Exploiting Joint Robustness to Adversarial Perturbations

Ali Dabouei, Sobhan Soleymani, Fariborz Taherkhani, Jeremy Dawson, Nasser M. Nasrabadi
West Virginia University
{ad0046, ssoleyma, ft0009}@mix.wvu.edu, {nasser.nasrabadi, jeremy.dawson}@mail.wvu.edu

Abstract

Recently, ensemble models have demonstrated empirical capabilities to alleviate the adversarial vulnerability. In this paper, we exploit first-order interactions within ensembles to formalize a reliable and practical defense. We introduce a scenario of interactions that certifiably improves the robustness according to the size of the ensemble, the diversity of the gradient directions, and the balance of the member's contribution to the robustness. We present a joint gradient phase and magnitude regularization (GPMR) as a vigorous approach to impose the desired scenario of interactions among members of the ensemble. Through extensive experiments, including gradient-based and gradient-free evaluations on several datasets and network architectures, we validate the practical effectiveness of the proposed approach compared to the previous methods. Furthermore, we demonstrate that GPMR is orthogonal to other defense strategies developed for single classifiers and their combination can further improve the robustness of ensembles.

1. Introduction

Deep neural networks (DNNs) have played an astonishing role in the evolution of modern machine learning by achieving state-of-the-art performance on many challenging tasks [22, 41]. Despite their excellent performance, scalability, and generalization to unseen test data, they suffer from a major drawback: slight manipulations of the input samples can form *adversarial examples* causing drastic changes in the predictions of the model [38, 19, 31]. Perturbations required for this aim are often quasi-imperceptible to the human eye and can transfer across classifiers [7, 14], data samples [30, 32], and input transformations [3, 5]. This issue has raised increasing concerns regarding the deployment of DNNs in security-sensitive applications such as autonomous vehicles, biometric identification, and e-commerce.

Initially, a large body of work has been devoted to addressing the problem by heuristic approaches built upon the empirically observed characteristics of perturbations, such as their noisy structure. However, the uncertainty of assumptions and lack of formal explanations for the phenomenon has caused the majority of the defense attempts to be compromised by more advanced attacks [7, 12, 11]. Recent studies have made significant progress in explaining the cause of adversarial vulnerability by demonstrating that adversarial examples are natural consequences of non-zero test error of classifiers in the data space [18, 10]. Particularly, due to the huge cardinality of the input space, a small number of misclassified points around a natural input sample forms a very close decision boundary which can be reached by adversarial perturbations. This suggests that adversarial robustness can only be certified for bounded perturbations [20, 10] since achieving zero error rate is nontrivial in general [18].

The majority of studies on adversarial robustness have concerned single classifiers [38, 19, 26, 18, 10, 20]. However, exploring interactions of multiple classifiers has highlighted the potential of ensembles for mitigating the adversarial vulnerability [33, 21, 1, 4]. In this paper, we exploit first-order interactions in ensembles to provably improve the robustness of the ensemble prediction. We illustrate that the diversity of the gradient directions and the balance of the gradient magnitudes are two key factors for enhancing the robustness of deep ensembles. Specifically, we make the following contributions:

- We introduce a practically feasible case of interactions within ensembles which is certified to improve the robustness of the model against *white-box* attacks.
- We propose a training framework termed joint gradient phase and magnitude regularization (GPMR) to impose the desired interactions among the members of the ensemble.
- We validate the effectiveness of the proposed method using extensive experiments including gradient-based and gradient-free evaluations.
- We demonstrate that the proposed training framework is orthogonal to previous approaches that aim to provide adversarial robustness by bounding the magnitude of the gradients, such as adversarial training.

2. Related Work

Myriad of studies have attempted to robustify DNNs employing approaches such as knowledge distillation [35], manifold learning [37, 29], data transformation and compression [40, 15], statistical analysis [44], and regularization [43]. However, the majority of the defense schemes in the literature are compromised by more sophisticated attacks [7, 6]. An effective approach for improving the robustness of DNNs is *adversarial training* in which the training set is augmented by adversarial examples crafted during the training process. This approach is widely studied using different types of adversarial examples [38, 19, 31, 26, 23]. A major limitation of adversarial training is its dependence on the type of adversarial examples used for training the model. Thus, this approach cannot provide reliable robustness against unseen adversarial examples and out-of-distribution samples, *e.g.* crafted by additive Gaussian noise [18].

A group of studies proposed to directly limit the variation of predictions against slight input changes by bounding the Lipschitz constant of networks [9, 20, 39]. However, controlling the Lipschitz constant involves incorporating highly non-linear and intractable losses to the training objective, which results in restrictive computational costs for large-scale DNNs. Besides, theoretical assumptions for regularizing the Lipschitz constant of DNNs reduces the effectiveness of these approaches against strong attacks [42].

Another body of work has considered interactions of multiple classifiers to alleviate adversarial vulnerability [1, 4, 33, 21]. The majority of these approaches propose a method to promote the diversity of predictions. Abbasi *et al*. [1] demonstrated that specializing members of the ensemble on different subsets of classes can provide robustness against adversarial examples. Bagnall *et al*. [4] proposed a joint optimization scheme to minimize the similarity of the classification scores on adversarial examples. Pang *et al*. [33] developed the adaptive diversity promoting (ADP) approach which diversifies the non-maximum predictions to maintain the accuracy of the model on natural examples.

However, diversifying the predictions does not provide reliable robustness in the *white-box* defense scenario, where all the parameters of the model are known by the adversary. In this setup, the adversary can use the gradients of diversified predictions to fool all classifiers at the same time. Moreover, we both theoretically and experimentally demonstrate that diversifying predictions does not improve the robustness in gradient-free evaluations since the gradient of classifiers can share similar directions. Recently, Kariyappa and Qureshi [21] considered the diversity of gradients in ensembles to provide adversarial robustness and proposed the gradient alignment loss (GAL).

However, this approach suffers from two limitations. First, GAL does not consider the optimal geometrical bounds for diversifying the gradient directions. This degrades the performance of the approach and causes significant fluctuations in the training process as discussed in section 4. Second, GAL does not equalize the magnitude of gradients of the members. Therefore, it is solely evaluated in the *black-box* threat model, where the attacker has no access to the model parameters or gradients. In the *white-box* attack scenario, the attacker can easily fool a few classifiers in the set which have the maximum gradient magnitudes at the input sample. In contrast, our work establishes a new theoretical framework for analyzing the joint robustness by finding the optimal first-order defensive interactions between the members of the ensemble in the white-box threat model.

