Computers, Materials & Continua

Tech Science Press

Doi:10.32604/cmc.2025.066370

ARTICLE

An Effective Adversarial Defense Framework: From Robust Feature Perspective

Baolin Li[1], **Tao Hu**[1,2,3,*], **Xinlei Liu**[1], **Jichao Xie**[1] **and Peng Yi**[1,2,3]

[1]Information Engineering University, Zhengzhou, 450000, China
[2]Key Laboratory of Cyberspace Security, Ministry of Education of China, Zhengzhou, 450000, China
[3]National Key Laboratory of Advanced Communication Networks, Zhengzhou, 450000, China

*Corresponding Author: Tao Hu. Email: hutaondsc@163.com

Received: 07 April 2025; Accepted: 24 July 2025; Published: 29 August 2025

ABSTRACT: Deep neural networks are known to be vulnerable to adversarial attacks. Unfortunately, the underlying mechanisms remain insufficiently understood, leading to empirical defenses that often fail against new attacks. In this paper, we explain adversarial attacks from the perspective of robust features, and propose a novel Generative Adversarial Network (GAN)-based Robust Feature Disentanglement framework (GRFD) for adversarial defense. The core of GRFD is an adversarial disentanglement structure comprising a generator and a discriminator. For the generator, we introduce a novel Latent Variable Constrained Variational Auto-Encoder (LVCVAE), which enhances the typical beta-VAE with a constrained rectification module to enforce explicit clustering of latent variables. To supervise the disentanglement of robust features, we design a Robust Supervisory Model (RSM) as the discriminator, sharing architectural alignment with the target model. The key innovation of RSM is our proposed Feature Robustness Metric (FRM), which serves as part of the training loss and synthesizes the classification ability of features as well as their resistance to perturbations. Extensive experiments on three benchmark datasets demonstrate the superiority of GRFD: it achieves 93.69% adversarial accuracy on MNIST, 77.21% on CIFAR10, and 58.91% on CIFAR100 with minimal degradation in clean accuracy. Codes are available at: https://github.com/brother2cat/GRFD (accessed on 23 July 2025).

KEYWORDS: Adversarial defense; robust features; disentanglement; VAE, GAN

1 Introduction

Deep neural networks (DNNs) provide an excellent end-to-end solution for face recognition [1], automatic driving [2], image segmentation [3], etc. However, DNN-based models are vulnerable to adversarial attacks, causing incorrect high-confidence predictions on modified inputs [4]. In particular, when applying DNNs in blockchain, adversarial attacks amplify the risks such as malicious transactions, loss of control in data ownership management, and damage to intelligent recommendation systems. Due to the black-box property [5] of DNNs, researchers have not yet clearly elucidated the underlying mechanism of adversarial attacks. As a result, adversarial attacks and defenses have become a crucial concern in the applications of DNN-based target models.

The usual methods of adversarial defense are empirical and lack researches on the internal mechanism of adversarial attacks. Therefore, they can only defend against specific attacks, resulting in a lack of generalization, and they can be easily compromised by well-designed adversarial attacks. Meanwhile, there are a few certificated defenses, such as Jacobian norm defense [6] and Random Smoothing [7]. However, these techniques are generally less effective compared to the former approaches. The problem must be

addressed at its root; therefore, we should design an adversarial defense based on the internal mechanism of adversarial attacks.

The causes of adversarial attacks have been examined through various hypotheses, including the linear hypothesis [8], the low flexibility hypothesis [9], the high pixel hypothesis [10], and the inherent uncertainty hypothesis [11]. However, all of these explanations have certain limitations. In 2019, Andrew Ilyas proposed the robust feature hypothesis [12] to explain the generation of adversarial samples. He demonstrated that the vulnerability of target models to adversarial attacks can be directly attributed to their learning of non-robust features: features (patterns in the data distribution) that are highly predictive yet vulnerable to perturbations. Since the standard training process of DNN-based target models involves minimizing classification loss, the features learned by the model primarily focus on classification ability, which are referred to as useful features. Consequently, the features that are not learned by the model are termed redundant features. Furthermore, the useful features consist of both robust and non-robust features, with the latter being the source of vulnerability to adversarial attacks. If target models are trained on the robust features directly, they would exhibit high robustness. There has been some research on disentangling features such as [13], but this work has only disentangled the useful features. Therefore, an intuitive problem is:

How can we disentangle the robust features from the training data for adversarial defense?

