A PENALIZED MODIFIED HUBER REGULARIZATION TO IMPROVE ADVERSARIAL ROBUSTNESS

Modeste Atsague Ashutosh Nirala Olukorede Fakorede Jin Tian*

Iowa State University,USA

ABSTRACT

Adversarial training (AT) is a learning procedure that trains a deep neural network with adversary examples to improve robustness. AT and its variants are widely considered the most empirically successful against adversary examples. Along the same line, this work proposes a new training objective, **PMHR-AT** (Penalized Modified Huber Regularization for Adversarial training) for improving adversarial robustness. PMHR-AT minimizes both natural and adversarial risk and introduces a modified Huber loss between the natural and adversarial logits as a regularization with the regularization strength adjusted based on the similarity between the predicted natural and adversarial class probabilities. Experimental results show that the proposed method recorded a better performance than existing methods on strong attacks and offers a better trade-off between the natural accuracy and adversarial robustness.

Index Terms— Adversarial Robustness, Adversarial Training, modified Huber loss.

1. INTRODUCTION

Despite the outstanding progress of deep neural networks (DNNs), debate remains on whether we should fully trust systems that employ DNNs in decision-making. It has been shown that DNNs are vulnerable to adversary examples [1], limiting their application in sensitive domains. The vulnerability problem relates to the non-flatness and non-smoothness of the loss landscapes of trained DNNs. Extensive works have been proposed to overcome such limitations, of which empirical defense methods are currently the most effective.

Among the empirical defense methods, adversarial training (AT) and its variants [2, 3, 4, 5, 6, 5, 7, 8] have been demonstrated to be the most promising approaches. Formally, [3] formulates the adversarial training procedure as an optimization problem, seeking to find the optimal network parameters θ that minimize the following risk:

$$\min_{\theta} \frac{1}{n} \sum_{i=1}^{n} l(f_\theta(x_i'), y_i), \tag{1}$$

The research was partially supported by NSF grant IIS-2231797.

where $l(.)$ is a loss function, $f_\theta(x_i)$ is the prediction of the neural network with parameters θ given an input x_i, and y_i is the class label. In (1), the adversarial examples x_i''s are generated using $x_i' = arg\max_{x' \in B_\epsilon[x_i]} l_I(f_\theta(x'), y_i)$, which are then used to train the model. $l_I(.)$ is the loss used to generate adversary examples (the cross-entropy loss is commonly used), and $B_\epsilon[x] = \{x' | \, \|x' - x\|_p < \epsilon\}$ is a neighborhood of x. A widely used approach in searching the optimal adversarial examples is the projected gradient descent (PGD) [3]. Assuming a starting point $x^{(0)} = x_i + Gaussian/Uniform\ noise$ in the input feature space with distance metric $||x - x'||_\infty$ and $t \in \mathbb{N}$, PGD generates adversarial examples using the following update rule:

$$x^{(t+1)} = \prod_{B[x_i]} (x^{(t)} + \alpha \cdot sign(\nabla_{x^{(t)}} l_I(f_\theta(x^{(t)}), y_i))) \tag{2}$$

In (2), α is a step size, $\prod_{B[x_i]}(.)$ is the projection function, and $x^{(t)}$ is the adversarial example at step t.

Along the line of adversarial training, existing works focus on designing better losses, regularization approaches that enhance robustness, perturbing the network weights, and assigning weights to losses during training, to name a few. We provide more details on existing works in Section 2.

The contributions of this paper are summarized as follows:

- We propose a new training objective, termed **PMHR-AT**, for improving the robustness of deep neural networks against adversarial examples. PMHR-AT consists of objectives for minimizing both natural and adversarial risk and two regularization terms: a modified Huber loss between the natural and adversarial logits that is penalized by the closeness between the predicted natural and adversarial probabilities, and the l_2 penalty to the network weights.

- We experimentally demonstrate that the proposed method (**PMHR-AT**) improves the state-of-the-art on adversarial robustness against common attacks and achieves a better trade-off between the natural accuracy and adversarial robustness.

