Learning to Generate Noise for Multi-Attack Robustness

Divyam Madaan [1] **Jinwoo Shin** [2 3] **Sung Ju Hwang** [1 3 4]

Abstract

Adversarial learning has emerged as one of the successful techniques to circumvent the susceptibility of existing methods against adversarial perturbations. However, the majority of existing defense methods are tailored to defend against a single category of adversarial perturbation (e.g. ℓ_∞-attack). In safety-critical applications, this makes these methods extraneous as the attacker can adopt diverse adversaries to deceive the system. Moreover, training on multiple perturbations simultaneously significantly increases the computational overhead during training. To address these challenges, we propose a novel meta-learning framework that explicitly learns to generate noise to improve the model's robustness against multiple types of attacks. Its key component is *Meta Noise Generator (MNG)* that outputs optimal noise to stochastically perturb a given sample, such that it helps lower the error on diverse adversarial perturbations. By utilizing samples generated by MNG, we train a model by enforcing the label consistency across multiple perturbations. We validate the robustness of models trained by our scheme on various datasets and against a wide variety of perturbations, demonstrating that it significantly outperforms the baselines across multiple perturbations with a marginal computational cost.

1. Introduction

Deep neural networks have demonstrated enormous success on multiple benchmark applications (Amodei et al., 2016; Devlin et al., 2018), by achieving super-human performance on certain tasks. However, to deploy them to safety-critical applications (Shen et al., 2017; Chen et al., 2015; Mao et al., 2019), we need to ensure that the model is *robust* as well as *accurate*, since incorrect predictions may lead to severe consequences. Notably, it is well-known that the existing neural networks are highly susceptible to *adversarial examples* (Szegedy et al., 2013), which are carefully crafted image perturbations that are imperceptible to humans but derail the predictions of these otherwise accurate networks.

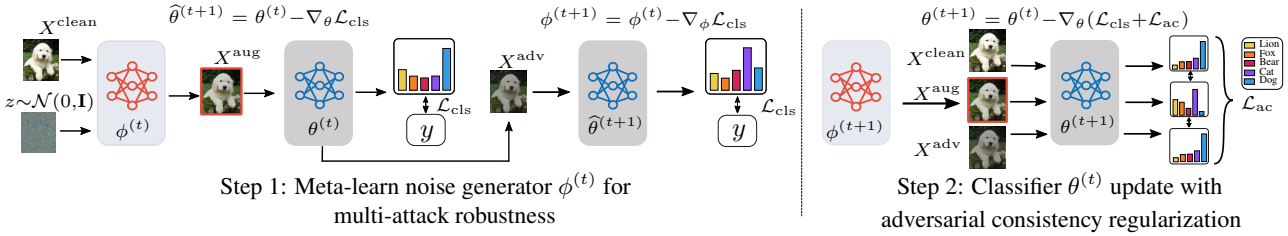
The emergence of adversarial examples has received significant attention in the research community, and several empirical (Madry et al., 2017; Dhillon et al., 2018; Song et al., 2018; Zhang et al., 2019; Carmon et al., 2019; Pang et al., 2020) and certified (Wong & Kolter, 2017; Raghunathan et al., 2018; Cohen et al., 2019) defense mechanisms have been proposed to circumvent this phenomenon. However, despite a large literature to improve upon the robustness of neural networks, most of the existing defenses leverage the knowledge of the adversaries and are based on the assumption of only a single type of adversarial perturbation. Consequently, many of the proposed defenses were circumvented by stronger attacks (Carlini & Wagner, 2017; Athalye et al., 2018; Uesato et al., 2018; Tramer et al., 2020).

Meanwhile, several recent works (Schott et al., 2018; Tramèr & Boneh, 2019) have demonstrated the vulnerability of existing defense methods against multiple perturbations. For the desired multi-attack robustness, Tramèr & Boneh (2019); Maini et al. (2020) proposed various strategies to aggregate multiple perturbations during training. However, training with multiple perturbations comes at an additional cost; it increases the training cost by a factor of four over adversarial training, which is already an order of magnitude more costly than standard training. This slowdown factor hinders the research progress of robustness against multiple perturbations due to the large computation overhead incurred during training. Some recent works reduce this cost by reducing the complexity of generating adversarial examples (Shafahi et al., 2019; Wong et al., 2020), however, they are limited to ℓ_∞ adversarial training.

To address the drawbacks of existing methods, we propose a novel training scheme, *Meta Noise Generator with Adversarial Consistency (MNG-AC)*, which learns instance-dependent noise to minimize the adversarial loss across multiple perturbations while enforcing label consistency between them, as illustrated in Figure 1 and explained in details below.

[1]School of Computing, KAIST, South Korea [2]School of Electrical Engineering, KAIST, South Korea [3]Graduate School of AI, KAIST, South Korea [4]AITRICS, South Korea. Correspondence to: Divyam Madaan <dmadaan@kaist.ac.kr>.

Proceedings of the 38^{th} International Conference on Machine Learning, PMLR 139, 2021.



Step 1: Meta-learn noise generator $\phi^{(t)}$ for multi-attack robustness

Step 2: Classifier $\theta^{(t)}$ update with adversarial consistency regularization

Figure 1. **Overview of Meta-Noise Generator with Adversarial Consistency (MNG-AC).** The generator $\phi^{(t)}$ takes stochastic noise and input X^{clean} to generate the noise-augmented sample X^{aug}, which is used for a temporary update of the classifier to increase the influence of the augmented examples. The generator is learnt via meta-learning by minimizing the loss on adversarial examples generated from an attack sampled uniformly from the perturbation set. The classifier $\theta^{(t)}$ then minimizes the stochastic adversarial classification loss $\mathcal{L}_{\text{cls}}$ and the adversarial consistency loss $\mathcal{L}_{\text{ac}}$.

First, we tackle the heavy computational overhead incurred by multi-perturbation training by proposing *Stochastic Adversarial Training (SAT)*, that uniformly samples from a distribution of perturbations during training, which significantly accelerates training for multiple perturbations[1]. Then, based on the assumption that the model should output the same predictions for different perturbations of the same image, we introduce *Adversarial Consistency (AC)* loss that enforces label consistency across multiple perturbations. Finally, motivated by the noise regularization techniques (Huang et al., 2016; Srivastava et al., 2014; Noh et al., 2017; Lee et al., 2020) which target generalization, we formulate a *Meta Noise Generator (MNG)* that learns to stochastically perturb a given sample in a meta-learning framework to explicitly improve the generalization and label consistency across multiple attacks. In particular, *Meta Noise Generator with Adversarial Consistency (MNG-AC)* utilizes the generated samples to enforce label consistency across the generated samples from MNG, adversarial samples, and clean samples. Consequently, it increases the smoothness of the model (see Figure 3) and pushes the decision boundary away from the data (see Table 5) to improve the robustness across multiple perturbations.

We extensively validate the robustness and computational efficiency of our proposed method by evaluating it on state-of-the-art attack methods and comparing it against existing state-of-the-art single and multi-perturbation adversarial defense methods on multiple benchmark datasets (CIFAR-10, SVHN, and Tiny-ImageNet dataset). The experimental results show that our method obtains significantly superior performance over all the baseline methods trained with multiple adversarial perturbations, generalizes to diverse perturbations, and substantially reduces the computational cost incurred by training with multiple adversarial perturbations. m In summary, the contributions of our paper are as follows:

[1]By a factor of four on a single machine with four GeForce RTX 2080Ti on CIFAR-10 and SVHN dataset using Wide ResNet 28-10 (Zagoruyko & Komodakis, 2016) architecture.

