Adversarial Attacks are Reversible with Natural Supervision

Chengzhi Mao[1] Mia Chiquier[1] Hao Wang[2] Junfeng Yang[1]
Carl Vondrick[1]
[1]Columbia University, [2]Rutgers University
{cm3797,mac2500}@columbia.edu, hoguewang@gmail.com, {junfeng, vondrick}@cs.columbia.edu

Abstract

We find that images contain intrinsic structure that enables the reversal of many adversarial attacks. Attack vectors cause not only image classifiers to fail, but also collaterally disrupt incidental structure in the image. We demonstrate that modifying the attacked image to restore the natural structure will reverse many types of attacks, providing a defense. Experiments demonstrate significantly improved robustness for several state-of-the-art models across the CIFAR-10, CIFAR-100, SVHN, and ImageNet datasets. Our results show that our defense is still effective even if the attacker is aware of the defense mechanism. Since our defense is deployed during inference instead of training, it is compatible with pre-trained networks as well as most other defenses. Our results suggest deep networks are vulnerable to adversarial examples partly because their representations do not enforce the natural structure of images.

1. Introduction

Deep networks achieve strong performance over a number of computer vision tasks, yet they remain brittle under adversarial attacks [11, 6, 16, 55]. With crafted perturbations, attackers can undermine predictions from the state-of-the-art models by changing the features in the representation [36]. These limitations prevent application of deep networks to sensitive and safety-critical applications [46, 52, 32, 53], underscoring the gap between current machine learning algorithms and human-level abilities [5].

A large body of work has studied how to *train* deep networks such that they are robust to adversarial attacks. Adversarial training and its variants [34, 36, 55, 42, 47], including multitask learning [35, 25] and semi-supervised learning [58], significantly improve robustness. However, while existing methods focus on improving the training algorithm, they are burdened because they need to find a single representation that *also* works for all possible corruptions and attacks. Training-based defenses cannot adapt to the individual characteristics of each attack at testing-time.

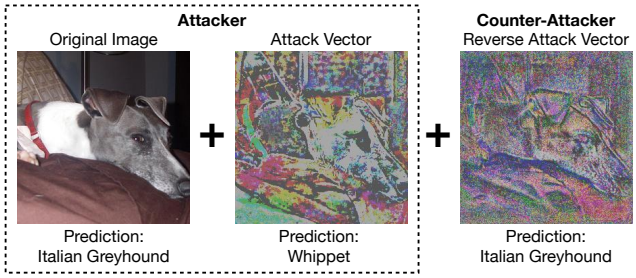


Figure 1: **Reverse Attacks:** Adversarial attacks are small perturbations that cause classification networks to fail [34, 55]. In this paper, we show there are intrinsic signals in natural images to reverse many types of attacks. In the right column, we visualize our reverse attack on an ImageNet image. Note that both attack vectors have been multiplied by ten for visualization purposes only.

In this paper, we introduce an approach for *reversing* the attack process, allowing us to formulate a defense strategy that adapts to each attack during the testing phase. Just as an attacker finds the right additive perturbation to *break* the input, our approach will find the right additive perturbation to *repair* the input. Figure 1 shows our reverse attack on a poisoned ImageNet image. However, reverse attacks are more challenging to produce than standard attacks because the category label is unknown to us during testing.

Our key insight is that images contain natural and intrinsic structure that we can leverage to reverse many types of adversarial attacks. We found that, although adversarial attacks aim to fool the image classifier, they also collaterally damage self-supervised objectives. Our approach shows how to capitalize on this incidental signal in order to create adversarial defenses. By using self-supervision for defense at test time, we can guarantee that even the strongest adversary cannot manipulate the intrinsic signals that naturally come with the images, providing a more robust defense than training-based methods.

A pivotal advantage of our framework is that it factors out the defense strategy from the visual representation. Since reverse attacks are adaptive, this defense is able to

efficiently scale to any corruption that violates the natural image manifold.

Moreover, the modularity of our approach allows it to work with any classifier and complements existing defense models. It can also be integrated into future defense models and defend against novel attacks that corrupt natural image structures.

Visualizations, empirical experiments, and theoretical analysis show that our reversal strategy significantly improves robust prediction for several established benchmarks and attacks. Under attacks with up to 200 steps, our method advances the state-of-the-art defense methods by a large margin across four natural image datasets including CIFAR-10 (over 2.9% gain), CIFAR-100 (over 1.6% gain), SVHN (over 6.8% gain), and ImageNet (over 3.0% gain). Our method is robust against established attacks, including PGD [34] and C&W [6]. In addition, our empirical results demonstrate that, even when the attacker is aware of our defense mechanism, our approach remains robust. Our models, data, and code are available at https://github.com/cvlab-columbia/SelfSupDefense.

2. Related Work

Self-supervised Learning: Natural images contain rich information for representation learning. Self-supervised learning enables us to learn high quality representations from images without annotations [13, 9, 69, 21, 10, 3, 7, 45]. By solving pretext tasks, such as jigsaw puzzles [40], image inpainting [45], rotation prediction [17], image colorization [69, 60], random walk [26], and clustering [8], the learned representations can generalize to unseen downstream tasks such as image recognition [9], and also allow domain adaptation at testing time [54]. Recently, contrastive learning has significantly advanced image recognition [9, 21, 37, 19]. In this paper, we leverage this incidental structure to correct adversarial attacks. Our defense uses the contrastive learning task [9], and it is extensible to existing self-supervised tasks as well [45, 40, 17].

Adversarial Robustness: A large number of adversarial attacks have been proposed to fool deep models [55, 1, 6, 31, 43, 38]. Special adversarial attacks that can be reversed to clean images are also proposed [65]. Different from the existing approach to construct reversible attacks [65], our approach aims to reverse any unknown attacks for defense. While many defense methods are proved to be not robust [48, 66, 64, 33, 2, 59, 41, 20, 50, 14, 4] as they relied on gradient obfuscation, gradient masking [1, 5], and weak adaptive attack evaluation [56], adversarial training and its variants are proved to achieve the true robustness [18, 34, 36, 68, 47, 42, 62, 61, 63]. Moreover, recent progress shows that unlabeled data [58, 24] and self-supervised learning [25] improve the robustness of deep models. While training a robust neural network to defense is vastly studied, no existing work investigates algorithms that improve robustness at inference time.

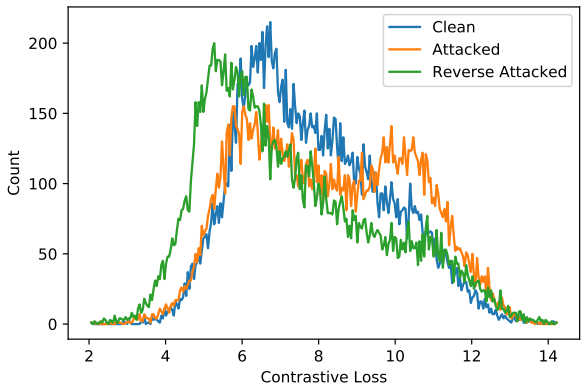


Figure 2: **Contrastive Score Distribution:** We histogram the constrastive loss value [9] for natural images (blue), after adversarial attack (orange), and after our reverse attack (green). This plot shows that adversarial attacks cause the contrastive loss to increase. We create a counter-attack by finding a perturbation that restores the self-supervised loss.

3. Method

We will first present a reverse attack that uses self-supervision at deployment time to defend against adversarial attacks. We then analyze the case where the attacker is aware of our defense, and show our defense remains effective. We finally provide theoretical justification for the robustness of our approach.

3.1. Attacks and Reverse Attacks

Let $\mathbf{x}$ be an input image, and $\mathbf{y}$ be its ground-truth category label. To perform classification, neural networks commonly learn to predict the category $\hat{\mathbf{y}} = F_\theta(\mathbf{x})$ by optimizing the cross entropy $H(\hat{\mathbf{y}}, \mathbf{y})$ between the predictions and the ground truth.

The network parameters θ are estimated by minimizing the expected value of the objective:

$$\mathcal{L}_c(\mathbf{x}, \mathbf{y}) = H\left(F_\theta(\mathbf{x}), \mathbf{y}\right), \tag{1}$$

which can be optimized with gradient-based descent.

The Attack: In order to corrupt this model, the adversarial attack finds additive perturbations $\boldsymbol{\delta}$ to the image such that $\mathbf{x}_a = \mathbf{x} + \boldsymbol{\delta}$ is no longer classified correctly by the trained network F_θ.

Attackers create these worst-case images by maximizing the objective:

$$\mathbf{x}_a = \underset{\mathbf{x}_a}{\operatorname{argmax}}\, \mathcal{L}_c(\mathbf{x}_a, \mathbf{y}), \quad \text{s.t.} \quad ||\mathbf{x}_a - \mathbf{x}||_q < \epsilon, \tag{2}$$

where the q norm bound of the perturbation $\boldsymbol{\delta} = \mathbf{x}_a - \mathbf{x}$ is less than ϵ, which keeps the perturbation minimal.

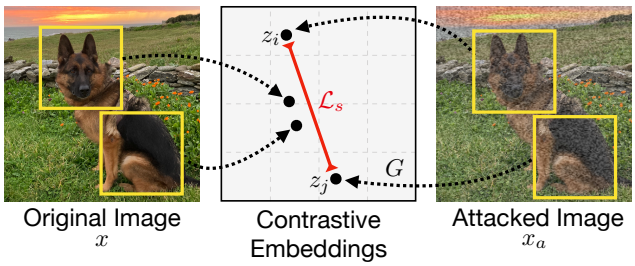


Figure 3: **Defense Overview:** We find that adversarial attacks on classification networks will also collaterally attack self-supervised contrastive networks [9]. Since self-supervision is available during deployment, we exploit this discrepancy to reverse adversarial attacks and provide a defense. Our approach modifies the potentially attacked input image such that contrastive distances $\mathcal{L}_s$ are restored.

The Reverse: We aim to defend against these attacks by reversing the attack process. Just as the attack finds an additive perturbation to *break* the input, we will find an additive perturbation to *repair* the input. However, we cannot simply flip Equation 2 from a maximization to a minimization because the category labels $\mathbf{y}$ are unknown at deployment.

The key observation is that self-supervised objectives are always available because they do not depend on the labels $\mathbf{y}$. While adversarial attacks aim to corrupt the classifier, they will also impact self-supervised representations, which is a signal we will leverage for reversal. Let $\mathcal{L}_s(\mathbf{x})$ be a self-supervised objective on the input $\mathbf{x}$. We create the reverse attack vector $\mathbf{r}$ by minimizing the objective:

$$\mathbf{r} = \underset{\mathbf{r}}{\operatorname{argmin}}\, \mathcal{L}_s(\mathbf{x}_a + \mathbf{r}), \quad \text{s.t.} \quad ||\mathbf{r}||_q < \epsilon_v, \qquad (3)$$

where ϵ_v defines the bound of our reverse attack. The solution $\mathbf{r}$ will modify the adversarial image $\mathbf{x}_a$ such that it satisfies our choice of self-supervised objective.

After finding the optimal $\mathbf{r}$, robust prediction is straightforward. Our defense adds the resulting perturbation vector $\mathbf{r}$ to the input $\mathbf{x}_a$ before predicting the classification result with the normal network forward pass: $\hat{\mathbf{y}} = F_\theta(\mathbf{x}_a + \mathbf{r})$.

An advantage of reverse attacks is that, since they do not rely on offline adversarial training, the defense will generalize to unseen adversarial attacks. Moreover, our defense is able to fortify existing models without re-training.

3.2. Natural Supervision for Defense

While any self-supervised task [45, 40, 17] can construct the loss $\mathcal{L}_s$, we use the contrastive loss as our natural supervision objective [9, 10], which is a state-of-the-art self-supervised representation learning approach. The contrastive objective creates features that maximize the agreement between positive pairs of examples while minimizing the agreement between negative pairs of examples. Pairs are typically created with an augmentation strategy [9]. In our case, when we receive a potentially adversarial image $\mathbf{x}$, we create the positive examples by sampling different augmentations from it to create multiple positive pairs. We create the negative pairs in a similar way, except applying augmentations to the randomly selected images.

Algorithm 1 Self-supervised Reverse Attack

Input: Potentially attacked image $\mathbf{x}$, step size η, number of iterations K, a classifier F, reverse attack bound ϵ_v, and self-supervised loss function $\mathcal{L}_s$.
Output: Class prediction $\hat{y}$
Inference:
$\mathbf{x}' \leftarrow \mathbf{x} + \mathbf{n}$, where $\mathbf{n}$ is the initial random noise
for $k = 1, ..., K$ **do**
  $\mathbf{x}' \leftarrow \mathbf{x}' - \eta \nabla_{\mathbf{x}'} \mathcal{L}_s(\mathbf{x}')$
  $\mathbf{x}' \leftarrow \Pi_{(\mathbf{x}, \epsilon_v)} \mathbf{x}'$, which projects the image back into the bounded region.
end for
Predict the final output by $\hat{y} = F(\mathbf{x}')$

Since these pairs are constructible at evaluation time, we create reverse attacks that minimize the term:

$$\mathcal{L}_s(\mathbf{x}) = -\mathbb{E}_{i,j}\left[\mathbf{y}_{ij}^{(s)} \log \frac{\exp(\cos(\mathbf{z}_i, \mathbf{z}_j)/\tau)}{\sum_k \exp(\cos(\mathbf{z}_i, \mathbf{z}_k)/\tau)}\right], \qquad (4)$$

where $\mathbf{z}$ are the contrastive features. We use $\mathbf{y}_{ij}^{(s)}$ to indicate which pairs are positive and which are negative. This indicator satisfies $\mathbf{y}_{ij}^{(s)} = 1$ if and only if the examples i and j are both from $\mathbf{x}$, and 0 otherwise. τ is a scalar hyper-parameter, and cos denotes cosine similarity.