3. Joint Adversarial Robustness

Altering the prediction of the classifier primarily changes the score of the predicted class. Therefore, in our theoretical analysis, we focus on the change of the final output of a differentiable classifier rather than the change in the index of the maximum argument of the output, *i.e.*, the predicted class. Considering ℓ_p-norm as the distance metric to measure the magnitude of perturbations, we define the robustness to adversarial examples or, more specifically, adversarial perturbations as follows:

Definition 1. *A function $f : \mathcal{X} \subset \mathbb{R}^n \rightarrow \mathbb{R}^m$ is said to be $(\epsilon, \delta)_p$ robust to adversarial perturbations over the set $\mathcal{X}$, if for all samples x, $x' \in \mathcal{X}$ and $||x - x'||_p \leq \epsilon$, f satisfies $||f(x) - f(x')||_2 \leq \delta$.*

To compare the robustness of different classification schemes, we analyze the ℓ_p-norm lower bound of the magnitude of perturbations, ϵ, that is needed to change the maximum prediction by a fixed δ.

We analyze the robustness of a single classifier in Section 3.1. In Section 3.2, we formalize the robustness of ensembles for the case where the adversary has to change the prediction of all members to fool the ensemble prediction. Here, we introduce a practically feasible scenario, *i.e.*, a set of conditions, for interactions between the members of the ensemble and prove that it enhances the robustness of the ensemble. Afterward in Section 3.3, we adopt the proposed scenario for the practical threat models in which fooling a subset of classifiers in the ensemble is sufficient to change the ensemble prediction. Finally, we present our approach for imposing the desired defensive interactions in Section 3.4.

3.1. Robustness of a Single Classifier

Let $f : \mathcal{X} \rightarrow \mathbb{R}^m$ be a differentiable classifier mapping data point $x \in \mathcal{X}$ to m classification scores $f_j(x)$, $j \in \{0, \ldots, m-1\}$. The true label for sample x is y, and the class predicted by the network is denoted by $c = \arg\max_j f_j(x)$. Any attempt to change the prediction of the network by translating the input sample x using perturbation r changes

$f_c(x)$. We develop our methodology based on the first-order approximation of $f_c(x)$: $f_c(x+r) - f_c(x) \approx \langle \nabla_x f_c, r \rangle$[1] since DNNs exhibit linear characteristics around the input samples [19, 16, 17] where we seek to enhance the robustness. The minimal ℓ_p-norm perturbation r_p^*, for $p \in [1, \infty)$, required to change the classification score by δ can be computed using the Hölder inequality and ℓ_p-norm projection [28, 13] as:

$$r^* \approx \frac{\delta}{||\nabla f_c||_q} \boldsymbol{\partial}(||\nabla f_c||_q), \tag{1}$$

where ℓ_p-norm and ℓ_q-norm are dual norms ($\frac{1}{p} + \frac{1}{q} = 1$), and $\boldsymbol{\partial}(.)$ denotes the subgradient of the argument. For a differentiable ℓ_q-norm at ∇f_c, the subgradient is equal to: $\partial ||\nabla f_c||_q / \partial \nabla f_c$, and Equation 1 can be rewritten as:

$$r^* \approx \Big(\frac{\delta}{||\nabla f_c||_q}\Big)\Big(\frac{|\nabla f_c|^{q-1} \odot sign(\nabla f_c)}{||\nabla f_c||_q^{q-1}}\Big), \tag{2}$$

where $\odot$ denotes element-wise multiplication. Similar first-order approximation of the lower bound of the ℓ_p-norm robustness has been previously derived [31, 20]. Equation 2 implies that the magnitude of the gradient plays a crucial role in the robustness of the classifier. Hence, significant efforts have been made to directly smooth out f_c by controlling the Lipschitz constant [9, 20, 39] or adversarial training [19, 26]. Here, we take an orthogonal approach to the previous studies and seek to increase the lower bound of Equation 2 by exploring the joint robustness of multiple classifiers.

3.2. Joint Robustness of Multiple Classifiers

Let $\mathcal{F}$ be an ensemble of k classifiers, $\mathcal{F}=\{f^i\}_{i=0}^{k-1}$, where $f^i : \mathcal{X} \rightarrow \mathbb{R}^m$ maps the data point $x \in \mathcal{X}$ to m classification scores $f_j^i(x)$, $j \in \{0, \ldots, m-1\}$. The class predicted for x by the classifier $f^i \in \mathcal{F}$ is denoted by $c_i = \arg\max_j f_j^i(x)$. Following the previous studies on the robustness of ensembles [33, 21, 1, 4], we assume that the ensemble prediction is the average of the prediction of individual classifiers as: $\mathcal{F}(x) = \frac{1}{k}\sum_{f \in \mathcal{F}} f(x)$, and the predicted class by the ensemble is: $c = \arg\max_j \mathcal{F}_j(x)$, where $\mathcal{F}_j$ is the predicted probability associated with the j^{th} class. In this section, we relax the problem by assuming that the adversary has to fool all members in order to fool the ensemble prediction, *i.e.*, the ensemble rejects the input sample when $\exists\, i, j : c_i \neq c_j$.

The ℓ_p-norm minimal perturbation r_p^* required to decrease the classification scores of all classifiers at x by at least $\delta > 0$ is the solution of the following optimization problem:

$$\min ||r||_p \text{ s.t. } \langle \nabla f_{c_i}^i, r \rangle \leq -\delta, \quad \forall i \in \{0, \ldots, k-1\}. \tag{3}$$

[1]We drop x from the gradient operator for the rest of the paper.

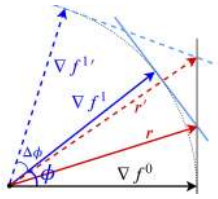


Figure 1: An illustration of Theorem 1 for $k = 2$. Increasing the angle between gradients ∇f^0 and ∇f^1 by $\Delta\phi$ increases the magnitude of the minimum perturbation from $||r||_2 = \frac{\sqrt{2}}{l\sqrt{\cos\phi+1}}$ to $||r'||_2 = \frac{\sqrt{2}}{l\sqrt{\cos(\phi+\Delta\phi)+1}}$.

This interprets that the joint robustness within the ensemble is associated with the gradients of each individual member. Analyzing such interactions rely on the solution of this optimization problem which does not have an analytic form for the general ℓ_p-norm case but can be computed using non-linear programming methods [27]. However, the ℓ_2-norm case when gradient vectors are linearly independent has the following closed-form solution:

$$r_2^* = -\delta \Omega^T (\Omega\, \Omega^T)^{-1} \mathbb{1}_{k\times 1}, \tag{4}$$

where $\Omega := [\nabla f_{c_0}^0, \nabla f_{c_1}^1, \ldots, \nabla f_{c_{k-1}}^{k-1}]^T$ and $\mathbb{1}_{k\times 1}$ is an all-one matrix of size $k \times 1$.

The worst-case scenario for the joint robustness of k classifiers occurs when gradients of classifiers, $\nabla f_{c_i}^i$ for any given sample x, share the same direction. For this case, the magnitude of the optimal ℓ_2-norm solution for Equation 3 is: $||r_2^*||_2 \approx \frac{\delta}{\max_i\{||\nabla f_{c_i}^i||_2\}}$. Therefore, the ℓ_2-norm joint robustness offered by k classifiers in the worst-case scenario is of the same order as the robustness of a single classifier depicted in Equation 2.