2. EXISTING WORKS

The discovery of adversary examples attracted many talents in the model robustness community, particularly adversarial robustness. Since then, extensive work has been done and is mainly classified into two subgroups: certified defense and empirical defense. Certified defense provides a provable guarantee of adversarial robustness to norm bounded (l_1 and l_2) perturbations [4, 9]. Empirical defense, the most successful, incorporates adversarial data into the training process [3, 10, 11, 4, 12, 5, 6, 8]. Our work aligns with adversarial training, so we provide a brief discussion of benchmark adversarial training approaches. The adversarial training is formulated as a minimax optimization problem. Madry et al. [3] utilize the standard cross-entropy loss. Adversarial Logit Pairing (ALP) [10] proposes a regularization term that minimizes the mean square error loss between two logits (natural and adversarial logits). MIMAE-AT [8] proposes two regularization terms, the mutual information between the probabilistic predictions of the natural example and its adversarial version and the mean absolute error between their logits. TRADES [4] theoretically characterizes the trade-off between accuracy and robustness of classification problems and proposes a regularization term that trades adversarial robustness off against accuracy. MMA [6] proposes a training objective that directly maximizes the input margin for each data point to achieve adversarial robustness. MART [5] proposes a regularization term that explicitly differentiates misclassified and correctly classified examples. Of the methods listed in this section, we will compare our work with the Vanilla AT [3] and its most prominent variants TRADES [4], MART [5], and MIMAE-AT [8].

3. NOTATION

Consider a C classes' classification problem over the data set $D = \{(x_i, y_i)\}_{i=1}^n$ where x_i is a natural input example associated with the label $y_i \in \{1, ..., C\}$. Let $f_c(x_i, \theta)$ be the *logit* output of the deep neural network with model parameters θ corresponding to class c and $p_c(x_i, \theta) = e^{f_c(x_i,\theta)} / \sum_{c'=1}^{C} e^{f_{c'}(x_i,\theta)}$ represent the probability that the network predicts class c given the input example x_i. We denote by $l(.)$ and $E[l(.)]$ the loss and expected loss, respectively. The loss of the network over the dataset D is defined by

$$E[l(.)] = \frac{1}{n} \sum_{i=1}^{n} l(f_\theta(x_i), y_i). \tag{3}$$

Where $f_\theta(x_i)$ is the class prediction of the network.

4. PROPOSED DEFENSE METHOD

4.1. Empirical Risk Formulation

Formally, consider the adversarial risk formulation of [3, 4] on the classifier $f_\theta(.)$ with the 0-1 loss over the dataset $D = \{(x_i, y_i)\}_{i=1}^n$ formulated as

$$Risk_{adv}(f_\theta(.)) = \frac{1}{n} \sum_{i=1}^{n} \max_{x_i' \in B_\epsilon[x_i]} \mathbb{1}(f_\theta(x_i') \neq y_i), \tag{4}$$

where $\mathbb{1}(.)$ is the indicator function. The natural risk is formulated as

$$Risk_{nat}(f_\theta(.)) = \frac{1}{n} \sum_{i=1}^{n} \mathbb{1}(f_\theta(x_i) \neq y_i). \tag{5}$$

Our model should be able to correctly classify both adversarial and natural examples. Therefore, we should minimize both natural and adversarial risks. In addition, we minimize $\mathbb{1}(f_\theta(x_i') \neq f_\theta(x_i))$ as a regularization, which encourages learning a model that makes the same classification decisions on the natural example and its adversarial counterpart and may further improve adversarial robustness.

In summary, we propose to minimize the following risk:

$$Risk_r(f_\theta(.)) = \frac{1}{n} \sum_{i=1}^{n} [\mathbb{1}(f_\theta(x_i) \neq y_i) + \mathbb{1}(f_\theta(x_i') \neq y_i) + \mathbb{1}(f_\theta(x_i') \neq f_\theta(x_i))], \tag{6}$$

where

$$x_i' = \arg\max_{x_i' \in B_\epsilon[x_i]} \mathbb{1}(f_\theta(x_i') \neq y_i). \tag{7}$$