- We introduce *Adversarial Consistency (AC)* loss that enforces label consistency across multiple perturbations to enforce smooth and robust networks.
- We formulate *Meta-Noise Generator (MNG)* that explicitly meta-learns an input-dependent noise generator, such that it outputs stochastic noise distribution to improve the model's robustness and adversarial consistency across multiple types of adversarial perturbations.
- We validate our proposed method on various datasets against diverse benchmark adversarial attacks, on which it achieves state-of-the-art performance, highlighting its practical impact.

We release our code with the pre-trained models for reproducing all the experiments at `https://github.com/divyam3897/MNG_AC`.

2. Related work

Robustness against single adversarial perturbation. In the past few years, multiple defenses have been proposed to defend against a single type of attack (Madry et al., 2017; Xiao et al., 2020; Zhang et al., 2019; Carmon et al., 2019; Madaan et al., 2020; Wu et al., 2020) and have been consequently circumvented by stronger attacks (Athalye et al., 2018; Brendel et al., 2018; Tramer et al., 2020; Croce & Hein, 2020). Adversarial-training based defenses (Madry et al., 2017; Zhang et al., 2019; Carmon et al., 2019; Madaan et al., 2020; Wu et al., 2020) have been the only exceptions that have withstood the intense scrutiny and have provided empirical gains in adversarial robustness. Recently, Croce & Hein (2021) proposed ℓ_1-APGD adversarial training to achieve better performance than MNG-AC on ℓ_1-attack. However, we believe that the comparison is unfair, as MNG-AC does not target ℓ_1-norm robustness only and its lower performance on ℓ_1-norm is due to the trade-off between multiple perturbations (Schott et al., 2018; Tramèr & Boneh,

2019). Moreover, all these prior single perturbation defenses are restricted to a single ℓ_p-threat model and are not capable to defend against multiple perturbations simultaneously.

Robustness against multiple adversarial perturbations. Schott et al. (2018) demonstrated that ℓ_∞ adversarial training is highly susceptible to ℓ_0/ℓ_2-norm adversarial perturbations and used multiple VAEs to defend against multiple perturbations on the MNIST dataset. However, it was not scalable and limited to the MNIST dataset. Tramèr & Boneh (2019) investigated the theoretical/empirical trade-offs between multiple perturbations and introduced adversarial training with worst/average perturbations to defend against multiple perturbations. Maini et al. (2020) incorporated multiple perturbations into a single adversary to maximize the adversarial loss. However, computing all the perturbations is impractical for multiple perturbations and large scale datasets. On the other hand, our proposed framework overcomes this limitation, with improved performance over these methods and has a negligible increase in training cost over multi-perturbation adversarial training.

Generative models for adversarial robustness. There have been various attempts that leverage the representative power of generative models to improve model robustness. Samangouei et al. (2018); Jalal et al. (2017) project an image onto the generator manifold, which is then classified by the discriminator. Song et al. (2018) uses the sensitivity of generative models to defend against a single perturbation. Yin et al. (2020) proposed a detection method based on input space partitioning. However, Samangouei et al. (2018); Jalal et al. (2017); Song et al. (2018) were shown to be ineffective by stronger attacks (Carlini & Wagner, 2017; Athalye et al., 2018). In contrast to learning the generative model to model the adversarial examples, we meta-learn the generator to explicitly learn an input-dependent optimal noise distribution to lower adversarial error across multiple perturbations, that does not necessarily correspond to any of the attack perturbations.

3. Robustness against multiple perturbations

We first briefly review single/multi-perturbation adversarial training and introduce *Stochastic Adversarial Training (SAT)* to reduce the computational cost incurred by training with multiple perturbations. We consider a dataset $\mathcal{D}$ over observations $x \in \mathbb{R}^d$ and labels $y \in \mathbb{R}^C$ with C classes. Let $f_\theta : \mathbb{R}^d \to \mathbb{R}^C$ be a classifier with parameters θ and classification loss $\mathcal{L}_{\text{cls}}$. Given an attack procedure $\mathcal{A}(x)$ with norm-ball $\mathcal{B}(x, \varepsilon)$ around x with radius ε for each example, which introduces a perturbation δ, we let $x^{\text{adv}} = x + \delta$ denote the corresponding adversarial examples. We consider the ℓ_p norm attacks and adopt the projected-gradient descent (PGD) (Madry et al., 2017) for crafting the ℓ_p perturbations:

$$x^{\text{adv}}_{(t+1)} = \underset{\mathcal{B}(x,\varepsilon)}{\text{proj}} \left(x^{\text{adv}}_{(t)} + \arg\max_{||v||} v^T \nabla_{x^{\text{adv}}_{(t)}} \mathcal{L}_{\text{cls}} \left(f_\theta \left(x^{\text{adv}}_{(t)} \right), y \right) \right), \tag{1}$$

where x^{adv}_0 is chosen at random within $\mathcal{B}(x, \varepsilon)$, α is the step size, proj is the projection operator projecting the input onto the norm ball $\mathcal{B}(x, \varepsilon)$, and $x^{\text{adv}}_{(t+1)}$ denotes the adversarial example at the t-th PGD step. We will refer the approximation of the maximum loss by an attack procedure $\mathcal{A}(x)$ as $\max_{\delta \in \mathcal{B}(x,\varepsilon)} \mathcal{L}_{\text{cls}} \left(f_\theta \left(x + \delta \right), y \right) \approx \mathcal{L}_{\text{cls}} \left(f_\theta \left(\mathcal{A} \left(x \right) \right), y \right)$ for the rest of our paper.

Single-perturbation adversarial training. In the standard adversarial training (Kurakin et al., 2016; Madry et al., 2017), the model optimizes the network using a min-max formulation. More formally, the inner maximization generates the adversarial perturbation by maximizing the loss, while the outer minimization minimizes the loss on the generated examples.

$$\min_\theta \mathbb{E}_{(x,y)\sim\mathcal{D}} \mathcal{L}_{\text{cls}} \left(f_\theta \left(\mathcal{A} \left(x \right) \right), y \right). \tag{2}$$

The majority of existing defenses are primarily able to defend against a single category of adversarial perturbation. However, this limits the generalization of these methods to perturbations that are unseen during training (Schott et al., 2018; Tramèr & Boneh, 2019), which has been referred to as *overfitting* on the particular type of training perturbation.

Multi-perturbation adversarial training. Tramèr & Boneh (2019) extended the adversarial training to multiple perturbations by optimizing the outer objective in Eq. (2) on the strongest/union of adversarial perturbations for each input example as follows:

1. **The maximum over all perturbations**: It optimizes the outer objective in Eq. (2) on the strongest adversarial perturbation from the perturbation set.

$$\min_\theta \mathbb{E}_{(x,y)\sim\mathcal{D}} \left[\arg\max_k \mathcal{L}_{\text{cls}} \left(f_\theta \left(\mathcal{A}_k \left(x \right) \right), y \right) \right]. \tag{3}$$

2. **The average over all perturbations:** It optimizes the outer objective in Eq. (2) on the set of n perturbations.

$$\min_\theta \mathbb{E}_{(x,y)\sim\mathcal{D}} \frac{1}{n} \sum_{k=1}^{k=n} \mathcal{L}_{\text{cls}} \left(f_\theta \left(\mathcal{A}_k \left(x \right), y \right) \right). \tag{4}$$

Recently, Maini et al. (2020) proposed "Multi Steepest Descent" (MSD) by incorporating the different perturbations into the direction of steepest descent. However, the practicality of all these methods is limited due to an increased computational overhead for training.

