Improving Adversarial Training with Informative Feature-based Attack

Chiu Wai YAN
CSE, HKUST
cwyan@connect.ust.hk

Dit-yan YEUNG
CSE, HKUST
dyyeung@cse.ust.hk

Abstract

Adversarial training has achieved great success in improving the robustness of machine learning models against adversarial attacks. However, recent studies showed that the regularization approach in adversarial training still has much room for improvement to further enhance model robustness. In this paper, we first analyze two types of regularization to improve adversarial training, one in the loss during adversarial training and the other in the loss during attack generation. Based on our findings, we propose a new attack, called the InfoFeature *attack, which perturbs the class activation map (CAM) generated from the high-level feature maps of the model. The proposed method based on the* InfoFeature *attack aims to provide additional information and regularization to the model during training and hence achieve parameter updates that can boost model robustness. With extra regularization incorporated,* InfoFeature *exhibits an outstanding attack strength when compared to the Projected Gradient Descent (PGD) attack. When using the adversarial examples generated from the* InfoFeature *attack for adversarial training, our experiments show that state-of-the-art model robustness against white-box attacks can be achieved.*

1. Introduction

With multiple breakthroughs, deep learning models have achieved outstanding performance in solving many computer vision tasks, including image classification [19, 12], semantic segmentation [31], image synthesis [15], and many more. However, recent studies [35, 11] showed that adversarial examples can be crafted to degrade the performance of deep learning models significantly. By adding small perturbations to normal images, the resulting adversarial examples can consistently fool the model to yield incorrect predictions [4]. Despite causing a huge difference to model prediction, the perturbations added to adversarial examples are often hardly perceptible by humans [6]. As a small change in the input leads to an unexpectedly huge change in the output, the model is also said to be lacking adversarial robustness.

Numerous defense approaches [24, 37] have been proposed to reduce the influence caused by adversarial examples or to improve the robustness of the target model. For many of the defense schemes, a phenomenon known as obfuscated gradient [2], as a kind of gradient masking [27], exists to give a false sense of robustness. Obfuscated gradient works by eliminating the gradient of the model in test time, causing most iterative white-box attacks to fail to update properly and hence contributing to its delusion of robustness. Defense schemes relying on obfuscated gradients usually can be circumvented and are vulnerable to black-box attacks which do not make use of the model gradient information [2]. Among the defense schemes, adversarial training [11, 24] is shown to be an example which does not rely on obfuscated gradients [2], granting it potential to be further explored.

In this paper, we investigate improvements made on top of adversarial training. The model update during adversarial training can be viewed as self-correction for enhancing robustness. Our goal in this work is to uncover a method that improves such self-correction so that the model converges towards a more robust state. We study two types of regularization that improve the self-correction, with consideration of the feature space. The first type of work involves regularization in the training loss function. This is originally used in model training to avoid overfitting, and multiple lines of work [40, 29, 17, 16] used regularization to increase model robustness. A generic formulation of this type of work can be defined as

$$\min_{\theta} \; L(\theta, x+\delta, y) + \lambda \, R_1(\theta) \tag{1}$$

where L is the adversarial training loss introduced in [24], R_1 is a regularization term for the loss, and λ is the weighting factor to adjust the influence of the regularization term.

The second type of work modifies the attack [38, 43, 44]. A naive training with random noise perturbation to the input is shown to be ineffective to improve model robustness, and the resulting perturbation added on the input sample should be semantically meaningful, by showing correlation to the representation and classification ability of the model . A

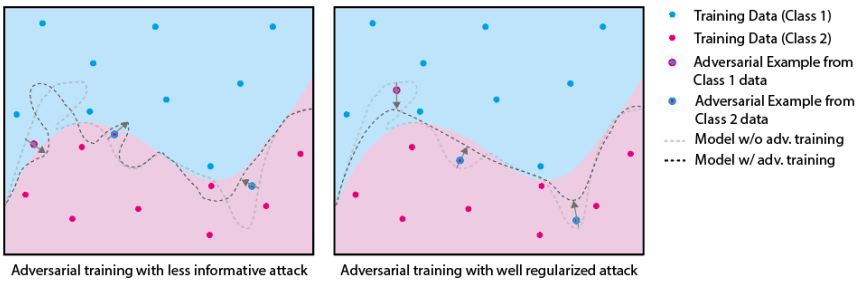


Figure 1: A simplistic illustration showing the influence on the decision boundaries between an attack with a less informative attack (left) and an ideal informative attack (right). A less informative attack (left) fails to generalize the adversarial example, resulting in extra vulnerability after model update. An ideal informative attack (right) regularizes the parameter to be more robust.

straightforward example is a strong white-box attack. However, a semantically meaningful example is not simply limited to the strength of the attack. Instead, the attacks used in this type of work usually take the knowledge representation of the model into account, and a typical implementation is to perturb the feature space. Similar to Equation 1, the generic formulation of the task is given by

$$\min_{\theta} \; L(\theta, x + \delta, y) \tag{2}$$

and

$$\delta = \arg\max_{\delta \in \mathcal{S}} \; \lambda_1 \, L(\theta, x + \delta, y) + \lambda_2 \, R_2(\theta) \tag{3}$$

where L is usually the same loss function as that used for adversarial training to control the attack strength, and R_2 is the regularization term used to control the information to perturb. We also consider the case when $\lambda_1 = 0$, in which the attack is generated without perturbing the prediction score but only the regularization term. Here we would like to focus on the attack formulation, and therefore augmentation techniques such as mixing attacks [25, 36] and mixup [42, 21] will not be considered in this work.