Figure 2 shows that adversarial attacks on classification objectives also attack the contrastive objective $\mathcal{L}_s$, even though the attacker never explicitly optimizes for it. When there is an attack, $\mathcal{L}_s(\mathbf{x}_a)$ will be larger than on clean images $\mathcal{L}_s(\mathbf{x})$. This gap provides the signal for reverse attacks.

Figure 3 provides an overview of this defense mechanism, and Algorithm 1 summarizes our procedure.

Contrastive Feature Estimation: To estimate the contrastive features $\mathbf{z}$, we take the features before logits from a backbone F and pass them to a two-layer network G. To compute the positive features, we sample augmentations conditioned on the input image $\mathbf{x}$. We follow a similar procedure to compute the contrastive features for the negative examples $\mathbf{z}_k$, sampling random images from a collection of images that form the negative set.

Offline, we fit the contrastive model G on a large set of clean images using the same procedure as [9, 10]. We sequentially apply two augmentations: random cropping then scale back to the original size, and random color distortions including color jittering and random gray-scale. We found removing the Gaussian blur from the augmentations

improved performance because it otherwise favored over-smooth perturbations. After G is trained on clean images, we use it during reverse attacks without any further training.

3.3. Analysis of Defense Aware Attack

In this section, we analyze the effectiveness of our approach when the attacker is aware of our defense.

Attack Model: Let us assume the attacker knows the contrastive model parameters and our defense strategy. In this setting, the attacker can adversarially optimize against our defense with the following alternating optimization:

$$\mathbf{r} = \underset{\mathbf{r}}{\operatorname{argmin}} \mathcal{L}_s(\mathbf{x} + \mathbf{r}), \tag{5}$$

$$\boldsymbol{\delta} = \underset{\boldsymbol{\delta}}{\operatorname{argmax}} \mathcal{L}_c(\mathbf{x} + \mathbf{r} + \boldsymbol{\delta}, \mathbf{y}). \tag{6}$$

From the attacker's perspective, the above procedure is not ideal because it involves an alternating, min-max optimization. Past work suggests that this leads to unstable gradient estimation, having a gradient obfuscation problem that reduces the attack efficiency [1].

Similar to C&W [6] and L-BFGS attack [55], the attacker can reformulate the above equation as a constrained optimization problem:

$$\text{maximize} \quad \mathcal{L}_c(\mathbf{x}_a, \mathbf{y}), \quad \text{s.t.} \quad \mathcal{L}_s(\mathbf{x}_a) \leq \epsilon', \tag{7}$$

where ϵ' is the same value as the converged loss $\mathcal{L}_s$ for natural images. Intuitively, the attacker should maximize the adversarial gain while respecting the self-supervised loss if they want to render our defense ineffective.

To optimize Equation 24, they in practice maximize the following equation w.r.t. $\mathbf{x}_a$:

$$\mathcal{L}_l(\mathbf{x}_a, \mathbf{y}, \lambda_s) = \mathcal{L}_c(\mathbf{x}_a, \mathbf{y}) - \lambda_s \mathcal{L}_s(\mathbf{x}_a). \tag{8}$$

We derive Equation 8 from Equation 24 via the Lagrange Penalty method [49], where $\mathcal{L}_l$ is the new loss for the adaptive attack. Full derivations are in the supplementary.

Multi-objective Trade-off: The above derivation shows that the attacker can attempt to bypass our reversal by also minimizing $\mathcal{L}_s(\mathbf{x}_a)$, so that attacks mimic the self-supervised features of clean examples. If the attacker produces examples that are as good as the clean examples in terms of $\mathcal{L}_s(\mathbf{x}_a)$, our defense would not be able to reverse the attack by further decreasing the loss $\mathcal{L}_s(\mathbf{x}_a + \mathbf{r})$.

However, as the attacker must solve a multi-objective optimization, they must trade-off between the two objectives. The scalar λ_s controls how aggressively the attacker will corrupt the self-supervised model. The attacker's ideal adversarial attack will first optimize for the Pareto frontier by maximizing $\mathcal{L}_l(\mathbf{x}_a, \mathbf{y}, \lambda_s)$ for each λ_s. They should select the λ_s that yields the most damage (the lowest robust accuracy), and use the corresponding generated attack $\mathbf{x}_a^*$.

A larger λ_s shifts the adversarial budget from attacking the classification loss $\mathcal{L}_c$ to attacking the self-supervised loss $\mathcal{L}_s$. If the attacker is to attack the self-supervised task, they would then reduce the effectiveness of their classification attack, undermining their goal. Attacking both $\mathcal{L}_s$ and $\mathcal{L}_c$ jointly requires creating adversarial images for multiple objectives, which is fundamentally more challenging [35].

Our defense creates a lose-lose situation for the attacker. If they ignore our defense, then we improve accuracy. If they account for our defense, then they hurt their attack.

3.4. Theoretical Analysis

We will show theoretical insights for why leveraging natural supervision improves adversarial robustness. Without our defense, the model predicts the category on an image with an incorrect estimate for the self-supervision label. With our defense, the model uses an image for which the self-supervision label is estimated correctly. We prove this increases the upper bound of the prediction accuracy.

A feed forward pass is equivalent to also including a latent self-supervised label to the model, since the information which the self-supervised network uses is in the image itself. We denote the ground-truth label of self-supervision as $\mathbf{y}^{(s)}$ and the predicted label of self-supervision under attack as $\mathbf{y}_a^{(s)}$. To make this latent label explicit in notation, we rewrite the loss functions as: $\mathcal{L}_s(\mathbf{x}) \rightarrow \mathcal{L}_s(\mathbf{x}, \mathbf{y}^{(s)})$ and $\mathcal{L}_s(\mathbf{x}_a) \rightarrow \mathcal{L}_s(\mathbf{x}_a, \mathbf{y}_a^{(s)})$.

Lemma 1. *The standard classifier under adversarial attack is equivalent to predicting with* $P(\mathbf{Y}|\mathbf{X} = \mathbf{x}_a, \mathbf{Y}^{(s)} = \mathbf{y}_a^{(s)})$*, and our approach is equivalent to predicting with* $P(\mathbf{Y}|\mathbf{X} = \mathbf{x}_a, \mathbf{Y}^{(s)} = \mathbf{y}^{(s)})$.

Proof. For the standard classifier under attack, we know that $P(\mathbf{Y}^{(s)} = \mathbf{y}_a^{(s)}|\mathbf{X} = \mathbf{x}_a) = 1$. Thus we know the standard classifier under adversarial attack is equivalent to

$$\begin{aligned} P(\mathbf{Y}|\mathbf{X} = \mathbf{x}_a) &= \sum_{\mathbf{Y}^{(s)}} P(\mathbf{Y}^{(s)}|\mathbf{X} = \mathbf{x}_a) P(\mathbf{Y}|\mathbf{Y}^{(s)}, \mathbf{X} = x_a) \\ &= P(\mathbf{Y}|\mathbf{Y}^{(s)} = \mathbf{y}_a^{(s)}, \mathbf{X} = \mathbf{x}_a). \end{aligned}$$

Our algorithm finds a new input image $\mathbf{x}_{\max}^{(n)}$ that

$$\begin{aligned} &\underset{\mathbf{x}^{(n)}}{\operatorname{argmax}} P(\mathbf{X}^{(n)} = \mathbf{x}^{(n)}|\mathbf{X} = \mathbf{x}_a) P(\mathbf{Y}^{(s)} = \mathbf{y}^{(s)}|\mathbf{X}^{(n)} = \mathbf{x}^{(n)}) \\ &= \underset{\mathbf{x}^{(n)}}{\operatorname{argmax}} P(\mathbf{X}^{(x)} = \mathbf{x}^{(n)}|\mathbf{X} = \mathbf{x}_a, \mathbf{Y}^{(s)} = \mathbf{y}^{(s)}). \end{aligned}$$

Our algorithm first estimate $\mathbf{x}_{\max}^{(n)}$ with adversarial image $\mathbf{x}_a$ and self-supervised label $\mathbf{y}^{(s)}$. We then predict the label $\mathbf{Y}$ using our new image $\mathbf{x}_{\max}^{(n)}$. Thus, our approach in fact estimates $P(\mathbf{Y}|\mathbf{X}^{(n)} = \mathbf{x}_{\max}^{(n)}) P(\mathbf{X}^{(n)} = \mathbf{x}_{\max}^{(n)}|\mathbf{X} =$

$\mathbf{x}_a, \mathbf{Y}^{(s)} = \mathbf{y}^{(s)}$). Note the following holds:

$$P(\mathbf{Y}|\mathbf{X} = \mathbf{x}_a, \mathbf{Y}^{(s)} = \mathbf{y}^{(s)}) \tag{9}$$

$$= \sum_{\mathbf{x}^{(n)}} P(\mathbf{Y}|\mathbf{x}^{(n)}) P(\mathbf{x}^{(n)}|\mathbf{X} = \mathbf{x}_a, \mathbf{Y}^{(s)} = \mathbf{y}^{(s)}) \tag{10}$$

$$\approx P(\mathbf{Y}|\mathbf{X}^{(n)} = \mathbf{x}^{(n)}_{\max}) P(\mathbf{X}^{(n)} = \mathbf{x}^{(n)}_{\max}|\mathbf{X} = \mathbf{x}_a, \mathbf{Y}^{(s)} = \mathbf{y}^{(s)}) \tag{11}$$

Thus our approach is equivalent to estimating $P(\mathbf{Y}|\mathbf{X} = \mathbf{x}_a, \mathbf{Y}^{(s)} = \mathbf{y}^{(s)})$. □

We use the maximum a posteriori (MAP) estimation $\mathbf{x}^{(n)}_{\max}$ to approximate the sum over $\mathbf{X}^{(n)}$ because: (1) sampling a large number of $\mathbf{X}^{(n)}$ is computationally expensive; (2) our results in Figure 7 shows that random sampling is ineffective; (3) our MAP estimate naturally produces a denoised image that can be useful for other downstream tasks.

Next we provide theoretical guarantees that our approach can strictly improve the bound for classification accuracy in Theorem 1. For convenience we introduce an additional random variable $\mathbf{X}_a$ representing the adversarial image.

Theorem 1. *Assume the base classifier operates better than chance and instances in the dataset are uniformly distributed over n categories. Let the prediction accuracy bounds be $P(\mathbf{Y}|\mathbf{Y}^{(s)}_a, \mathbf{X}_a) \in [b_1, c_1]$ and $P(\mathbf{Y}|\mathbf{Y}^{(s)}, \mathbf{X}_a) \in [b_2, c_2]$. If the conditional mutual information $I(\mathbf{Y}; \mathbf{Y}^{(s)}|\mathbf{X}_a) > 0$, we have $b_2 \geq b_1$ and $c_2 > c_1$, which means our approach strictly improves the bound for classification accuracy.*

Proof. If $I(\mathbf{Y}; \mathbf{Y}^{(s)}|\mathbf{X} = \mathbf{x}_a) > 0$, then it is straightforward that:

$$I(\mathbf{Y}; \mathbf{Y}^{(s)}, \mathbf{X}_a) > I(\mathbf{Y}; \mathbf{Y}^{(s)}_a, \mathbf{X}_a) = I(\mathbf{Y}; \mathbf{X}_a).$$

We define $H(\epsilon_p) = -\epsilon_p \log \epsilon_p - (1 - \epsilon_p) \log(1 - \epsilon_p)$. Using the *Fano's Inequality* [51] and the fact that $Q(\epsilon_p) = H(\epsilon_p) + \epsilon_p \log(n - 1)$ is a monotonically increasing function when error rate $\epsilon_p < 1 - \frac{1}{n}$, i.e., accuracy higher than random guessing,[1] we derive the upper bound of accuracy c_1 and c_2 to be:

$$1 - \epsilon_p \leq c_1 = 1 - Q^{-1}(-I(\mathbf{Y}; \mathbf{X}_a) + H(\mathbf{Y})),$$
$$1 - \epsilon_p \leq c_2 = 1 - Q^{-1}(-I(\mathbf{Y}; \mathbf{Y}^{(s)}, \mathbf{X}_a) + H(\mathbf{Y})),$$

where the upper bound is a function of the mutual information. Since $H(\mathbf{Y})$ is a constant, a larger mutual information will strictly increase the bound. Detailed proof is in the supplementary material. □

[1]The validity of this fact are explained in the supplementary.

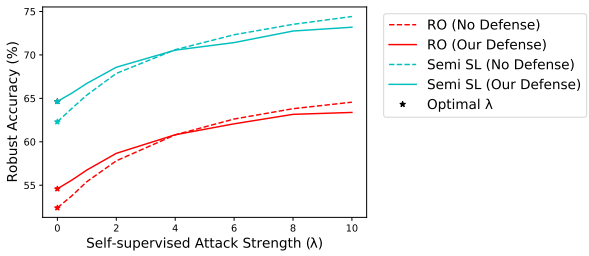


Figure 4: **The Trade-off:** The robust accuracy for defense aware attack under different λ_s setup. We increase the λ_s of defense aware attack from 0 to 10, and show robustness accuracy on two robust models, RO and Semi-SL, on CIFAR-10 dataset. The plot shows a trade-off. While increasing the value of λ_s decreases the gain of Ours compared with Baseline, it also decreases the attacker's effectiveness. To achieve the best attack effectiveness, the attacker should use $\lambda_s = 0$, which is the standard attack without attempting to corrupt the self-supervised task.