Stochastic Adversarial Training (SAT). To overcome this limitation, we propose Stochastic Adversarial Training to defend against multiple adversarial perturbations. Specifically, we conjecture that it is essential to cover the threat model during training, not utilizing all the perturbations simultaneously. We formulate the threat model as a random attack $\mathcal{A}(x)$ sampled uniformly from a perturbation set S during each episode (or batch) of training which prevents overfitting on a particular adversarial perturbation. In this work, we consider the ℓ_p-bounded perturbation set, and we sample the attack procedure $\mathcal{A}(x)$ from the perturbation set S as follows:

通过随机抽样真的很拉胯耶，我觉得这里可以参考autoattack

$$\begin{aligned} S &= \{\mathcal{A}_1(x), \ldots, \mathcal{A}_n(x)\}, \\ k &\sim \mathrm{Cat}\left((1/n, \ldots, 1/n)\right), \\ \mathcal{A}(x) &= S_k(x), \end{aligned} \tag{5}$$

where Cat is the categorical distribution and n is the number of attacks in the perturbation set S. Our proposed SAT optimizes the outer objective in Eq. (2) using the sampled attack procedure $\mathcal{A}(x)$ and is a drastic simplification of the average strategy in Eq. (4), which makes it highly efficient for multiple perturbations. It is important to note that unlike the average and max strategy SAT can be applied to any perturbation set with a constant cost and it promotes generalization and convergence (due to its stochasticity) by preventing over-fitting on a single type of perturbation.

4. Learning to generate noise for multi-attack robustness

In this section, we introduce our framework MNG-AC, which leverages an *adversarial consistency loss (AC)* and a *meta-noise generator (MNG)* to help the model generalize to multiple perturbations. Let x_θ^{adv} be the adversarial examples generated from the network f_θ for a uniformly sampled attack $\mathcal{A}(x)$ with norm-ball $\mathcal{B}(x, \varepsilon)$ from a perturbation set S and $g_\phi : \mathbb{R}^d \to \mathbb{R}^d$ denotes the generator with parameters ϕ. We input $z \sim \mathcal{N}(0, \mathbf{I})$ to our generator jointly with the clean examples x to generate the noise-augmented samples x_ϕ^{aug} projected on the same norm-ball $\mathcal{B}(x, \varepsilon)$ as:

$$x_\phi^{\mathrm{aug}} = \underset{\mathcal{B}(x,\varepsilon)}{\mathrm{proj}} \left(x + g_\phi(z, x)\right), \quad \text{where } z \sim \mathcal{N}(0, \mathbf{I}). \tag{6}$$

The total loss function $\mathcal{L}_{\mathrm{total}}$ for the classifier consists exclusively of two terms: SAT classification loss and an adversarial consistency loss:

$$\begin{aligned} \mathcal{L}_{\mathrm{total}} = \frac{1}{B} \sum_{i=1}^{B} & \underbrace{\mathcal{L}_{\mathrm{cls}}\left(\theta \mid x_\theta^{\mathrm{adv}}(i), y(i)\right)}_{\text{SAT classification loss}} \\ & + \underbrace{\beta \cdot \mathcal{L}_{\mathrm{ac}}\left(p^{\mathrm{clean}}(i); p^{\mathrm{adv}}(i); p^{\mathrm{aug}}(i)\right)}_{\text{adversarial consistency loss}}, \end{aligned} \tag{7}$$

Algorithm 1 Algorithm for MNG-AC

input Dataset $\mathcal{D}$, T epochs, batch size B, perturbation set S, classifier θ and noise-generator ϕ.
output Final model parameters θ.

for $t = \{1, \ldots, T\}$ **do**
  Sample mini-batch of size B.
  Sample an attack procedure $\mathcal{A}(x)$ from S (Eq. (5)).
  Generate adversarial examples for $\mathcal{A}(x)$ (Eq. (1)).
  Generate x_ϕ^{aug} using $\phi^{(t)}$ by Eq. (6).
  Temporary update of $\theta^{(t)}$ using Eq. (9).
  Update the parameters $\phi^{(t)}$ of MNG by Eq. (10).
  Generate x_ϕ^{aug} using $\phi^{(t+1)}$ by Eq. (6).
  Update $\theta^{(t)}$ by Eq. (11).
end for

where B is the batch-size, β is the hyper-parameter determining the strength of the AC loss denoted by $\mathcal{L}_{\mathrm{ac}}$ and $p^{\mathrm{clean}}, p^{\mathrm{adv}}, p^{\mathrm{aug}}$ represent the posterior distributions $p(y \mid x^{\mathrm{clean}}), p(y \mid x_\theta^{\mathrm{adv}}), p(y \mid x_\phi^{\mathrm{aug}})$ computed using the softmax function on the logits for x^{clean}, x^{adv}, and x^{aug} respectively. Specifically, $\mathcal{L}_{\mathrm{ac}}$ represents the Jensen-Shannon Divergence (JSD) among the posterior distributions:

$$\begin{aligned} \mathcal{L}_{\mathrm{ac}} = \frac{1}{3} \big(& D_{\mathrm{KL}}(p^{\mathrm{clean}} \parallel M) + D_{\mathrm{KL}}(p^{\mathrm{adv}} \parallel M) \\ & + D_{\mathrm{KL}}(p^{\mathrm{aug}} \parallel M)\big), \end{aligned} \tag{8}$$

where $M = \left(p^{\mathrm{clean}} + p^{\mathrm{adv}} + p^{\mathrm{aug}}\right)/3$. Consequently, $\mathcal{L}_{\mathrm{ac}}$ enforces stability and insensitivity across a diverse range of inputs based on the assumption that the classifier should output similar predictions when fed perturbed versions of the same image.

Recently, Rusak et al. (2020) formulated an adversarial noise generator to learn the adversarial noise to improve the robustness on common corruptions. However, our goal is different; the robustness against multiple adversarial attacks is a much more challenging task than that against common corruptions. To generate the augmented samples for our purpose, MNG meta-learns (Thrun & Pratt, 1998; Finn et al., 2017) the parameters ϕ of the noise generator g_ϕ to generate an input-dependent noise distribution to alleviate the issue of generalization across multiple adversaries. The standard approach to train our adversarial classifier jointly with MNG is to use bi-level optimization (Finn et al., 2017). However, bi-level optimization for adversarial training would be computationally expensive.

To tackle this challenge, we adopt an online approximation (Ren et al., 2018; Shu et al., 2019) to update θ and ϕ using a single-optimization loop. We alternatively update the parameters θ of the classifier with the parameters ϕ of MNG using the following training scheme:

1. **Temporary model update on augmented samples.** First, we update θ to minimize $\mathcal{L}_{\text{cls}}(\theta \mid x_{\phi}^{\text{aug}}, y)$, which ensures the learning of the classifier using the generated samples constructed by MNG. It explicitly increases the influence of the noise-augmented samples on the classifier. More specifically, for a learning rate α, projection operator proj, current $\theta^{(t)}$ moves along the following descent direction:

$$\widehat{\theta}^{(t)} = \theta^{(t)} - \alpha \cdot \frac{1}{B} \sum_{i=1}^{B} \nabla_{\theta} \mathcal{L}_{\text{cls}} \left(\theta^{(t)} \mid x_{\phi}^{\text{aug}}(i), y(i) \right). \tag{9}$$

2. **Update generator parameters.** After receiving feedback from the classifier, we adapt ϕ to minimize the SAT loss (adversarial loss on the uniformly sampled attack $\mathcal{A}(x)$). In particular, ϕ facilitates the classifier parameters θ in the next step with the update step[2]:

$$\phi^{(t+1)} = \phi^{(t)} - \alpha \cdot \frac{1}{B} \sum_{i=1}^{B} \nabla_{\phi} \mathcal{L}_{\text{cls}} \left(\widehat{\theta}^{(t)} \mid x_{\theta}^{\text{adv}}(i), y(i) \right). \tag{10}$$

3. **Update model parameters.** Finally, we update $\theta^{(t)}$ to minimize loss from Eq. (7). This step explicitly models the adaptation of adversarial model parameters in the presence of the noise-augmented data using the adversarial consistency loss:

$$\theta^{(t+1)} = \theta^{(t)} - \frac{1}{B} \sum_{i=1}^{B} \Big(\mathcal{L}_{\text{cls}} \left(\theta \mid x_{\theta}^{\text{adv}}(i), y(i) \right) + \beta \cdot \mathcal{L}_{\text{ac}} \left(p^{\text{clean}}(i); p^{\text{adv}}(i); p^{\text{aug}}(i) \right) \Big). \tag{11}$$

To summarize, MNG-AC utilizes perturbation sampling to generate the adversarial examples. The generator perturbs the clean examples in a meta-learning framework to explicitly lower the loss on the generated adversarial examples. Lastly, the adversarial classifier utilizes the generated samples, adversarial samples and clean samples to optimize the adversarial classification and adversarial consistency loss.