4.2. Surrogate losses

Directly minimizing the empirical risk in Eq. (6) with 0-1 loss is intractable. The 0-1 loss is usually replaced by an appropriate convex surrogate loss in practice. The most commonly used surrogate loss for the $\mathbb{1}(f_\theta(x_i) \neq y_i)$ term in Eq. (6) is the cross-entropy (CE) loss, defined by

$$CE(p(x_i, \theta), y_i) = -\log p_{y_i}(x_i, \theta), \tag{8}$$

where $p_{y_i}(x_i, \theta)$ is the probability that the network predicts class y_i given the input example x_i. We use $-log(1 - \max_{k \neq y_i} p_k(x_i', \theta))$ as a surrogate loss for the $\mathbb{1}(f_\theta(x_i') \neq y_i)$ term in Eq. (6). In related work, MART [5], as well as MIMAE-AT [8], used the boosted cross-entropy (BCE) loss formulated as

$$BCE(p(x_i', \theta), y_i) = -\log p_{y_i}(x_i', \theta) - log(1 - \max_{k \neq y_i} p_k(x_i', \theta)). \tag{9}$$

However, BCE improves adversarial robustness at the expense of natural performance. Instead, we consider a formulation that minimizes both natural and adversarial risk, leading us to the following loss

$$l_1 = -\log p_{y_i}(x_i, \theta) - log(1 - \max_{k \neq y_i} p_k(x_i', \theta)). \tag{10}$$

For the regularization term $\mathbb{1}(f_\theta(x'_i) \neq f_\theta(x_i))$ in Eq. (6), we minimize the difference between the natural and adversarial logits as a surrogate. Intuitively, logits contain information about a class prediction, so minimizing the error between natural and adversarial logits will also improve the model's robustness to class-based attacks. In related work, MIMAE-AT [8] proposes a regularization term that minimizes the mean absolute error between the logits of the natural example x and its adversarial version x', given by $\|f(x'_i, \theta) - f(x_i, \theta)\|_1$. Similarly, ALP (Adversarial Logit Pairing) [10] minimizes $\|f(x'_i, \theta) - f(x_i, \theta)\|_2$. However, the mean square error loss is known to be sensitive to outliers and could lead to unexpected outcomes [13].

With this in mind, to minimize the difference between the natural and adversarial logits, we propose to exploit the mHuber model developed in [14] and showed to be more robust to outliers and noisy data than the original Huber[15].

mHuber Loss: Consider two vectors $u = [u_1, ..., u_n]$ and $v = [v_1, ..., v_n]$. The element-wise subtraction is $u - v = [u_1 - v_1, ..., u_n - v_n]$, and $|u - v| = [|u_1 - v_1|, ..., |u_n - v_n|]$. Let $c = [c_1, .., c_n]$ such that c_i is True if $|u_i - v_i|/\alpha \leq \pi/2$, and False otherwise. In addition, let $A = [A_1, \ldots, A_n] = \alpha^2(1 - cos((u - v))/\alpha)$, $B = [B_1, \ldots, B_n] = \alpha|u - v| + (1 - \frac{\pi}{2})\alpha^2$, and $H = [H_1, ..., H_n]$ such that $H_i = A_i$ if c_i is True, and $H_i = B_i$ if c_i is False. Then $mHuber(u, v, \alpha) = mean(H) \equiv (H_1 + \ldots + H_n)/n$.

Specifically, we propose the following mHuber loss as a surrogate for the regularization term $\mathbb{1}(f_\theta(x'_i) \neq f_\theta(x_i))$ in Eq. (6):

$$l_2 = mHuber(f(x'_i, \theta), f(x_i, \theta), \alpha). \tag{11}$$

Using ResNet-18 and the Optuna hyperparameter optimization package, we experimented with different α values and recorded the optimal result at 5.345, which is then used across models and datasets for better generalization.