To compare the two types of regularization, we raise three research questions in particular to measure the influence brought by regularization:

- During attack generation, what is the influence of the attack regularizer on the attack generated?
- How much does a loss regularizer in the feature space contribute to a model's final robustness?
- How much does an attack regularizer in the feature space contribute to a model's final robustness?

Based on our preliminary findings, the attack regularizer consistently results in models with better robustness when compared to the loss regularizer. We then propose a new attack called the `InfoFeature` attack, which attempts to provide additional regularization to model update through scattering the high-level feature space. Different from previous methods in which attacks are performed on the raw features, our proposed method applies postprocessing to the feature maps with visualization technique and performs attacks on the class activation map (CAM) obtained. We expect training the model with `InfoFeature` delivers more information to the model regarding feature extraction, and hence to enhance the model to have robust parameter update. In terms of the classification decision boundary, the parameter update from a well-regularized attack should push the boundaries towards the border of the ground-truth distribution, as shown in Figure 1.

We summarize the contributions of this work here:

- We studied the influence of common feature-based regularization to model parameter update in two situations, applied to the loss function and to attack generation.
- We proposed the `InfoFeature` attack as an attack that exploits the high-level feature space with a regularization term.
- We integrated `InfoFeature` into adversarial training to achieve comparable performance to state-of-the-art methods in the same line of exploration. When the model is small in capacity, our method outperforms previous methods.
- We studied how the attack during adversarial training influences the resulting model through various experimental evaluations.

2. Related Work

Since the amount of work in improving adversarial training is large, we only include in this short literature review

the existing methods that are most relevant to ours – to improve the robust accuracy of the trained model against different attacks.

Adversarial Training Adversarial training was first proposed by Goodfellow et al. [11], by training the model with both clean and adversarial examples perturbed with Fast Gradient Sign Method (FGSM). It targets to minimize $\alpha\, J(\theta, x, y) + (1-\alpha) J(\theta, x + \epsilon\, sign(\nabla_x\, J(\theta, x, y))$, where the second term is the loss computed from predicting adversarial examples.

Madry et al. [24] refined the method and replaced the objective loss function to be

$$\min_{\theta} \mathbb{E}_{(x,y)\sim\mathcal{D}} \left[\max_{\delta\in\mathcal{S}} L(\theta, x+\delta, y) \right] \tag{4}$$

where the inner maximization is realized with Projected Gradient Descent (PGD) attack proposed in the same paper.

Despite being effective to strengthen model robustness, adversarial training suffers from a few drawbacks. Due to the iterative nature of the attack, training takes much longer than that with clean inputs, causing it impractical for large-scale datasets [34]. On the other hand, overfitting in adversarial training can harm to a very large degree in robustness [32], while a phenomenon known as catastrophic overfitting [39] can occur by reducing the robust accuracy drastically. Some studies [28, 7, 8] reveal that a slight change in hyper-parameters can cause a huge difference in the final result, raising the difficulty to optimize the model. Furthermore, models trained in an adversarial manner suffer from label leaking [20], which the models learn to exploit regularities in the adversarial example construction process.

Regularization in Loss Function One type of work attempts to improve model robustness by adding a regularization term to the loss function. The most relevant methods are those appending a term regularizing the feature-based representation in the loss function. [17] proposed Adversarial Logit Pairing (ALP), by introducing a regularization term on the logits $L(f(x;\theta), f(\tilde{x};\theta))$. Following ALP, [22] proposed Feature Matching, with the regularization term coupling the high-level feature maps. Some works [40, 29, 10, 30] proposed to regularize the spectral norm, which is the largest singular value of the weight matrices. Some works proposed to enforce regularization in the backward gradient [16, 33], which is sometimes also called Jacobian regularization [14]. TRADES [45] worked on the trade-off between robustness and accuracy, with regularization $\max_{X'\in\,\mathbb{B}(X,\epsilon)} \phi(f(X)f(X')/\lambda)$, where ϕ is a surrogate loss that enhances the theoretical guarantee. Another type of work [1] employed an unsupervised perspective on improving model robustness, with the regularization term playing the core role in the loss function.

Variation of the Attack The work most relevant to ours focuses on variation of the attack. Bilateral Adversarial Training [38] generates adversarial labels with a closed-form formula besides the adversarial image to train the model. Feature Scattering [43] trains the model with an attack that scatters features in the latent space. Instead of comparing an image to its adversarial sample, it compares to the feature of another image in the same class, taking inter-sample relationships into consideration. Similarly but in an opposite way, Adversarial Interpolation Training [44] advocates training with attacks that minimize the feature space distance between two different classes. In the context of our work, these training methods are said to be *informative* in the attack.

3. Informative Adversarial Training

We first define the term *informative*. Given a model which is trained in an adversarial manner against two different attacks, δ_1 and δ_2, to result in accuracy a_1 and a_2, respectively, we say δ_1 is more *informative* than δ_2 if the resultant robust accuracy $a_1 > a_2$. Conceptually, we use the term *informative* to describe how robust a model ends up converging, with the assumption that the model obtains information from the attack during adversarial training. The adversarial example indicates a confusing input to the model, while the model learns to overcome misclassification through appropriate parameter update.