Analyzing the characteristics of the optimal perturbation, r_2^*, for the multiple classifier framework is not analytically possible without considering additional constraints. In Theorem 1, we assume that the gradient vectors of all classifiers have an equal magnitude at each x, and they are equiangular, *i.e.*, the angle of any two gradient vectors is equal to ϕ. Hence, we derive a lower bound for the joint robustness of k classifiers with equiangular gradients.

Theorem 1. *Let $\nabla f^0, \ldots, \nabla f^{k-1}$ be k vectors in $\mathbb{R}^n$ with an equal length l, and for any $i \neq j \in \{0, \ldots, k-1\}$, $\langle \nabla f^i, \nabla f^j \rangle = l^2 \cos\phi$, and let $r \in \mathbb{R}^n$ be a vector such that $|\langle \nabla f^i, r \rangle| \geq |\delta|$ holds for any i, then:*

$$||r||_2 \geq \frac{|\delta|\sqrt{k}}{l\sqrt{((k-1)cos\phi + 1)}}. \tag{5}$$

Proof. Rewriting Equation 4 gives the minimal ℓ_2-norm solution, r, satisfying $|\langle \nabla f^i, r \rangle| \geq |\delta|$ as: $r_2^* = \sum_i \alpha_i \nabla f^i$, where α_i is the i^{th} element of the vector

$\alpha = \delta(\Omega\ \Omega^T)^{-1}\mathbb{1}_{k\times 1}$, and $\Omega = [\nabla f^0, \ldots, \nabla f^{k-1}]^T$. Applying the equiangular condition, we have: $\alpha_i = \frac{\delta}{l^2((k-1)cos\phi + 1)}$, which is independent of i. On the other hand, $||\sum_{i=0}^{k-1} \nabla f^i||_2^2 = \langle \sum_{i=0}^{k-1} \nabla f^i, \sum_{i=0}^{k-1} \nabla f^i \rangle = kl^2((k-1)\cos\phi + 1)$. Combining these two equations concludes the proof. □

For the equiangular case, when k classifiers are identical, *i.e.*, $\phi = 0$, members of the ensemble have the minimum defensive interactions since the joint robustness is equal to the robustness of a single classifier obtained in Equation 2. For k classifiers with orthogonal gradient vectors, the lower bound is equal to $||r||_2 = \delta\frac{\sqrt{k}}{l}$ and the robustness is of $O(\frac{\sqrt{k}}{l})$. As ϕ grows, the robustness increases and approaches infinity when $\phi \to \arccos(\frac{-1}{k-1})$. The robustness for an arbitrary set of gradients, $\{\nabla f^0, \ldots, \nabla f^{k-1}\}$, is lower bounded by the robustness of any set of inscribed equiangular vectors with $\phi = \min_{i\neq j} \angle(\nabla f^i, \nabla f^j)$ and $l = \max_i ||\nabla f^i||_2$. Therefore, Theorem 1 provides a lower bound to the robustness of the general case of the gradients. This implies that the robustness of ensembles can be improved by increasing the minimum angle between gradients and decreasing the maximum gradient magnitude. Figure 1 illustrates how promoting the gradient diversity improves the robustness.

Diversity of the gradient directions has been studied before in GAL [21] as a heuristic methodology to improve the robustness against *black-box* attacks. Theorem 1 highlights two shortcomings of GAL limiting its effectiveness against *white-box* attacks. First, GAL does not consider the optimal bound $\arccos(\frac{-1}{k-1})$ for the gradient diversity. We observe that this causes a fluctuation in the training of GAL and reduces the effectiveness of diversifying the gradient directions. Second, GAL does not regularize the gradient magnitudes among members. Consequently, any *white-box* attack to the ensemble prediction can easily circumvent the defensive strategy by targeting the least robust members.

3.3. Threat Model in Practice

In the previous section, we formalized a geometric framework to analyze the robustness according to the size of the ensemble, $k = |\mathcal{F}|$, and the extent of the diversity of the gradient directions. This methodology is built upon the optimization problem in Equation 3 which assumes that the adversary must fool all classifiers at the input sample. However, it is not practical to reject all samples which do not have the full agreement of the members. In real-world applications, changing the prediction of a subset of the ensemble, $\mathcal{F}' \subset \mathcal{F}$, is enough to alter the prediction of the ensemble. In this case, the lower bound of the robustness, presented in Theorem 1, reduces based on $k' = |\mathcal{F}'|$. Previous defenses based on diversifying predictions [33, 1, 4] or gradients [21] do not control the magnitude of the gradients of the members. Thus, the subset required to be fooled in order to fool the ensemble is often smaller than $\lfloor \frac{|\mathcal{F}|}{2} \rfloor + 1$ since the adversary can fool a set of locally weak classifiers, *i.e.*, members with large gradient magnitudes. In the next section, we propose a gradient magnitude equalization loss that alleviate this problem by enforcing: $|\mathcal{F}'| \geq \lfloor \frac{|\mathcal{F}|}{2} \rfloor + 1$.

3.4. Joint Gradient Regularization

Here, we present the *joint gradient phase and magnitude regularization (GPMR)* scheme as a theoretically-grounded approach for improving the robustness of the ensemble against bounded alterations of the input domain. GPMR maximizes a lower bound to the robustness of the ensemble, according to Theorem 1, by jointly regularizing the gradient directions and magnitudes. First, we define the gradient diversity promoting loss to increase the angle between the gradients by forcing the cosine similarity of gradients to approach $\frac{-1}{k-1}$ as:

$$\mathcal{L}_{div} = \frac{2}{k(k-1)} \sum_{0\leq i<j\leq k-1} \left(\frac{\langle \nabla f^i_{c_i}, \nabla f^j_{c_j} \rangle}{||\nabla f^i_{c_i}||_2 ||\nabla f^j_{c_j}||_2} + \frac{1}{k-1}\right)^2, \tag{6}$$

where $\frac{2}{k(k-1)}$ normalizes the loss over the number of the pairs in the ensemble. Second, we define the gradient magnitude regularization loss. To focus on regularizing the joint interactions of the members, we opt to equalize the gradient magnitudes rather than minimizing them as:

$$\mathcal{L}_{eq} = \frac{1}{k}\sum_i (||\nabla f^i_{c_i}||_2 - \frac{1}{k}\sum_j ||\nabla f^j_{c_j}||_2)^2. \tag{7}$$

This forces the gradient magnitudes to be roughly equal at each input sample and equalizes the contribution of the members to the ensemble robustness. Consequently, the adversary must fool at least the majority of classifiers and cannot fool the ensemble prediction by fooling a few classifiers with the maximum magnitude of gradient at the input sample. The equalization loss also makes GPMR orthogonal to defenses developed for single classifiers controlling the smoothness of the predictions [36, 38, 19, 31, 26, 23, 9, 20, 39]. Hence, other defenses can be employed to further robustify the ensemble by alleviating the vulnerability of individual members. It must be noted that $\mathcal{L}_{eq}$ does not constrain the magnitude of gradients at two different input samples.

