Adversarial Robustness through Disentangled Representations

Shuo Yang, [1] **Tianyu Guo,** [2] **Yunhe Wang,** [3] **Chang Xu** [1]

[1]School of Computer Science, Faculty of Engineering, The University of Sydney, Australia
[2]Key Laboratory of Machine Perception (MOE), CMIC, School of EECS, Peking University, China
[3]Huawei Noah's Ark Lab
syan9630@uni.sydney.edu.au, tianyuguo@pku.edu.cn, yunhe.wang@huawei.com, c.xu@sydney.edu.au

Abstract

Despite the remarkable empirical performance of deep learning models, their vulnerability to adversarial examples has been revealed in many studies. They are prone to make a susceptible prediction to the input with imperceptible adversarial perturbation. Although recent works have remarkably improved the model's robustness under the adversarial training strategy, an evident gap between the natural accuracy and adversarial robustness inevitably exists. In order to mitigate this problem, in this paper, we assume that the robust and non-robust representations are two basic ingredients entangled in the integral representation. For achieving adversarial robustness, the robust representations of natural and adversarial examples should be disentangled from the non-robust part and the alignment of the robust representations can bridge the gap between accuracy and robustness. Inspired by this motivation, we propose a novel defence method called Deep Robust Representation Disentanglement Network (DRRDN). Specifically, DRRDN employs a disentangler to extract and align the robust representations from both adversarial and natural examples. Theoretical analysis guarantees the mitigation of the trade-off between robustness and accuracy with good disentanglement and alignment performance. Experimental results on benchmark datasets finally demonstrate the empirical superiority of our method.

Introduction

With the rapid development of deep learning, revolutionary breakthroughs have been made in various fields during the past few years, such as computer vision (He et al. 2016; Guo et al. 2019b; Wang et al. 2018), natural language processing (Collobert and Weston 2008), transfer learning (Yang et al. 2020; Guo et al. 2019a) and so on. However, in parallel with remarkable achievements, DNNs were revealed to be brittle to certain maliciously manipulated inputs named adversarial examples (Biggio et al. 2013; Szegedy et al. 2013). Compared with the natural ones, only imperceptible perturbations are introduced to adversarial examples, yet entirely different predictions can be induced even for the well-trained deep learning models. It poses threats to the real-world, especially the security-critical (e.g., autonomous driving (Tian et al. 2018; Bojarski et al. 2016)) deployments of deep learning models due to the existence of adversarial examples.

In response to adversarial examples, tremendous effort has been made towards the robustness of deep learning models against adversarial attacks. Common defense strategies include adversarial examples detecting and filtering methods (Metzen et al. 2017; Grosse et al. 2017; Xie et al. 2019), pre-processing and dimension reduction methods (Bhagoji, Cullina, and Mittal 2017; Buckman et al. 2018; Prakash et al. 2018). These methods may improve the robustness of DNN models to some extent, however, they are only applicable to a narrow range of attacks or model architectures (Athalye, Carlini, and Wagner 2018; Xie et al. 2019).

Recently, a novel defense framework call adversarial training (Madry et al. 2017) has drawn significant attention. Adversarial training takes the adversarial defense as a mini-max game between the attacker and the classifier, and reinforces the model's robustness by alternatively generating adversarial examples and training the models with both adversarial and natural examples. Adversarial training has shown to be one of the most effective defense strategies according to (Athalye, Carlini, and Wagner 2018). Based on the adversarial training framework, some more advanced works like (Zhang et al. 2019) have been proposed and achieved state-of-the-art performance. However, (Zhang et al. 2019) also reveals an intrinsic negative impact, i.e., the trade-off between the accuracy and robustness.

More recently, the seminal work (Ilyas et al. 2019) suggests that the existence of adversarial examples is a natural consequence of the non-robust (predictive, yet brittle) representation set, which exists in the natural representation distribution independently along with the robust representation set. The model's robustness can be improved by separating the robust representation set from the non-robust one. However, the hypothesis of independence between robust and non-robust representation seems to be questionable, because it is possible to generate adversarial examples for any instance including the robust set. Thus, in this paper, we postulate that the robust and non-robust representations are two basic ingredients entangled in the integral representation. The robust representation is specified to the classification task (i.e., class-specific), in contrast, the non-robust representation is not capable of classification (i.e., class-irrelevant), but contains the information about whether the natural domain or the adversarial domain the examples is sampled from. No matter whether the example is natural or adversarial example, its representation always has the robust and non-robust ingredients. The reason why the model has incorrect prediction to the adversarial example is that the adversarial perturbation changes the class-irrelevant part (ideally, unchangeable to the class-specific part) so that the integral representation will be misclassified by the classifier. Thus, if we can disentangle the class-specific representation from the integral representation, better robustness could be achieved. Further more, if a good alignment between the natural and adversarial class-specific representations can be realized, the gap between the natural accuracy and adversarial robustness should be minimized.

Inspired by the motivation above, we propose a novel defense method called Deep Robust Representation Disentanglement Network (DRRDN). Concretely, DRRDN follows the framework of the Auto-Encoder (AE) which extracts the hidden representation by the so-called encoder and restores the original input with the decoder. Different from the typical AEs, DRRDN particularly disentangles the hidden representation into two branches by two disentanglers, one of which is to extract the class-specific representation for classification, while another aims at deriving the class-irrelevant representation to determine whether the natural domain or the adversarial domain the representation is from. For better disentanglement performance, we further regularize the disentanglers with a mutual information minimizer. Similar to the traditional AEs, a reconstructor is also employed to restore the integral representation before disentanglement for consistency. During the inference phase, we only utilize the class-specific representation for prediction, therefore, no matter what domain the input example comes from, the extracted representation for classification would be consistently robust against adversarial attacks. Furthermore, we theoretically proof that, given satisfying disentanglement and alignment performance, a smaller distribution discrepancy between the natural and adversarial class-specific representation can be realized, therefore, the natural accuracy and adversarial accuracy should be closer. Finally, we empirically verify the effectiveness of DRRDN on MNIST and CIFAR10 datasets with various adversarial attacks. It is worthy to highlight the main contributions of this work as follows:

- We propose a novel adversarial defense method DRRDN which eliminates the effect of adversarial perturbations by disentangling and aligning the robust representation from both natural and adversarial examples.
- The proposed DRRDN is model-agnostic, thus can adapt to any model architecture. Furthermore, DRRDN exploits the information from the perturbation for complete disentanglement performance.
- We prove a smaller gap between the natural accuracy and adversarial robustness with theoretical analysis.
- Experimental results on benchmark datasets demonstrate the superiority of DRRDN.