Intuitively, the adversarial attack $\mathbf{x}_a$ will corrupt some mutual information between the label $\mathbf{Y}$ and natural structure $\mathbf{Y}^{(s)}$. Thus, there is additional mutual information between $\mathbf{Y}$ and $\mathbf{Y}^{(s)}$ given $\mathbf{x}_a$, i.e., $I(\mathbf{Y}; \mathbf{Y}^{(s)}|\mathbf{X} = \mathbf{x}_a) > 0$. Theorem 1 shows that by restoring information from the correct $\mathbf{Y}^{(s)}$, the prediction accuracy can be improved.

Theoretically, by optimizing the self-supervision loss, the defense aware attack is in fact predicting classification label given the right self-supervision label $\mathbf{Y}^{(s)}$, i.e., $P(\mathbf{Y}|\mathbf{X} = \mathbf{x}_a, \mathbf{Y}^{(s)} = \mathbf{y}^{(s)})$. According to our theory, the robust accuracy should increase due to the restored information. Overall, our defense is robust even under a defense aware adversary.

4. Experiments

Our experiments evaluate the robustness at image classification on four datasets: CIFAR-10 [29], CIFAR-100 [30], SVHN [39], and ImageNet [12]. We compare with the state-of-the-art defense methods, under several strong adversarial attacks including a defense aware attack.

4.1. Baselines

We apply our method to seven established, scrutinized [1] defense methods including the state-of-the-art adversarial robust model. All studied methods are trained with adversarial training [34], but achieve higher robust accuracy than the initial version of Madry et al. [34].

TRADES [68] is the winning solution for NeurIPS 2018 Adversarial Vision Challenge. It introduces a KL-divergence term to regularize the representation of adversarial examples to match the ones of clean examples.

Robust Overfit (RO) [47] re-examines the existing adversarial robust models through overfitting, which is the

Inference Type	Model Architecture	Adversarial Attack $L_\infty = 8/255$ PGD 50 Standard	PGD 50 Ours	PGD 200 Standard	PGD 200 Ours	BIM 200 Standard	BIM 200 Ours	C&W 200 Standard	C&W 200 Ours
TRADES [68]	WRN-34-10	55.05%	**57.00%**	55.02%	**57.18%**	55.06%	**57.33%**	53.72%	**56.56%**
RO [47]	PreRes-18	52.40%	**54.59%**	52.34%	**54.62%**	52.32%	**54.54%**	50.33%	**53.48%**
BagT [42]	WRN-34-10	56.44%	**58.33%**	56.40%	**58.47%**	56.38%	**58.55%**	54.82%	**57.35%**
MART [61]	WRN-28-10	62.72%	**64.40%**	62.63%	**64.26%**	62.54%	**64.28%**	58.96%	**62.18%**
AWP [63]	WRN-28-10	63.67%	**64.21%**	63.64 %	**64.07%**	63.64%	**63.69%**	60.82%	**61.90%**
Semi-SL [58]	WRN-28-10	62.30%	**64.64%**	62.22%	**64.44%**	62.18%	**64.68%**	60.90%	**63.83%**

Table 1: Adversarial robust accuracy on the CIFAR-10 test set. Our method improves robustness of established work across different adversarial attack setups, including the state-of-the-art method, by over **2.9%**. The lower bound for robustness of the state-of-the-art and ours are boxed.

Inference Type		Adversarial Attack $L_\infty = 8/255$ PGD 50 Standard	PGD 50 Ours	PGD 200 Standard	PGD 200 Ours	BIM 200 Standard	BIM 200 Ours	C&W 200 Standard	C&W 200 Ours
RO [47]	ResNet18	21.90%	**23.83%**	20.90%	**22.23%**	21.00%	**22.09%**	20.42%	**22.19%**
TRADES* [68]	WRN34-10	27.04%	**28.55%**	26.94%	**28.16%**	26.94%	**28.18%**	26.57%	**27.88%**
BagT * [42]	WRN34-10	29.87%	**31.21%**	28.81%	**31.15%**	29.81%	**31.44%**	29.72%	**31.40%**

Table 2: Adversarial robust accuracy on the CIFAR-100 test set. Our method consistently improves robustness by over 1.6%.

state-of-the-art model trained with Pre-ResNet18.

Bag of Tricks (BagT) [42] conducts extensive experiments on the effect of hyper-parameters on adversarial training [34]. It is the state-of-the-art adversarial robust model without additional unlabeled data for training.

Semi-supervised Learning (Semi-SL) [58] significantly improves adversarial robustness using unlabeled data. By training with pseudo labels of unlabeled images, the model achieved the state-of-the-art robustness. However, this work neglects the information of natural images beyond the pseudo classification label.

MART [61] uses misclassification aware adversarial training to achieve improved robustness. We use its best version trained on top of Semi-SL [58].

Adversarial Weight Perturbation (AWP) [63] trains robust model by smoothing the weights' loss landscape. We use its best version trained on top of Semi-SL [58].

Fast is Better than Free (FBF) [62] is the state-of-the-art solution for training robust ImageNet classifier in reasonable training budget and time.

4.2. Attack Methods

Fast Gradient Sign Method (FGSM) [18] is a one-step adversarial attack to fool neural networks.

Projected Gradient Descent (PGD) [34] is the standard evaluation for adversarial robustness, which optimizes the adversarial noise with gradient descent for iterations, and project the noise back to the nearest boundary if it is out of the given bound.

Basic Iterative Attack (BIM) [31] is a variant of PGD attack without the initial random start.

C&W Attack [6] is a powerful iterative attack that has been widely used for robustness evaluation. It reduces the logit value for the right class while increasing that for the second best class to fool the classifier.

Defense Aware Attack is discussed in Section 3.3, which is theoretically the optimal adaptive white-box attack to bypass our defense algorithm.

4.3. Experimental Settings

Backbone Architectures. Following prior literature, we conduct experiments with Pre-ResNet18 [23], ResNet50 [22], and WideResNet [67]. We download the pretrained models' weights online. [2]

Self-supervised Learning Branch. We use a network with two fully connected layers that takes in the features from the penultimate layer of the backbone network.

Implementation Details. We train our self-supervision model with the Adam [28] optimizer. We use a learning rate of 0.001. When training the self-supervised model, we use temperature $\tau = 0.2$ for the contrastive loss, with a batch size of 128. For CIFAR-10 and CIFAR-100, we train the self-supervised branch for 200 epochs. For SVHN, we train it for 600 epochs. For ImageNet, we train for 30 epochs. We set the reverse attack bound to be $\epsilon_v = 2\epsilon$ and optimization iterations to be $K = 40$. We implement our model with Pytorch [44]. Please see supplementary for details.

4.4. Results of Defense Aware Adversarial Attacks

In Section 3.3, we discussed the strongest adaptive attack that can be used to bypass our defense. We show the results of the adaptive attacker on CIFAR-10 in Figure 4. We use attacks with 50 steps, with perturbation bound $L_\infty = 8/255$. We vary the value of the λ_s from 0 to 10, where 0 corresponds to the standard PGD attack without

[2] We reproduce a few models that are not available and denote by *

Inference Type	Model Architecture	Adversarial Attack $L_\infty = 8/255$ PGD 50 Standard	PGD 50 Ours	PGD 200 Standard	PGD 200 Ours	BIM 200 Standard	BIM 200 Ours	C&W 200 Standard	C&W 200 Ours
RO [47]	PreRes-18	51.03%	**53.21%**	50.82%	**53.06%**	50.84%	**53.26%**	48.90%	**54.62%**
Semi-SL [58]	WRN-28-10	55.69%	**62.40%**	55.29%	**62.12%**	55.43%	**62.53%**	57.03%	**63.34%**

Table 3: Adversarial robust accuracy on the SVHN test set under different attacks. Our method improves robustness including the state-of-the-art semi-supervised learning model, by over **6.8%**.

	Adversarial Attack $L_\infty = 4/255$ FGSM	PGD10	PGD 20	BIM 20	C&W 20
FBF [62]	32.47%	28.68%	28.52%	28.49%	27.81%
Ours + FBF	**33.51%**	**31.36%**	**31.32%**	**31.27%**	**30.88%**

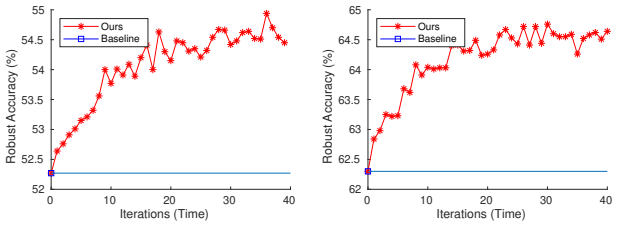
Table 4: Adversarial robust accuracy on the ImageNet test set. Our method uses the same model and parameters as the baseline FBF [62]. After reversing the adversarial attack with our natural supervison, we improve robustness on ImageNet by over **3.07%**.

Inference Type	Adversarial Attack $L_2 = 256/255$ PGD 200 Standard	PGD 200 Ours	C&W 200 Standard	C&W 200 Ours
TRADES [68]	36.90%	**39.16%**	35.89%	**38.07%**
RO [47]	40.00%	**41.98%**	38.31%	**40.17%**
BagT [42]	38.80%	**41.28%**	37.01%	**39.20%**
Semi-SL [58]	42.07%	**44.47%**	39.97%	**42.72%**
AWP [63]	44.39%	**47.15%**	40.97%	**43.72%**
MART [61]	43.24%	**45.85%**	43.23%	**45.85%**

Table 5: L_2 norm bounded adversarial robust accuracy on the CIFAR-10. Our natural supervision is agnostic to the attack type and can improve the robustness by over 2.6% for L_2 attack without retraining the defense model. The lower bound for the best achieved robustness on L_2 is boxed.

considering our defense strategy. The results show that increasing λ_s and focusing more attack budget to the self-supervised defense, the gain of our approach is reduced (full line falls under dotted line). However, as λ_s gets larger, the attack for classification task also gets weaker (line goes up). While adaptive attacks successfully reduce the additional gain bought by our approach, it too significantly sacrifices the initial attack success rate on the classification task. Minimizing the $\mathcal{L}_s(\mathbf{x}_a, \mathbf{y}^{(s)})$ hurts the original classification attack so much that it is not worth it for the attacker to account for our defense. We also show results with 500 steps in the supplementary, where our conclusion also holds.

This finding matches our initial theory in Section 3.4, where leveraging the incidental structure in the images improves robustness. The decrease of attack success rate on the target classification task is also consistent with prior work [35], which suggests that it is harder to simultaneously attack multiple tasks at once. In fact, the attacker is trading off between classification attack success rate and fooling the self-supervised defense. For the attacker, the optimal attack is $\lambda_s = 0$, which is the standard adversarial attack without considering our self-supervised defense. Therefore, we use this setup in the remaining experiments.



(a) RO [47] (b) Semi-SL [58]

Figure 5: **Speed versus Robust Accuracy:** Trade-off between inference time and adversarial robustness on CIFAR-10 dataset. As our approach is iterative, we can stop early if the application prefers speed over the robustness.

4.5. Results with Optimal Adversarial Attacks

In the optimal setup, $\lambda_s = 0$. The attack is equivalent to the standard adversarial attack without considering the defense branch. Thus the gain from *Standard* to *Ours* is the lower bound of our self-supervised correction. We now show the gain on four datasets.

CIFAR-10 [29] contains 10 categories. In Table 1, we add our approach to six existing robust models including the state-of-the-art, where we constantly correct by up to 2.9% of the adversarial examples, as shown in the gain from Standard to Ours.

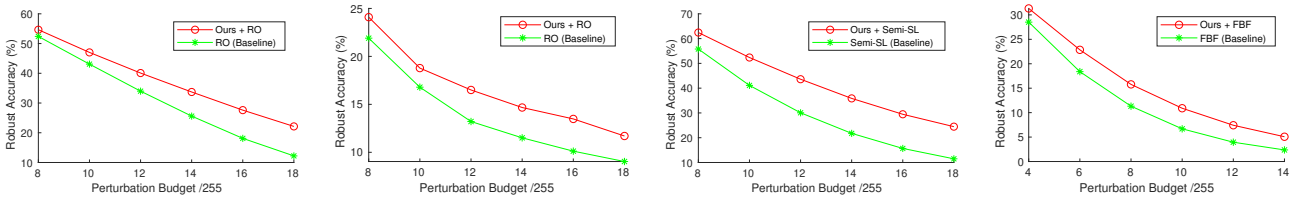
CIFAR-100 [30] contains 100 categories. In Table 2, we show over 1.9% gain compared with the baselines.

SVHN [39] is a 10 category street view house number dataset. In Table 3, we experiment on two methods that have pretrained models available, including the state-of-the-art semi-supervised learning [58]. Ours demonstrate over 6.8% gain compared with the original defense method.

ImageNet [12] contains 1000 categories. We use the pretrained model from Wong et al. [62], which is the state-of-the-art ResNet50 robust model. In Table 4, we use 5 different attacks to access the adversarial robustness of the original model and our model with natural supervision defense. Our approach achieved over 3% gain on robustness.