The final loss function for training the ensemble is:

$$\mathcal{L}_t = \mathcal{L}_{xent} + \lambda_{eq}\mathcal{L}_{eq} + \lambda_{div}\mathcal{L}_{div}, \tag{8}$$

where λ_{div} and λ_{eq} are Lagrangian coefficients controlling the importance of the regularization terms. $\mathcal{L}_{xent}$ is the classification loss function which computes the average of the cross-entropy loss over all members. The classification loss can be defined on natural or adversarial examples. The latter case combines the proposed framework with adversarial

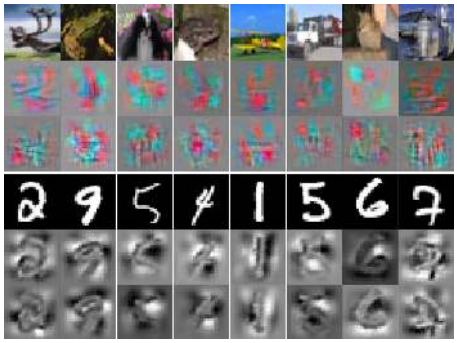


Figure 2: Visualizing gradients for an ensemble of $k = 2$ classifiers trained on CIFAR-10 (top block) and MNIST (bottom block) using GPMR. First, second, and third row in each block illustrates the inputs to the model and gradients of the first and second classifiers, respectively.

training to further improve the robustness of the ensemble as studied in Section 4.3. It may be noted that GPMR does not improve the robustness of the member classifiers. Indeed, it regularizes the interactions of the members to mitigate the adversarial behavior using the joint evaluation of the members. Moreover, GPMR aims to construct an ensemble classifier for which the perturbations crafted for one classifier have less effect on other classifiers or even increase the score corresponding to their predicted class.

4. Experiments

Here, we provide the experimental results to evaluate the effectiveness of GPMR. We evaluate the joint robustness of multiple classifiers on the MNIST, CIFAR-10, and CIFAR-100 datasets. We consider two base network architectures detailed in Table 1. We train models using stochastic gradient descent with momentum equal to 0.9 and weight decay of $5e-4$. The initial learning rate is set to 10^{-1}, and decayed with the factor of 0.2 every 30 epochs until the final learning rate 10^{-4}. We run the training process for 60 epochs on MNIST, and 200 epochs on CIFAR-10 and CIFAR-100. The batch size for training models is set to 64 for all experiments. We observe that λ_{div} directly affects the classification accuracy of the ensembles as depicted in Figure 3a. Consequently, the diversity loss coefficient, λ_{div}, is set to 0.1 for MNIST and 0.04 for CIFAR-10 and CIFAR-100. We also observe that the accuracy of the ensembles on natural examples is roughly independent of λ_{eq}. This is expected since the equalization loss does not minimize the magnitude of gradients. Hence, we select $\lambda_{eq} = 10$ for all network architectures and datasets based on the experiments conducted in Section 4.2 and Figure 3d.

We compare our method to three ensemble models. The first ensemble is trained without any diversity encouraging criterion, *i.e.*, GPMR with $\lambda_{eq} = \lambda_{div} = 0$. The second ensemble is trained using GAL [21] which diversifies the gradient of predictions. ADP [33] is used as the third method to promote the prediction diversity among classifiers. Due to the constraints on the number of classifiers in ADP, we conduct the comparisons with this baseline on ensembles of $k = 3$ classifiers. For GAL, the coefficient of the diversity loss, λ, is set to 0.5. For ADP coefficients α and β are set to 2 and 0.5, respectively. These values are associated with the best performance reported by the authors. Major parts of our experiments are adapted from Pang *et al*. [33] to provide consistent evaluations for the future works. The results are the average of 10 independent runs.

Table 1: Network architecture of the base classifiers, consisting of Convolution (C), Max-pooling (M) and Fully-connected (F) layers. Each RES block consists of two (C) with a skip connection. All layers, except the last (F), are followed by ReLU. The number of classes is denoted by m.

Model	Structure
Conv	2×C64-M-2×C128-M-2×C256-M-2×C256-F512-F(m)
ResNet-20	C16-3×RES16-3×RES32-3×RES64-F512-F(m)

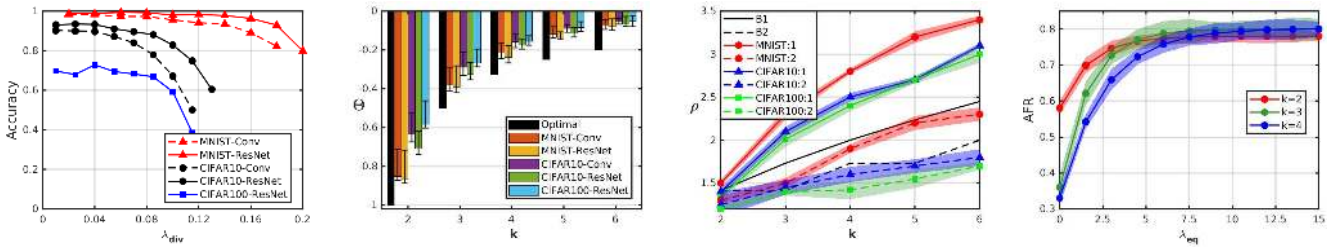
4.1. Performance on Natural Examples

Table 2 presents the classification error rate of the member classifiers and the ensembles. Promoting diversity of gradient directions slightly degrades the classification performance on natural examples. This is attributed to that by diversifying gradients classifiers learn to discriminate input samples based on distinct sets of representative features, illustrated in Figure 2. Minimizing the similarity of salient regions using the gradient diversity loss divides important features between classifiers which reduces the accuracy on natural examples. However, this enhances the robustness against adversarial examples as presented in experiments on white-box defense performance. Table 2 also highlights the superior performance of GPMR compared to GAL. As discussed in Section 3.4, GAL does not consider the optimal bound for the similarity of gradients. During the training, it forces the cosine similarity of gradients for k classifiers to approach -1 while the optimal bound is $\frac{-1}{k-1}$. Consequently, GAL suffers from the fluctuation in the loss and accuracy of individual classifiers during the training.