4.3. Proposed training objective

Further, we propose to vary the strength of the mHuber regularization term based on the similarity between the predicted natural and adversarial class probability distributions. Specifically, for the C class classification problem, denote by $P_C = [p_1, p_2, \ldots, p_{|C|}]$ and $P'_C = [p'_1, p'_2, \ldots, p'_{|C|}]$ the predicted class probability distributions of natural and adversarial examples, respectively. Let $1 - P_C = [1 - p_1, \ldots, 1 - p_{|C|}]$ and $1 - P'_C = [1 - p'_1, \ldots, 1 - p'_{|C|}]$. We define the following penalty for the mHuber regularization term l_2:

$$k = mean((1 - P_C) \odot (1 - P'_C)), \tag{12}$$

where $(1 - P_C) \odot (1 - P'_C)$ is the element-wise multiplication between the two vectors $1 - P_C$ and $1 - P'_C$. k measures the similarity between the natural probability distributions P_C and its adversarial counterpart P'_C, such that k is larger when the predicted class probabilities are the same over the natural and adversarial examples, and the value is smaller when the predictions are different. Therefore, $k * l_2$ assigns a bigger penalty to the mHuber loss during the training process when the predicted classes are the same over the natural and adversarial examples and assigns a smaller penalty otherwise. Intuitively, since there is a trade-off between the natural and adversarial accuracy [4] and the mHuber regularization may increase the adversarial accuracy at the expense of the natural accuracy, we only increase the regularization strength when the network performs well to achieve better trade-off between the natural and adversarial accuracy.

In summary, we propose the following *penalized modified Huber regularization for adversarial training (PMHR-AT)* objective:

$$\min_\theta \frac{1}{n} \sum_{i=1}^{n} [l_1 + \lambda * k * l_2 + \beta \|\theta\|_2], \tag{13}$$

where we have applied the L_2 regularization on the model weights to prevent overfitting and improve generalizability, and λ and β are regularization hyperparameters. The adversarial examples x'_i will be generated following the standard AT method [3] as follows:

$$x'_i = \underset{x'_i \in B_\epsilon[x_i]}{\arg\max} CE(p(x'_i, \theta), y_i). \tag{14}$$

5. EXPERIMENTS

We conducted a series of experiments and compared our method with the state-of-the-art defenses on benchmark datasets CIFAR-10 [16], and CIFAR-100 [16]. We tested on two model architectures: ResNet-18 [17] and a larger capacity network, WideResNet-34-10 [18].

Baselines: We compare PMHR-AT with top-performing variants of the adversarial training defense to date: Vanilla AT [3], TRADES [4], MART [5], and MIMAE-AT [8]. MMA [6] is shown to be outperformed by MART in [5] on most stronger attacks and hence is not compared with.

Training settings: The regularization parameter in TRADES and MART is set to 6.0 and 5.0 respectively as specified in their original papers. We fine-tuned PMHR-AT on CIFAR-10 and observed our best performance when the regularization parameter λ is set to 26.5 and 20.5 for WideResNet and ResNet-18, respectively. On CIFAR-100, λ is 64.5. β is $2\text{x}10^{-6}$ for WideResNet and $2\text{x}10^{-4}$ for ResNet18. The initial learning rate is set to 0.1 for WideResNet and 0.01 for ResNet-18, then decayed by a factor of ten at the 75th and further decayed at the 90th epoch. We used a batch size of 128 for training and 100 for testing for both models. Adversarial data used in training are generated using PGD with: random start, maximum perturbation ϵ set to 0.031, step size as 0.007, and the number of steps as 10. We train all the models for 100 epochs using SGD with a momentum of 0.9.

Table 1. Clean and robust accuracy on **ResNet-18**. We perform six runs and report the average performance with 95% confidence intervals. The 'Clean' column represents accuracy on natural examples.