Intuition behind our framework. Unlike existing defenses that aim for robustness against a single perturbation, our proposed approach targets for a realistic scenario of robustness against multiple perturbations. Our motivation is that meta-learning the noise distribution to minimize the SAT loss allows to learn the optimal noise to improve multi-perturbation generalization. Based on the assumption that the model should output similar predictions for perturbed versions of the same image, we enforce the AC loss, which enforces the label consistency across multiple perturbations.

[2]Note that ϕ is a variable in this case, which makes the loss in Eq. 10 a function of ϕ, allowing the the gradients' computation.

5. Experiments

5.1. Experimental setup

Datasets. We evaluate on multiple benchmark datasets:

1. **CIFAR-10.** This dataset (Krizhevsky, 2012) contains 60,000 images with 5,000 images for training and 1,000 images for test for each class. Each image is sized 32×32, we use the Wide ResNet 28-10 architecture (Zagoruyko & Komodakis, 2016) as a base network for this dataset.

2. **SVHN.** This dataset (Netzer et al., 2011) contains 73257 training and 26032 testing images of digits and numbers in natural scene images containing ten-digit classes. Each image is sized 32×32, and we use the Wide ResNet 28-10 architecture similar to the CIFAR-10 dataset as the base network for this dataset.

3. **Tiny-ImageNet.** This dataset[3] is a subset of ImageNet (Russakovsky et al., 2015) dataset, consisting of 500, 50, and 50 images for training, validation, and test dataset, respectively. This dataset contains 64×64 size images from 200 classes, we use ResNet50 (He et al., 2016) as a base network for this dataset.

Baselines and our model. We compare MNG-AC with the standard network (Nat) and single-perturbation baselines including Madry et al. (2017) (Adv_{p}) for ℓ_{∞}, ℓ_1, and ℓ_2 norm, TRADES_{∞} (Zhang et al., 2019) for ℓ_{∞} norm. We consider state-of-the-art multi-perturbation baselines: namely, we consider Adversarial training with the maximum (see Eq. (3)) (Adv_{max}), average (Adv_{avg}) (Tramèr & Boneh, 2019) (see Eq. (4)) strategies, and Multiple steepest descent (MSD) (Maini et al., 2020). We additionally consider Adversarial Noise Training (Rusak et al., 2020) that learns adversarial noise to improve robustness against common corruptions (Hendrycks & Dietterich, 2019).

Evaluation setup. We have evaluated the proposed defense scheme and baselines against perturbations generated by state-of-the-art attack methods. We validate the clean accuracy ($\text{Acc}_{\text{clean}}$), the worst ($\text{Acc}_{\text{adv}}^{\text{union}}$) and average ($\text{Acc}_{\text{adv}}^{\text{avg}}$) adversarial accuracy across all the perturbation sets for all the models. For ℓ_{∞} attacks, we use PGD (Madry et al., 2017), Brendel and Bethge (Brendel et al., 2019), and AutoAttack (Croce & Hein, 2020). For ℓ_2 attacks, we use CarliniWagner (Carlini & Wagner, 2017), PGD (Madry et al., 2017), Brendel and Bethge (Brendel et al., 2019), and AutoAttack (Croce & Hein, 2020). For ℓ_1 attacks, we use SLIDE (Tramèr & Boneh, 2019), Salt and pepper (Rauber et al., 2017), and EAD attack (Chen et al., 2018). We provide a detailed description of the training and evaluation setup in Appendix A.

[3]https://tiny-imagenet.herokuapp.com/

Table 1. Comparison of robustness against multiple types of perturbations. All the values are measured by computing mean, and standard deviation across three trials, the best and second-best results are highlighted in **bold** and underline respectively. Time denotes the training time in hours. For CIFAR-10 and SVHN, we use $\varepsilon = \{\frac{8}{255}, \frac{2000}{255}, \frac{128}{255}\}$ for ℓ_∞, ℓ_1, and ℓ_2 attacks respectively. For Tiny-ImageNet, we use $\varepsilon = \{\frac{4}{255}, \frac{2000}{255}, \frac{80}{255}\}$ for ℓ_∞, ℓ_1, and ℓ_2 attacks respectively. We report the worst-case accuracy for all the attacks and defer the breakdown of all attacks to Appendix B.