4.2. Theory vs. Practice

Here, we evaluate the gap between the theory and practice of the proposed approach. To this aim, we first measure the diversity of the gradient directions within the trained ensemble using the expected cosine similarity as:

$$\Theta(\mathcal{F}) = \frac{2}{k(k-1)} \mathbb{E}_x \Big[\sum_{0 \le i < j \le k-1} \frac{\langle \nabla f^i_{c_i}, \nabla f^j_{c_j} \rangle}{||\nabla f^i_{c_i}|| \cdot ||\nabla f^j_{c_j}||} \Big]. \quad (9)$$



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

Figure 3: (a) Classification accuracy of ensembles versus λ_{div} on natural examples, (b) expected cosine similarity of gradients versus the number of classifiers, (c) robustness of the ensemble versus the number of classifiers, where B1 and B2 denote the optimal and practical robustness at $\phi = \frac{\pi}{2}$, and the solid and dashed plots show the results for GPMR with λ_{eq} equal to 10 and 0, respectively, (d) average fooling rate of members in the ensemble versus λ_{eq}.

Table 2: Classification error rate (%) on natural examples. Ensembles consist of $k = 3$ Conv/ResNet-20 classifiers, *i.e.*, Net 1, Net 2, and Net 3. The maximum standard deviation of error is 0.5%, 0.6%, 5.3%, and 0.7% for Base., ADP, GAL, and GPMR, respectively.

Dataset	Classifier	Base.	ADP	GAL	GPMR
MNIST	Net 1	0.76/0.40	0.70/0.41	1.18/0.91	0.72/0.53
	Net 2	0.78/0.43	0.67/0.48	1.14/0.96	0.78/0.57
	Net 3	0.75/0.45	0.69/0.44	1.12/0.93	0.71/0.52
	Ensemble	0.73/0.36	**0.66/0.31**	1.02/0.86	0.71/0.51
CIFAR-10	Net 1	10.43/8.50	10.72/8.93	11.92/9.58	10.37/9.11
	Net 2	10.18/8.15	10.25/9.38	11.58/10.33	10.87/9.52
	Net 3	10.50/8.72	10.80/9.28	11.45/10.19	11.05/9.83
	Ensemble	9.28/6.94	**9.12/6.85**	11.40/9.16	9.30/7.22
CIFAR-100	Net 1	39.83/34.00	43.33/39.35	45.03/39.61	40.46/36.85
	Net 2	38.65/35.58	43.45/40.53	42.32/37.77	40.47/37.48
	Net 3	40.51/35.82	42.94/40.81	43.49/41.13	41.66/37.15
	Ensemble	36.35/30.72	**35.48/30.41**	39.61/36.42	36.76/31.05

Figure 3b presents the empirical values of the cosine similarity computed over $1,000$ test samples. In all experiments, the cosine similarity is negative and close to the optimal value which implies that the diversity of the gradient directions is better than the orthogonal case where all gradients are perpendicular. Diversifying the gradients on MNIST achieves closer values to the optimal bound compared to CIFAR-10 and CIFAR-100. We attribute this to the capacity of members compared to the complexity of the task. Increasing the size of the ensemble enlarges the gap between the practice and the optimal bound of the gradient diversity.

For the second evaluation, we adapt the robustness measure proposed by Moosavi-Dezfooli *et al.* [31] for the ensemble framework as:

$$\rho(\mathcal{F}) := \mathbb{E}_x \Big[\frac{\Delta(x;\mathcal{F})}{\max_{f \in \mathcal{F}} \Delta(x;f)} \Big], \tag{10}$$

where $\Delta(x;\mathcal{F})$ is the minimum ℓ_2-norm adversarial perturbation for the given classifier $\mathcal{F}$ at x, and we approximate it using ℓ_2-DeepFool [31]. Indeed, $\rho(\mathcal{F})$ measures the expected ratio of the robustness of the ensemble over the robustness of the most robust classifier in the set. This measure can reliably characterize the effectiveness of a defense based on ensembles since it measures the relative robustness of the set compared to its members. We compute this measure over $1,000$ test examples. Figure 3c illustrates the results for this evaluation. GPMR improves the robustness on all datasets as the size of the ensemble grows. For instance, with $k = 4$ classifiers, GPMR increases the magnitude of the minimum ℓ_2 perturbation by $\mathbf{2.75}$, $\mathbf{2.5}$, and $\mathbf{2.4}$ on MNIST, CIFAR-10, and CIFAR-100, respectively. We also ablate the role of gradient magnitude equalization by repeating this evaluation using GPMR with $\lambda_{eq} = 0$. As depicted in Figure 3c, diversifying the gradients without equalizing the gradient magnitudes significantly limits the effectiveness of GPMR.

We further analyze the role of the gradient equalization loss by measuring the average ratio of the number of classifiers that are fooled by DeepFool [31] over the number of members in the ensemble. We refer to this ratio as the average fooling ratio (AFR) of the members. We train ensembles consist of $k = \{2, 3, 4\}$ ResNet-20 networks on CIFAR-10. Figure 3d presents the results for these experiments. We observe that without the equalization loss ($\lambda_{eq} = 0$) AFR is 0.58, 0.37, and 0.34 for k equal to 2, 3, and 4, respectively. This illustrates that the attack targets merely 1 or 2 classifiers at each input sample to fool the ensemble prediction. However, by increasing λ_{eq} AFR improves significantly, which validates the effectiveness of the gradient equalization loss for regularizing the contribution of members.

4.3. White-box Defense Performance

We evaluate the performance of GPMR against several well-known and powerful white-box attacks including fast gradient sign method (FGSM) [19], basic iterative method (BIM) [23], projected gradient descent (PGD) [26], momentum iterative method (MIM) [14], Jacobian-based saliency map attack (JSMA) [34], Carlini & Wagner (C&W) [7], and

Table 3: Classification accuracy (%) for adversarial examples on MNIST and CIFAR-10. The results for Conv and ResNet architectures are separated by '/'. The coefficient ϵ for JSMA is set to 0.1 and 0.2 for MNIST and CIFAR-10, respectively. The coefficient β of the EAD attack is set to 0.01. The maximum standard deviation of results is 6.3% and 0.20% for GAL and other methods, respectively.