It is obvious that robust and non-robust features are often intertwined in data distributions, lacking clear mathematical definitions or quantifiable separation criteria. Moreover, the extremely high dimensionality of real-world image data contains numerous redundant features that may obscure robust features, making disentanglement challenging. Previous theories have shown that feature disentangling cannot be achieved solely through unsupervised models or without inductive bias on the data [14]. Hence, we propose GRFD, a new Robust Feature Disentanglement framework based on Generative Adversarial Network (GAN) [15]. GRFD comprises a generator and a discriminator. The key advantage of our proposed GRFD defense method over other GAN-based approaches lies in its explainability-driven design. Without requiring any prior knowledge of adversarial attacks, GRFD can directly extract robust features from samples to train the robust model. In contrast, most comparable methods adopt denoising-based strategies that typically require pre-collected adversarial samples as training data throughout the defense process. For the generator in GRFD, inspired by disentangled representation learning and image translation [16], the beta Variational Auto-Encoder (beta-VAE) [17] is well suited as an image generation model for feature disentanglement. However, considering that the distribution of latent variables in typical beta-VAE is disorderly and scattered, which does not align with the expectation of its design, we improve it and propose the Latent Variable Constrained VAE (LVCVAE) to serve as the generator in GRFD. In LVCVAE, latent variables of samples within the same class are clustered together by adding a constrained rectification module. To supervise the disentanglement of robust features, we design a Robust Supervisory Model (RSM) as the discriminator. Since the target model can learn the features in the dataset well, the architecture of RSM is consistent with that of the target model. It is important to note that the robust features we disentangle are independent of the target model. The key innovation of RSM is its Feature Robustness Metric (FRM), which serves as the training loss and directly determines the defense effectiveness of GRFD. Motivated by the theory of robust features [12], FRM jointly optimizes two key properties: (1) classification performance through standard classification loss, and (2) perturbation resistance through local perturbation gradient similarity—implemented via a finite-difference approximation of loss surface curvature. During the training stage, LVCVAE and RSM learn against each other. Upon convergence, LVCVAE acquires the ability to disentangle the images and generate their robust features. Then the robust features training set disentangled by GRFD will be input into the target model for training, resulting in a robust target model capable of resisting adversarial attacks.

Therefore, the main contributions of this paper are as follows:

- We propose GRFD, a novel GAN-based Robust Feature Disentanglement framework for adversarial defense. The GAN architecture plays an important role in providing robust supervision to address the issue that feature disentanglement cannot be achieved solely by unsupervised models or without an inductive bias.
- By adding a constrained rectification module to the typical beta-VAE, we propose the LVCVAE to serve as the generator. Meanwhile, we design the Robust Supervisory Model (RSM) to play as the discriminator. The key component of RSM is the Feature Robustness Metric (FRM), which quantitatively evaluates feature robustness and serves as the training loss.
- We have conducted several convincing experiments on three benchmark datasets: MNIST, CIFAR-10, and CIFAR-100. The results show that GRFD effectively disentangles robust features, and achieves higher adversarial accuracy of 93.69% on MNIST, 77.21% on CIFAR10, and 58.91% on CIFAR100 with minimal degradation in clean accuracy compared with other state-of-the-art adversarial defenses.

The rest of this article is organized as follows. Preliminaries and related works are presented in Section 2. In Section 3, our proposed framework GRFD and its detailed composition are introduced. Experiments are presented in Section 4. Finally, Section 5 concludes this article.