For convolutional neural networks, adversarial examples generated from white-box attacks usually consist of low-level features not belonging to the class, causing the convolutional layers to output perturbed feature maps in the beginning. Then the perturbed features grow significantly across different unbounded hidden layers [11], and finally cause an incorrect prediction. Intuitively, an *informative* attack used in adversarial training leads to feature maps with misleading perturbation, such that during parameter update, the model learns to classify against similar ambiguous features. This motivates us to study regularization in the feature space. Since adversarial vulnerability is more likely to exist in the hidden layers which work as a universal approximator [11] but not the nonlinear mapper such as the softmax layer, one possible option for picking where to apply perturbation is the final convolutional layer. This ensures that all hidden layers of the model are backpropagated and have gradient update, which maximizes the information used as self-correction.

3.1. Informative Features

A common definition of features is the intermediate output of the model. Previous works mainly focus on either the logit output [17] or the activation after the last convolutional layer [44]. Although the feature maps capture high-level semantics from the model and provide feasible gradient for

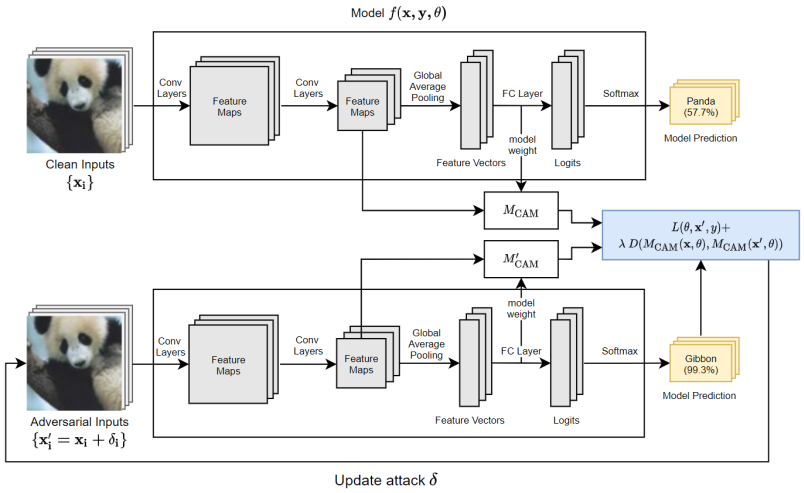


Figure 2: Generation procedure of the `InfoFeature` attack.

backpropagation, the relationship between the numeral values and the representation is unclear. With simply an intermediate feature map, one has no idea how the feature maps contribute to particular activation. When the entire feature space is scattered or interpolated, the perturbation is likely biased towards disrupting less significant information, resulting in a drop of attack strength. This may also hinder the corresponding improvement to be made by the trained model. To resolve this issue, visualization techniques can be applied to the features such that the resulting features contain more significant information, especially for those contributing to a high degree to the final prediction.

Therefore, instead of a naive intermediate feature map, we are more interested in attacking the saliency of the model, which indicates the location the model is most influenced. We obtain the **class activation map (CAM)** [46], a simplistic postprocessing to understand the saliency of the model. The formula to compute CAM is given by

$$M_{\text{CAM}}(x, \theta) = \sum_k w_{\text{fc}}^{(k)} M_{\text{conv}}^{(k)}(x, \theta) \tag{5}$$

where k is iterated along the channel dimension, w_{fc} is the weight of the last fully connected layer in the model, and M_{conv} is the activation after the last convolutional layer and before the global average pooling, given the input image x and model parameters θ. The computation only consists of addition and multiplication, so it is differentiable and simple to calculate without much overhead.

CAM not only provides an indication of object localization for visualization, but also suggests the significance of feature maps by weighting across the channel dimension. In other words, more relevant features are magnified while less relevant features are suppressed. With these benefits, CAM is also widely adopted in weakly-supervised learning tasks [3].

We are not the first work to apply CAM to adversarial training. Similar work was first proposed in [22]. However, our method exhibits a substantial difference with it. [22] couples the CAM features during model training in the loss function, while the CAM in our work helps in strengthening and regularizing the attack. We also compared the difference between the two types of work, and our work outperforms the former by a large margin as shown in Section 4.1.

3.2. InfoFeature Attack

Here we present the `InfoFeature` attack as an example to make the attack more *informative*. During the generation of the attack, besides fooling the model prediction to be away from the target distribution, we add additional regularization such that the feature space is also perturbed in terms of the activation with respect to the classes.

We do not generate CAM directly from Equation 5. The weight $w_{\text{fc}}^{(k)}$ in the model can be negative, and thus the finally computed CAM can also contain negative values. The goal during the parameter update should be highlighted by significant information from positive CAM values, while a reverse update from negative CAM values may lead to possible deviation in the update directions. Hence we clip away all negative values before the values are summed up. In particular, the CAM computed in our proposed attack is changed as follows:

$$M_{\text{CAM}}(x, \theta) = \sum_k \max(w_{\text{fc}}^{(k)} M_{\text{conv}}^{(k)}(x, \theta), 0) \tag{6}$$

With the modified formulation of M_{CAM}, the formal formu-

lation of the attack loss function is to maximize L_{Info} which is given by

$$L_{\text{PGD}}(\theta, x+\delta, y)+\lambda\, D(M_{\text{CAM}}(x,\theta), M_{\text{CAM}}(x+\delta,\theta)) \quad (7)$$

where L_{PGD} is the cross-entropy loss adopted in the original PGD attack [24], λ is the weighting factor of the regularization, and D is a distance metric used to compare the two CAMs. A diagram illustrating the attack generation procedure can be found in Figure 2.