Related Works

Adversarial Attacks

The phenomenon of adversarial examples is initially revealed by (Szegedy et al. 2013; Goodfellow, Shlens, and Szegedy 2014). Based on their observation, following works propose various adversarial attacks, including the gradient-based methods FGSM (Szegedy et al. 2013), PGD (Madry et al. 2017), C&W (Carlini and Wagner 2017), the evolutionary-based extreme one pixel attack (Su, Vargas, and Sakurai 2019), and the universal adversarial perturbations (Moosavi-Dezfooli et al. 2017), among which the gradient-based methods empirically show stronger attack ability.

Adversarial Defenses

Traditional defense strategies include adversarial examples detecting and filtering methods (Metzen et al. 2017; Grosse et al. 2017; Xie et al. 2019), pre-processing and dimension reduction methods (Bhagoji, Cullina, and Mittal 2017; Buckman et al. 2018; Prakash et al. 2018). These methods may improve the robustness of DNN models to some extent, however, they are only applicable to a narrow range of attacks or model architectures (Athalye, Carlini, and Wagner 2018; Xie et al. 2019). The recently proposed adversarial training (Madry et al. 2017) framework improves the model's robustness by augmenting the training set with generated adversarial examples. Although adversarial training-based methods such as ALP (Kannan, Kurakin, and Goodfellow 2018) and TRADES (Zhang et al. 2019) competitive performance, a trade-off between the model's natural accuracy and adversarial robustness ubiquitously exists. Our model share similar idea with the feature denoising (FD) (Xie et al. 2019) method. However, FD directly modify the convolutional filter and highly rely on the model architecture.

Representation Disentanglement

Representation disentanglement aims at modeling the independent factors of data variation. It has a wide range of applications including image generation and translation (Mathieu et al. 2016; Liu et al. 2018), domain adaptation (Peng et al. 2019). Unlike the unsupervised representation disentanglement which is proved to be fundamentally impossible without proper inductive biases (Locatello et al. 2019), our disentanglement is conducted under the supervised manner, which provides more guarantees for its performance.

Method

The proposed DRRDN method is updated under the adversarial training framework, and for simplicity, we use PGD method to generate adversarial examples during training stage. Thus, before we introduce our proposed method, it is necessary to make a brief review of adversarial training and PGD attack procedure in the following preliminary subsection.

Preliminary

We consider a deep neural network $f(\cdot) = c\big(g(\cdot;\theta);w\big) \in \mathbb{R}^k$, where $g(\cdot;\theta)$ is the representation encoder parameterized by θ, $c(\cdot;w)$ is the classifier parameterized by w, and k is the number of classes. The adversarial example $x' \in \mathbb{R}^{H\times W\times C}$ is generated from the natural example $x \in \mathbb{R}^{H\times W\times C}$ by adding the adversarial perturbation δ which is constrained in a ball with small radius ϵ, i.e., $x' = x + \delta \in \mathcal{B}^p_\epsilon(x) :=$

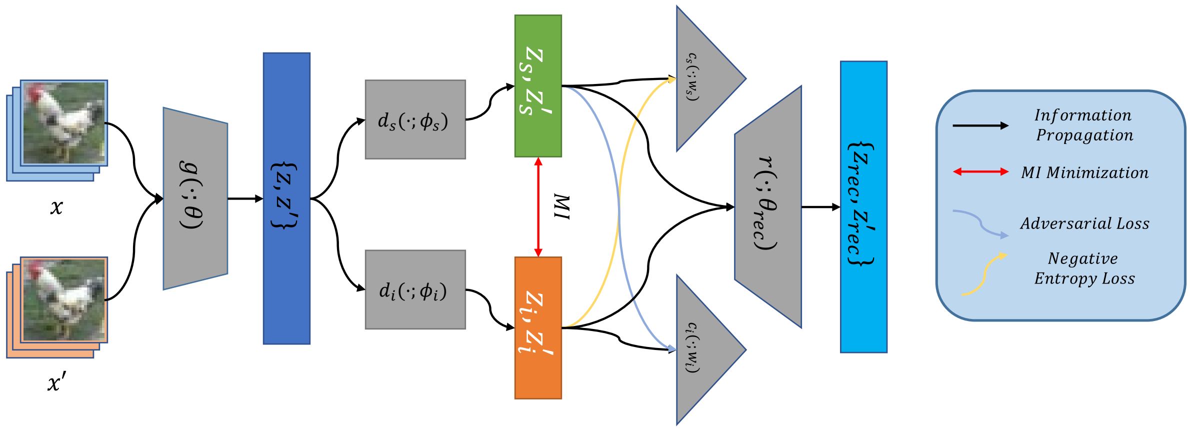

Figure 1: Illustration of proposed DRRDN architecture. $g(\cdot;\theta)$ represents the feature extractor parameterized by θ, $d_s(\cdot;\phi_s)$ and $d_i(\cdot;\phi_i)$ are the disentanglers for the class-specific and class-irrelevant representations with ϕ_s and ϕ_i as parameters, respectively, c_s is the classifier to predict the category of the input representation, c_i is a discriminator identifying which domain the representation comes from, r is a reconstruction decoder, and z represents the output representation of each module.