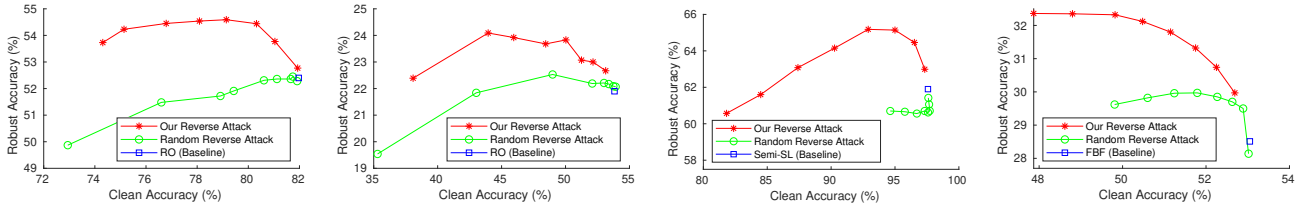
4.6. Analysis

Accuracy vs. Time Budget. As our method is iterative, we can adjust the number of iterations according to different time budget. We use the number of iterations conducted as a indicator of time, and plot the accuracy vs. time budget in Figure 5, where we can see even a few updates can significantly improves the robustness.



(a) CIFAR-10 (up to 9.2% gain) (b) CIFAR-100 (up to 3.3% gain) (c) SVHN (up to 14.1% gain) (d) ImageNet (up to 4.4% gain)

Figure 6: The adversarial robust accuracy vs. perturbation budget curves on CIFAR-10, CIFAR-100, SVHN, and ImageNet, under the L_∞ norm. The red line is applying our inference algorithm to the baseline models [47, 58, 62]. Using our inference algorithm significantly improves the robustness.



(a) CIFAR-10 (b) CIFAR-100 (c) SVHN (d) ImageNet

Figure 7: The trade-off between adversarial robust accuracy vs. clean accuracy on CIFAR-10, CIFAR-100, and SVHN under the L_∞ norm. We increase the noise budget ϵ_v from small to large, which causes the clean accuracy to drop from right to left. Our method produces a better reversal of the adversarial perturbation than just adding random noise to reverse it.

Robustness Curve. We adopt the robustness curve evaluation of adversarial robustness accuracy vs. the perturbation budget [15]. We show the trend in Figure 6. We apply our inference algorithm as additional defense to existing robust models [47, 62], where our approach achieved up to **14%** robustness gain compared with standard inference method, especially when the attack gets stronger with larger perturbation bound.

Trade-off between clean accuracy and adversarial robustness. It has been proved that there exists a natural trade-off between clean accuracy and adversarial robustness given a classifier [57, 68]. In Figure 7, we compare our natural supervised reverse defense with the random reverse defense (baseline). We increase the additive noise level ϵ_v that is applied to reverse the adversarial examples as well as the clean examples (we use the same algorithm to clean examples because during inference we cannot distinguish adversarial ones from clean ones). The clean accuracy often drops as the noise level goes up. While there is a trade-off between clean accuracy and robust accuracy [57, 68], our approach achieved a better trade-off between them.

Robustness on L_2 norm bounded adversarial attacks. We measure whether our defense can also generalize beyond the L_∞ bounded attack. Table 5 shows results under L_2 norm bounded attack on CIFAR-10, where our approach consistently improves robustness under L_2 norm bounded attacks by over 2.6%.

Feature visualization. Figure 8 visualizes the trajectory of images' penultimate layer's feature as it transitions through an attack and the reversal. We use PCA to project the features onto a plane. The plot demonstrates that the attack shifts the feature embedding from the right class to the wrong class. Then, the reverse attack often returns the features back to the right class.

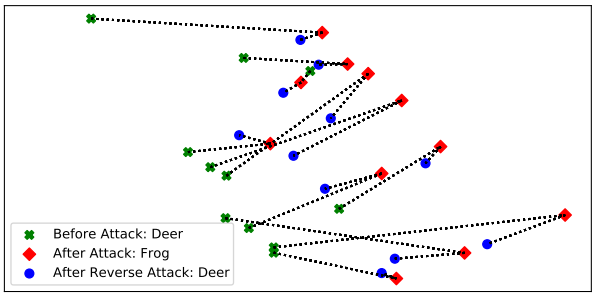


Figure 8: **Feature Trajectories:** We project the features onto a plane with PCA to visualize their trajectory under attack and our reverse attack. The green cross indicates the clean examples, the red square indicates the misclassified adversarial examples, and the blue dot indicates our reversal method. Our approach pushes the misclassified examples (red) back to the original features (green), improving adversarial robustness.

To quantify this effect, we take the Euclidean distance between the clean embedding and the attacked embedding, denoted D_{c_a}, as well as the Euclidean distance between the clean embedding and the inverse attacked embedding, denoted D_{c_i}, for the triples that have the same clean class and inverse attacked class. For all but one combination of categories, $D_{c_a} > D_{c_i}$. Additionally, across all triplets, we checked how much the average distance from clean to re-

verse attacked is reduced from the average distance from clean to attacked, and obtained a value of roughly 17% decrease. These results together demonstrate that, on average, our reverse attack returns the attacked embedding closer to the original embedding.

5. Conclusions

We introduce an approach to use natural supervision to reverse adversarial attacks on images. Our results demonstrate improved robustness across several benchmarks and several state-of-the-art attacks. Our findings suggest integrating defense mechanisms into the inference algorithm is a promising direction to improve adversarial robustness.

Acknowledgements: This research is based on work partially supported by NSF CRII #1850069, NSF grant CNS-15-64055; ONR grants N00014-16-1- 2263 and N00014-17-1-2788; a JP Morgan Faculty Research Award; and a DiDi Faculty Research Award. MC is supported by a CAIT Amazon PhD fellowship. We thank NVidia for GPU donations. The views and conclusions contained herein are those of the authors and should not be interpreted as necessarily representing the official policies, either expressed or implied, of the sponsors.

References

[1] Anish Athalye, Nicholas Carlini, and David Wagner. Obfuscated gradients give a false sense of security: Circumventing defenses to adversarial examples. In *Proceedings of the 35th International Conference on Machine Learning*, volume 80, pages 274–283. PMLR, 2018. 2, 4, 5, 13

[2] Mitali Bafna, Jack Murtagh, and Nikhil Vyas. Thwarting adversarial examples: An l_0-robust sparse fourier transform. In S. Bengio, H. Wallach, H. Larochelle, K. Grauman, N. Cesa-Bianchi, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 31. Curran Associates, Inc., 2018. 2

[3] Sagie Benaim, Ariel Ephrat, Oran Lang, Inbar Mosseri, William T. Freeman, Michael Rubinstein, Michal Irani, and Tali Dekel. Speednet: Learning the speediness in videos. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2020. 2

[4] Jacob Buckman, Aurko Roy, Colin Raffel, and Ian J. Goodfellow. Thermometer encoding: One hot way to resist adversarial examples. In *6th International Conference on Learning Representations*, 2018. 2

[5] Nicholas Carlini, Anish Athalye, Nicolas Papernot, Wieland Brendel, Jonas Rauber, Dimitris Tsipras, Ian J. Goodfellow, Aleksander Madry, and Alexey Kurakin. On evaluating adversarial robustness. *CoRR*, abs/1902.06705, 2019. 1, 2

[6] Nicholas Carlini and David A. Wagner. Towards evaluating the robustness of neural networks. In *2017 IEEE Symposium on Security and Privacy*, pages 39–57, 2017. 1, 2, 4, 6, 13

[7] Mathilde Caron, Ishan Misra, Julien Mairal, Priya Goyal, Piotr Bojanowski, and Armand Joulin. Unsupervised learning of visual features by contrasting cluster assignments. 2020. 2

[8] Mathilde Caron, Ishan Misra, Julien Mairal, Priya Goyal, Piotr Bojanowski, and Armand Joulin. Unsupervised learning of visual features by contrasting cluster assignments, 2021. 2

[9] Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. A simple framework for contrastive learning of visual representations, 2020. 2, 3

[10] Xinlei Chen, Haoqi Fan, Ross Girshick, and Kaiming He. Improved baselines with momentum contrastive learning. *arXiv preprint arXiv:2003.04297*, 2020. 2, 3

[11] Francesco Croce and Matthias Hein. Reliable evaluation of adversarial robustness with an ensemble of diverse parameter-free attacks. In *ICML*, 2020. 1

[12] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei. ImageNet: A Large-Scale Hierarchical Image Database. In *CVPR09*, 2009. 5, 7

[13] Virginia R. DeSa. Learning classification with unlabeled data. In *Proceedings of the 6th International Conference on Neural Information Processing Systems*, NIPS'93, page 112–119, San Francisco, CA, USA, 1993. Morgan Kaufmann Publishers Inc. 2

[14] Guneet S. Dhillon, Kamyar Azizzadenesheli, Zachary C. Lipton, Jeremy Bernstein, Jean Kossaifi, Aran Khanna, and Animashree Anandkumar. Stochastic activation pruning for robust adversarial defense. In *6th International Conference on Learning Representations*, 2018. 2

[15] Yinpeng Dong, Qi-An Fu, Xiao Yang, Tianyu Pang, Hang Su, Zihao Xiao, and Jun Zhu. Benchmarking adversarial robustness on image classification. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 321–331, 2020. 8

[16] Yinpeng Dong, Fangzhou Liao, Tianyu Pang, Hang Su, Jun Zhu, Xiaolin Hu, and Jianguo Li. Boosting adversarial attacks with momentum. In *CVPR*, pages 9185–9193, 2018. 1

[17] Spyros Gidaris, Praveer Singh, and Nikos Komodakis. Unsupervised representation learning by predicting image rotations, 2018. 2, 3

[18] Ian J. Goodfellow, Jonathon Shlens, and Christian Szegedy. Explaining and harnessing adversarial examples. *arXiv:1412.6572*, 2014. 2, 6

[19] Jean-Bastien Grill, Florian Strub, Florent Altché, Corentin Tallec, Pierre H. Richemond, Elena Buchatskaya, Carl Doersch, Bernardo Avila Pires, Zhaohan Daniel Guo, Mohammad Gheshlaghi Azar, Bilal Piot, Koray Kavukcuoglu, Rémi Munos, and Michal Valko. Bootstrap your own latent: A new approach to self-supervised learning, 2020. 2

[20] Chuan Guo, Mayank Rana, Moustapha Cissé, and Laurens van der Maaten. Countering adversarial images using input transformations. *CoRR*, abs/1711.00117, 2017. 2

[21] Kaiming He, Haoqi Fan, Yuxin Wu, Saining Xie, and Ross Girshick. Momentum contrast for unsupervised visual representation learning. *arXiv preprint arXiv:1911.05722*, 2019. 2

[22] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. *arXiv 1512.03385*, 2015. 6

[23] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Identity mappings in deep residual networks, 2016. 6

[24] Dan Hendrycks, Kimin Lee, and Mantas Mazeika. Using pre-training can improve model robustness and uncertainty. *Proceedings of the International Conference on Machine Learning*, 2019. 2

[25] Dan Hendrycks, Mantas Mazeika, Saurav Kadavath, and Dawn Song. Using self-supervised learning can improve model robustness and uncertainty. *Advances in Neural Information Processing Systems (NeurIPS)*, 2019. 1, 2

[26] Allan Jabri, Andrew Owens, and Alexei A. Efros. Space-time correspondence as a contrastive random walk. In *Advances in Neural Information Processing Systems*, 2020. 2

[27] William Karush. Minima of functions of several variables with inequalities as side constraints. *M. Sc. Dissertation. Dept. of Mathematics, Univ. of Chicago*, 1939. 13

[28] Diederik P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization, 2017. 6

[29] Alex Krizhevsky, Vinod Nair, and Geoffrey Hinton. Cifar-10 (canadian institute for advanced research). 5, 7

[30] Alex Krizhevsky, Vinod Nair, and Geoffrey Hinton. Cifar-100 (canadian institute for advanced research). 5, 7

[31] Alexey Kurakin, Ian J. Goodfellow, and Samy Bengio. Adversarial examples in the physical world. *CoRR*, abs/1607.02533, 2017. 2, 6

[32] Alex H. Lang, Sourabh Vora, Holger Caesar, Lubing Zhou, Jiong Yang, and Oscar Beijbom. Pointpillars: Fast encoders for object detection from point clouds. *CoRR*, abs/1812.05784, 2018. 1

[33] Yingzhen Li, John Bradshaw, and Yash Sharma. Are generative classifiers more robust to adversarial attacks? 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 3804–3814. PMLR, 09–15 Jun 2019. 2

[34] Aleksander Madry, Aleksandar Makelov, Ludwig Schmidt, Dimitris Tsipras, and Adrian Vladu. Towards deep learning models resistant to adversarial attacks. In *ICLR*, 2018. 1, 2, 5, 6

[35] Chengzhi Mao, Amogh Gupta, Vikram Nitin, Baishakhi Ray, Shuran Song, Junfeng Yang, and Carl Vondrick. Multi-task learning strengthens adversarial robustness. In Andrea Vedaldi, Horst Bischof, Thomas Brox, and Jan-Michael Frahm, editors, *Computer Vision – ECCV 2020*, pages 158–174, Cham, 2020. Springer International Publishing. 1, 4, 7

[36] Chengzhi Mao, Ziyuan Zhong, Junfeng Yang, Carl Vondrick, and Baishakhi Ray. Metric learning for adversarial robustness. In *Advances in Neural Information Processing Systems*, volume 32. Curran Associates, Inc., 2019. 1, 2

[37] Ishan Misra and Laurens van der Maaten. Self-supervised learning of pretext-invariant representations, 2019. 2

[38] Seyed-Mohsen Moosavi-Dezfooli, Alhussein Fawzi, and Pascal Frossard. Deepfool: a simple and accurate method to fool deep neural networks, 2016. 2