	Model	$\text{Acc}_{\text{clean}}$	ℓ_∞	ℓ_1	ℓ_2	$\text{Acc}^{\text{union}}_{\text{adv}}$	$\text{Acc}^{\text{avg}}_{\text{adv}}$	Time (h)
CIFAR-10	Nat (Zagoruyko & Komodakis, 2016)	**94.7± 0.1**	0.0± 0.0	0.0± 0.0	0.4± 0.2	0.0± 0.0	0.0± 0.0	**0.4**
	Adv_∞ (Madry et al., 2017)	86.8± 0.1	44.9± 0.7	26.2± 0.4	55.0± 0.9	25.6± 0.6	41.9± 0.6	4.5
	Adv_1	93.3± 0.4	0.0± 0.0	**80.7± 0.7**	0.0± 0.0	0.0± 0.00	26.8± 0.6	8.1
	Adv_2	89.4± 0.2	28.8± 1.3	54.2± 0.4	**65.8± 0.3**	28.6± 1.4	49.6± 0.3	3.7
	TRADES_∞ (Zhang et al., 2019)	84.7± 0.3	**48.9± 0.7**	32.3± 1.0	57.8± 0.6	31.5± 1.2	46.3± 0.7	5.2
	Adv_{avg} (Tramèr & Boneh, 2019)	86.0± 0.1	34.1± 0.5	61.3± 0.6	65.7± 0.4	34.1± 0.1	53.7± 0.3	16.9
	Adv_{max} (Tramèr & Boneh, 2019)	84.2± 0.1	39.9± 0.5	57.9± 0.7	64.5± 0.1	39.7± 0.5	54.1± 0.4	16.3
	MSD (Maini et al., 2020)	82.7± 0.1	43.5± 0.5	54.3± 0.4	63.1± 0.5	**42.7± 0.5**	53.6± 0.2	16.7
	ANT (Rusak et al., 2020)	94.6± 0.0	0.0± 0.0	0.0± 0.0	0.0± 0.0	0.0± 0.0	0.0± 0.0	0.7
	MNG-AC (Ours)	81.7± 0.3	41.4± 0.7	65.4± 0.3	65.2± 0.5	41.4± 0.7	**57.2± 0.4**	8.4
SVHN	Nat (Zagoruyko & Komodakis, 2016)	**96.8± 0.1**	0.0± 0.0	9.4± 0.5	3.8± 0.7	0.0± 0.0	4.5± 0.2	**0.6**
	Adv_∞ (Madry et al., 2017)	92.8± 0.2	46.2± 0.6	8.2± 0.9	30.2± 0.5	8.1± 0.9	28.3± 0.1	6.2
	Adv_1	92.4± 0.9	0.0± 0.0	**77.2± 2.9**	0.0± 0.0	0.0± 0.0	25.7± 1.0	11.8
	Adv_2	93.0± 0.1	21.7± 0.4	44.7± 0.5	62.9± 0.2	21.0± 0.4	43.1± 0.3	6.1
	TRADES_∞ (Zhang et al., 2019)	93.9± 0.1	**49.9± 1.7**	4.2± 0.4	26.7± 2.0	4.1± 0.4	26.9± 1.1	7.9
	Adv_{avg}(Tramèr & Boneh, 2019)	91.6± 0.3	21.5± 2.7	61.2± 4.1	56.1± 2.3	20.4± 2.7	45.9± 0.9	24.1
	Adv_{max} (Tramèr & Boneh, 2019)	86.9± 0.3	28.8± 0.2	48.9± 0.9	56.3± 0.8	28.8± 0.2	44.7± 0.4	22.7
	MSD (Maini et al., 2020)	81.8± 0.3	34.1± 0.3	43.4± 0.5	54.1± 0.2	34.1± 0.3	44.0± 0.1	23.7
	ANT (Rusak et al., 2020)	96.7± 0.0	0.2± 0.0	15.6± 0.6	7.7± 0.6	0.0± 0.0	7.8± 0.4	1.1
	MNG-AC (Ours)	92.6± 0.2	34.2± 1.0	71.3± 1.7	**66.7 ± 0.9**	**34.2± 1.0**	**57.4± 0.4**	11.9
Tiny-ImageNet	Nat (He et al., 2016)	**62.8± 0.4**	0.0± 0.0	2.7± 0.3	12.6± 0.8	0.0± 0.0	5.1± 0.4	**0.9**
	Adv_∞ (Madry et al., 2017)	54.2± 0.4	29.6± 0.1	38.2± 0.7	42.5± 0.6	29.4± 0.1	36.7± 0.4	4.3
	Adv_1	57.8± 0.2	10.5± 0.7	44.6± 0.1	41.9± 0.0	10.1± 0.7	32.2± 0.4	12.9
	Adv_2	59.5± 0.1	5.2± 0.6	44.1± 0.4	**44.9± 0.1**	5.2± 0.6	31.7± 0.5	3.7
	TRADES_∞ (Zhang et al., 2019)	48.2± 0.2	28.7± 0.9	33.2± 0.4	35.8± 0.7	26.1± 0.9	32.8± 0.1	5.8
	Adv_{avg} (Tramèr & Boneh, 2019)	56.0± 0.0	23.7± 0.2	43.3± 0.5	44.6± 1.8	23.6± 0.3	37.2± 0.2	26.8
	Adv_{max} (Tramèr & Boneh, 2019)	53.5± 0.0	**29.8± 0.1**	39.5± 0.4	42.4± 1.0	**29.8± 0.3**	37.3± 0.4	20.8
	MSD (Maini et al., 2020)	45.5± 0.1	29.4± 0.3	35.3± 0.8	33.9± 0.8	29.4± 0.3	33.5± 0.6	25.2
	ANT (Rusak et al., 2020)	62.8± 0.0	0.2± 0.0	3.4± 0.1	13.4± 0.3	0.0± 0.0	5.6± 0.1	1.2
	MNG-AC (Ours)	53.1± 0.3	28.1± 0.7	**45.1± 0.5**	44.4± 0.1	28.1± 0.8	**39.1± 0.6**	10.4

5.2. Comparison of robustness against multiple perturbations

Results with CIFAR-10 dataset. Table 1 shows the experimental results for the CIFAR-10 dataset. It is evident from the results that MNG-AC achieves a relative improvement of $\sim 31\%$ and $\sim 24\%$ on the $\text{Acc}^{\text{union}}_{\text{adv}}$ and $\text{Acc}^{\text{avg}}_{\text{adv}}$ metric over the best single-perturbation adversarial training methods. Furthermore, MNG-AC achieves a relative improvement of $\sim 6\%$ on the $\text{Acc}^{\text{avg}}_{\text{adv}}$ metric over the state-of-the-art methods trained on multiple perturbations. Moreover, MNG-AC achieves $\sim 50\%$ reduction in training time compared to all the multi-perturbations training baselines. It is also worth mentioning that, MNG-AC also shows an improvement over Adv_{max}, which is fundamentally designed to defend against the worst perturbation in the perturbation set.

Results with SVHN dataset. The results for the SVHN dataset are shown in Table 1. We make the following observations from the results: (1) Firstly, MNG-AC significantly outperforms Adv_{avg}, Adv_{max} by $\sim 67.6\%$ and $\sim 18.8\%$ on the $\text{Acc}^{\text{union}}_{\text{adv}}$ metric respectively. Furthermore, it achieves an absolute improvement of $+12\%$ on the $\text{Acc}^{\text{avg}}_{\text{adv}}$ metric over the multi-perturbation adversarial training baselines.

Table 2. Robustness evaluation using semi-supervised learning against multiple perturbations. All the values are measured by computing mean, and standard deviation across three trials, the best results are highlighted in **bold**. Due to the computational constraints, we use efficient training techniques (Smith, 2017; Wong et al., 2020) for training all the methods, which result in a slightly lower performance compared to the results in the original paper (Carmon et al., 2019).

	Model	$\text{Acc}_{\text{clean}}$	ℓ_∞	ℓ_1	ℓ_2	$\text{Acc}_{\text{adv}}^{\text{union}}$	$\text{Acc}_{\text{adv}}^{\text{avg}}$	Time (h)
CIFAR-10	RST_∞ (Carmon et al., 2019)	**88.9± 0.2**	**54.9± 1.8**	36.0± 0.9	59.5± 0.2	35.7± 0.6	50.1± 0.8	73.5
	MNG-AC (Ours)	81.7± 0.3	41.4± 0.7	65.4± 0.3	65.2± 0.5	41.4± 0.7	57.2± 0.4	8.4
	MNG-AC + RST (Ours)	88.7± 0.2	47.2± 0.8	**73.8± 0.7**	**73.7± 0.2**	**47.2± 0.7**	**64.9± 0.3**	78.5
SVHN	RST_∞ (Carmon et al., 2019)	95.6± 0.0	**60.9± 2.0**	3.5± 0.5	28.8± 0.9	3.5± 0.5	31.1± 0.6	81.0
	MNG-AC (Ours)	92.6± 0.2	34.2± 1.0	71.3± 1.7	66.7 ± 0.9	34.2± 1.0	57.4± 0.4	11.9
	MNG-AC + RST (Ours)	**96.3± 0.3**	43.8± 1.5	**78.9± 2.0**	**72.6± 0.2**	**43.8± 1.5**	**65.1± 0.4**	85.0

Table 3. Ablation study analyzing the significance of SAT, Adversarial Consistency loss (AC) and Meta Noise Generator (MNG). The best results are highlighted in **bold**.