$\{x' : \|x - x'\|_p = \|\delta\|_p \leq \epsilon\}$. Besides, x' should lead to a prediction alteration of the network, i.e., $\arg\max_i f_i(x') \neq \arg\max_j f_j(x) = y$, where $y \in \mathbb{R}^k$ represents the one-hot embedding of the ground-truth label. The basic idea of adversarial training is to alternately generating the adversarial examples and training with them. The overall objective can be formally described as the following optimization problem:

$$\min_{\theta,w} \mathbb{E}_{p(x,y)} \max_{x' \in \mathcal{B}_\epsilon^p(x)} \mathcal{L}^{\theta,w}(f(x'), y). \tag{1}$$

However, the inner maximization is normally intractable for complex high dimensional data. Therefore, PGD uses the following T-step projection to approximately generate the strongest adversarial example within the ϵ-ball:

$$x'^{\{t+1\}} = \mathrm{Proj}_{\mathcal{B}_\epsilon^p(x)} \left(x'^{\{t\}} + \alpha \mathrm{sign}(\nabla_x \mathcal{L}^{\theta,w}(x'^{\{t\}}, y)) \right), \tag{2}$$

where $\mathrm{Proj}_{\mathcal{B}_\epsilon^p(x)}(\cdot)$ indicates the projection operation which projects the perturbed example to a ϵ-radius ball $\mathcal{B}_\epsilon^p(x)$ around x to ensure the perceptual similarity, $t \in [1, T]$ is the number of projection steps, α is the step size, and $\mathrm{sign}(\cdot)$ returns the sign of each element in the gradient. Note that if the total update step T equals to one, PGD method degrades to the weaker attack FGSM. After x' is obtained by Eq. (2), it will be fed to the model to optimize the outer minimization as in Eq. (1) for robustness.

Deep Robust Representation Disentanglement Network (DRRDN)

The basic philosophy of our DRRDN is we suppose that the integral representation $z = g(x;\theta) \sim p_\theta(z)$ (similarly, $z' = g(x';\theta) \sim p_\theta(z')$) is composed of two entangled branches which are the class-specific representation z_s and class-irrelevant representation z_i, where $p_\theta(z)$ is a push-forward measure of the original data distribution $p(x)$. Concretely, the pure z_s is specified to classification task, thus, no matter for z_s or z_s', they all should be capable to be correctly classified. In DRRDN, we use a disentangler $d_s(\cdot;\phi_s)$ to separate the class-specific representation from z and z' as:

$$\begin{aligned} z_s &= d_s(z;\phi_s), z \sim p_\theta(z), \\ z_s' &= d_s(z';\phi_s), z' \sim p_\theta(z'). \end{aligned} \tag{3}$$

Recall the job of the class-specific representation, we impose classification loss (e.g, cross-entropy (CE)) on $d_s(\cdot;\phi_s)$:

$$\mathcal{L}_{\mathrm{CE}}^{\theta,\phi_s,w_s} = -\mathbb{E}_{\substack{z\sim p_\theta(z) \\ z'\sim p_\theta(z')}} \left[y^{\mathrm{T}} \cdot \log(c_s(\{z_s, z_s'\}, w_s)) \right], \tag{4}$$

where $c_s(\cdot, w_s) \in \mathbb{R}^k$ is the class-specific classifier parameterized by w_s.

However, only the separation of the class-specific representation maybe not enough for complete disentanglement. The class-irrelevant part which captures the perturbation noise information also should be modelled and wiped out from the integral representation. Similar to d_s, we design a class-irrelevant disentangler $d_i(\cdot;\phi_i)$ to model the information irrelevant to classification:

$$\begin{aligned} z_i &= d_i(z;\phi_i), z \sim p_\theta(z), \\ z_i' &= d_i(z';\phi_i), z' \sim p_\theta(z'). \end{aligned} \tag{5}$$

Note that the class-irrelevant representation should capture the characteristics of the natural representation and adversarial representation, which is to say z_i is supposed to be distinguishable from z_i'. Therefore, we apply the binary classification loss on d_i as:

$$\begin{aligned} \mathcal{L}_{\mathrm{BC}}^{\theta,\phi_i,w_i} = &-\mathbb{E}_{z\sim p_\theta(z)} \left[\log(c_i(\{z_i, z_s\}, w_i))\right] \\ &- \mathbb{E}_{z'\sim p_\theta(z')} \left[\log(1 - c_i(\{z_i', z_s'\}, w_i))\right], \end{aligned} \tag{6}$$

where $c_i(, w_i) \in [0, 1]$ is the class-irrelevant classifier parameterized by w_i, which tries to distinguish the natural representations from the adversarial ones no matter class-specific or class-irrelevant.

A good disentanglement should produce mutually independent parts which contain no shared information from each other. In order to achieve this objective, we consider the following two aspects: functional exclusiveness, and mutual information minimization. Firstly, functional exclusiveness means that one disentangled branch should and only should be competent for one specific job but incompetent for another. Eqs. (4) and (6) have ensured that each disentangled representation is capable for its corresponding task, but no constraints are imposed on the counterpart, which may cause the remaining of perturbation information or the loss of the robust information in the class-specific representation. Thus, inspired by the generative adversarial networks (GANs) (Goodfellow et al. 2014), we update d_s to fool the well-trained c_i^* by using an adversarial loss:

$$\begin{aligned}\mathcal{L}_{\text{ADV}}^{\theta,\phi_s} = \mathbb{E}_{z\sim p_\theta(z)}\left[\log(c_i^*(z_s, w_i^*))\right] \\ + \mathbb{E}_{z'\sim p_\theta(z')}\left[\log(1 - c_i^*(z'_s, w_i^*))\right],\end{aligned} \quad (7)$$

where $c_i^*(\cdot, w_i^*)$ is the class-irrelevant classifier well-trained by Eq. (6). Besides, we regularize the prediction entropy of optimized c_s^* on d_i to be large to make sure that d_i cannot be correctly classified:

$$\mathcal{L}_{\text{ENT}}^{\theta,\phi_i} = \mathbb{E}_{\substack{z\sim p_\theta(z)\\ z'\sim p_\theta(z')}}\left[c_s^*(z_i, w_s^*)^{\text{T}} \cdot \log(c_s^*(z'_i, w_s^*))\right], \quad (8)$$

where $c_s^*(\cdot, w_s^*)$ is optimized by Eq. (4), and Eq. (8) is to minimize the negative information entropy of $c_s^*(\cdot, w_s^*)$ on d_i.