[39] Yuval Netzer, Tao Wang, Adam Coates, Alessandro Bissacco, Bo Wu, and Andrew Y. Ng. Reading digits in natural images with unsupervised feature learning. In *NIPS Workshop on Deep Learning and Unsupervised Feature Learning 2011*, 2011. 5, 7

[40] Mehdi Noroozi and Paolo Favaro. Unsupervised learning of visual representations by solving jigsaw puzzles. In Bastian Leibe, Jiri Matas, Nicu Sebe, and Max Welling, editors, *Computer Vision – ECCV 2016*, pages 69–84, Cham, 2016. Springer International Publishing. 2, 3

[41] Tianyu Pang, Kun Xu, Chao Du, Ning Chen, and Jun Zhu. Improving adversarial robustness via promoting ensemble diversity. 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 4970–4979. PMLR, 09–15 Jun 2019. 2

[42] Tianyu Pang, Xiao Yang, Yinpeng Dong, Hang Su, and Jun Zhu. Bag of tricks for adversarial training, 2020. 1, 2, 6, 7

[43] Nicolas Papernot, Patrick D. McDaniel, Somesh Jha, Matt Fredrikson, Z. Berkay Celik, and Ananthram Swami. The limitations of deep learning in adversarial settings. *arXiv:1511.07528*, 2015. 2

[44] Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, Alban Desmaison, Andreas Kopf, Edward Yang, Zachary DeVito, Martin Raison, Alykhan Tejani, Sasank Chilamkurthy, Benoit Steiner, Lu Fang, Junjie Bai, and Soumith Chintala. Pytorch: An imperative style, high-performance deep learning library. In *Advances in Neural Information Processing Systems*, volume 32. Curran Associates, Inc., 2019. 6

[45] Deepak Pathak, Philipp Krähenbühl, Jeff Donahue, Trevor Darrell, and Alexei A. Efros. Context encoders: Feature learning by inpainting. *CoRR*, abs/1604.07379, 2016. 2, 3

[46] Kexin Pei, Yinzhi Cao, Junfeng Yang, and Suman Jana. Deepxplore. *Proceedings of the 26th Symposium on Operating Systems Principles*, Oct 2017. 1

[47] Leslie Rice, Eric Wong, and J. Zico Kolter. Overfitting in adversarially robust deep learning, 2020. 1, 2, 5, 6, 7, 8, 13, 14

[48] Kevin Roth, Yannic Kilcher, and Thomas Hofmann. The odds are odd: A statistical test for detecting adversarial examples. 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 5498–5507. PMLR, 09–15 Jun 2019. 2

[49] A. M. Rubinov, X. Q. Yang, and Y. Y. Zhou. A lagrange penalty reformulation method for constrained optimization. *Optimization Letters*, 2007. 4

[50] Pouya Samangouei, Maya Kabkab, and Rama Chellappa. Defense-gan: Protecting classifiers against adversarial attacks using generative models. *CoRR*, abs/1805.06605, 2018. 2

[51] Jonathan Scarlett and Volkan Cevher. An introductory guide to fano's inequality with applications in statistical estimation, 2019. 5, 12

[52] Florian Schroff, Dmitry Kalenichenko, and James Philbin. Facenet: A unified embedding for face recognition and clustering. *CoRR*, abs/1503.03832, 2015. 1

[53] Pei Sun, Henrik Kretzschmar, Xerxes Dotiwalla, Aurelien Chouard, Vijaysai Patnaik, Paul Tsui, James Guo, Yin Zhou, Yuning Chai, Benjamin Caine, Vijay Vasudevan, Wei Han, Jiquan Ngiam, Hang Zhao, Aleksei Timofeev, Scott Ettinger, Maxim Krivokon, Amy Gao, Aditya Joshi, Yu Zhang, Jonathon Shlens, Zhifeng Chen, and Dragomir Anguelov. Scalability in perception for autonomous driving: Waymo open dataset. *CoRR*, abs/1912.04838, 2019. 1

[54] Yu Sun, Xiaolong Wang, Zhuang Liu, John Miller, Alexei Efros, and Moritz Hardt. Test-time training with self-supervision for generalization under distribution shifts. In *International Conference on Machine Learning*, pages 9229–9248. PMLR, 2020. 2

[55] Christian Szegedy, Wojciech Zaremba, Ilya Sutskever, Joan Bruna, Dumitru Erhan, Ian J. Goodfellow, and Rob Fergus. Intriguing properties of neural networks. *arXiv:1312.6199*, 2013. 1, 2, 4, 13

[56] Florian Tramer, Nicholas Carlini, Wieland Brendel, and Aleksander Madry. On adaptive attacks to adversarial example defenses. In H. Larochelle, M. Ranzato, R. Hadsell, M. F. Balcan, and H. Lin, editors, *Advances in Neural Information Processing Systems*, volume 33, pages 1633–1645. Curran Associates, Inc., 2020. 2

[57] Dimitris Tsipras, Shibani Santurkar, Logan Engstrom, Alexander Turner, and Aleksander Madry. Robustness may be at odds with accuracy, 2019. 8

[58] Jonathan Uesato, Jean-Baptiste Alayrac, Po-Sen Huang, Robert Stanforth, Alhussein Fawzi, and Pushmeet Kohli. Are labels required for improving adversarial robustness? *CoRR*, 2019. 1, 2, 6, 7, 8, 14

[59] Gunjan Verma and Ananthram Swami. Error correcting output codes improve probability estimation and adversarial robustness of deep neural networks. In *Advances in Neural Information Processing Systems*, volume 32. Curran Associates, Inc., 2019. 2

[60] Carl Vondrick, Abhinav Shrivastava, Alireza Fathi, Sergio Guadarrama, and Kevin Murphy. Tracking emerges by colorizing videos, 2018. 2

[61] Yisen Wang, Difan Zou, Jinfeng Yi, James Bailey, Xingjun Ma, and Quanquan Gu. Improving adversarial robustness requires revisiting misclassified examples. In *ICLR*, 2020. 2, 6, 7

[62] Eric Wong, Leslie Rice, and J. Zico Kolter. Fast is better than free: Revisiting adversarial training, 2020. 2, 6, 7, 8

[63] Dongxian Wu, Shu-Tao Xia, and Yisen Wang. Adversarial weight perturbation helps robust generalization. In *NeurIPS*, 2020. 2, 6, 7

[64] Chang Xiao, Peilin Zhong, and Changxi Zheng. Enhancing adversarial defense by k-winners-take-all, 2019. 2

[65] Zhaoxia Yin, Hua Wang, Li Chen, Jie Wang, and Weiming Zhang. Reversible adversarial example based on reversible image transformation, 2021. 2

[66] Tao Yu, Shengyuan Hu, Chuan Guo, Weilun Chao, and Kilian Weinberger. A new defense against adversarial images: Turning a weakness into a strength. In *Proceedings of the 33rd Conference on Neural Information Processing Systems (NeurIPS 2019)*, Oct. 2019. 2

[67] Sergey Zagoruyko and Nikos Komodakis. Wide residual networks. *CoRR*, abs/1605.07146, 2016. 6

[68] Hongyang Zhang, Yaodong Yu, Jiantao Jiao, Eric P. Xing, Laurent El Ghaoui, and Michael I. Jordan. Theoretically principled trade-off between robustness and accuracy. *arXiv abs/1901.08573*, 2019. 2, 5, 6, 7, 8

[69] Richard Zhang, Phillip Isola, and Alexei A. Efros. Colorful image colorization, 2016. 2

6. Theoretical Results

6.A. Detailed Proof of Lemma 1

Lemma 2. *The standard classifier under adversarial attack is equivalent to predicting with $P(Y|X = x_a, Y^{(s)} = y_a^{(s)})$, and our approach is equivalent to predicting with $P(Y|X = x_a, Y^{(s)} = y^{(s)})$.*

First we show that

$$P(Y|X = x_a) = P(Y|X = x_a, Y^{(s)} = y^{(s)_a}).$$

It is easy to see that

$$\begin{aligned} P(Y|X = x_a) &= \sum_{Y^{(s)}} P(Y^{(s)}|X = x_a)P(Y|Y^{(s)}, X = x_a) \\ &= P(Y|Y^{(s)} = y_a^{(s)}, X = x_a), \end{aligned}$$

where the last equality is due to the neural network's deterministic nature, i.e.,

$$P(Y^{(s)} = y_a^{(s)}|X = x_a) = 1$$

where $y_a^{(s)}$ is the latent self-supervised label prediction. Thus the probability is 0 otherwise.

Intuitively, this demonstrates that the attack is equivalent to using the classifier

$$P(Y|X = x_a, Y^{(s)} = y_a^{(s)})$$

to predict the label.

Next we show that our algorithm is equivalent to using the following classifier

$$P(Y|X = x_a, Y^{(s)} = y^{(s)}).$$

Our algorithm finds a new input image $X^{(n)}$ that

$$\begin{aligned} x_{\max}^{(n)} &= \underset{\mathbf{x}^{(n)}}{\operatorname{argmax}} P(\mathbf{X}^{(n)} = \mathbf{x}^{(n)}|\mathbf{X} = \mathbf{x})P(\mathbf{Y}^{(s)} = \mathbf{y}^{(s)}|\mathbf{X}^{(n)} = \mathbf{x}^{(n)}) \\ &= \underset{\mathbf{x}^{(n)}}{\operatorname{argmax}} P(\mathbf{X}^{(x)} = \mathbf{x}^{(n)}|\mathbf{X} = \mathbf{x}, \mathbf{Y}^{(s)} = \mathbf{y}^{(s)}) \end{aligned}$$

Note that

$$\begin{aligned} &P(\mathbf{Y}|\mathbf{X} = \mathbf{x}_a, \mathbf{Y}^{(s)} = \mathbf{y}^{(s)}) \\ &= \sum_{\mathbf{x}^{(n)}} P(\mathbf{Y}|\mathbf{x}^{(n)})P(\mathbf{x}^{(n)}|\mathbf{X} = \mathbf{x}_a, \mathbf{Y}^{(s)} = \mathbf{y}^{(s)}) \\ &\approx P(\mathbf{Y}|\mathbf{X}^{(n)} = \mathbf{x}_{\max}^{(n)})P(\mathbf{X}^{(n)} = \mathbf{x}_{\max}^{(n)}|\mathbf{X} = \mathbf{x}_a, \mathbf{Y}^{(s)} = \mathbf{y}^{(s)}) \end{aligned}$$

The last formulation is our algorithm's inference procedure, where we first estimate $x_{\max}^{(n)}$ with adversarial image x_a and self-supervised label $y^{(s)}$. We then predict the label Y using our new image $x_{\max}^{(n)}$. We now have proved that our algorithm is equivalent to using

$$P(\mathbf{Y}|\mathbf{X} = \mathbf{x}_a, \mathbf{Y}^{(s)} = \mathbf{y}^{(s)}).$$

Here, we use the maximum a posteriori (MAP) estimate $X_{\max}^{(n)}$ to approximate the marginalization over $\mathbf{x}^{(n)}$ because: first, sampling a large number of $X^{(n)}$ is computational expensive, second, our our shows the sampling is ineffective, lastly, our MAP estimation also produce a denoised image that can also be useful for other downstream tasks.

6.B. Detailed Proof of Theorem 1

Theorem 2. *Assume the base classifier operates better than chance and instances in the dataset are uniformly distributed over n categories. Let the prediction accuracy bounds be $P(\mathbf{Y}|\mathbf{Y}_a^{(s)}, \mathbf{X}_a) \in [b_1, c_1]$ and $P(\mathbf{Y}|\mathbf{Y}^{(s)}, \mathbf{X}_a) \in [b_2, c_2]$. If the conditional mutual information $I(\mathbf{Y}; \mathbf{Y}^{(s)}|\mathbf{X}_a) > 0$, we have $b_2 \geq b_1$ and $c_2 > c_1$, which means our approach strictly improves the bound for classification accuracy.*

Proof. If $I(\mathbf{Y}; \mathbf{Y}^{(s)}|\mathbf{X} = \mathbf{x}_a) > 0$, we have:

$$I(\mathbf{Y}; \mathbf{Y}^{(s)}, \mathbf{X}_a) > I(\mathbf{Y}; \mathbf{Y}_a^{(s)}, \mathbf{X}_a) = I(\mathbf{Y}; \mathbf{X}_a)$$

We let the predicted label to be $\hat{\mathbf{Y}}$, we assume there are n categories, and let the lower bound for prediction accuracy to be $Pr(\hat{\mathbf{Y}} = \mathbf{Y}) \geq 1 - \epsilon_p$. We define $H(\epsilon_p) = -\epsilon_p log\epsilon_p - (1 - \epsilon_p)log(1 - \epsilon_p)$. Use the *Fano's Inequality* [51], we have

$$H(\mathbf{Y}|\mathbf{X}_a) \leq H(\epsilon_p) + \epsilon_p \cdot \log(n - 1) \tag{12}$$

$$-\epsilon_p \cdot \log(n - 1) \leq H(\epsilon_p) - H(\mathbf{Y}|\mathbf{X}_a) \tag{13}$$

We add $H(\mathbf{Y})$ to both side

$$H(\mathbf{Y}) - \epsilon_p \cdot \log(n - 1) \leq H(\epsilon_p) + I(\mathbf{Y}; \mathbf{X}_a) \tag{14}$$

because $I(\mathbf{Y}; \mathbf{X}_a) = H(\mathbf{Y}) - H(\mathbf{Y}|\mathbf{X}_a)$.