Our observation shows that setting $\lambda = 0.1$ results in a good level of robustness. We use the L2 distance for D, while other distance metrics such as the L1 distance and cosine distance are also expected to have similar performance. Although some methods [39, 38] achieved satisfying robustness by training with non-iterative attacks, we found that our attack works most effectively between 3 to 7 iterations. To balance the training time required and the final model robustness, we set the number of iterations to 7 as a default hyperparameter. Unlike previous works including Feature Scattering [43] and Adversarial Interpolation Training [44], our proposed attack does not take the inter-sample relationship into consideration. Instead, our experiments show that an attack that perturbs feature information suffices to contribute to a high degree of robustness.

3.3. Utilizing the Attack in Training

We further leverage the proposed `InfoFeature` attack into adversarial training. The final formulation of the training is defined to be:

$$\min_{\theta} \mathbb{E}_{(x,y)\sim\mathcal{D}} \left[\max_{\delta\in\mathcal{S}} L_{\text{Info}}(\theta, x+\delta, y) \right] \quad (8)$$

where L_{info} is generated from Equation 7. Note that such formulation of the setup is almost the same as Equation 4 proposed by Madry et al. [24], except that the inner maximization of the attack is replaced by our proposed `InfoFeature` attack. Within our proposed attack, the only difference between ours and that of [24] is that we have an additional weighted regularization to scatter the CAMs. We will show that such a subtle change in the formulation induces a large contribution to both the attack strength and the resulting adversarial robustness.

4. Experiments

For all adversarial settings in training, the model is trained against ℓ_∞ norm adversaries. The maximum perturbation budget $\epsilon = 8$, and all iterative attacks are generated in 7 iterations with step size 2. We perform data augmentation that consists of random crops with a padding of 4 pixels, followed by random horizontal flip. We follow the order of preprocessing in [43], which performs data normalization before applying the attack. With such a setup, the images to perturb lie in the range $[-1, 1]$, and hence the actual perturbation enforced should be doubled (i.e., $\epsilon_{\text{impl}} = 16$). To avoid confusion and unify with previous works, such perturbation strength in our experiments will be reported as $\epsilon = 8$.

Unless otherwise specified, all ResNet18 models are trained with an initial learning rate of 0.1 and a decay factor of 0.1 at the [60, 90]-th epoch. We use the SGD optimizer with momentum 0.9 and weight decay 2e-4.

4.1. Regularization in Loss versus Attacks

We first report our findings comparing to state-of-the-art feature-based loss regularization and attack regularization. Our baselines include standard training on clean data (Standard) [18] and adversarial training with PGD attacks (Madry) [24]. We pick two previous works on loss regularization, namely, Adversarial Logit Pairing (ALP) [17] and Adversarial Feature Matching (FM) [22]. Following the suggestions in the papers, we set $\lambda = 0.1$ for ALP and $\lambda = 10$ for FM. We compare the two regularization methods in the loss with another two works using regularization in the attack, known as Feature Scattering Adversarial Training (Feascatter) [43] and Adversarial Interpolation Training (Advinterp) [44]. Finally, we also append an entry of our proposed `InfoFeature` adversarial training. All of our selected methods focus on regularizing or perturbing the feature space of the model, and all of them are trained for 120 epochs in total.

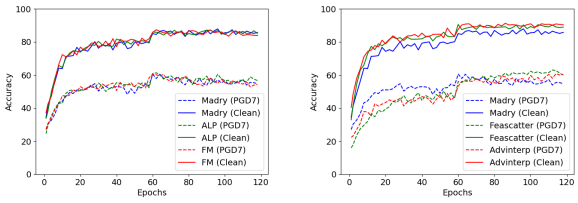
The initial experiments are performed on ResNet18 [13] models using the CIFAR10 [18] dataset. The CIFAR10 dataset is an object classification dataset with 10 classes, containing 50K training images and 10K test images. We attack the trained model with the most common white-box attacks, including FGSM [11], PGD [24] with 7 iterations and 20 iterations, and the CW [5] attack. Due to computational complexity concerns, we do not use the optimization-based CW attack in the original paper [5], but the implementation with projected gradient descent using a surrogate loss. Table 1 shows the preliminary experimental result for CIFAR10 trained on ResNet18 and Figure 3 shows a plot of the validation accuracy recorded over different epochs.

From both Table 1 and Figure 3, we observe that regularization applied to the loss contributes very little to improvement in model robustness. The resulting accuracy under most the of attacks remains very similar to the original adversarial training proposed by Madry et al. [24], which agrees with the result in an evaluation [9] of adversarial logit pairing. In contrast, regularization applied to attack generation provides a more apparent improvement to the model, with more than 8% increase in adversarial accuracy against the PGD20 attack.

After showing that regularizing the training loss function provides limited contribution to adversarial robustness, we would like to further verify the influence of the attack regu-

Methods	Clean	ℓ_∞-norm white-box attacks			
		FGSM	PGD7	PGD20	CW20
Standard	**94.7**	30.7	1.8	0.0	0.0
Madry	85.6	68.6	56.1	40.3	41.5
Loss Regularization					
ALP (0.1)	85.7	67.6	56.3	41.3	43.0
FM (10)	85.6	68.7	58.0	42.9	44.2
Attack Regularization					
FeaScatter	89.5	69.8	61.0	**50.2**	44.2
AdvInterp	90.0	**70.4**	61.0	47.5	44.1
Ours	85.1	69.3	**61.3**	49.3	**48.0**

Table 1: Accuracy comparison on CIFAR10 trained on ResNet18.