Second, for the complete disentanglement, the dependence between the output of d_s and d_i should be as small as possible. As denoted by (Kinney and Atwal 2014), mutual information (MI) can serve as a measure of true dependence for it capturing non-linear statistical dependencies between variables. Thus, in our method, we impose MI minimization on d_s and d_i for better decoupling the effect adversarial perturbation. However, the computation of MI is usually intractable. Thus, in our model, we adopt the Mutual Information Neural Estimator (MINE) (Belghazi et al. 2018) for unbiased estimation of MI. Specifically, given n samples of jointly distributed class-specific and class-irrelevant representation pairs $\{(d_s^j, d_i^j)_i\}_{i=1}^n$, and n pairs from the product of margins $\{(d_s^j, d_i^m)_i\}_{i=1}^n$ (note that, the disentangled representation can be either from natural or adversarial representations), MINE empirically estimates MI between d_s and d_i by Monte-Carlo integration as below:

$$\begin{aligned}MI^{\phi,\psi} = \frac{1}{n}\sum_{i=1}^{n} T\left((d_s^j, d_i^j)_i; \psi\right) \\ - \log\left(\frac{1}{n}\sum_{i=1}^{n} e^{T\left((d_s^j, d_i^m)_i;\psi\right)}\right),\end{aligned} \quad (9)$$

where $T(\cdot;\psi)$ is a neural network parameterized by ψ.

Finally, following the standard AE architecture, we employ a reconstructor to restore the integral representation to keep the cross-cycle consistency of the disentangled representations. In this paper, we use an L2 reconstructor for simplicity:

$$\begin{aligned}\mathcal{L}_{\text{REC}}^{\phi,\theta_{rec}} = \mathbb{E}_{\substack{z\sim p_\theta(z)\\ z'\sim p_\theta(z')}}\Big[||r(z_s, z_i; \theta_{rec}) - z||_2^2 \\ + ||r(z'_s, z'_i; \theta_{rec}) - z'||_2^2\Big],\end{aligned} \quad (10)$$

where $r(\cdot, \cdot; \theta_{rec})$ is the reconstructor parameterized by θ_{rec}.

Algorithm 1: Training Process.

```
Input: Natural training data $(x, y)$, initialized
  parameters for different modules in DRRDN
  $\theta, \phi_{\{s,i\}}, w_{\{s,i\}}, \psi, \theta_{rec}$, trade-off coefficient
  $\lambda_{\{mi,rec\}}$, initialized learning rate $\eta$.
for each training iteration do
    Draw a mini-batch of natural example from an
      empirical distribution $(x, y) \sim \hat{p}(x, y)$;
    Generate adversarial examples $x'$ by iteratively
      optimizing Eq. (2) with $\mathcal{L}_{\text{CE}}$ (Eq. (4));
    Disentangle the class-specific representation $z_s, z'_s$
      and class-irrelevant representation $z_i, z'_i$ with $d_s$
      and $d_i$ in Eqs. (3) and (5), respectively;
    Update the class-specific and class-irrelevant
      classifiers by minimizing $\mathcal{L}_{\text{CE}}$ (Eq. (4)) and $\mathcal{L}_{\text{BC}}$
      (Eq. (6)), respectively;
    Adversarially update the class-specific and
      class-irrelevant disentanglers by minimizing
      $\mathcal{L}_{\text{ADV}}$ (Eq. (7)) and $\mathcal{L}_{\text{ENT}}$ (Eq. (8)), respectively;
    Estimate and minimize the MI between
      class-specific and class-irrelevant representations
      with the MINE loss in Eq. (9);
    Minimize the reconstruction loss $\mathcal{L}_{\text{REC}}$ (Eq. (10));
```

Note that, during the inference phase and when generating the adversarial example, we only use the basic representation extractor $g(\cdot;\phi)$, class-specific disentangler $d_s(\cdot;\phi)$, and class-specific classifier $c_s(\cdot, w_s)$. For a better understanding of the DRRDN structure, we provide an illustration in Fig. 1. We also summarize the training steps of DRRDN in Algorithm 1. Specifically, the overall objective of DRRDN can be summarized as follows:

$$\begin{aligned}\mathcal{L}_{\text{DRRDN}} = \mathcal{L}_{\text{CE}} + \mathcal{L}_{\text{BC}} + \mathcal{L}_{\text{ADV}} + \mathcal{L}_{\text{ENT}} \\ + \lambda_{mi}\mathcal{L}_{\text{MI}} + \lambda_{rec}\mathcal{L}_{\text{REC}},\end{aligned} \quad (11)$$

Theoretical Analysis

In this section, we provide theoretical understanding of the proposed method. Zhang et al. (2019) indicate that there exists a trad-off between the model's accuracy on the natural examples and the robustness against adversarial examples. We hypothesize a possible reason for the trade-off is that the classifier cannot generalize to the representations from both natural and adversarial distribution. Recall that in Eqs. (6) and (7), we have an adversarial loss to regularize the class-specific disentangler. In the following proposition, we prove that by optimizing the objectives in Eqs. (6) and (7), the distribution of class-specific representation from natural and adversarial example can be aligned to each other. Because we only use the class-specific representation for inference, the gap between the model's accuracy and robustness can then be reduced.