Then we get

$$H(\epsilon_p) + \epsilon_p \log(n - 1) \geq -I(\mathbf{Y}; \mathbf{X}_a) + H(\mathbf{Y}) \tag{15}$$

Now we define a new function $G(\epsilon_p) = H(\epsilon_p) + \epsilon_p log(n - 1)$. Given that in classification task, the number of category $n \geq 2$. We know $log(n - 1) \geq 0$. Given that the entropy function $H(\epsilon_p)$ first increase and then decrease, the function $G(\epsilon_p)$ should also first increase, peak at some point, and then decrease.

We calculate the ϵ_p for the peak value via calculate the first order derivative $G'(\epsilon_p) = 0$. By solving this, we have:

$$\epsilon_p = 1 - \frac{1}{n} \tag{16}$$

which shows that the function $G(\epsilon_p)$ is monotonically increasing when $\epsilon_p \in [0, 1 - \frac{1}{n}]$.

Given that we know, the base classifier already achieves accuracy better than random guessing, thus the given classifier satisfies $\epsilon_p \in [0, 1 - \frac{1}{n}]$. Now, the function $G(\epsilon_p) = H(\epsilon_p) + \epsilon_p log(n-1)$ is a monotonically increasing function in our studied region, which has the inverse function G^{-1}.

By rewritting the equation 15 We then have

$$G(\epsilon_p) \geq -I(\mathbf{Y}; \mathbf{X}_a) + H(\mathbf{Y}) \tag{17}$$

We apply the inverse function G^{-1} to both side:

$$\epsilon_p \geq G^{-1}(-I(\mathbf{Y}; \mathbf{X}_a) + H(\mathbf{Y})) \tag{18}$$

$$1 - \epsilon_p \leq 1 - G^{-1}(-I(\mathbf{Y}; \mathbf{X}_a) + H(\mathbf{Y})) \tag{19}$$

Note that $(1 - \epsilon_p)$ is our defined accuracy, thus $b_1 = 1 - G^{-1}(-I(\mathbf{Y}; \mathbf{X}_a) + H(\mathbf{Y}))$

The above derivation also applies to $P(\mathbf{Y}|\mathbf{Y}^{(s)}, \mathbf{X} = \mathbf{x}_a) \in [a_2, b_2]$, thus $b_2 = 1 - G^{-1}(-I(\mathbf{Y}; \mathbf{X}_a, \mathbf{Y}^{(s)}) + H(\mathbf{Y}))$.

Since $I(\mathbf{Y}; \mathbf{Y}^{(s)}, \mathbf{X}_a) > I(\mathbf{Y}; \mathbf{X}_a)$ Thus $b_2 > b_1$, using our approach, the upper bound for robust accuracy is improved.

To prove the lower bound $a_2 \geq a_1$, we divide the joint set of $\mathbf{Y}^{(s)} \cup \mathbf{X}_a$ into set $\mathbf{X}_a$ and $(\mathbf{Y}^{(s)} \cup \mathbf{X}_a) - \mathbf{X}_a$, given the additional information from $(\mathbf{Y}^{(s)} \cup \mathbf{X}_a) - \mathbf{X}_a$, the accuracy will not get worse, thus the new lower bound a_2 should not be smaller than a_1.

□

6.C. Defense Aware Attack

We derive the defense aware attack in our main paper in details. We make the latent label $y^{(s)}$ explicit in our notation.

The straight forward adaptive attack is to optimize the attack in an adversary way against the defense.

$$\boldsymbol{r} = \underset{\boldsymbol{r}}{\operatorname{argmin}} \mathcal{L}_s(\mathbf{x} + \boldsymbol{r}, \mathbf{y}^{(s)}) \tag{20}$$

$$\boldsymbol{\delta} = \underset{\boldsymbol{\delta}}{\operatorname{argmax}} \mathcal{L}_c(\mathbf{x} + \boldsymbol{r} + \boldsymbol{\delta}, \mathbf{y}) \tag{21}$$

From the attacker perspective, the above optimization is not ideal, as it involves iterative optimization of two directions, thus the gradient estimated maybe not stable enough, even having gradient obfuscation [1]. Following the standard constrained optimization attack practice from [55, 6], the attacker reformulates the above equation as a constrained optimization problem:

$$\text{maximize} \quad \mathcal{L}_c(\mathbf{x}_a + \boldsymbol{r}, \mathbf{y}), \tag{22}$$

$$\text{s.t.} \quad \mathcal{L}_s(\mathbf{x}_a + r, \mathbf{y}^{(s)}) \leq \epsilon' \tag{23}$$

where ϵ' is the same value as the converged loss $\mathcal{L}_s$ for natural images. Intuitively, the attacker should maximize the adversarial gain while respecting the self-supervised loss if they want to render our defense ineffective.

The above equation is equivalent to:

$$\text{minimize} \quad -\mathcal{L}_c(\mathbf{x}_a + \boldsymbol{r}, \mathbf{y}), \tag{24}$$

$$\text{s.t.} \quad \mathcal{L}_s(\mathbf{x}_a + r, \mathbf{y}^{(s)}) - \epsilon' \leq 0 \tag{25}$$

We can use the Lagrangian Penalty Method to derive the following:

$$\mathcal{L}_l(\mathbf{x}_a, \lambda_s) = -\mathcal{L}_c(\mathbf{x}_a, \mathbf{y}) + \lambda_s(\mathcal{L}_s(\mathbf{x}_a + \boldsymbol{r}, \mathbf{y}^{(s)}) - \epsilon') \tag{26}$$

Thus the optimal value for the attack $\mathbf{x}_a^*$ is:

$$\mathbf{x}_a^* = \min_{\mathbf{x}_a} \max_{\lambda_s \geq 0} \mathcal{L}_l(\mathbf{x}_a, \lambda_s) \tag{27}$$

which is the primal.

Using the Weak Duality Theorem we have the following upper bound for the optimal solution of the above optimization problem [27]:

$$\mathbf{x}_a^* = \min_{\lambda_s \geq 0} \max_{\mathbf{x}_a} \mathcal{L}_l(\mathbf{x}_a, \lambda_s) \tag{28}$$

Removing the negative sign, we have:

$$\mathbf{x}_a^* = \max_{\lambda_s \geq 0} \max_{\mathbf{x}_a} (\mathcal{L}_c(\mathbf{x}_a, \mathbf{y}) - \lambda_s(\mathcal{L}_s(\mathbf{x}_a + \boldsymbol{r}, \mathbf{y}^{(s)}) - \epsilon')). \tag{29}$$

which is equivalent to first maximizing the followings under different λ_s:

$$\mathcal{L}_l(\mathbf{x}_a, \lambda_s) = \mathcal{L}_c(\mathbf{x}_a, \mathbf{y}) - \lambda_s \mathcal{L}_s(\mathbf{x}_a, \mathbf{y}^{(s)}) \tag{30}$$

And then select the λ_s that yields the most damage with the lowest robust accuracy, and use the corresponding generated attack $\mathbf{x}_a^*$.

7. Experimental Results

7.A. Defense Aware Adversarial Attack

We show the numerical results for the defense aware attack in Table 6. In addition to the 50 steps we used in our main paper, we also show results using 500 steps of adaptive attack. We apply 500 steps to the RO (robust overfitting method by Rice et al. [47]). The results clearly show that using more steps does not change the conclusion. 500 steps of attack achieve almost the same robust accuracy as the 50 steps baseline, which suggests that the attack is almost converged. In addition, our approach still efficiently improves the robust accuracy by over 2%. Lastly, the attacker needs a $\lambda_s = 4$ in order to bypass our defense through the reverse attack, however, at a cost that the attack for classification task gets weaker by over 7%, which itself helps our defense. Overall, even under more attack steps, our defense is still effective.

	Defense Aware Adversarial Attack							
λ_S	0	0.5	1	2	4	6	8	10
RO [47] 50 steps Baseline	**52.40%**	53.81%	55.41%	57.81%	60.80%	62.62%	63.81%	64.58%
RO [47] 50 steps with Ours	**54.59%**	55.61%	56.75%	58.67%	60.81%	62.07%	63.16%	63.68%
RO [47] 500 steps Baseline	**52.23%**	53.47%	54.89%	57.07%	59.86%	61.89%	63.09%	63.90%
RO [47] 500 steps with Ours	**54.70%**	55.61%	56.68%	58.17%	60.51%	61.26%	62.40%	63.06%
Semi-SL [58] 50 steps Baseline	**62.30%**	63.87%	65.38%	67.87%	70.60%	72.32%	73.52%	74.42%
Semi-SL [58] 50 steps with Ours	**64.64%**	65.62%	66.72%	68.58%	70.55%	71.43%	72.75%	73.19%

Table 6: The robust accuracy for defense aware attack under different λ_S setup. We increase the λ_S of defense aware attack from none to 10, and show robustness accuracy on two robust model RO and Semi-SL on CIFAR-10 dataset. While increasing the value of λ_S decrease the gain of Ours compared with Baseline, it also increase the robust accuracy of baseline methods. To achieve the best attack efficiency, the attack should use $\lambda_S = 0$, the standard attack without considering fooling the self-supervised branch.

7.B. Defending a Stand-alone Non-robust Network

In our main paper, we apply our approach to a list of existing state-of-the-art models, now we show results on applying our approach to an undefended neural network.

We train a PreRes-18 model on pure clean images without any adversarial training, which yields 0% robust accuracy under $L_\infty = 8/255$ adversarial attack. We then use our defense to reverse the attack via natural supervision. We achieve an improvement in robust accuracy of 34.4%.

7.C. Feature Input for Self-supervised Models

We investigate which layer's feature should be the input for the self-supervision model. We conduct an ablation study that read out the latent features from low to high layers. Results in Figure 9 show that read our feature from the top layer for self-supervision achieved the most robustness gain.

7.D. Implementation Details

We run our experiments with 8 RTX 2080 Ti GPU. For CIFAR-10, CIFAR-100, and SVHN, the input dimension is all $32 \times 32 \times 3$. We maximize the GPU space usage to speed up our inference. For the PreRes-18 model, we use a batch size of 1024 during inference. For the Wide Residual Network model, we use a batch size of 512. For the ImageNet dataset, the input size is $256 \times 256 \times 3$, we use the ResNet-50 model, and due to the larger input dimension and model capacity, we use a batch size of 87 to maximize the GPU usage. For all the contrastive learning, we sample 4 positive views for each given image instance.

8. Visualization

8.A. Attack Vector Visualization:

We visualize more examples of the adversarial attack vector and the inverse attack vector in Figure 10. Our reverse attack vector is highly structured, reversing the mispredicted examples back to the right one.

8.B. Feature Visualization:

We show more visualizations of the feature trajectories of our approach in Figure 11, Figure 12, Figure 13, Figure 14, and Figure 15. We project the features onto a plane with PCA under the same setup as Figure 8 in the main paper. We can see that our approach pushes the misclassified examples (red) back to the original features (green), improving adversarial robustness.

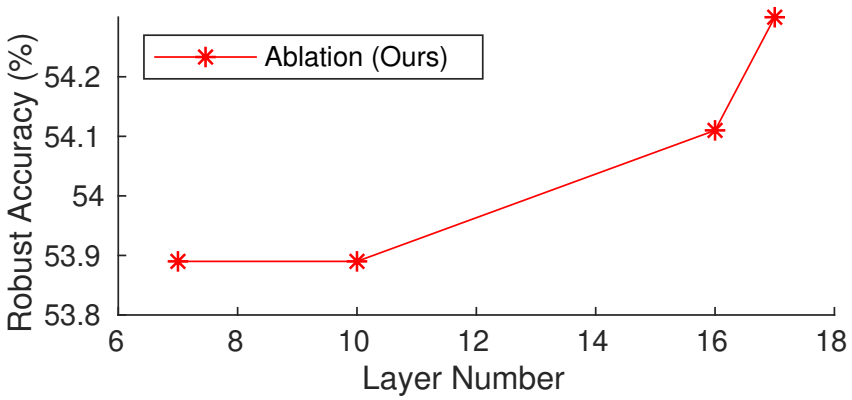


Figure 9: On CIFAR-10 dataset, we experiment on reading out from differnt layers and plot the robustness gain here.

