Table 5: Result of SAD without updating statistics of BN on CIFAR10 with ResNet-18.

Accuracy \ Ratio / Data	0.2	0.5	0.7	1.0	1.2
Nat.Image	94.66%	93.36%	87.84%	81.12%	68.1%
StrongPGD	0.74%	4.88%	6.23%	9.11%	11.16%

Secondly, we modify the batch normalization layers' statistics using the adversarial samples without overlaying saliency maps. The result shows that the accuracy of the clean samples decreases, while the robustness is too small to make sense, see Appendix 1.1 for detailed results. The ablation experiments verify the importance of saliency maps and update of BN statistics.

5 Conclusions and Future works

We have proposed an interpretable defense method called SAD, which outperforms state-of-the-art adversarial training methods against multiple attacks without adversarial training. By adjusting the strength of saliency maps overlaid to the input data, we can obtain different defense effects and strike a good balance between robustness and accuracy. Our method provides a novel view to understanding model robustness and adversarial samples, which may be illuminating for future research. We also realized that fine-tuning with processed data and learning a sample-dependent strength of saliency map are two promising ideas to further improve our results which are left for future works.

References

Anish Athalye, Logan Engstrom, Andrew Ilyas, and Kevin Kwok. Synthesizing robust adversarial examples. volume 80 of *Proceedings of Machine Learning Research*, pp. 284–293, Stockholmsmässan, Stockholm Sweden, 10–15 Jul 2018. PMLR.

Pranjal Awasthi, Natalie Frank, and Mehryar Mohri. Adversarial learning guarantees for linear hypotheses and neural networks. *ICML*, 2020.

Osbert Bastani, Yani Ioannou, Leonidas Lampropoulos, Dimitrios Vytiniotis, Aditya V. Nori, and Antonio Criminisi. Measuring neural net robustness with constraints. *CoRR*, abs/1605.07262, 2016. URL http://arxiv.org/abs/1605.07262.

Jacob Buckman, Aurko Roy, Colin Raffel, and Ian Goodfellow. Thermometer encoding: One hot way to resist adversarial examples. In *International Conference on Learning Representations*, 2018. URL https://openreview.net/forum?id=S18Su--CW.

Nicholas Carlini and David Wagner. Towards evaluating the robustness of neural networks, 2016.

Aditya Chattopadhyay, Anirban Sarkar, Prantik Howlader, and Vineeth N. Balasubramanian. Grad-cam++: Generalized gradient-based visual explanations for deep convolutional networks. *CoRR*, abs/1710.11063, 2017. URL http://arxiv.org/abs/1710.11063.

Lin Chen, Yifei Min, Mingrui Zhang, and Amin Karbasi. More data can expand the generalization gap between adversarially robust and standard models. *ICML*, 2020.

Pin-Yu Chen, Huan Zhang, Yash Sharma, Jinfeng Yi, and Cho-Jui Hsieh. Zoo. *Proceedings of the 10th ACM Workshop on Artificial Intelligence and Security - AISec '17*, 2017. doi: 10.1145/3128572.3140448. URL http://dx.doi.org/10.1145/3128572.3140448.

Edgar Dobriban, Hamed Hassani, David Hong, and Alexander Robey. Provable tradeoffs in adversarially robust classification. *arXiv preprint arXiv:2006.05161*, 2020.

Christian Etmann, Sebastian Lunz, Peter Maass, and Carola Schoenlieb. On the connection between adversarial robustness and saliency map interpretability. volume 97 of *Proceedings of Machine Learning Research*, pp. 1823–1832, Long Beach, California, USA, 09–15 Jun 2019. PMLR.

Ian J Goodfellow, Jonathon Shlens, and Christian Szegedy. Explaining and harnessing adversarial examples. *ICLR*, 2015.

Chuan Guo, Mayank Rana, Moustapha Cisse, and Laurens van der Maaten. Countering adversarial images using input transformations. *CoRR*, abs/1711.00117, 2017. URL http://arxiv.org/abs/1711.00117.

Matt Jordan, Justin Lewis, and Alexandros G Dimakis. Provable certificates for adversarial examples: Fitting a ball in the union of polytopes. In *Advances in Neural Information Processing Systems 32*, pp. 14082–14092. Curran Associates, Inc., 2019.

Alex Krizhevsky, Vinod Nair, and Geoffrey Hinton. Cifar-10 and cifar-100 datasets. *URl: https://www. cs. toronto. edu/kriz/cifar. html*, 6, 2009.

Bailin Li, B. Wu, Jiang Su, Guangrun Wang, and L. Lin. Eagleeye: Fast sub-net evaluation for efficient neural network pruning. *ArXiv*, abs/2007.02491, 2020a.

Yanghao Li, Naiyan Wang, Jianping Shi, Jiaying Liu, and Xiaodi Hou. Revisiting batch normalization for practical domain adaptation. *CoRR*, abs/1603.04779, 2016. URL http://arxiv.org/abs/1603.04779.

Yueru Li, Shuyu Cheng, Hang Su, and Jun Zhu. Defense against adversarial attacks via controlling gradient leaking on embedded manifolds. *ECCV*, 2020b.

Xuanqing Liu, Yao Li, Chongruo Wu, and Cho-Jui Hsieh. Adv-bnn: Improved adversarial defense through robust bayesian neural network, 2018.

Aleksander Madry, Aleksandar Makelov, Ludwig Schmidt, Dimitris Tsipras, and Adrian Vladu. Towards deep learning models resistant to adversarial attacks, 2017a.

Aleksander Madry, Aleksandar Makelov, Ludwig Schmidt, Dimitris Tsipras, and Adrian Vladu. Towards deep learning models resistant to adversarial attacks, 2017b.