Proposition 1. *$p_{\phi_s}(z_s)$ equals to $p_{\phi_s}(z'_s)$ when the global minimums of $\mathcal{L}_{BC}^{\theta,\phi_i,w_i}$ and $\mathcal{L}_{\text{ADV}}^{\theta,\phi_s}$ are achieved.*

Proof. Following (Goodfellow et al. 2014), given natural and adversarial class-specific representations $z_s \sim p_{\phi_s}(z_s), z'_s \sim p_{\phi_s}(z'_s)$, the optimal class-irrelevant classifier $c_i^*(\cdot, w_i^*)$ satisfies:

$$c_i^*(\cdot, w_i^*) = \frac{p_{\phi_s}(z_s)}{p_{\phi_s}(z_s) + p_{\phi_s}(z'_s)}. \tag{12}$$

With the fixed optimal class-irrelevant classifier $c_i^*(\cdot, w_i^*)$, Eq. (7) can be transformed as:

$$\begin{aligned}
\mathcal{L}_{\text{ADV}}^{\theta,\phi_s} &= \mathbb{E}_{z_s \sim p_{\phi_s}(z_s)}\left[\frac{p_{\phi_s}(z_s)}{p_{\phi_s}(z_s) + p_{\phi_s}(z'_s)}\right] \\
&+ \mathbb{E}_{z'_s \sim p_{\phi_s}(z'_s)}\left[\frac{p_{\phi_s}(z'_s)}{p_{\phi_s}(z_s) + p_{\phi_s}(z'_s)}\right] \\
&= \mathbb{E}_{z_s \sim p_{\phi_s}(z_s)}\left[2\frac{p_{\phi_s}(z_s)}{p_{\phi_s}(z_s) + p_{\phi_s}(z'_s)}\right] \\
&+ \mathbb{E}_{z'_s \sim p_{\phi_s}(z'_s)}\left[2\frac{p_{\phi_s}(z'_s)}{p_{\phi_s}(z_s) + p_{\phi_s}(z'_s)}\right] - \log 4 \\
&= KL\left(p_{\phi_s}(z_s) \middle|\middle| {}^{(p_{\phi_s}(z_s)+p_{\phi_s}(z'_s))}\!/_2\right) \\
&+ KL\left(p_{\phi_s}(z'_s) \middle|\middle| {}^{(p_{\phi_s}(z_s)+p_{\phi_s}(z'_s))}\!/_2\right) - \log 4 \\
&= 2 \cdot JS(p_{\phi_s}(z_s) || p_{\phi_s}(z'_s)) - \log 4.
\end{aligned} \tag{13}$$

Thus, the optimal $\mathcal{L}_{\text{ADV}}^{\theta^*,\phi_s^*}$ can be achieved when $p_{\phi_s}(z_s) = p_{\phi_s}(z'_s)$, which completes the proof. □

Experimental Results

In this section, we empirically verify the effectiveness of our proposed method with various experiments. We first implement white- and black- box attacks to compare the robustness of DRRDN with state-of-the-art methods. Further analysis of the effect of disentanglement is also conducted in this section.

Settings

Datasets. In this paper, we use MNIST [1] and CIFAR10 [2] which are benchmark datasets for the evaluation of defense methods. MNIST contains 70,000 examples of handwritten digits with the resolution of 28×28, while CIFAR10 consists of 60,000 32×32 real-world object images in 10 classes. We keep the default training/testing set splits in practice.

Implementation. Following (Zhang et al. 2019), we utilize a four-layer convolutional network as the basic representation extractor for MNIST dataset. The disentanglers and classifiers of DRRDN are fully-connected layers, and for a fair comparison, the classifiers of DRRDN share the same capacity of that in baseline models. For the CIFAR10 dataset, we employed the WRN-28-10 (Zagoruyko and Komodakis 2016) as the representation encoder. We followed the same principle when designing the disentanglers and classifiers as in the MNIST dataset. In the MINE and reconstructor modules, we also employ the dense layers for simplicity. We defer the detailed architecture of DRRDN model for different datasets in the supplementary materials. As for the optimization, we use the SGD optimizer with an initial learning rate of $1e^{-2}$ and $1e^{-1}$ for CIFAR10. The number of total training epoch is 100 during which a learning rate decay of 0.1 is imposed at 55, 75, 90 epochs, respectively. Also, the searching for the optimal hyper-parameters is included in the supplementary materials. We use Pytorch[3] to implement our model and the code can be found here[4].

Defense Setting. We train DRRDN under the adversarial training framework. For the training attacks, we typically set the perturbation $\epsilon = 0.3$, step size $\alpha = 0.01$ for MNIST; $\epsilon = 0.031$, step size $\alpha = 0.007$ for CIFAR10. The update iterations for generating the training attacks is 40 and 10 for MNIST and CIFAR10, respectively. Note that, we only verify the l_∞-norm attack for simplicity.

Adversarial Defense Performance under Different Attacks

White-box Attacks The white-box attack assumes that the attacker has full access to the architecture and parameters of the target model, and directly generate adversarial examples based on the gradient as described in Eq. (2). Thus, the robustness of the standard training model could collapse rapidly. In this work, three types of attacks are employed for verification, which are FGSM, PGD, and CW (Carlini and Wagner 2017) (with various update steps). PGD and CW generate attacks with the step size of 0.01 and 0.003 for MNIST and CIFAR, respectively, and the step size of FGSM is magnified 10 times. We follow (Carlini and Wagner 2017) to generate the CW attacks. We compare the performance of DRRDN with the model trained under *Standard* manner (only trained with natural data), and robust models trained under *Madry* (Madry et al. 2017) and *TRADES* (Zhang et al. 2019) . We summarize the natural accuracy and robust accuracy in the left seven columns of Table 1 and 2 for comparison.