(a) (b)

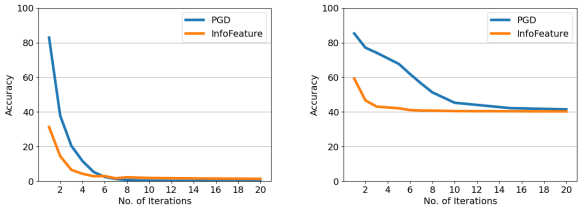
Figure 3: Validation accuracy over training epochs for (a) feature-based loss regularization methods and (b) feature-based attack regularization methods.

larizer. To answer our first question: "What is the influence of the attack regularizer on the final attack?", we measure the L2 distance between the PGD perturbed example with and without our regularization term, and also test their attack strength against different models. The result is reported in Table 2 and Figure 4.

Since both of our compared methods drop the original cross-entropy loss (i.e., setting $\lambda_1 = 0$ in Equation 3) in the PGD attack and use their corresponding loss function as the core to regularize the attack update, the perturbed outputs are expected to show disparity with that from the PGD attack. Table 2 indicates that our proposed `InfoFeature` attack also exhibits a substantial difference from the original PGD attack. Moreover, Figure 4 reflects an apparent difference in attack strength between the adversarial examples generated by `InfoFeature` and PGD, especially when the number of attack iterations is small, further verifying the effectiveness of attack regularization.

4.2. Performance Evaluation on White-box Attacks

We extend our experiment to cover more datasets and model architectures. Apart from the CIFAR10 dataset in Section 4.1, we evaluate the performance of our proposed method with two more benchmark datasets: CIFAR100 [18] and SVHN [26]. In this section we only compare our work with attack regularization methods in the feature space, that is, Feature Scattering [43] and Adversarial Interpolation Training [44]. Following previous work, we adopt WideResNet28-10 (WRN-28-10) [41], a 10-times wide variant of the ResNet, as the model to train. WideResNet28-10 contains around 36.5M parameters and achieves the best classification result in CIFAR10 and CIFAR100 among the tests in [41]. Different from the training of ResNes18, we train the WideResNet28-10 model for 200 epochs, with the remaining hyperparameters unchanged.



(a) (b)

Figure 4: Accuracy against PGD and InfoFeature attacks on (a) a model without defense and (b) a model trained with standard 7-iteration PGD adversarial training.

In an adversarial setting, the robustness of the final model can be fluctuating. Under the same testing environment, a slight difference in weight decay can reduce the model robust accuracy by more than 7% [28]. We follow some of the implementation details suggested in [28], particularly smoothing the labels with strength 0.5, configuring batch normalization to the evaluation mode during attack generation, and applying early stopping. We noticed that the performance of the methods compared is unstable due to sensitivity to randomness. Therefore, for CIFAR10 with WideResNet28-10, we perform multiple restarts of the training process, using the same hyperparameter setting but different initializations, and report both the best and worst models in different restarts in Table 3. Note that the worst model presented here already achieves the highest adversarial accuracy among all of the 200 training epochs in its restart.

CIFAR100 Similar to CIFAR10, CIFAR100 is an extended dataset with 100 classes, containing 50K training images and 10K test images. The classification task for this dataset is much more difficult as the number of classes increases but the number of images for each class is 10 times smaller than that of CIFAR10 [43]. Considering the difficulty of the task, we only evaluate the performance of our method on the large ResNet28-10 model, and the result is reported in Table 4. We also trained all of the models for

Methods	L2 Distance to n-iter PGD Perturbation ($\epsilon = 8$) ($\times$1e$-$5)													
	1-iter				2-iter		5-iter		7-iter		10-iter		20-iter	
	Feascatter	Advinterp	CW	Ours	CW	Ours	CW	Ours	CW	Ours	CW	Ours	CW	Ours
Clean	2.1	2.0	2.8	2.0	1.8	2.2	1.6	1.9	1.5	2.0	1.5	2.0	1.6	2.1
Madry	2.1	2.0	2.8	2.0	1.8	2.7	1.6	2.7	1.8	2.8	2.0	2.8	2.1	2.8

Table 2: The L2 distance between examples generated from the regularized attack and the original attack. The PGD with CW loss (CW) is a reference for an attack with a different loss function. Our proposed `InfoFeature` attack, as well as other regularization methods in attack generation such as Feature Scattering (Feascatter) and Adversarial Interpolation Training (Advinterp), exhibit a substantial difference from the original PGD attack.

Models	Clean	ℓ_∞-norm white-box attacks			
		FGSM	PGD7	PGD20	CW20
Standard	95.8	42.1	3.1	0.0	0.0
Madry	88.7	70.8	59.5	45.0	45.7
Best Model					
FeaScatter	**92.8**	**88.9**	66.8	55.0	54.9
AdvInterp	90.6	75.1	**71.6**	**67.5**	**61.4**
Ours	90.6	73.0	64.8	50.0	49.5
Worst Model					
FeaScatter	**90.3**	71.4	59.1	44.4	44.5
AdvInterp	88.1	66.8	58.2	48.4	49.1
Ours	89.9	**72.6**	**61.6**	**48.5**	**49.2**

Table 3: Accuracy comparison on CIFAR10 trained on WideResNet28-10. We perform multiple restarts of training, and report both the models performing the best and the worst.