Puneet Mangla, Vedant Singh, and Vineeth N Balasubramanian. On saliency maps and adversarial robustness, 2020.

Seyed-Mohsen Moosavi-Dezfooli, Alhussein Fawzi, and Pascal Frossard. Deepfool: a simple and accurate method to fool deep neural networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 2574–2582, 2016.

Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, Alban Desmaison, Andreas Kopf, Edward Yang, Zachary DeVito, Martin Raison, Alykhan Tejani, Sasank Chilamkurthy, Benoit Steiner, Lu Fang, Junjie Bai, and Soumith Chintala. Pytorch: An imperative style, high-performance deep learning library. In H. Wallach, H. Larochelle, A. Beygelzimer, F. dAlché-Buc, E. Fox, and R. Garnett (eds.), *Advances in Neural Information Processing Systems 32*, pp. 8024–8035. Curran Associates, Inc., 2019. URL http://papers.neurips.cc/paper/9015-pytorch-an-imperative-style-high-performance-deep-learning-library.pdf.

Marco Túlio Ribeiro, Sameer Singh, and Carlos Guestrin. "why should I trust you?": Explaining the predictions of any classifier. *CoRR*, abs/1602.04938, 2016. URL http://arxiv.org/abs/1602.04938.

Ramprasaath R. Selvaraju, Abhishek Das, Ramakrishna Vedantam, Michael Cogswell, Devi Parikh, and Dhruv Batra. Grad-cam: Why did you say that? visual explanations from deep networks via gradient-based localization. *CoRR*, abs/1610.02391, 2016. URL http://arxiv.org/abs/1610.02391.

Ali Shafahi, Mahyar Najibi, Mohammad Amin Ghiasi, Zheng Xu, John Dickerson, Christoph Studer, Larry S Davis, Gavin Taylor, and Tom Goldstein. Adversarial training for free! In *Advances in Neural Information Processing Systems*, pp. 3353–3364, 2019.

Karen Simonyan, Andrea Vedaldi, and Andrew Zisserman. Deep inside convolutional networks: Visualising image classification models and saliency maps. In *Workshop at International Conference on Learning Representations*, 2014.

Yang Song, Taesup Kim, Sebastian Nowozin, Stefano Ermon, and Nate Kushman. Pixeldefend: Leveraging generative models to understand and defend against adversarial examples. *CoRR*, abs/1710.10766, 2017.

Jost Tobias Springenberg, A. Dosovitskiy, T. Brox, and Martin A. Riedmiller. Striving for simplicity: The all convolutional net. *CoRR*, abs/1412.6806, 2015.

Jiawei Su, Danilo Vasconcellos Vargas, and Kouichi Sakurai. One pixel attack for fooling deep neural networks. *IEEE Transactions on Evolutionary Computation*, 23(5):828–841, Oct 2019. ISSN 1941-0026. doi: 10.1109/tevc.2019.2890858. URL http://dx.doi.org/10.1109/TEVC.2019.2890858.

Christian Szegedy, Wojciech Zaremba, Ilya Sutskever, Joan Bruna, Dumitru Erhan, Ian Goodfellow, and Rob Fergus. Intriguing properties of neural networks. *arXiv preprint arXiv:1312.6199*, 2013.

Christian Szegedy, Wei Liu, Yangqing Jia, Pierre Sermanet, Scott Reed, Dragomir Anguelov, Dumitru Erhan, Vincent Vanhoucke, and Andrew Rabinovich. Going deeper with convolutions, 2014.

Thomas Tanay and Lewis D. Griffin. A boundary tilting persepective on the phenomenon of adversarial examples. *CoRR*, abs/1608.07690, 2016. URL http://arxiv.org/abs/1608.07690.

Dimitris Tsipras, Shibani Santurkar, Logan Engstrom, Alexander Turner, and Aleksander Madry. Robustness may be at odds with accuracy. In *International Conference on Learning Representations*, 2019. URL https://openreview.net/forum?id=SyxAb30cY7.

Eric Wong, Leslie Rice, and J Zico Kolter. Fast is better than free: Revisiting adversarial training. *arXiv preprint arXiv:2001.03994*, 2020.

Cihang Xie, Jianyu Wang, Zhishuai Zhang, Zhou Ren, and Alan L. Yuille. Mitigating adversarial effects through randomization. *CoRR*, abs/1711.01991, 2017.

Cihang Xie, Mingxing Tan, Boqing Gong, Jiang Wang, Alan Yuille, and Quoc V. Le. Adversarial examples improve image recognition, 2019.

Dong Yin, Ramchandran Kannan, and Peter Bartlett. Rademacher complexity for adversarially robust generalization. 2019.

Dinghuai Zhang, Tianyuan Zhang, Yiping Lu, Zhanxing Zhu, and Bin Dong. You only propagate once: Accelerating adversarial training via maximal principle. In *Advances in Neural Information Processing Systems*, pp. 227–238, 2019a.

Hongyang Zhang, Yaodong Yu, Jiantao Jiao, Eric P. Xing, Laurent El Ghaoui, and Michael I. Jordan. Theoretically principled trade-off between robustness and accuracy, 2019b.

Tianyuan Zhang and Zhanxing Zhu. Interpreting adversarially trained convolutional neural networks. *CoRR*, abs/1905.09797, 2019. URL http://arxiv.org/abs/1905.09797.

Bolei Zhou, Aditya Khosla, Àgata Lapedriza, Aude Oliva, and Antonio Torralba. Learning deep features for discriminative localization. *CoRR*, abs/1512.04150, 2015. URL http://arxiv.org/abs/1512.04150.