From Table 1 and 2 we can find that the *Standard* model can barely resist to any kind of attacks. When testing with the adversarial examples, the robust accuracy of the *Standard* model will abruptly plunge to almost zero, except for FGSM which is a slightly weaker attack. *Madry* and *TRADES* demonstrate their effectiveness on defending the adversarial attacks, however, there is a large margin between their natural accuracy and robust accuracy. We attribute this phenomenon to the misalignment of the representation distributions between natural examples and adversarial examples so that it is not easy for the model to generalize to both natural and adversarial examples. In contrast, our proposed DRRDN not only achieves higher performance on adversarial classification accuracy (boldface) but also maintains a smaller gap between the natural and adversarial accuracies. This suggests that by

[1] http://yann.lecun.com/exdb/mnist/

[2] https://www.cs.toronto.edu/ kriz/cifar.html

[3] https://pytorch.org/

[4] https://github.com/AAAI2021DRRDN/DRRDN.git

MNIST										
Models	Natural Accuracy	Robust Accuracy White-box Attack					Black-box Attack from Surrogate Models			
		FGSM	PGD40	PGD100	CW40	CW100	*Standard*	*Madry*	*TRADES*	DRRDN
Standard	99.49	86.44	0.36	0.30	0.00	0.00	–	90.26	83.83	98.24
Madry	99.45	98.88	96.58	95.32	96.62	95.43	97.74	–	97.02	98.03
TRADES	99.50	98.98	96.95	95.73	96.88	95.88	97.77	96.93	–	97.95
DRRDN	**99.56**	**99.13**	**97.55**	**96.78**	**97.52**	**96.93**	**98.05**	**97.38**	**97.59**	–

Table 1: Model performance under white-box and black-box adversarial attacks on MNIST dataset.

CIFAR10										
Models	Natural Accuracy	Robust Accuracy White-box Attack					Black-box Attack from Surrogate Models			
		FGSM	PGD40	PGD100	CW40	CW100	*Standard*	*Madry*	*TRADES*	DRRDN
Standard	**95.32**	37.22	0.0	0.0	0.0	0.0	–	45.22	47.79	50.40
Madry	86.53	55.86	45.49	44.97	47.29	46.95	79.55	–	68.66	71.42
TRADES	84.66	60.30	50.91	50.32	49.73	49.11	81.60	70.79	–	74.52
DRRDN	85.76	**62.81**	**52.32**	**52.21**	**51.88**	**51.33**	**82.65**	**72.26**	**72.33**	–

Table 2: Model performance under white-box and black-box adversarial attacks on CIFAR10 dataset.

disentangling and aligning the class-specific representation, a tighter bond between the natural accuracy and adversarial robustness can be achieved.

Black-box Attacks We further verify our model with black-box attacks. Just as the name implies, this type of attack assumes that the detail of the target defender is totally unknown to the attacker. Even though the black-box attacks are generally much weaker than the white-box attacks, the setting of black-box attack is more natural in the real-world applications where the specific design of the model is not clear and can demonstrate the transferability of the model robustness against different attacks.

Regarding the design of black-box attacks, we first independently train surrogate models including *Standard*, *Madry*, *TRADES* and DRRDN to generate adversarial examples with PGD method based on their gradients. Then the adversarial examples generated by certain surrogate model will be used to attack other target models. Specifically, when generating the black-box attacks, we use the same model structures as in white-box attacks. For convenience, the PGD method also share the same set of hyper-parameters (i.e., $\epsilon = 0.3$, $\alpha = 0.01$ with 40 iterations for MNIST; $\epsilon = 0.031$, $\alpha = 0.003$ with 20 iterations for CIFAR10). The classification results under black-box attacks can be found in the right four columns of Table 1 and 2.

By vertically viewing the right four columns of Table 1 and 2, we can see that DRRDN empowers the model with the strongest robustness (boldface) against black-box attacks, which means that the robustness of DRRDN is not specialized for a certain attack, but is a universal characteristic across different attacks. The possible reason may be that the class-irrelevant representation can well-distinguish the natural and adversarial distribution and induce different attacks into one non-natural distribution. However, an interesting finding is that DRRDN provides the weakest attacks (underlined) if we observe the right four columns of Table 1 and 2 horizontally. In fact, there is a natural and intuitive explanation for this phenomenon. We generated adversarial perturbations with DRRDN based on the gradient of the class-specific branch. The class-specific representation can capture the basic and essential characteristic of the input with respect to its categories (ideally, purer than the representation extracted from the *Standard* model which only trained with the natural data, because the redundant information about the natural distribution is also eliminated). Thus, the adversarial perturbation generated by DRRDN is supposed to be slightly weaker than that generated by *Standard* model.

The Effectiveness of Disentanglement

Aforementioned experiments demonstrate that DRRDN is able to empower the DNN models with stronger robustness against both white- and black-box adversarial attacks. We may wonder how and whether the elaborate disentanglement mechanism really benefits the robustness of the model. In this subsection, we conduct the following two exploratory experiments to verify the effectiveness of disentanglement.