Attack	Setting	MNIST					Setting	CIFAR-10				
		Baseline	ADP$^{k=3}$	GAL$^{k=3}$	GPMR$^{k=2}$	GPMR$^{k=3}$		Baseline	ADP$^{k=3}$	GAL$^{k=3}$	GPMR$^{k=2}$	GPMR$^{k=3}$
FGSM	$\epsilon=0.1$	65.9/75.2	83.5/95.2	57.4/84.3	85.0/92.2	**90.8/97.6**	$\epsilon=0.02$	30.3/35.2	50.5/60.4	27.6/34.9	53.2/55.2	**61.0/66.8**
	$\epsilon=0.2$	18.2/20.6	45.1/51.2	31.9/39.2	38.9/54.1	**58.7/65.4**	$\epsilon=0.04$	17.6/18.0	44.0/48.7	31.3/32.9	43.1/45.8	**56.0/60.5**
BIM	$\epsilon=0.1$	46.5/50.0	72.5/88.9	40.0/59.2	75.1/80.9	**89.0/92.4**	$\epsilon=0.01$	15.4/16.9	41.8/43.9	31.8/33.8	45.5/48.5	**50.3/55.2**
	$\epsilon=0.15$	12.1/13.8	68.0/72.7	41.5/51.3	67.0/74.3	**73.8/79.6**	$\epsilon=0.02$	5.7/7.2	23.6/32.5	20.4/23.9	33.1/34.7	**38.6/46.2**
PGD	$\epsilon=0.1$	48.6/49.4	78.7/82.4	53.8/54.6	72.8/77.0	**84.8/87.6**	$\epsilon=0.01$	16.9/23.1	44.4/49.2	27.4/36.9	44.8/53.8	**62.5/64.9**
	$\epsilon=0.15$	4.3/7.6	38.2/41.1	26.7/30.8	36.9/42.5	**51.4/59.3**	$\epsilon=0.02$	6.5/7.5	23.1/31.6	20.4/21.5	24.0/33.9	**35.8/49.2**
MIM	$\epsilon=0.1$	54.1/57.6	88.7/91.5	75.0/84.2	90.8/92.1	**91.3/93.5**	$\epsilon=0.01$	22.3/24.2	46.3/54.6	43.3/47.6	49.0/58.4	**63.4/66.8**
	$\epsilon=0.15$	6.4/15.9	70.9/76.8	64.4/69.6	74.1/79.8	**76.5/82.4**	$\epsilon=0.02$	6.8/7.4	25.0/33.7	21.2/28.9	29.3/35.5	**47.2/51.9**
JSMA	$\gamma=0.3$	79.5/83.1	90.1/95.0	76.4/83.6	90.7/94.0	**95.0/96.7**	$\gamma=0.05$	25.9/29.0	40.1/43.7	35.7/36.7	43.3/45.9	**52.9/55.4**
	$\gamma=0.6$	73.2/75.0	86.2/89.8	72.4/81.3	85.9/87.6	**92.8/93.3**	$\gamma=0.1$	23.9/26.2	31.2/38.2	32.8/35.7	35.6/40.2	**48.1/50.6**
C&W	$c=1.0$	25.4/31.3	73.2/78.5	52.7/55.2	77.0/79.4	**80.4/82.4**	$c=0.01$	41.8/46.3	50.9/54.8	32.3/36.6	53.5/58.0	**60.9/66.9**
	$c=10.0$	4.6/5.8	20.1/24.0	10.9/15.7	22.3/27.8	**28.5/33.4**	$c=0.1$	15.8/18.5	22.6/25.4	18.7/20.4	22.1/27.3	**32.9/35.1**
EAD	$c=5.0$	25.1/28.4	90.2/93.0	72.4/74.4	90.2/90.5	**92.8/96.1**	$c=1.0$	12.3/17.1	65.6/70.4	52.6/54.2	63.0/68.8	**76.6/79.8**
	$c=10.0$	7.1/7.4	86.6/89.6	68.9/72.3	83.2/85.1	**87.6/91.9**	$c=5.0$	2.4/3.3	30.1/30.3	10.5/18.5	27.4/31.2	**45.8/50.2**

Table 4: Classification accuracy (%) on adversarial examples for CIFAR-100 with ResNet-20 architecture .

Attack	ϵ	Base.	ADP$^{k=3}$	GAL$^{k=3}$	GPMR$^{k=2}$	GPMR$^{k=3}$
BIM	0.005	23.6	27.3	21.8	34.2	**37.8**
	0.01	11.7	13.6	12.8	19.5	**24.2**
PGD	0.005	25.2	32.4	30.2	36.1	**38.5**
	0.01	11.4	17.8	14.0	25.5	**29.2**
MIM	0.005	23.4	31.2	26.4	32.8	**37.1**
	0.01	10.3	18.9	16.7	22.5	**28.6**

Table 5: Classification accuracy (%) of combined defenses on ResNet-20. The maximum standard deviation is 1.4%.

Defense	FGSM	BIM	PGD	MIM
Def$_A$	41.7	19.6	25.6	28.5
Def$_A$ + GPMR	**70.9**	**54.0**	**55.1**	**58.9**
Def$_B$	41.3	25.4	32.1	33.8
Def$_B$ + GPMR	**66.2**	**68.5**	**57.7**	**62.3**

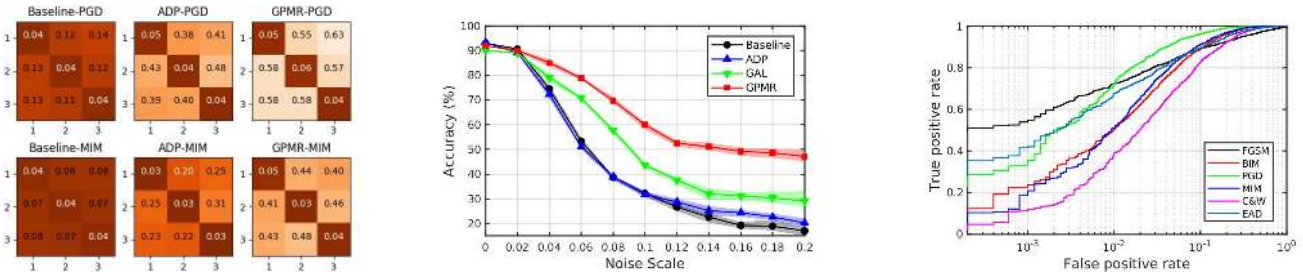
elastic-net attack (EAD) [8]. A brief summary of these attacks can be found in [33]. For each attack, as detailed in Tables 3 and 4, we consider two settings to demonstrate the effectiveness of our approach against a wide range of adversaries. For BIM, PGD, and MIM, the iteration of attack is set to 10 and the step size is set to $\frac{\epsilon}{10}$. Both C&W and EAD are implemented with the learning rate of 0.01 and $1,000$ iterations.

Tables 3 and 4 present the classification accuracy of ensemble models on adversarial examples. GPMR consistently outperforms other ensemble-based defenses on both network architectures and all datasets. Ensembles consist of $k = 2$ classifiers trained with GPMR outperform GAL ensembles with $k = 3$ classifiers, and provide comparable performance to ADP ensembles with $k = 3$ classifiers on MNIST and CIFAR-10. On CIFAR-100, our ensemble model with $k = 2$ classifiers surpasses all other ensembles consisted of $k = 3$ classifiers. This can better demystify the effectiveness of GPMR since its functionality is independent of the number of classes in the task. However, the number of classifiers required by ADP increases as the number of classes grows.