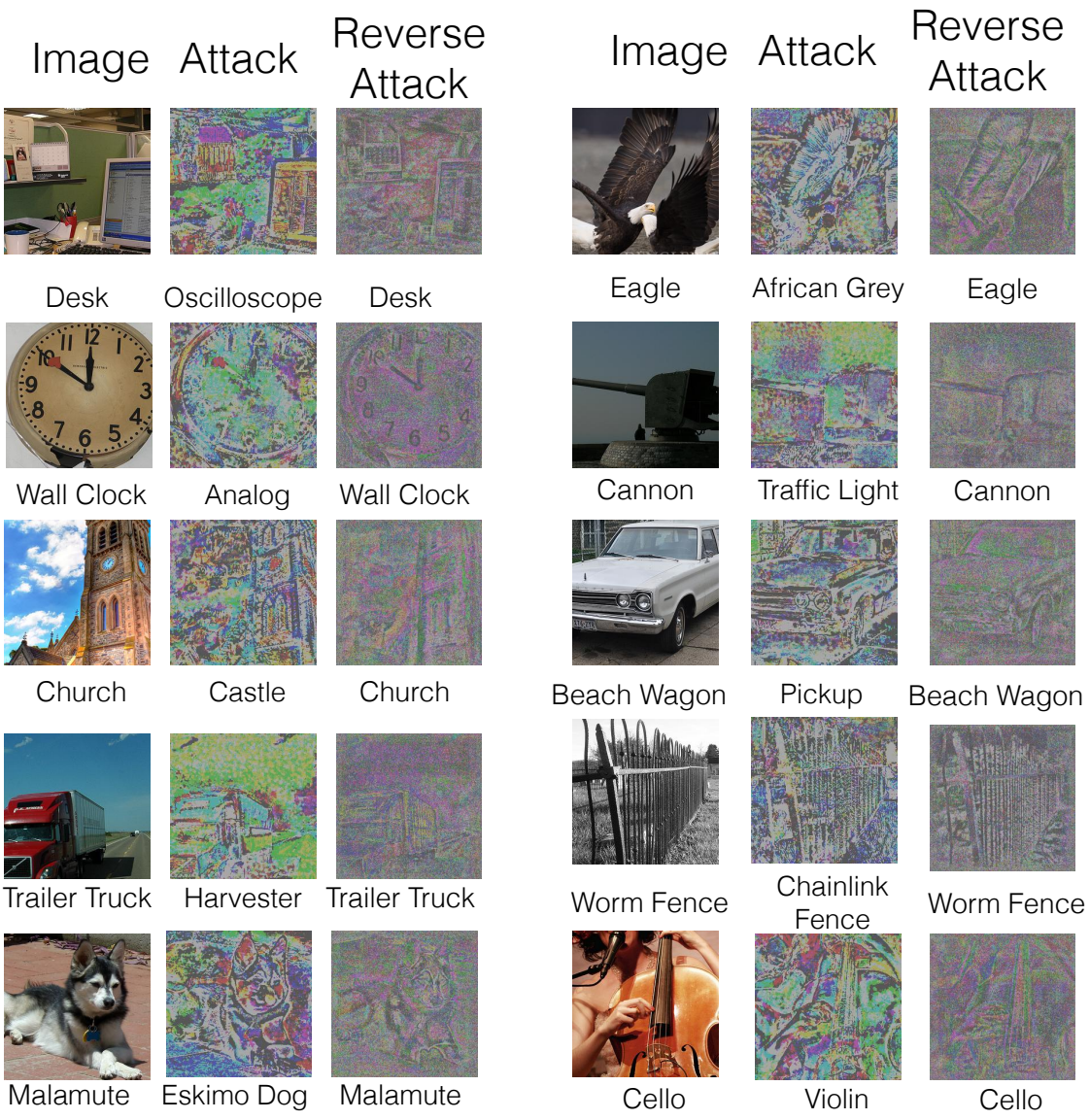


Figure 10: We show ImageNet's clean examples, attack vector, and our reverse attack vector. The adversarial attack is bounded by $\mathbf{L}_\infty = 4/255$. By adding our reverse attack to the attack image, we can correct the misprediction on ImageNet classifier. As we can see, the reverse attack vector is also highly structured, which explains the reason that our approach is more efficient than adding random noise. The attack and reverse attack vectors have been multiplied by ten for visualization purposes only.

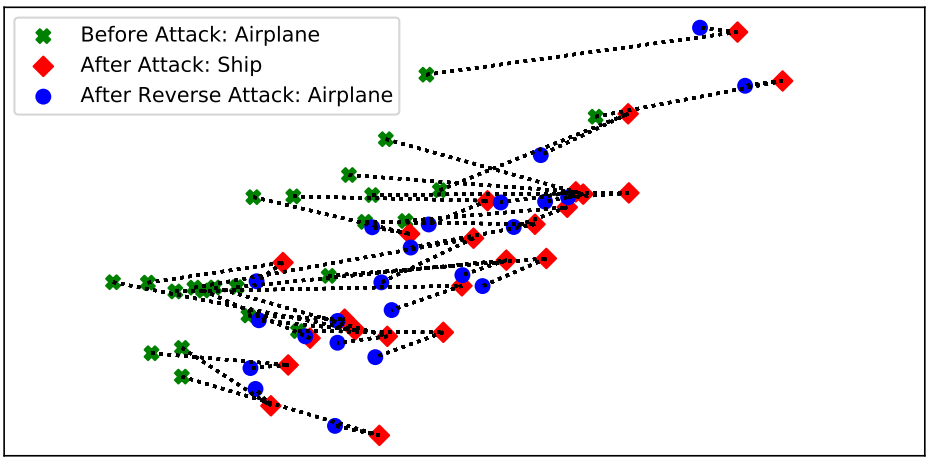


Figure 11: Feature Trajectories under attack and our reverse attack. We plot the figure in the same way as Figure 8 in the main paper with PCA.

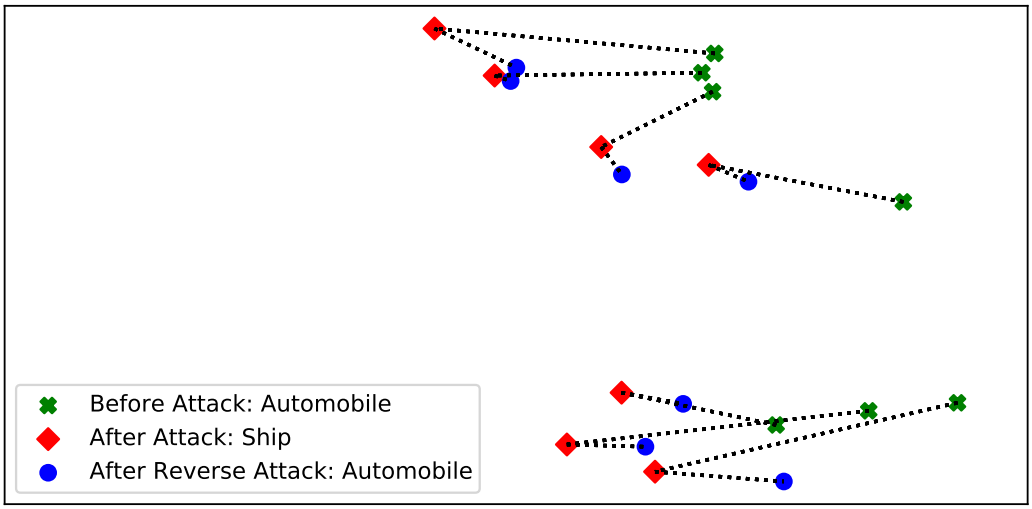


Figure 12: Feature Trajectories under attack and our reverse attack. We plot the figure in the same way as Figure 8 in the main paper with PCA.

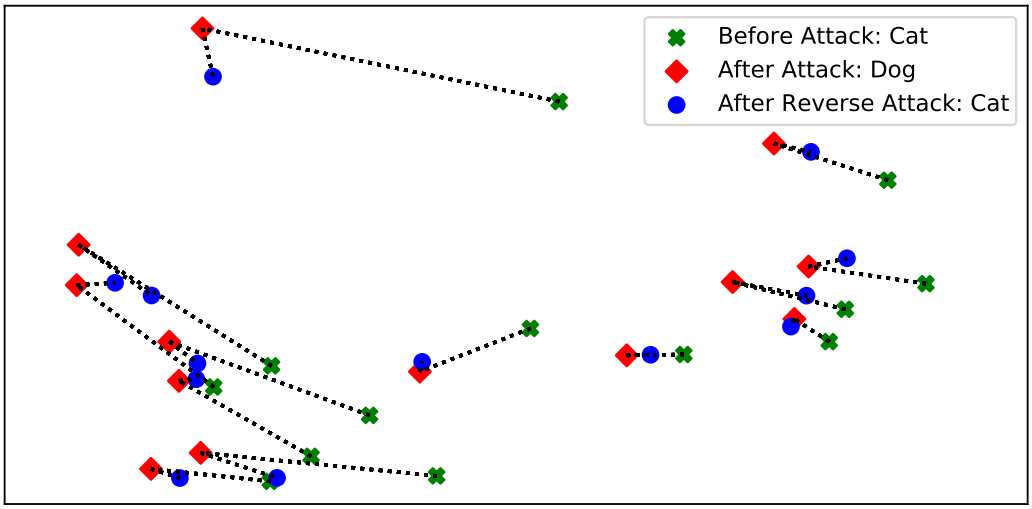


Figure 13: Feature Trajectories under attack and our reverse attack. We plot the figure in the same way as Figure 8 in the main paper with PCA.

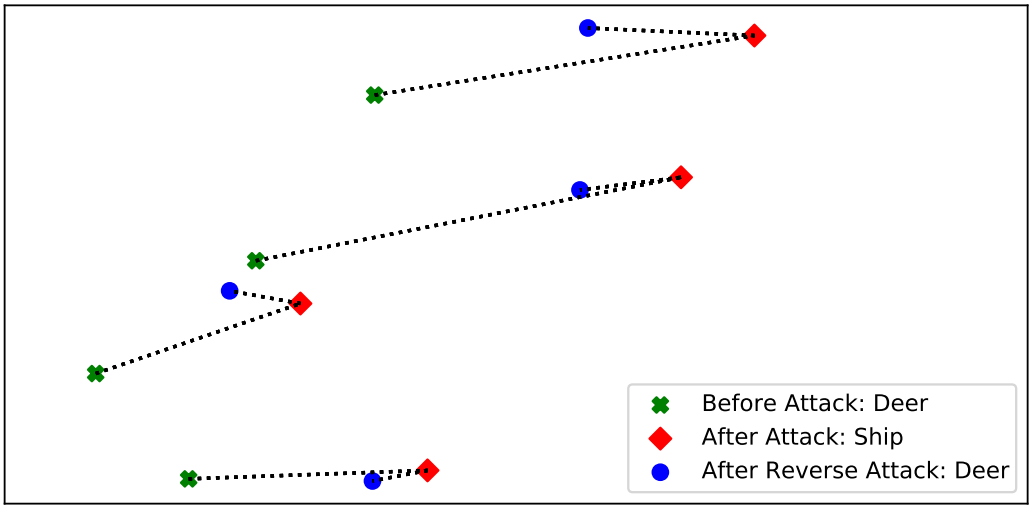


Figure 14: Feature Trajectories under attack and our reverse attack. We plot the figure in the same way as Figure 8 in the main paper with PCA.

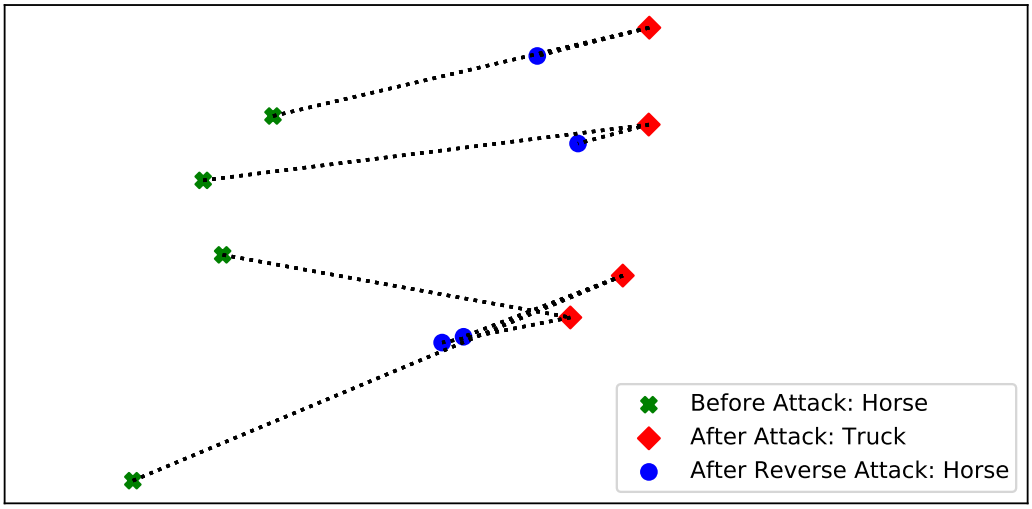


Figure 15: Feature Trajectories under attack and our reverse attack. We plot the figure in the same way as Figure 8 in the main paper with PCA.