Hybrid Reconstructed Representations In the method section, we introduce the function of class-specific and class-irrelevant representations in detail. Recall that the job of class-specific disentangler is to retain the representation which only contains the information for classification, meanwhile the class-irrelevant representation aims at absorbing the perturbation information. Naturally, the class-specific representations disentangled from the natural and adversarial examples are supposed to generalize well from one to another because they are basically from the same distribution. Thus, should the disentanglement mechanism of DRRDN really works, it is not difficult to imagine that if we combine the class-

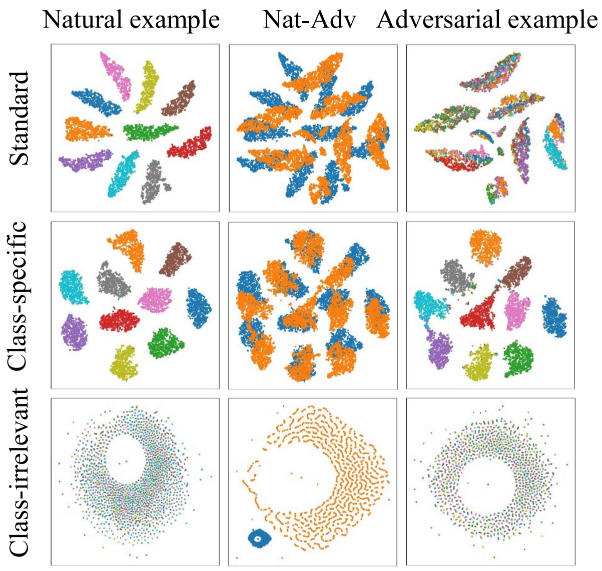

Figure 2: Visualization of class-specific, class-irrelevant and standard representations on MNIST.

Hybrid Representation	$z_s + z_i$	$z'_s + z_i$	$z_s + z'_i$	$z'_s + z'_i$
Accuracy	99.31%	99.02%	95.82%	95.01%

Table 3: Classification performance on hybrid representations.

specific and class-irrelevant representations with different distribution labels (i.e., natural distribution and adversarial distribution) to reconstruct the hybrid representations, we will receive distinct classification results by the classifier of the $Standard$ model fine-tuned on the hybrid representations. Because the hybrid representations actually restore the original representation from two distributions. With the same setting as the white-box attack on MNIST dataset, we conduct classification experiments based on the hybrid reconstructed representations with a standard training model and the results are shown in Table 3. Note that, by permutation, we can reconstruct four kinds of combinations of hybrid representations which are $z_s + z_i$, $z'_s + z_i$, $z_s + z'_i$, and $z'_s + z'_i$, respectively.

In Table 3, it is obvious that the reconstructed representations hybridized with z'_i (i.e., the class-irrelevant representation from adversarial examples) are generally more difficult be correctly classified by the standard model. On the contrary, the hybrid representations which contain z_i (i.e., the class-irrelevant representation from natural examples) can be well-distinguished. This phenomenon implies that the distribution characteristic has been successfully disentangled into the class-irrelevant representation.

Visualization of Disentangled Representations In order to better demonstrate the effectiveness of the disentanglement, we conduct a visualization of the disentangled representations. The visualization experiment is also based on the experiment setting of the white-box attack on MNIST dataset. After the robust model trained with DRRDN is obtained, we first extract the class-specific and class-irrelevant representations from the natural and adversarial testing data, then we compress the high-dimensional representation into a 2D feature by t-SNE (Maaten and Hinton 2008). We summarize the visualization results in Figure 2. Compressed features belonging to different categories are represented with different colours. Except for the class-specific and class-irrelevant representation, we also visualize the representations from the *Standard* model for comparison.

Model	DRRDN	DRRDN-MI	DRRDN-REC
PGD40	97.55%	96.88%	97.32%
PGD100	96.78%	96.25%	96.43%

Table 4: Ablation study on MNIST dataset. DRRDN (the final algorithm), DRRDN-MI (without the mutual information item), and DRRDN-REC (without the reconstruction item).

In Figure 2, the fist and third row demonstrate the compressed representations of examples from natural distribution and adversarial distribution, respectively, and the middle row is a combination of compressed representations form both distributions to illustrate the distribution alignment. Regarding the compressed representation in Figure 2, we have the following three significant discoveries: 1) only the class-specific representation is classifiable concerning both natural and adversarial examples; 2) only the class-irrelevant representation can distinguish between the natural and adversarial distribution; 3) in the middle row, the class-specific representation extracted from natural and adversarial examples is more compact and aligns better than the standard representation. These aforementioned characteristics also justify the effect of disentanglement.

Ablation Study We further conducted ablation study by removing the mutual information item (Eq. (9)) or the reconstruction item (Eq. (10)) from the whole objective function, respectively. As shown in Table 4, we find that without mutual information, DRRDN suffers a much worse accuracy drop than that without the reconstruction item, which implies the importance of the complete disentanglement.

Conclusion

In this paper, we investigate the problem of adversarial defense against adversarial attacks. We propose the DRRDN model to disentangle the class-specific representation where the information about natural and adversarial domain characteristics has been fully eliminated. We also provide a theoretical guarantee of our proposed model by taking advantage of the GAN theories. Empirical evaluations based on benchmark datasets further demonstrate the superiority of DRRDN compared with state-of-the-art defense methods.

Acknowledgments

This work was supported in part by the Australian Research Council under Projects DE180101438 and DP210101859, and Ant Financial Special Scientific Research Funds.

References

Athalye, A.; Carlini, N.; and Wagner, D. 2018. Obfuscated gradients give a false sense of security: Circumventing defenses to adversarial examples. *arXiv preprint arXiv:1802.00420* .

Belghazi, M. I.; Baratin, A.; Rajeshwar, S.; Ozair, S.; Bengio, Y.; Courville, A.; and Hjelm, D. 2018. Mutual information neural estimation. In *International Conference on Machine Learning*, 531–540.

Bhagoji, A. N.; Cullina, D.; and Mittal, P. 2017. Dimensionality reduction as a defense against evasion attacks on machine learning classifiers. *arXiv preprint arXiv:1704.02654* 2.

Biggio, B.; Corona, I.; Maiorca, D.; Nelson, B.; Šrndić, N.; Laskov, P.; Giacinto, G.; and Roli, F. 2013. Evasion attacks against machine learning at test time. In *Joint European conference on machine learning and knowledge discovery in databases*, 387–402. Springer.