Dataset	Method	Clean	FGSM	PGD-20	PGD-100	CW	AA	SQUARE	SPSA
CIFAR-10	*Vanilla AT*	**85.80**$_{\pm 0.001}$	57.87$_{\pm 0.0023}$	52.05$_{\pm 0.003}$	49.28$_{\pm 0.0022}$	51.08$_{\pm 0.001}$	46.62$_{\pm 0.004}$	55.69$_{\pm 0.0014}$	56.17$_{\pm 0.001}$
	TRADES	82.46$_{\pm 0.0012}$	58.26$_{\pm 0.0030}$	54.78$_{\pm 0.0010}$	53.45 $_{\pm 0.0032}$	51.65$_{\pm 0.0021}$	49.08$_{\pm 0.0031}$	55.64$_{\pm 0.001156.50}$	56.50$_{\pm 0.0020}$
	MART	81.30 $_{\pm 0.003}$	58.06$_{\pm 0.001}$	54.73$_{\pm 0.006}$	53.28$_{\pm 0.005}$	51.86$_{\pm 0.0031}$	49.01$_{\pm 0.0020}$	55.66$_{\pm 0.0031}$	56.15$_{\pm 0.0040}$
	MIMAE-AT	81.19$_{\pm 0.0010}$	58.86$_{\pm 0.0031}$	54.89 $_{\pm 0.0024}$	53.18$_{\pm 0.0011}$	51.15$_{\pm 0.0033}$	47.60$_{\pm 0.0020}$	54.66$_{\pm 0.0021}$	55.42$_{\pm 0.0017}$
	PMHR-AT	83.12$_{\pm 0.0022}$	**60.34**$_{\pm 0.0010}$	**56.13**$_{\pm 0.0021}$	**54.45**$_{\pm 0.0031}$	**52.16**$_{\pm 0.0010}$	**49.42**$_{\pm 0.0020}$	**56.54**$_{\pm 0.0021}$	**57.16**$_{\pm 0.0030}$

Table 2. Clean and robust accuracies on **WRN-34-10**. We perform six runs and report the average performance with 95% confidence intervals. The 'Clean' column represents accuracy on natural examples.

Dataset	Method	Clean	FGSM	PGD-20	PGD-100	CW	AA	SQUARE	SPSA
CIFAR-10	*Vanilla AT*	**86.46**$_{\pm 0.0013}$	61.62$_{\pm 0.0021}$	56.75$_{\pm 0.002}$	54.72 $_{\pm 0.001}$	55.63$_{\pm 0.0012}$	51.06 $_{\pm 0.0023}$	59.68$_{\pm 0.0012}$	60.66$_{\pm 0.002}$
	TRADES	84.58$_{\pm 0.0021}$	60.60$_{\pm 0.001}$	57.71$_{\pm 0.0012}$	56.69$_{\pm 0.002}$	55.01$_{\pm 0.0013}$	52.57$_{\pm 0.002}$	59.45$_{\pm 0.0024}$	61.09$_{\pm 0.0023}$
	MART	84.25$_{\pm 0.001}$	62.03$_{\pm 0.00}$	58.29$_{\pm 0.0032}$	55.56$_{\pm 0.0011}$	54.82$_{\pm 0.00}$	51.40$_{\pm 0.00}$	58.21$_{\pm 0.00}$	59.87$_{\pm 0.00}$
	MIMAE-AT	85.33$_{\pm 0.0022}$	**63.06**$_{\pm 0.003}$	58.23$_{\pm 0.001}$	55.68$_{\pm 0.0011}$	55.20$_{\pm 0.005}$	51.42$_{\pm 0.002}$	59.37$_{\pm 0.004}$	61.27$_{\pm 0.0023}$
	PMHR-AT	84.87$_{\pm 0.0020}$	63.05$_{\pm 0.0010}$	**59.26**$_{\pm 0.0021}$	**57.60**$_{\pm 0.0031}$	**56.36**$_{\pm 0.0010}$	**53.58**$_{\pm 0.0020}$	**59.87**$_{\pm 0.0021}$	**61.71**$_{\pm 0.0030}$

Table 3. Clean and robust accuracies on **ResNet-18** . We perform six runs and report the average performance with 95% confidence intervals. The 'Clean' column represents accuracy on natural examples.