	SAT	AC	MNG	$\text{Acc}_{\text{clean}}$	ℓ_∞	ℓ_1	ℓ_2	$\text{Acc}_{\text{adv}}^{\text{union}}$	$\text{Acc}_{\text{adv}}^{\text{avg}}$	Time (h)
CIFAR-10	✓	-	-	**86.6± 0.0**	35.1± 0.5	61.8± 1.1	**66.9± 0.4**	35.0± 0.5	54.6± 0.2	**5.5**
	✓	✓	-	80.3± 0.2	40.6± 0.8	62.0± 0.2	63.5± 0.6	40.6± 0.1	55.4± 0.1	6.8
	✓	✓	✓	81.7± 0.3	**41.4± 0.7**	**65.2± 0.3**	65.4± 0.5	**41.4± 0.7**	**57.2± 0.4**	8.4
SVHN	✓	-	-	92.3± 0.1	26.2± 0.8	64.4± 0.2	63.2± 0.8	26.2± 0.8	51.0± 0.1	**7.6**
	✓	✓	-	92.2± 0.3	31.4± 1.3	65.2± 3.6	63.9± 0.5	31.1± 1.4	53.5± 0.8	8.7
	✓	✓	✓	**92.6± 0.2**	**34.2± 1.0**	**71.3± 1.7**	**66.7 ± 0.9**	**34.2± 1.0**	**57.4± 0.4**	11.9

(2) Additionally, we note that compared to MNG-AC, ANT does not improve the performance across adversarial perturbations as it does not utilize them during training. (3) Interestingly, MNG-AC achieves better performance over the standard ℓ_2 training with comparable training time, which implies that our method leverages the knowledge across multiple perturbations, illustrating the utility of our method over standard adversarial training.

Results with Tiny-ImageNet dataset. We also evaluate our method on Tiny-ImageNet in Table 1 to verify that it performs well on complex datasets. We observe that MNG-AC outperforms the multi-perturbation training baselines and achieves comparable performance to the single-perturbation baselines. Only against ℓ_∞ perturbations, we notice that $\text{Adv}_{\max}$ achieves marginally better performance. We believe this is an artefact of the inherent trade-off across multiple perturbations (Tramèr & Boneh, 2019; Schott et al., 2018). Interestingly, MNG-AC even achieves comparable performance to the single perturbation baselines trained on ℓ_1 and ℓ_2 norm. This demonstrates the effectiveness of MNG in preventing overfitting over a single attack, and it's generalization ability to diverse types of attacks.

Results with semi-supervised learning. The efficiency of MNG-AC allows us to utilize semi-supervised data augmentation techniques (Carmon et al., 2019; Alayrac et al., 2019) for multi-perturbation adversarial training with a marginal increase in computation. In Table 2, we can observe that Robust-Self Training (RST_∞) (Carmon et al., 2019) overfits to ℓ_∞-norm perturbations, while MNG-AC + RST leads to a significant absolute gain of $+11.5\%$ and $+40.3\%$ on the $\text{Acc}_{\text{adv}}^{\text{union}}$ metric on CIFAR-10 and SVHN respectively. Furthermore, MNG-AC + RST also improves the absolute performance on the $\text{Acc}_{\text{adv}}^{\text{avg}}$ metric by $+14.8\%$ and $+34\%$ on CIFAR-10 and SVHN respectively with comparable training time compared to the RST_∞ training.

5.3. Ablation studies

Component analysis. Table 3 dissects the effectiveness of various components in MNG-AC. First, we examine that SAT leads to a $\sim 68\%$ and $\sim 35\%$ relative reduction in training time over multiple perturbations baselines and MNG-AC for both the datasets; however, it does not improve the adversarial robustness. Then, we analyze the impact of our meta-noise generator by injecting random noise $z \sim \mathcal{N}(0, \mathbf{I})$ to the inputs for the generation of augmented samples. We observe that it significantly improves the performance over SAT with a marginal increase in the training time. Furthermore, leveraging MNG, our combined framework MNG-AC achieves consistent improvements over all the baselines, demonstrating the efficacy of our meta-learning scheme to defend against multiple perturbations.

Table 4. Spatial attack evaluation.

Model	CIFAR-10		SVHN	
	Accuracy	Time (h)	Accuracy	Time (h)
Adv_{avg}	**54.2 ± 0.2**	16.9	58.7 ± 1.8	24.1
Adv_{max}	**54.3 ± 0.2**	16.3	54.4 ± 0.4	22.7
MSD	52.5 ± 0.8	16.7	45.9 ± 0.8	23.7
MNG-AC	52.4 ± 0.4	8.4	**65.0 ± 0.7**	11.9
MNG-AC (seen)	**68.9 ± 0.8**	**6.0**	**83.0 ± 1.1**	**10.8**

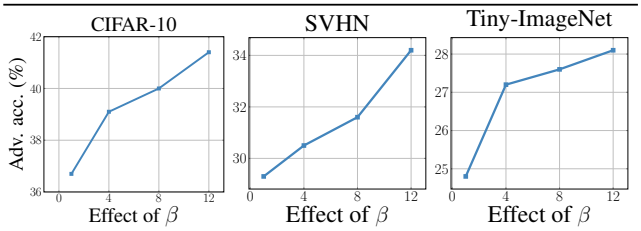


Figure 2. Ablation study on the impact of $\mathcal{L}_{\text{ac}}$ on union robustness ($\text{Acc}_{\text{adv}}^{\text{union}}$) against ℓ_p attacks on various datasets. With an increase in β in Eq. 7, the robustness against the adversarial attacks increases across all the datasets.

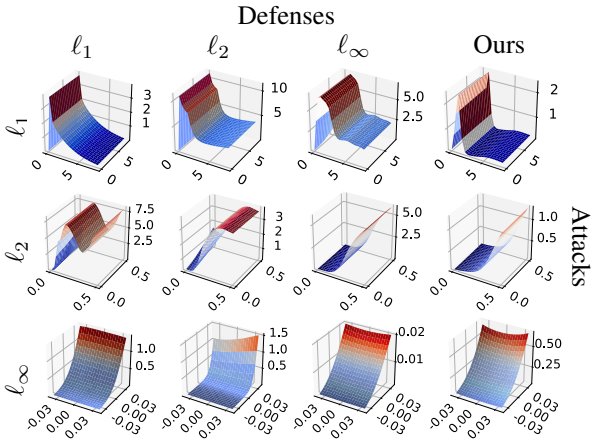
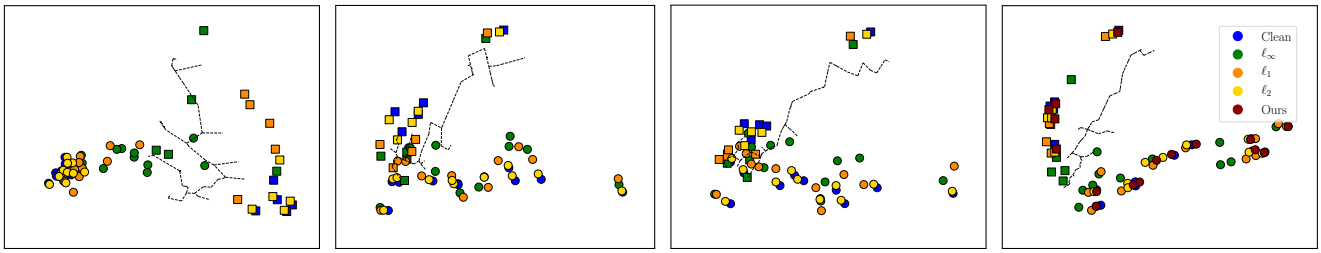


Figure 3. Visualization of the loss landscapes for the ℓ_1, ℓ_2, and ℓ_∞-norm attacks on the CIFAR-10 dataset. The rows represent the attacks and columns represent different defenses.



(a) Adv_{avg} (b) Adv_{max} (c) MSD (d) MNG-AC

Table 5. Visualization of the decision boundary in the penultimate latent-feature space for multi-perturbation methods for SVHN dataset on Wide ResNet 28-10 architecture. The two shapes represent different classes in binary classification.