Bojarski, M.; Del Testa, D.; Dworakowski, D.; Firner, B.; Flepp, B.; Goyal, P.; Jackel, L. D.; Monfort, M.; Muller, U.; Zhang, J.; et al. 2016. End to end learning for self-driving cars. *arXiv preprint arXiv:1604.07316* .

Buckman, J.; Roy, A.; Raffel, C.; and Goodfellow, I. 2018. Thermometer Encoding: One Hot Way To Resist Adversarial Examples. In *International Conference on Learning Representations*. URL https://openreview.net/forum?id=S18Su--CW.

Carlini, N.; and Wagner, D. 2017. Towards evaluating the robustness of neural networks. In *2017 ieee symposium on security and privacy (sp)*, 39–57. IEEE.

Collobert, R.; and Weston, J. 2008. A unified architecture for natural language processing: Deep neural networks with multitask learning. In *Proceedings of the 25th international conference on Machine learning*, 160–167.

Goodfellow, I.; Pouget-Abadie, J.; Mirza, M.; Xu, B.; Warde-Farley, D.; Ozair, S.; Courville, A.; and Bengio, Y. 2014. Generative adversarial nets. In *Advances in neural information processing systems*, 2672–2680.

Goodfellow, I. J.; Shlens, J.; and Szegedy, C. 2014. Explaining and harnessing adversarial examples. *arXiv preprint arXiv:1412.6572* .

Grosse, K.; Manoharan, P.; Papernot, N.; Backes, M.; and McDaniel, P. 2017. On the (statistical) detection of adversarial examples. *arXiv preprint arXiv:1702.06280* .

Guo, T.; Xu, C.; He, S.; Shi, B.; Xu, C.; and Tao, D. 2019a. Robust student network learning. *IEEE Transactions on Neural Networks and Learning Systems* .

Guo, T.; Xu, C.; Shi, B.; Xu, C.; and Tao, D. 2019b. Learning from Bad Data via Generation. In *Advances in Neural Information Processing Systems*, 6044–6055.

He, K.; Zhang, X.; Ren, S.; and Sun, J. 2016. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 770–778.

Ilyas, A.; Santurkar, S.; Tsipras, D.; Engstrom, L.; Tran, B.; and Madry, A. 2019. Adversarial examples are not bugs, they are features. In *Advances in Neural Information Processing Systems*, 125–136.

Kannan, H.; Kurakin, A.; and Goodfellow, I. 2018. Adversarial logit pairing. *arXiv preprint arXiv:1803.06373* .

Kinney, J. B.; and Atwal, G. S. 2014. Equitability, mutual information, and the maximal information coefficient. *Proceedings of the National Academy of Sciences* 111(9): 3354–3359.

Liu, A. H.; Liu, Y.-C.; Yeh, Y.-Y.; and Wang, Y.-C. F. 2018. A unified feature disentangler for multi-domain image translation and manipulation. In *Advances in neural information processing systems*, 2590–2599.

Locatello, F.; Bauer, S.; Lucic, M.; Raetsch, G.; Gelly, S.; Schölkopf, B.; and Bachem, O. 2019. Challenging common assumptions in the unsupervised learning of disentangled representations. In *international conference on machine learning*, 4114–4124.

Maaten, L. v. d.; and Hinton, G. 2008. Visualizing data using t-SNE. *Journal of machine learning research* 9(Nov): 2579–2605.

Madry, A.; Makelov, A.; Schmidt, L.; Tsipras, D.; and Vladu, A. 2017. Towards deep learning models resistant to adversarial attacks. *arXiv preprint arXiv:1706.06083* .

Mathieu, M. F.; Zhao, J. J.; Zhao, J.; Ramesh, A.; Sprechmann, P.; and LeCun, Y. 2016. Disentangling factors of variation in deep representation using adversarial training. In *Advances in neural information processing systems*, 5040–5048.

Metzen, J. H.; Genewein, T.; Fischer, V.; and Bischoff, B. 2017. On detecting adversarial perturbations. *arXiv preprint arXiv:1702.04267* .

Moosavi-Dezfooli, S.-M.; Fawzi, A.; Fawzi, O.; and Frossard, P. 2017. Universal adversarial perturbations. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 1765–1773.

Peng, X.; Huang, Z.; Sun, X.; and Saenko, K. 2019. Domain agnostic learning with disentangled representations. *arXiv preprint arXiv:1904.12347* .

Prakash, A.; Moran, N.; Garber, S.; DiLillo, A.; and Storer, J. 2018. Protecting JPEG images against adversarial attacks. In *2018 Data Compression Conference*, 137–146. IEEE.

Su, J.; Vargas, D. V.; and Sakurai, K. 2019. One pixel attack for fooling deep neural networks. *IEEE Transactions on Evolutionary Computation* 23(5): 828–841.

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

Tian, Y.; Pei, K.; Jana, S.; and Ray, B. 2018. Deeptest: Automated testing of deep-neural-network-driven autonomous cars. In *Proceedings of the 40th international conference on software engineering*, 303–314.

Wang, F.; Zhao, L.; Li, X.; Wang, X.; and Tao, D. 2018. Geometry-aware scene text detection with instance transformation network. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 1381–1389.

Xie, C.; Wu, Y.; Maaten, L. v. d.; Yuille, A. L.; and He, K. 2019. Feature denoising for improving adversarial robustness. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 501–509.

Yang, Y.; Qiu, J.; Song, M.; Tao, D.; and Wang, X. 2020. Distilling Knowledge From Graph Convolutional Networks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*.

Zagoruyko, S.; and Komodakis, N. 2016. Wide residual networks. *arXiv preprint arXiv:1605.07146* .

Zhang, H.; Yu, Y.; Jiao, J.; Xing, E. P.; Ghaoui, L. E.; and Jordan, M. I. 2019. Theoretically principled trade-off between robustness and accuracy. *arXiv preprint arXiv:1901.08573* .