Models	Clean	ℓ_∞-norm white-box attacks			
		FGSM	PGD7	PGD20	CW20
Standard	**76.8**	0.1	0.0	0.0	0.0
Madry	63.8	42.7	33.7	23.7	24.2
FeaScatter	72.8	**46.0**	**36.7**	25.3	22.4
AdvInterp	70.3	44.7	33.9	28.2	19.2
Ours	63.0	44.9	34.4	**28.6**	**26.6**

Table 4: Accuracy comparison on CIFAR100 trained on WideResNet28-10. The best models are reported.

200 epochs, with the same set of hyperparameters as in the previous experiments. The performance of all models is poor in general, especially under strong attack, while our proposed method results in slightly higher robustness.

SVHN SVHN is an object classification dataset composing of digits cropped from street view images. It contains 73,257 training examples and 26,032 test examples, along with an extra training set of over 531K examples. We do not use the extra training dataset in the experiment. Different from CIFAR10 and CIFAR100, we do not perform horizontal flip as a data augmentation operation on this dataset. Following previous works [43], we set the initial learning rate to be 0.01, and the decay of the learning weight retains to be 0.1 at the [60, 90]-th epoch. The result can be found in Table 5.

Models	Clean	ℓ_∞-norm white-box attacks			
		FGSM	PGD7	PGD20	CW20
Standard	**96.1**	40.6	1.4	0.0	0.0
Madry	91.9	80.4	71.8	43.1	45.4
FeaScatter	94.5	89.1	71.5	52.8	49.0
AdvInterp	95.0	**91.1**	69.2	49.1	48.0
Ours	94.4	89.3	**72.4**	**54.4**	**50.7**

Table 5: Accuracy comparison on SVHN trained on ResNet18. The best models are reported. Ours outperform other works in a model with less capacity.

In the above experiments, the performance of the reproduced state-of-the-art methods to compare is much worse than what was reported in their corresponding papers. Similar observation also exists in other works [7] evaluating the robustness of the defenses. According to references reproducing the Feature Scatter (FeaScatter) adversarial training [7, 21], together with our experiment findings, feature-based regularization in the attack can result in very different model robustness, under the same hyperparameter setting but different initial model parameters and batched input. In other words, the performance of the trained model to a large extent depends on the initial setting, which remarkably increases the difficulty to reproduce an outstanding performance from previous works. For example, in Table 3, although the best-case performance of the previous methods can be promising, the model can also end up in a poor state with much lower adversarial robustness.

Even in the worst case, our proposed `InfoFeature` adversarial training method always achieves better robustness in comparison to the standard adversarial training. Although being less competitive in the best-case performance for large-scale models such as WideResNet28-10, our method outperforms the state-of-the-art methods when

Models	Random Noise	Transferred ℓ_∞-norm black-box attacks			
		Standard		Madry	
		PGD10	CW10	PGD10	CW10
Standard	92.1	35.8	30.1	82.4	81.1
Madry	88.7	88.6	88.4	75.3	74.6
Ours	89.8	89.8	88.3	76.6	75.2

Table 6: Accuracy comparison against transferred black-box attacks. Models in columns are ResNet18 models being attacked under white-box setting. Models in rows are the target WideResNet28-10 models to evaluate.

the model's capacity is small, and shows a lower variance with different training restarts. The partial superiority highlights the success of creating an *informative* attack by perturbing postprocessed high-level feature maps. However, one possible drawback of our proposed method is the trade-off of clean accuracy, due to the strong nature of `InfoFeature` attack. This issue is also noticeable in standard adversarial training with PGD attacks. One suggestion we could provide is to reduce the number of iterations to generate `InfoFeature` attack, which results in a drop in robustness and a rise in clean accuracy.

4.3. Performance Evaluation on Black-box Attacks

It is shown that some works in the community rely on obfuscated gradients [2] to seemingly make the model look secure, while these types of methods are usually vulnerable to gradient-free attacks such as typical black-box attacks. Although adversarial training is already known to be free of obfuscated gradients [2], we would like to also report the robustness of our model against black-box attacks. In this section, we focus on transferred black-box attacks [27] from other models. Specifically, we first generate white-box attacks on multiple ResNet18 models, and apply the same set of perturbed images to our target WideResNet28-10 model, trained with different methods. To simplify the experiment, we use the untargeted attack in all of the cases as it is known to have better transferability [23].

From Table 6, our proposed training method is able to resist transferred black-box attacks, with similar robustness to adversarial training from Madry et al. [24]. This implies that model robustness is originated from the ability of the model to classify the correct class under strong attacks, not by disabling gradient information to bypass white-box attacks.

4.4. Discussions on the Result

In this section, we summarize and discuss the experimental result reported above.

Effect of different types of regularization Feature-based regularization in the loss function has a limited influence on the adversarial training procedure. However, feature-based regularization in attack generation brings about changes to a much higher degree, in terms of both the attack strength and the final model robustness.

Performance of `InfoFeature` adversarial training We proposed `InfoFeature` adversarial training as an example to achieve an *informative* attack. Our method consistently outperforms standard adversarial training. When the model is small or the training examples are scarce, it outperforms state-of-the-art methods based on regularization in the feature space. We noticed instability of performance in the previous works, while `InfoFeature` was shown to alleviate such instability. One drawback of `InfoFeature` is the drop in classification accuracy with clean input. This issue can be alleviated by reducing the number of attack iterations and applying data augmentation such as mixup [42, 21].

Ablation study and more results For the studies on the effect of hyperparameters such as the regularization weight λ, the number of iterations in the attack, and more figures illustrating the performance of `InfoFeature` adversarial training, please refer to the supplemental material.