Effect of hyperparameters. We further analyze the impact of β in our augmentation loss (see Eq. (7)) in Figure 2. In particular, we evaluate the robustness on the $\text{Acc}_{\text{adv}}^{\text{union}}$ metric across all the ℓ_p norm adversarial perturbations for various datasets on Wide ResNet 28-10 architecture. Our results show that as the value of β increases the performance improves on the $\text{Acc}_{\text{adv}}^{\text{union}}$ metric across all the datasets. Specifically, the absolute performance of $\text{Acc}_{\text{adv}}^{\text{union}}$ improves by $\sim 5\%$ on all the datasets with an increase in the weight of adversarial consistency loss, highlighting the efficiency of our AC loss. However, we observe that increasing the weight of the consistency loss decreases the clean accuracy by $\sim 3\%$ for all the datasets (see Figure 4). We believe that the drop in the clean accuracy is an outcome of the trade-off between clean accuracy and robustness as observed in previous works (Tsipras et al., 2019; Zhang et al., 2019), which also holds for adversarial training with multiple perturbations and exploring ways to reduce this trade-off would be an interesting direction for future work.

5.4. Further analysis of our defense

Results with spatial attack. We further validated the flexibility and effectiveness of MNG-AC on unseen spatial attacks (Engstrom et al., 2019) in Table 4. We oberve that MNG-AC largely outperforms the multi-perturbation baselines on the SVHN dataset, and obtains comparable performance to them on CIFAR-10, with significantly smaller training cost. Further, MNG-AC achieves 26.1% and 36.6% relative higher robustness on CIFAR-10 and SVHN respectively when trained jointly with ℓ_p-norms and spatial attack projected on ℓ_2-norm (**MNG-AC (seen)**), which is not feasible for baselines due the prohibitive training cost. Moreover, we want to emphasize that MNG-AC (seen) leads to similar performance on ℓ_p-norms on both the datasets with lower computational cost. Additionally, we evaluate the effectiveness of MNG-AC on CIFAR10-C (Hendrycks & Dietterich, 2019) across five severity levels and unforeseen perturbations (Kang et al., 2019) (Elastic, ℓ_∞-JPEG, ℓ_1-JPEG and ℓ_2-JPEG attacks) on SVHN dataset in the Appendix B.

Visualization of loss landscape. As further qualitative analysis of the effect of MNG-AC, we compare the loss surface of various methods against ℓ_∞, ℓ_1, and ℓ_2 norm attack in Figure 3. We vary the input along a linear space defined by the ℓ_p-norm of the gradient where x and y-axes represent the perturbation added in each direction, and the z-axis represents the loss. We can observe that in most of the instances when trained with a single adversary, the adversary can find a direction orthogonal to that explored during training; for example, ℓ_1 attack results in a non-smooth loss surface for both ℓ_∞ and ℓ_2 adversarial training. On the contrary, MNG-AC achieves smoother loss surface across all types of attacks which suggests that the gradients modelled by our model are closer to the optimum global landscape. See Figure 6 in Appendix B for the the loss landscape on multiple ℓ_p-norm attacks for SVHN dataset.

Visualization of decision boundary. We visualize the learned decision boundary in the penultimate latent-feature space on binary-classification task across multiple ℓ_p-norm attacks in Table 5. In particular, we use use TSNE embedding for obtaining the penultimate latent-features followed by visualizing the decision boundary using the open-source DBPlot library. We observe that MNG-AC obtains the least error against all the attacks compared to the baselines trained on multiple adversarial perturbations. Furthermore, note that the adversarial consistency regularization embeds multiple perturbations onto the same latent space, which pushes them away from the decision boundary that in turn improves the overall robustness. Additionally, Figure 5 in Appendix B provides the visualization of the examples generated by our input-dependent meta-noise generator for CIFAR-10 and SVHN dataset.

6. Conclusion

We tackled the problem of robustness against multiple adversarial perturbations. Existing defense methods are tailored to defend against single adversarial perturbation which is an artificial setting to evaluate in real-life scenarios where the adversary will attack the system in any way possible. To this end, we propose a novel *Meta-Noise Generator (MNG)* that learns to stochastically perturb the clean examples by generating output noise across diverse perturbations. Then we train the model using *Adversarial Consistency (AC)* loss that accounts for label consistency across clean, adversarial, and augmented samples. Additionally, to resolve the problem of computation overhead with conventional adversarial training methods for multiple perturbations, we introduce a *Stochastic Adversarial Training (SAT)* which samples a perturbation from the distribution of perturbations. We believe that our method can be a strong guideline when other researchers pursue similar tasks in the future.

Acknowledgements

We thank the anonymous reviewers for their insightful comments and suggestions. This work was supported by Institute of Information & communications Technology Planning & Evaluation (IITP) grant funded by the Korea government (MSIT) (No.2020-0-00153), Penetration Security Testing of ML Model Vulnerabilities and Defense), Engineering Research Center Program through the National Research Foundation of Korea (NRF) funded by the Korean Government MSIT (NRF-2018R1A5A1059921), Institute of Information & communications Technology Planning & Evaluation (IITP) grant funded by the Korea government (MSIT) (No.2019-0-00075), and Artificial Intelligence Graduate School Program (KAIST). Any opinions, findings, and conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect the views of the funding agencies.

References

Alayrac, J.-B., Uesato, J., Huang, P.-S., Fawzi, A., Stanforth, R., and Kohli, P. Are labels required for improving adversarial robustness? In *NeurIPS*, 2019.

Amodei, D., Ananthanarayanan, S., Anubhai, R., Bai, J., Battenberg, E., Case, C., Casper, J., Catanzaro, B., Cheng, Q., Chen, G., et al. Deep speech 2: End-to-end speech recognition in english and mandarin. In *ICML*, 2016.

Athalye, A., Carlini, N., and Wagner, D. A. Obfuscated gradients give a false sense of security: Circumventing defenses to adversarial examples. In *ICML*, 2018.

Brendel, W., Rauber, J., and Bethge, M. Decision-based adversarial attacks: Reliable attacks against black-box machine learning models. In *ICLR*, 2018.

Brendel, W., Rauber, J., Kümmerer, M., Ustyuzhaninov, I., and Bethge, M. Accurate, reliable and fast robustness evaluation. In *NeurIPS*, 2019.

Carlini, N. and Wagner, D. Towards evaluating the robustness of neural networks. In *2017 ieee symposium on security and privacy (sp)*, 2017.

Carmon, Y., Raghunathan, A., Schmidt, L., Duchi, J. C., and Liang, P. S. Unlabeled data improves adversarial robustness. In *NeurIPS*, 2019.

Chen, C., Seff, A., Kornhauser, A., and Xiao, J. Deepdriving: Learning affordance for direct perception in autonomous driving. In *ICCV*, 2015.

Chen, P.-Y., Sharma, Y., Zhang, H., Yi, J., and Hsieh, C.-J. Ead: Elastic-net attacks to deep neural networks via adversarial examples. In *AAAI*, 2018.

Cohen, J., Rosenfeld, E., and Kolter, Z. Certified adversarial robustness via randomized smoothing. In *ICML*, 2019.

Croce, F. and Hein, M. Reliable evaluation of adversarial robustness with an ensemble of diverse parameter-free attacks. In *ICML*, 2020.

Croce, F. and Hein, M. Mind the box: l_1-apgd for sparse adversarial attacks on image classifiers. In *icml*, 2021.

Devlin, J., Chang, M.-W., Lee, K., and Toutanova, K. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*, 2018.

Dhillon, G. S., Azizzadenesheli, K., Bernstein, J. D., Kossaifi, J., Khanna, A., Lipton, Z. C., and Anandkumar, A. Stochastic activation pruning for robust adversarial defense. In *ICLR*, 2018.