Dataset	Method	Clean	FGSM	PGD-20	PGD-100	CW	AA	SQUARE
CIFAR-100	*Vanilla AT*	56.87$_{\pm 0.0031}$	31.21$_{\pm 0.0021}$	29.33 $_{\pm 0.0010}$	28.46 $_{\pm 0.0010}$	26.33$_{\pm 0.0030}$	23.69$_{\pm 0.0012}$	30.06$_{\pm 0.0030}$
	TRADES	57.16$_{\pm 0.0010}$	31.45$_{\pm 0.0021}$	30.32$_{\pm 0.0021}$	29.48$_{\pm 0.0011}$	25.16$_{\pm 0.0011}$	25.18$_{\pm 0.0031}$	30.46$_{\pm 0.0022}$
	MART	54.02 $_{\pm 0.0013}$	32.81$_{\pm 0.0020}$	31.13$_{\pm 0.0014}$	30.14$_{\pm 0.0011}$	26.98 $_{\pm 0.0010}$	24.83$_{\pm 0.0012}$	31.17 $_{\pm 0.0016}$
	MIMAE-AT	57.55$_{\pm 0.0012}$	32.63$_{\pm 0.0023}$	31.07$_{\pm 0.0010}$	31.09$_{\pm 0.0030}$	27.44$_{\pm 0.0040}$	24.74$_{\pm 0.0020}$	31.10$_{\pm 0.00}$
	PMHR-AT	**58.45**$_{\pm 0.0021}$	**34.33**$_{\pm 0.0031}$	**32.25**$_{\pm 0.0021}$	**31.35**$_{\pm 0.0014}$	**27.78**$_{\pm 0.0011}$	**25.96**$_{\pm 0.0031}$	**31.32**$_{\pm 0.0015}$

The weight decay is $3.5\text{x}10^{-3}$ for ResNet-18 and $7\text{x}10^{-4}$ for WideResNet.

Evaluation details: We evaluated our method under White-box attack threats including the L_∞ PGD-20/100 [3], FGSM [1], CW (PGD optimized with CW loss, confidence level K=50) [19], and AutoAttack [20]. The perturbation size is set to ϵ=0.031 and the step size 0.003. Additionally, we evaluated on strong Black-box attacks SQUARE [21] and SPSA [22] with the perturbation size of 0.001 (for gradient estimation), sample size of 128, 80 iterations, and learning rate 0.01.

Experimental results: Table 1 reports the results on CIFAR-10 for ResNet-18, Table 2 reports the results on CIFAR-10 for WideResNet-34-10, and Table 3 reports the results on CIFAR-100 for ResNet-18. The results show that our proposed PMHR-AT method significantly outperforms the Vanilla AT across common white-box and black-box attacks while Vanilla AT is better on natural accuracy on CIFAR-10. Our PMHR-AT method consistently outperforms other variants of AT (TRADES, MART, and MIMAE-AT), albeit often slightly, on both natural accuracy and across white-box and black-box attacks. These results show that *our proposed PMHR-AT method achieves a better trade-off between the natural accuracy and adversarial robustness than these existing AT-based defense methods it was compared to.*

It has been shown that some defense methods gave a false sense of security against adversarial examples by intentionally or inadvertently using obfuscated gradients [23]. Our experimental results indicate that PMHR-AT has no characteristic behavior of obfuscated gradients specified in [23]. Specifically, the results show that PMHR-AT achieves better robustness on a one-step attack FGSM than on a multi-step attack PGD. In addition, white-box attacks are more successful than black-box attacks.

6. CONCLUSION

This paper proposed a new adversarial training objective termed **PMHR-AT** to defend deep neural networks against adversarial examples. Experimental results on benchmark datasets and network architectures show that the proposed method consistently improves adversarial robustness over existing methods across common white-box and black-box attacks, and offers a better trade-off between the natural accuracy and adversarial robustness.

7. REFERENCES

[1] Ian J Goodfellow, Jonathon Shlens, and Christian Szegedy, "Explaining and harnessing adversarial exam-

ples," *arXiv preprint arXiv:1412.6572*, 2014.

[2] Ian J. Goodfellow, Jonathon Shlens, and Christian Szegedy, "Explaining and harnessing adversarial examples," *CoRR*, vol. abs/1412.6572, 2015.

[3] Aleksander Madry, Aleksandar Makelov, Ludwig Schmidt, Dimitris Tsipras, and Adrian Vladu, "Towards deep learning models resistant to adversarial attacks," in *International Conference on Learning Representations*, 2018.

[4] Hongyang Zhang, Yaodong Yu, Jiantao Jiao, Eric Xing, Laurent El Ghaoui, and Michael Jordan, "Theoretically principled trade-off between robustness and accuracy," in *International Conference on Machine Learning*. PMLR, 2019, pp. 7472–7482.