In another set of experiments, we evaluate the orthogonality of GPMR to other defenses. We consider adversarial training on FGSM (Def$_A$) [19] and PGD (Def$_B$) [26] to combine with GPMR. Both defenses are implemented using the same training setup as GPMR. The ℓ_∞-norm magnitude of perturbations, ϵ, is uniformly sampled from the interval $[0.01, 0.05]$ as suggested by Kurakin *et al.* [24]. Table 5 shows the classification accuracy of combined defenses against FGSM ($\epsilon = 0.04$), BIM ($\epsilon = 0.02$), PGD ($\epsilon = 0.02$), and MIM ($\epsilon = 0.02$) on CIFAR-10. As we observe, the combination of other defenses with GPMR consistently improves the performance of the defense. This is attributed to GPMR equalizing gradients and not reducing their magnitudes, while the conventional defense methods seek to reduce the magnitude of gradients. Hence, they can be combined to simultaneously diversify gradient directions, equalize gradient magnitudes, and reduce gradient magnitude. Combining GPMR with adversarial training further improves the robustness since the lower bound in Theorem 1 improves when the magnitude of gradients decreases.

4.4. Transferability Across Individual Classifiers

Defensive interactions between several classifiers can be characterized by the transferability of adversarial examples



(a) (b) (c)

Figure 4: (a) Transferability of adversarial examples in ensembles of size $k = 3$ on CIFAR-10. The rows and columns illustrate the source and target networks, respectively. (b) Gradient-free evaluation of robustness using Gaussian random noise. (c) ROC curves for detecting adversarial examples using the standard deviation of predictions.

among them. We perform transferability experiments using PGD and MIM which are powerful attacks for the black-box setting [25, 33]. We compute adversarial examples for each member classifier and then evaluate their transferability across other members by computing the classification accuracy of the target classifier. The perturbation size for both attacks is set to $\epsilon = 0.05$. Figure 4a presents the results for ResNet-20 architecture on CIFAR-10 and suggests that diversifying gradients is an effective approach to reduce the transferability of adversarial examples among members in the ensemble. However, it may be noted that minimizing the transferability among every two members does not lead to the maximum robustness of the ensemble since for $k > 2$ the optimal cosine similarity of pair of gradients is greater than -1.

4.5. Gradient-free Evaluation of the Robustness

White-box attacks are not sufficient to assess the performance of a defense method since the defense may cause obfuscated gradients and mislead the evaluation [2]. Therefore, in Figure 4b, we evaluate the performance of ensembles consist of $k = 3$ ResNet-20 classifiers on CIFAR-10 samples augmented with random noise. The maximum standard deviation of the results is 2.1%, 2.3%, 4.5%, and 2.3% for baseline, ADP, GAL, and GPMR, respectively. Results suggest that diversifying the predictions in ensembles does not improve the robustness to random perturbations since the performance of ADP is similar to the baseline. However, diversifying gradients improves the robustness to noisy input samples which demonstrates the superiority of gradient diversity compared to prediction diversity.

4.6. Joint Robustness for Detecting Adversaries

Here, we adopt a measure based on the prediction of all members to evaluate the detection performance of ensembles trained by GPMR. We compute the standard deviation of the class probability scores associated with the predicted class over all the classifiers and compare it with a predefined threshold to accept or reject the input example. Figure 4c and Table 6 present the ROC curves and AUC scores for the detection performance on $1,000$ natural examples and $1,000$ adversarial examples from the CIFAR-10 dataset. All ensembles consist of $k = 3$ classifiers. Results validate the performance of our model on detecting alterations of the input samples. Moreover, ADP outperforms GAL due to the disparity in the robustness of subsets of classifiers in GAL. We observe that GAL causes a notable robustness gap between the most and least robust sets of classifiers in the ensemble since it does not regularize the contribution of members in the ensemble.

Table 6: Detection performance of ensembles on CIFAR-10 using AUC (10^{-2}) score. Results for ADP are cited from the original paper.

Attack	Setting	ADP	GAL	GPMR
FGSM	$\epsilon = 0.1$	91.19	90.98	**95.29**
BIM	$\epsilon = 0.1$	93.14	90.54	**96.32**
PGD	$\epsilon = 0.1$	97.03	93.15	**98.45**
MIM	$\epsilon = 0.1$	94.09	91.24	**94.13**
C&W	$c = 1.0$	90.98	88.46	**93.67**
EAD	$c = 20.0$	94.84	91.52	**96.46**

5. Conclusion

In this paper, we introduced a practically feasible scenario of first-order defensive interactions between members of an ensemble. We both theoretically and empirically demonstrated that imposing these interactions significantly improves the robustness of ensembles. We proposed the joint gradient phase and magnitude regularization (GPMR) as an empirical tool to regularize the interaction between members and equalize their role in the ensemble decision. Furthermore, we concluded that the superior performance of GPMR is due to its capability to increase the effective number of members contributing to the robustness.

References

[1] Mahdieh Abbasi and Christian Gagné. Robustness to adversarial examples through an ensemble of specialists. *arXiv preprint arXiv:1702.06856*, 2017.

[2] Anish Athalye, Nicholas Carlini, and David Wagner. Obfuscated gradients give a false sense of security: Circumventing defenses to adversarial examples. *arXiv preprint arXiv:1802.00420*, 2018.

[3] Anish Athalye, Logan Engstrom, Andrew Ilyas, and Kevin Kwok. Synthesizing robust adversarial examples. In *International Conference on Machine Learning*, pages 284–293, 2018.

[4] Alexander Bagnall, Razvan Bunescu, and Gordon Stewart. Training ensembles to detect adversarial examples. *arXiv preprint arXiv:1712.04006*, 2017.

[5] Tom B Brown, Dandelion Mané, Aurko Roy, Martín Abadi, and Justin Gilmer. Adversarial patch. *arXiv preprint arXiv:1712.09665*, 2017.

[6] Nicholas Carlini and David Wagner. Adversarial examples are not easily detected: Bypassing ten detection methods. In *Proceedings of the 10th ACM Workshop on Artificial Intelligence and Security*, pages 3–14. ACM, 2017.

[7] Nicholas Carlini and David Wagner. Towards evaluating the robustness of neural networks. In *2017 IEEE Symposium on Security and Privacy (SP)*, pages 39–57. IEEE, 2017.

[8] Pin-Yu Chen, Yash Sharma, Huan Zhang, Jinfeng Yi, and Cho-Jui Hsieh. EAD: elastic-net attacks to deep neural networks via adversarial examples. In *Thirty-second AAAI conference on artificial intelligence*, 2018.

[9] Moustapha Cisse, Piotr Bojanowski, Edouard Grave, Yann Dauphin, and Nicolas Usunier. Parseval networks: Improving robustness to adversarial examples. In *Proceedings of the 34th International Conference on Machine Learning-Volume 70*, pages 854–863. JMLR. org, 2017.

[10] Jeremy M Cohen, Elan Rosenfeld, and J Zico Kolter. Certified adversarial robustness via randomized smoothing. *arXiv preprint arXiv:1902.02918*, 2019.