5. Conclusion

In this paper, we first evaluated the regularization in the training loss function and the regularization in attack generation. We showed that regularizing the loss with information from the feature space provides limited influence to the eventual model robustness. Nevertheless, regularization in attack generation makes a huge difference to both the generated attack and the model convergence state. We also proposed `InfoFeature` attack, an attack that regularizes the model update by perturbing the class activation maps during adversarial training. After integrating the attack into adversarial training, the final model achieves state-of-the-art robustness in relatively small models, outperforming previous works on the same line of exploration. We also discovered that the models with regularization in attack generation are sensitive to the inherent randomness of initialization. Changing the parameter initialization, batched samples and actual attack formulation can lead to a huge gap in performance. We leave the study of such phenomenon for future work.

References

[1] Jean-Baptiste Alayrac, Jonathan Uesato, Po-Sen Huang, Alhussein Fawzi, Robert Stanforth, and Pushmeet Kohli. Are labels required for improving adversarial robustness? In H.

Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. Fox, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 32. Curran Associates, Inc., 2019. 3

[2] Anish Athalye, Nicholas Carlini, and David A. Wagner. Obfuscated gradients give a false sense of security: Circumventing defenses to adversarial examples. In *ICML*, pages 274–283, 2018. 1, 8

[3] Wonho Bae, Junhyug Noh, and Gunhee Kim. Rethinking class activation mapping for weakly supervised object localization. In *European Conference on Computer Vision (ECCV)*, August 2020. 4

[4] Battista Biggio, Igino Corona, Davide Maiorca, Blaine Nelson, Nedim Šrndić, Pavel Laskov, Giorgio Giacinto, and Fabio Roli. Evasion attacks against machine learning at test time. In Hendrik Blockeel, Kristian Kersting, Siegfried Nijssen, and Filip Železný, editors, *Machine Learning and Knowledge Discovery in Databases*, pages 387–402, Berlin, Heidelberg, 2013. Springer Berlin Heidelberg. 1

[5] N. Carlini and D. Wagner. Towards evaluating the robustness of neural networks. In *2017 IEEE Symposium on Security and Privacy (SP)*, pages 39–57, 2017. 5

[6] Anirban Chakraborty, Manaar Alam, Vishal Dey, Anupam Chattopadhyay, and Debdeep Mukhopadhyay. Adversarial attacks and defences: A survey. *arXiv preprint*, arXiv: 1810.00069, 2018. 1

[7] Francesco Croce and Matthias Hein. Reliable evaluation of adversarial robustness with an ensemble of diverse parameter-free attacks. In Hal Daumé III and Aarti Singh, editors, *Proceedings of the 37th International Conference on Machine Learning*, volume 119 of *Proceedings of Machine Learning Research*, pages 2206–2216. PMLR, 13–18 Jul 2020. 3, 7

[8] Evelyn Duesterwald, Anupama Murthi, Ganesh Venkataraman, Mathieu Sinn, and Deepak Vijaykeerthy. Exploring the hyperparameter landscape of adversarial robustness. *Safe Machine Learning workshop at ICLR*, 2019. 3

[9] Logan Engstrom, Andrew Ilyas, and Anish Athalye. Evaluating and understanding the robustness of adversarial logit pairing. *arXiv preprint*, arXiv: 1807.10272, 2018. 5

[10] Farzan Farnia, Jesse Zhang, and David Tse. Generalizable adversarial training via spectral normalization. In *International Conference on Learning Representations*, 2019. 3

[11] Ian Goodfellow, Jonathon Shlens, and Christian Szegedy. Explaining and harnessing adversarial examples. In *International Conference on Learning Representations*, 2015. 1, 3, 5

[12] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2016. 1

[13] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2016. 5

[14] Judy Hoffman, Daniel A. Roberts, and Sho Yaida. Robust learning with jacobian regularization. *arXiv preprint*, arXiv: 1908.02729, 2019. 3

[15] Phillip Isola, Jun-Yan Zhu, Tinghui Zhou, and Alexei A. Efros. Image-to-image translation with conditional adversarial networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, July 2017. 1

[16] Daniel Jakubovitz and Raja Giryes. Improving dnn robustness to adversarial attacks using jacobian regularization. In *Proceedings of the European Conference on Computer Vision (ECCV)*, September 2018. 1, 3

[17] Harini Kannan, A. Kurakin, and Ian J. Goodfellow. Adversarial logit pairing. *arXiv preprint*, arXiv: 1803.06373, 2018. 1, 3, 5

[18] Alex Krizhevsky. Learning multiple layers of features from tiny images. Technical report, University of Toronto, 05 2012. 5, 6

[19] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. Imagenet classification with deep convolutional neural networks. In F. Pereira, C. J. C. Burges, L. Bottou, and K. Q. Weinberger, editors, *Advances in Neural Information Processing Systems*, volume 25. Curran Associates, Inc., 2012. 1

[20] Alexey Kurakin, Ian J. Goodfellow, and Samy Bengio. Adversarial machine learning at scale. In *International Conference on Learning Representations*, 2017. 3

[21] Saehyung Lee, Hyungyu Lee, and Sungroh Yoon. Adversarial vertex mixup: Toward better adversarially robust generalization. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2020. 2, 7, 8

[22] Z. Li, C. Feng, J. Zheng, M. Wu, and H. Yu. Towards adversarial robustness via feature matching. *IEEE Access*, 8:88594–88603, 2020. 3, 4, 5

[23] Yanpei Liu, Xinyun Chen, Chang Liu, and Dawn Songo. Delving into transferable adversarial examples and black-box attacks. In *International Conference on Learning Representations*, 2017. 8