[5] Yisen Wang, Difan Zou, Jinfeng Yi, James Bailey, Xingjun Ma, and Quanquan Gu, "Improving adversarial robustness requires revisiting misclassified examples," in *International Conference on Learning Representations*, 2020.

[6] Gavin Weiguang Ding, Yash Sharma, Kry Yik Chau Lui, and Ruitong Huang, "Mma training: Direct input space margin maximization through adversarial training," in *International Conference on Learning Representations*, 2020.

[7] Cihang Xie, Mingxing Tan, Boqing Gong, Alan Yuille, and Quoc V Le, "Smooth adversarial training," *arXiv preprint arXiv:2006.14536*, 2020.

[8] Modeste Atsague, Olukorede Fakorede, and Jin Tian, "A mutual information regularization for adversarial training," in *Asian Conference on Machine Learning*. PMLR, 2021, pp. 188–203.

[9] Mislav Balunović and Martin Vechev, "Adversarial training and provable defenses: Bridging the gap," in *8th International Conference on Learning Representations (ICLR)*, 2020.

[10] Harini Kannan, Alexey Kurakin, and Ian Goodfellow, "Adversarial logit pairing," *arXiv preprint arXiv:1803.06373*, 2018.

[11] Qi-Zhi Cai, Chang Liu, and Dawn Song, "Curriculum adversarial training," in *Proceedings of the Twenty-Seventh International Joint Conference on Artificial Intelligence (IJCAI)*, 2018, pp. 3740–3747.

[12] Yisen Wang, Xingjun Ma, James Bailey, Jinfeng Yi, Bowen Zhou, and Quanquan Gu, "On the convergence and robustness of adversarial training.," in *ICML*, 2019.

[13] K. Liano, "Robust error measure for supervised neural network learning with outliers," *IEEE Transactions on Neural Networks*, vol. 7, no. 1, pp. 246–250, 1996.

[14] William JJ Rey et al., "Introduction to robust and quasi-robust statistical methods," 1983.

[15] Ziyang Guo, Anyou Min, Bing Yang, Junhong Chen, and Hong Li, "A modified huber nonnegative matrix factorization algorithm for hyperspectral unmixing," *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, vol. 14, pp. 5559–5571, 2021.

[16] Alex Krizhevsky and Geoffrey Hinton, "Learning multiple layers of features from tiny images," Tech. Rep. 0, University of Toronto, Toronto, Ontario, 2009.

[17] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun, "Deep residual learning for image recognition," in *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016, pp. 770–778.

[18] Sergey Zagoruyko and Nikos Komodakis, "Wide residual networks," in *Proceedings of the British Machine Vision Conference (BMVC)*, Edwin R. Hancock Richard C. Wilson and William A. P. Smith, Eds. September 2016, pp. 87.1–87.12, BMVA Press.

[19] Nicholas Carlini and David Wagner, "Towards evaluating the robustness of neural networks," in *2017 ieee symposium on security and privacy (sp)*. IEEE, 2017, pp. 39–57.

[20] Francesco Croce and Matthias Hein, "Reliable evaluation of adversarial robustness with an ensemble of diverse parameter-free attacks," in *International Conference on Machine Learning*. PMLR, 2020, pp. 2206–2216.

[21] Maksym Andriushchenko, Francesco Croce, Nicolas Flammarion, and Matthias Hein, "Square attack: a query-efficient black-box adversarial attack via random search," 2020.

[22] Jonathan Uesato, Brendan O'Donoghue, Pushmeet Kohli, and Aaron van den Oord, "Adversarial risk and the dangers of evaluating against weak attacks," in *Proceedings of the 35th International Conference on Machine Learning*, Jennifer Dy and Andreas Krause, Eds. 10–15 Jul 2018, vol. 80 of *Proceedings of Machine Learning Research*, pp. 5025–5034, PMLR.

[23] Anish Athalye, Nicholas Carlini, and David Wagner, "Obfuscated gradients give a false sense of security: Circumventing defenses to adversarial examples," in *Proceedings of the 35th International Conference on Machine Learning, ICML 2018*, July 2018.