[11] Ali Dabouei, Sobhan Soleymani, Jeremy Dawson, and Nasser M Nasrabadi. Fast geometrically-perturbed adversarial faces. In *IEEE Winter Conference on Applications of Computer Vision (WACV)*, 2019.

[12] Ali Dabouei, Sobhan Soleymani, Fariborz Taherkhani, Jeremy Dawson, and Nasser Nasrabadi. Smoothfool: An efficient framework for computing smooth adversarial perturbations. In *The IEEE Winter Conference on Applications of Computer Vision*, pages 2665–2674, 2020.

[13] Xuan Vinh Doan and Stephen Vavasis. Finding approximately rank-one submatrices with the nuclear norm and \ell_1-norm. *SIAM Journal on Optimization*, 23(4):2502–2540, 2013.

[14] Yinpeng Dong, Fangzhou Liao, Tianyu Pang, Hang Su, Jun Zhu, Xiaolin Hu, and Jianguo Li. Boosting adversarial attacks with momentum. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 9185–9193, 2018.

[15] Gintare Karolina Dziugaite, Zoubin Ghahramani, and Daniel M Roy. A study of the effect of jpg compression on adversarial images. *arXiv preprint arXiv:1608.00853*, 2016.

[16] Alhussein Fawzi, Seyed Mohsen Moosavi Dezfooli, and Pascal Frossard. The robustness of deep networks-a geometric perspective. *IEEE Signal Processing Magazine*, 34, 2017.

[17] Alhussein Fawzi, Seyed-Mohsen Moosavi-Dezfooli, Pascal Frossard, and Stefano Soatto. Empirical study of the topology and geometry of deep networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018.

[18] Nic Ford, Justin Gilmer, Nicolas Carlini, and Dogus Cubuk. Adversarial examples are a natural consequence of test error in noise. *arXiv preprint arXiv:1901.10513*, 2019.

[19] Ian J Goodfellow, Jonathon Shlens, and Christian Szegedy. Explaining and harnessing adversarial examples. In *International Conference on Learning Representations (ICLR)*, 2015.

[20] Matthias Hein and Maksym Andriushchenko. Formal guarantees on the robustness of a classifier against adversarial manipulation. In *Advances in Neural Information Processing Systems (NIPS)*, pages 2266–2276, 2017.

[21] Sanjay Kariyappa and Moinuddin K Qureshi. Improving adversarial robustness of ensembles with diversity training. *arXiv preprint arXiv:1901.09981*, 2019.

[22] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. Imagenet classification with deep convolutional neural networks. In *Advances in neural information processing systems*, pages 1097–1105, 2012.

[23] Alexey Kurakin, Ian Goodfellow, and Samy Bengio. Adversarial examples in the physical world. *arXiv preprint arXiv:1607.02533*, 2016.

[24] Alexey Kurakin, Ian Goodfellow, and Samy Bengio. Adversarial machine learning at scale. *arXiv preprint arXiv:1611.01236*, 2016.

[25] Alexey Kurakin, Ian Goodfellow, Samy Bengio, Yinpeng Dong, Fangzhou Liao, Ming Liang, Tianyu Pang, Jun Zhu, Xiaolin Hu, Cihang Xie, et al. Adversarial attacks and defences competition. In *The NIPS'17 Competition: Building Intelligent Systems*, pages 195–231. Springer, 2018.

[26] 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 (ICLR)*, 2018.

[27] Olvi L Mangasarian. *Nonlinear programming*. SIAM, 1994.

[28] Olvi L Mangasarian. Arbitrary-norm separating plane. *Operations Research Letters*, 24(1-2):15–23, 1999.

[29] Dongyu Meng and Hao Chen. Magnet: a two-pronged defense against adversarial examples. In *Proceedings of the 2017 ACM SIGSAC Conference on Computer and Communications Security*, pages 135–147. ACM, 2017.

[30] Seyed-Mohsen Moosavi-Dezfooli, Alhussein Fawzi, Omar Fawzi, and Pascal Frossard. Universal adversarial perturbations. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1765–1773, 2017.

[31] 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*, pages 2574–2582, 2016.

[32] Konda Reddy Mopuri, Utsav Garg, and R Venkatesh Babu. Fast feature fool: A data independent approach to universal adversarial perturbations. In *Proceedings of the British Machine Vision Conference (BMVC)*, 2017.

[33] Tianyu Pang, Kun Xu, Chao Du, Ning Chen, and Jun Zhu. Improving adversarial robustness via promoting ensemble diversity. *arXiv preprint arXiv:1901.08846*, 2019.

[34] Nicolas Papernot, Patrick McDaniel, Somesh Jha, Matt Fredrikson, Z Berkay Celik, and Ananthram Swami. The limitations of deep learning in adversarial settings. In *Security and Privacy (EuroS&P), 2016 IEEE European Symposium on*, pages 372–387. IEEE, 2016.

[35] Nicolas Papernot, Patrick McDaniel, Xi Wu, Somesh Jha, and Ananthram Swami. Distillation as a defense to adversarial perturbations against deep neural networks. In *2016 IEEE Symposium on Security and Privacy (SP)*, pages 582–597. IEEE, 2016.

[36] Andrew Slavin Ross and Finale Doshi-Velez. Improving the adversarial robustness and interpretability of deep neural networks by regularizing their input gradients. In *Thirty-second AAAI conference on artificial intelligence*, 2018.

[37] Pouya Samangouei, Maya Kabkab, and Rama Chellappa. Defense-GAN: Protecting classifiers against adversarial attacks using generative models. In *International Conference on Learning Representations (ICLR)*, 2018.

[38] 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.

[39] Yusuke Tsuzuku, Issei Sato, and Masashi Sugiyama. Lipschitz-margin training: Scalable certification of perturbation invariance for deep neural networks. In *Advances in Neural Information Processing Systems*, pages 6541–6550, 2018.

[40] Weilin Xu, David Evans, and Yanjun Qi. Feature squeezing: Detecting adversarial examples in deep neural networks. *arXiv preprint arXiv:1704.01155*, 2017.

[41] Sijie Yan, Yuanjun Xiong, and Dahua Lin. Spatial temporal graph convolutional networks for skeleton-based action recognition. In *Thirty-Second AAAI Conference on Artificial Intelligence*, 2018.

[42] Ziang Yan, Yiwen Guo, and Changshui Zhang. Deep defense: Training dnns with improved adversarial robustness. In *Advances in Neural Information Processing Systems*, pages 419–428, 2018.

[43] Xiaoyong Yuan, Pan He, Qile Zhu, and Xiaolin Li. Adversarial examples: Attacks and defenses for deep learning. *IEEE transactions on neural networks and learning systems*, 2019.

[44] Zhihao Zheng and Pengyu Hong. Robust detection of adversarial attacks by modeling the intrinsic properties of deep neural networks. In *Advances in Neural Information Processing Systems*, pages 7913–7922, 2018.