Engstrom, L., Tran, B., Tsipras, D., Schmidt, L., and Madry, A. Exploring the landscape of spatial robustness. In Chaudhuri, K. and Salakhutdinov, R. (eds.), *icml*, 2019.

Finn, C., Abbeel, P., and Levine, S. Model-agnostic meta-learning for fast adaptation of deep networks. In *ICML*, 2017.

He, K., Zhang, X., Ren, S., and Sun, J. Identity mappings in deep residual networks. In *ECCV*, 2016.

Hendrycks, D. and Dietterich, T. Benchmarking neural network robustness to common corruptions and perturbations. In *ICLR*, 2019.

Huang, G., Sun, Y., Liu, Z., Sedra, D., and Weinberger, K. Q. Deep networks with stochastic depth. In *ECCV*, 2016.

Jalal, A., Ilyas, A., Daskalakis, C., and Dimakis, A. G. The robust manifold defense: Adversarial training using generative models. *arXiv preprint arXiv:1712.09196*, 2017.

Kang, D., Sun, Y., Hendrycks, D., Brown, T., and Steinhardt, J. Testing robustness against unforeseen adversaries. *arXiv preprint arXiv:1908.08016*, 2019.

Krizhevsky, A. Learning multiple layers of features from tiny images. *University of Toronto*, 05 2012.

Kurakin, A., Goodfellow, I., and Bengio, S. Adversarial examples in the physical world. *arXiv preprint arXiv:1607.02533*, 2016.

Lee, H. B., Nam, T., Yang, E., and Hwang, S. J. Meta dropout: Learning to perturb latent features for generalization. In *ICLR*, 2020.

Madaan, D., Shin, J., and Hwang, S. J. Adversarial neural pruning with latent vulnerability suppression. In *ICML*, 2020.

Madry, A., Makelov, A., Schmidt, L., Tsipras, D., and Vladu, A. Towards deep learning models resistant to adversarial attacks. In *ICLR*, 2017.

Maini, P., Wong, E., and Kolter, J. Z. Adversarial robustness against the union of multiple perturbation models. In *ICML*, 2020.

Mao, C., Zhong, Z., Yang, J., Vondrick, C., and Ray, B. Metric learning for adversarial robustness. In *AAAI*, 2019.

Netzer, Y., Wang, T., Coates, A., Bissacco, A., Wu, B., and Ng, A. Y. Reading digits in natural images with unsupervised feature learning. In *Workshop on Deep Learning and Unsupervised Feature Learning, NeurIPS*, 2011.

Noh, H., You, T., Mun, J., and Han, B. Regularizing deep neural networks by noise: Its interpretation and optimization. In *NeurIPS*, 2017.

Pang, T., Xu, K., Dong, Y., Du, C., Chen, N., and Zhu, J. Rethinking softmax cross-entropy loss for adversarial robustness. In *ICLR*, 2020.

Paszke, A., Gross, S., Massa, F., Lerer, A., Bradbury, J., Chanan, G., Killeen, T., Lin, Z., Gimelshein, N., Antiga, L., et al. Pytorch: An imperative style, high-performance deep learning library. In *NeurIPS*, 2019.

Raghunathan, A., Steinhardt, J., and Liang, P. Certified defenses against adversarial examples. *arXiv preprint arXiv:1801.09344*, 2018.

Rauber, J., Brendel, W., and Bethge, M. Foolbox: A python toolbox to benchmark the robustness of machine learning models. In *Reliable Machine Learning in the Wild Workshop, ICML*, 2017.

Ren, M., Zeng, W., Yang, B., and Urtasun, R. Learning to reweight examples for robust deep learning. In *ICML*, 2018.

Rusak, E., Schott, L., Zimmermann, R., Bitterwolf, J., Bringmann, O., Bethge, M., and Brendel, W. A simple way to make neural networks robust against diverse image corruptions. In *ECCV*, 2020.

Russakovsky, O., Deng, J., Su, H., Krause, J., Satheesh, S., Ma, S., Huang, Z., Karpathy, A., Khosla, A., Bernstein, M., et al. Imagenet large scale visual recognition challenge. *International journal of computer vision*, 2015.

Samangouei, P., Kabkab, M., and Chellappa, R. Defense-GAN: Protecting classifiers against adversarial attacks using generative models. In *ICLR*, 2018.

Schott, L., Rauber, J., Bethge, M., and Brendel, W. Towards the first adversarially robust neural network model on mnist. In *ICLR*, 2018.

Shafahi, A., Najibi, M., Ghiasi, M. A., Xu, Z., Dickerson, J., Studer, C., Davis, L. S., Taylor, G., and Goldstein, T. Adversarial training for free! In *NeurIPS*, 2019.

Shen, D., Wu, G., and Suk, H.-I. Deep learning in medical image analysis. *Annual review of biomedical engineering*, 2017.

Shu, J., Xie, Q., Yi, L., Zhao, Q., Zhou, S., Xu, Z., and Meng, D. Meta-weight-net: Learning an explicit mapping for sample weighting. In *NeurIPS*, 2019.

Smith, L. N. Cyclical learning rates for training neural networks. In *WACV*, 2017.

Song, Y., Kim, T., Nowozin, S., Ermon, S., and Kushman, N. Pixeldefend: Leveraging generative models to understand and defend against adversarial examples. In *ICLR*, 2018.

Srivastava, N., Hinton, G., Krizhevsky, A., Sutskever, I., and Salakhutdinov, R. Dropout: a simple way to prevent neural networks from overfitting. *JMLR*, 2014.

Szegedy, C., Zaremba, W., Sutskever, I., Bruna, J., Erhan, D., Goodfellow, I., and Fergus, R. Intriguing properties of neural networks. *arXiv preprint arXiv:1312.6199*, 2013.

Thrun, S. and Pratt, L. (eds.). *Learning to Learn*. Kluwer Academic Publishers, 1998.

Tramèr, F. and Boneh, D. Adversarial training and robustness for multiple perturbations. In *NeurIPS*, 2019.

Tramer, F., Carlini, N., Brendel, W., and Madry, A. On adaptive attacks to adversarial example defenses. In *NeurIPS*, 2020.

Tsipras, D., Santurkar, S., Engstrom, L., Turner, A., and Madry, A. Robustness may be at odds with accuracy. In *ICLR*, 2019.

Uesato, J., O'Donoghue, B., Oord, A. v. d., and Kohli, P. Adversarial risk and the dangers of evaluating against weak attacks. *arXiv preprint arXiv:1802.05666*, 2018.

Wong, E. and Kolter, J. Z. Provable defenses against adversarial examples via the convex outer adversarial polytope. *arXiv preprint arXiv:1711.00851*, 2017.

Wong, E., Rice, L., and Kolter, J. Z. Fast is better than free: Revisiting adversarial training. In *ICLR*, 2020.

Wu, D., Xia, S.-T., and Wang, Y. Adversarial weight perturbation helps robust generalization. In *NeurIPS*, 2020.

Xiao, C., Zhong, P., and Zheng, C. Enhancing adversarial defense by k-winners-take-all. In *ICLR*, 2020.

Yin, X., Kolouri, S., and Rohde, G. K. Gat: Generative adversarial training for adversarial example detection and robust classification. In *ICLR*, 2020.

Zagoruyko, S. and Komodakis, N. Wide residual networks. In *British Machine Vision Conference*, 2016.

Zhang, H., Yu, Y., Jiao, J., Xing, E. P., Ghaoui, L. E., and Jordan, M. I. Theoretically principled trade-off between robustness and accuracy. In *ICML*, 2019.