[24] Aleksander Madry, Aleksandar Makelov, Ludwig Schmidt, Dimitris Tsipras, and Adrian Vladu. Towards deep learning models resistant to adversarial attacks. In *International Conference on Learning Representations*, 2018. 1, 3, 5, 8

[25] Pratyush Maini, Eric Wong, and Zico Kolter. Adversarial robustness against the union of multiple perturbation models. In Hal Daumé III and Aarti Singh, editors, *Proceedings of the 37th International Conference on Machine Learning*, volume 119 of *Proceedings of Machine Learning Research*, pages 6640–6650. PMLR, 13–18 Jul 2020. 2

[26] Yuval Netzer, Tao Wang, Adam Coates, Alessandro Bissacco, Bo Wu, and Andrew Ng. Reading digits in natural images with unsupervised feature learning. *NIPS Workshop*, 01 2011. 6

[27] Papernot Nicolas, McDaniel Patrick, Goodfellow Ian, Jha Somesh, Celik Z. Berkay, and Swami Ananthram. Practical black-box attacks against machine learning. In *Proceedings of the 2017 ACM on Asia Conference on Computer and Communications Security*, ASIA CCS '17, page 506–519, New York, NY, USA, 2017. Association for Computing Machinery. 1, 8

[28] Tianyu Pang, Xiao Yang, Yinpeng Dong, Hang Su, and Jun Zhu. Bag of tricks for adversarial training. In *International Conference on Learning Representations*, 2021. 3, 6

[29] Haifeng Qian and Mark N. Wegman. L2-nonexpansive neural networks. In *International Conference on Learning Representations*, 2019. 1, 3

[30] Arash Rahnama, Andre T. Nguyen, and Edward Raff. Robust design of deep neural networks against adversarial attacks based on lyapunov theory. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2020. 3

[31] Joseph Redmon, Santosh Divvala, Ross Girshick, and Ali Farhadi. You only look once: Unified, real-time object detection. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2016. 1

[32] Leslie Rice, Eric Wong, and Zico Kolter. Overfitting in adversarially robust deep learning. In Hal Daumé III and Aarti Singh, editors, *Proceedings of the 37th International Conference on Machine Learning*, volume 119 of *Proceedings of Machine Learning Research*, pages 8093–8104. PMLR, 13–18 Jul 2020. 3

[33] A. Ross and Finale Doshi-Velez. Improving the adversarial robustness and interpretability of deep neural networks by regularizing their input gradients. In *AAAI Conference on Artificial Intelligence*, 2018. 3

[34] Ali Shafahi, Mahyar Najibi, Mohammad Amin Ghiasi, Zheng Xu, John Dickerson, Christoph Studer, Larry S Davis, Gavin Taylor, and Tom Goldstein. Adversarial training for free! In H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. Fox, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 32. Curran Associates, Inc., 2019. 3

[35] Christian Szegedy, Wojciech Zaremba, Ilya Sutskever, Joan Bruna, Dumitru Erhan, Ian Goodfellow, and Rob Fergus. Intriguing properties of neural networks. In *International Conference on Learning Representations*, 2014. 1

[36] Florian Tramer and Dan Boneh. Adversarial training and robustness for multiple perturbations. In H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. Fox, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 32. Curran Associates, Inc., 2019. 2

[37] Florian Tramèr, Alexey Kurakin, Nicolas Papernot, Ian Goodfellow, Dan Boneh, and Patrick McDaniel. Ensemble adversarial training: Attacks and defenses. In *International Conference on Learning Representations*, 2018. 1

[38] Jianyu Wang and Haichao Zhang. Bilateral adversarial training: Towards fast training of more robust models against adversarial attacks. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, October 2019. 1, 3, 5

[39] Eric Wong, Leslie Rice, and J. Zico Kolter. Fast is better than free: Revisiting adversarial training. In *International Conference on Learning Representations*, 2020. 3, 5

[40] Yuichi Yoshida and Takeru Miyato. Spectral Norm Regularization for Improving the Generalizability of Deep Learning. *arXiv preprint*, arXiv:1705.10941, May 2017. 1, 3

[41] Sergey Zagoruyko and Nikos Komodakis. Wide residual networks. *arXiv preprint*, abs/1605.07146, 05 2016. 6

[42] Hongyi Zhang, Moustapha Cisse, Yann N. Dauphin, and David Lopez-Paz. mixup: Beyond empirical risk minimization. In *International Conference on Learning Representations*, 2018. 2, 8

[43] Haichao Zhang and Jianyu Wang. Defense against adversarial attacks using feature scattering-based adversarial training. In H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. Fox, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 32. Curran Associates, Inc., 2019. 1, 3, 5, 6, 7

[44] Haichao Zhang and Wei Xu. Adversarial interpolation training: A simple approach for improving model robustness. 2020. 1, 3, 5, 6

[45] Hongyang Zhang, Yaodong Yu, Jiantao Jiao, Eric Xing, Laurent El Ghaoui, and Michael Jordan. Theoretically principled trade-off between robustness and accuracy. In Kamalika Chaudhuri and Ruslan Salakhutdinov, editors, *Proceedings of the 36th International Conference on Machine Learning*, volume 97 of *Proceedings of Machine Learning Research*, pages 7472–7482. PMLR, 09–15 Jun 2019. 3

[46] Bolei Zhou, Aditya Khosla, Agata Lapedriza, Aude Oliva, and Antonio Torralba. Learning deep features for discriminative localization. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2016. 4