References

[1] Anish Athalye, Nicholas Carlini, and David Wagner. Obfuscated gradients give a false sense of security: Circumventing defenses to adversarial examples. In *Proceedings of the 35th International Conference on Machine Learning*, volume 80, pages 274–283. PMLR, 2018. 2, 4, 5, 13

[2] Mitali Bafna, Jack Murtagh, and Nikhil Vyas. Thwarting adversarial examples: An l_0-robust sparse fourier transform. In S. Bengio, H. Wallach, H. Larochelle, K. Grauman, N. Cesa-Bianchi, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 31. Curran Associates, Inc., 2018. 2

[3] Sagie Benaim, Ariel Ephrat, Oran Lang, Inbar Mosseri, William T. Freeman, Michael Rubinstein, Michal Irani, and Tali Dekel. Speednet: Learning the speediness in videos. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2020. 2

[4] Jacob Buckman, Aurko Roy, Colin Raffel, and Ian J. Goodfellow. Thermometer encoding: One hot way to resist adversarial examples. In *6th International Conference on Learning Representations*, 2018. 2

[5] Nicholas Carlini, Anish Athalye, Nicolas Papernot, Wieland Brendel, Jonas Rauber, Dimitris Tsipras, Ian J. Goodfellow, Aleksander Madry, and Alexey Kurakin. On evaluating adversarial robustness. *CoRR*, abs/1902.06705, 2019. 1, 2

[6] Nicholas Carlini and David A. Wagner. Towards evaluating the robustness of neural networks. In *2017 IEEE Symposium on Security and Privacy*, pages 39–57, 2017. 1, 2, 4, 6, 13

[7] Mathilde Caron, Ishan Misra, Julien Mairal, Priya Goyal, Piotr Bojanowski, and Armand Joulin. Unsupervised learning of visual features by contrasting cluster assignments. 2020. 2

[8] Mathilde Caron, Ishan Misra, Julien Mairal, Priya Goyal, Piotr Bojanowski, and Armand Joulin. Unsupervised learning of visual features by contrasting cluster assignments, 2021. 2

[9] Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. A simple framework for contrastive learning of visual representations, 2020. 2, 3

[10] Xinlei Chen, Haoqi Fan, Ross Girshick, and Kaiming He. Improved baselines with momentum contrastive learning. *arXiv preprint arXiv:2003.04297*, 2020. 2, 3

[11] Francesco Croce and Matthias Hein. Reliable evaluation of adversarial robustness with an ensemble of diverse parameter-free attacks. In *ICML*, 2020. 1

[12] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei. ImageNet: A Large-Scale Hierarchical Image Database. In *CVPR09*, 2009. 5, 7

[13] Virginia R. DeSa. Learning classification with unlabeled data. In *Proceedings of the 6th International Conference on Neural Information Processing Systems*, NIPS'93, page 112–119, San Francisco, CA, USA, 1993. Morgan Kaufmann Publishers Inc. 2

[14] Guneet S. Dhillon, Kamyar Azizzadenesheli, Zachary C. Lipton, Jeremy Bernstein, Jean Kossaifi, Aran Khanna, and Animashree Anandkumar. Stochastic activation pruning for robust adversarial defense. In *6th International Conference on Learning Representations*, 2018. 2

[15] Yinpeng Dong, Qi-An Fu, Xiao Yang, Tianyu Pang, Hang Su, Zihao Xiao, and Jun Zhu. Benchmarking adversarial robustness on image classification. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 321–331, 2020. 8

[16] Yinpeng Dong, Fangzhou Liao, Tianyu Pang, Hang Su, Jun Zhu, Xiaolin Hu, and Jianguo Li. Boosting adversarial attacks with momentum. In *CVPR*, pages 9185–9193, 2018. 1

[17] Spyros Gidaris, Praveer Singh, and Nikos Komodakis. Unsupervised representation learning by predicting image rotations, 2018. 2, 3

[18] Ian J. Goodfellow, Jonathon Shlens, and Christian Szegedy. Explaining and harnessing adversarial examples. *arXiv:1412.6572*, 2014. 2, 6

[19] Jean-Bastien Grill, Florian Strub, Florent Altché, Corentin Tallec, Pierre H. Richemond, Elena Buchatskaya, Carl Doersch, Bernardo Avila Pires, Zhaohan Daniel Guo, Mohammad Gheshlaghi Azar,

Bilal Piot, Koray Kavukcuoglu, Rémi Munos, and Michal Valko. Bootstrap your own latent: A new approach to self-supervised learning, 2020. 2

[20] Chuan Guo, Mayank Rana, Moustapha Cissé, and Laurens van der Maaten. Countering adversarial images using input transformations. *CoRR*, abs/1711.00117, 2017. 2

[21] Kaiming He, Haoqi Fan, Yuxin Wu, Saining Xie, and Ross Girshick. Momentum contrast for unsupervised visual representation learning. *arXiv preprint arXiv:1911.05722*, 2019. 2

[22] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. *arXiv 1512.03385*, 2015. 6

[23] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Identity mappings in deep residual networks, 2016. 6

[24] Dan Hendrycks, Kimin Lee, and Mantas Mazeika. Using pre-training can improve model robustness and uncertainty. *Proceedings of the International Conference on Machine Learning*, 2019. 2

[25] Dan Hendrycks, Mantas Mazeika, Saurav Kadavath, and Dawn Song. Using self-supervised learning can improve model robustness and uncertainty. *Advances in Neural Information Processing Systems (NeurIPS)*, 2019. 1, 2

[26] Allan Jabri, Andrew Owens, and Alexei A. Efros. Space-time correspondence as a contrastive random walk. In *Advances in Neural Information Processing Systems*, 2020. 2

[27] William Karush. Minima of functions of several variables with inequalities as side constraints. *M. Sc. Dissertation. Dept. of Mathematics, Univ. of Chicago*, 1939. 13

[28] Diederik P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization, 2017. 6

[29] Alex Krizhevsky, Vinod Nair, and Geoffrey Hinton. Cifar-10 (canadian institute for advanced research). 5, 7

[30] Alex Krizhevsky, Vinod Nair, and Geoffrey Hinton. Cifar-100 (canadian institute for advanced research). 5, 7

[31] Alexey Kurakin, Ian J. Goodfellow, and Samy Bengio. Adversarial examples in the physical world. *CoRR*, abs/1607.02533, 2017. 2, 6

[32] Alex H. Lang, Sourabh Vora, Holger Caesar, Lubing Zhou, Jiong Yang, and Oscar Beijbom. Pointpillars: Fast encoders for object detection from point clouds. *CoRR*, abs/1812.05784, 2018. 1

[33] Yingzhen Li, John Bradshaw, and Yash Sharma. Are generative classifiers more robust to adversarial attacks? 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 3804–3814. PMLR, 09–15 Jun 2019. 2

[34] Aleksander Madry, Aleksandar Makelov, Ludwig Schmidt, Dimitris Tsipras, and Adrian Vladu. Towards deep learning models resistant to adversarial attacks. In *ICLR*, 2018. 1, 2, 5, 6

[35] Chengzhi Mao, Amogh Gupta, Vikram Nitin, Baishakhi Ray, Shuran Song, Junfeng Yang, and Carl Vondrick. Multitask learning strengthens adversarial robustness. In Andrea Vedaldi, Horst Bischof, Thomas Brox, and Jan-Michael Frahm, editors, *Computer Vision – ECCV 2020*, pages 158–174, Cham, 2020. Springer International Publishing. 1, 4, 7

[36] Chengzhi Mao, Ziyuan Zhong, Junfeng Yang, Carl Vondrick, and Baishakhi Ray. Metric learning for adversarial robustness. In *Advances in Neural Information Processing Systems*, volume 32. Curran Associates, Inc., 2019. 1, 2

[37] Ishan Misra and Laurens van der Maaten. Self-supervised learning of pretext-invariant representations, 2019. 2

[38] Seyed-Mohsen Moosavi-Dezfooli, Alhussein Fawzi, and Pascal Frossard. Deepfool: a simple and accurate method to fool deep neural networks, 2016. 2

[39] Yuval Netzer, Tao Wang, Adam Coates, Alessandro Bissacco, Bo Wu, and Andrew Y. Ng. Reading digits in natural images with unsupervised feature learning. In *NIPS Workshop on Deep Learning and Unsupervised Feature Learning 2011*, 2011. 5, 7

[40] Mehdi Noroozi and Paolo Favaro. Unsupervised learning of visual representations by solving jigsaw puzzles. In Bastian Leibe, Jiri Matas, Nicu Sebe, and Max Welling, editors, *Computer Vision – ECCV 2016*, pages 69–84, Cham, 2016. Springer International Publishing. 2, 3

[41] Tianyu Pang, Kun Xu, Chao Du, Ning Chen, and Jun Zhu. Improving adversarial robustness via promoting ensemble diversity. 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 4970–4979. PMLR, 09–15 Jun 2019. 2

[42] Tianyu Pang, Xiao Yang, Yinpeng Dong, Hang Su, and Jun Zhu. Bag of tricks for adversarial training, 2020. 1, 2, 6, 7

[43] Nicolas Papernot, Patrick D. McDaniel, Somesh Jha, Matt Fredrikson, Z. Berkay Celik, and Ananthram Swami. The limitations of deep learning in adversarial settings. *arXiv:1511.07528*, 2015. 2

[44] Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, Alban Desmaison, Andreas Kopf, Edward Yang, Zachary DeVito, Martin Raison, Alykhan Tejani, Sasank Chilamkurthy, Benoit Steiner, Lu Fang, Junjie Bai, and Soumith Chintala. Pytorch: An imperative style, high-performance deep learning library. In *Advances in Neural Information Processing Systems*, volume 32. Curran Associates, Inc., 2019. 6

[45] Deepak Pathak, Philipp Krähenbühl, Jeff Donahue, Trevor Darrell, and Alexei A. Efros. Context encoders: Feature learning by inpainting. *CoRR*, abs/1604.07379, 2016. 2, 3

[46] Kexin Pei, Yinzhi Cao, Junfeng Yang, and Suman Jana. Deepxplore. *Proceedings of the 26th Symposium on Operating Systems Principles*, Oct 2017. 1

[47] Leslie Rice, Eric Wong, and J. Zico Kolter. Overfitting in adversarially robust deep learning, 2020. 1, 2, 5, 6, 7, 8, 13, 14

[48] Kevin Roth, Yannic Kilcher, and Thomas Hofmann. The odds are odd: A statistical test for detecting adversarial examples. 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 5498–5507. PMLR, 09–15 Jun 2019. 2

[49] A. M. Rubinov, X. Q. Yang, and Y. Y. Zhou. A lagrange penalty reformulation method for constrained optimization. *Optimization Letters*, 2007. 4

[50] Pouya Samangouei, Maya Kabkab, and Rama Chellappa. Defense-gan: Protecting classifiers against adversarial attacks using generative models. *CoRR*, abs/1805.06605, 2018. 2

[51] Jonathan Scarlett and Volkan Cevher. An introductory guide to fano's inequality with applications in statistical estimation, 2019. 5, 12

[52] Florian Schroff, Dmitry Kalenichenko, and James Philbin. Facenet: A unified embedding for face recognition and clustering. *CoRR*, abs/1503.03832, 2015. 1

[53] Pei Sun, Henrik Kretzschmar, Xerxes Dotiwalla, Aurelien Chouard, Vijaysai Patnaik, Paul Tsui, James Guo, Yin Zhou, Yuning Chai, Benjamin Caine, Vijay Vasudevan, Wei Han, Jiquan Ngiam, Hang Zhao, Aleksei Timofeev, Scott Ettinger, Maxim Krivokon, Amy Gao, Aditya Joshi, Yu Zhang, Jonathon Shlens, Zhifeng Chen, and Dragomir Anguelov. Scalability in perception for autonomous driving: Waymo open dataset. *CoRR*, abs/1912.04838, 2019. 1

[54] Yu Sun, Xiaolong Wang, Zhuang Liu, John Miller, Alexei Efros, and Moritz Hardt. Test-time training with self-supervision for generalization under distribution shifts. In *International Conference on Machine Learning*, pages 9229–9248. PMLR, 2020. 2

[55] Christian Szegedy, Wojciech Zaremba, Ilya Sutskever, Joan Bruna, Dumitru Erhan, Ian J. Goodfellow, and Rob Fergus. Intriguing properties of neural networks. *arXiv:1312.6199*, 2013. 1, 2, 4, 13

[56] Florian Tramer, Nicholas Carlini, Wieland Brendel, and Aleksander Madry. On adaptive attacks to adversarial example defenses. In H. Larochelle, M. Ranzato, R. Hadsell, M. F. Balcan, and H. Lin, editors, *Advances in Neural Information Processing Systems*, volume 33, pages 1633–1645. Curran Associates, Inc., 2020. 2

[57] Dimitris Tsipras, Shibani Santurkar, Logan Engstrom, Alexander Turner, and Aleksander Madry. Robustness may be at odds with accuracy, 2019. 8

[58] Jonathan Uesato, Jean-Baptiste Alayrac, Po-Sen Huang, Robert Stanforth, Alhussein Fawzi, and Pushmeet Kohli. Are labels required for improving adversarial robustness? *CoRR*, 2019. 1, 2, 6, 7, 8, 14

[59] Gunjan Verma and Ananthram Swami. Error correcting output codes improve probability estimation and adversarial robustness of deep neural networks. In *Advances in Neural Information Processing Systems*, volume 32. Curran Associates, Inc., 2019. 2

[60] Carl Vondrick, Abhinav Shrivastava, Alireza Fathi, Sergio Guadarrama, and Kevin Murphy. Tracking emerges by colorizing videos, 2018. 2

[61] Yisen Wang, Difan Zou, Jinfeng Yi, James Bailey, Xingjun Ma, and Quanquan Gu. Improving adversarial robustness requires revisiting misclassified examples. In *ICLR*, 2020. 2, 6, 7

[62] Eric Wong, Leslie Rice, and J. Zico Kolter. Fast is better than free: Revisiting adversarial training, 2020. 2, 6, 7, 8

[63] Dongxian Wu, Shu-Tao Xia, and Yisen Wang. Adversarial weight perturbation helps robust generalization. In *NeurIPS*, 2020. 2, 6, 7

[64] Chang Xiao, Peilin Zhong, and Changxi Zheng. Enhancing adversarial defense by k-winners-take-all, 2019. 2

[65] Zhaoxia Yin, Hua Wang, Li Chen, Jie Wang, and Weiming Zhang. Reversible adversarial example based on reversible image transformation, 2021. 2

[66] Tao Yu, Shengyuan Hu, Chuan Guo, Weilun Chao, and Kilian Weinberger. A new defense against adversarial images: Turning a weakness into a strength. In *Proceedings of the 33rd Conference on Neural Information Processing Systems (NeurIPS 2019)*, Oct. 2019. 2

[67] Sergey Zagoruyko and Nikos Komodakis. Wide residual networks. *CoRR*, abs/1605.07146, 2016. 6

[68] Hongyang Zhang, Yaodong Yu, Jiantao Jiao, Eric P. Xing, Laurent El Ghaoui, and Michael I. Jordan. Theoretically principled trade-off between robustness and accuracy. *arXiv abs/1901.08573*, 2019. 2, 5, 6, 7, 8

[69] Richard Zhang, Phillip Isola, and Alexei A. Efros. Colorful image colorization, 2016. 2