2 Preliminaries and Related Work

2.1 Adversarial Attacks

The vulnerability of DNNs to adversarial attacks was first systematically demonstrated by Szegedy et al. [4], who showed that imperceptible perturbations could cause misclassification in otherwise high-accuracy models. The general paradigm of adversarial attacks is

$$x' = x + \delta \ \rightarrow f\left(x'\right) \neq y \tag{1}$$

where x denotes a clean sample with label y, δ represents the adversarial perturbation, x' is the resulting adversarial example, and f is the target DNN model. Notably, such perturbations exhibit transferability across different models. Adversarial attacks are commonly categorized into white-box attacks and black-box attacks depending on how much information the attacker has.

In white-box attacks, the attacker can usually obtain the parameter gradient and other information of the model. For example, FGSM [8] uses single step gradient as the image perturbation in adversarial attacks. After that, many gradient-based attacks have been derived. BIM [18] introduces multi-step iteration, RFGSM [19] proposes random initialization noise, and PGD [20] is a combination of the two. Since then, there have been many researches based on PGD, such as EOTPGD [21] which studies the generation of adversarial examples under different input transformations, APGD [22] which introduces adaptive strategy adjustment. Meanwhile, VMIFGSM [23] enhances the stability by reducing the gradient variance. HMCAM [24] is also a gradient-based method, but it incorporates stochastic sampling techniques to generate a distribution of adversarial examples. Besides, there are also optimization-based adversarial attacks, such as CW [25]. Generating adversarial samples using generative networks is also a promising approach, such as SEAdvGAN [26]. For black-box attacks, attackers cannot obtain the parameter information and can only get the output of the model. Pixle [27] rearranges a small number of pixels within the images, and Square [28] selects localized square at random positions to generate adversarial samples. There are also attacks which integrate a few methods, such as AA [22] which combines APGD and Square, etc. This integration results in a parameter-free, computationally affordable and user-independent combination of attacks. AA becomes a common attack benchmark for DNN robustness testing.

2.2 Adversarial Defense

Adversarial defense aims to protect DNN models against perturbation interference while maintaining task performance. Current approaches fall into two main categories: data-based defenses and model-based defenses. For data-based adversarial defenses, defenders achieve model robustness on the data level including the training set and test set. Goodfellow et al. [8] first proposed the adversarial training which was an effective defense method, and then there are many defenses based on adversarial training with different generation methods of adversarial samples, such as the method [29] of Zhang et al. Another approach involves removing adversarial perturbations during the inference stage, such as feature squeezing [30], input transformation (such as JPEG Composing, Bit-depth Reduction [31]), etc. Defenders can also achieve this by denoising (e.g., HGD [32]) and images reconstruction methods that earn clean data distributions through generative models (e.g., APE-GAN [33], DiffPure [34]). For model-based adversarial defenses, they focus on intrinsic model properties. These defenses increase the model robustness not only at the macro level (model classification boundaries), but also at the micro structure (model architecture components). For macro level, most of the defenses are based on defense distillation [35] and gradient regularization [6]. Random Smoothing [7] is also a good idea based on classification boundaries. There are also some robustness analyses based on Bayesian neural networks, such as the argument by Bortolussi et al [36]. Meanwhile, there are also lots of researches to achieve the adversarial defense by adopting the robustness component [37]. Beyond individual model improvements, ensemble methods like TRS [38] combine multiple models for enhanced defense. While feature disentanglement approaches like CD-VAE [13] share conceptual similarities with our work, they focus on useful feature separation whereas our GRFD specifically targets robust feature extraction, representing a critical advancement in defense strategy.

3 GRFD Design

3.1 GRFD Overview

Previous theories have shown that feature disentanglement cannot be realized simply through unsupervised models or without inductive bias on the data. Building upon this theoretical limitations, we propose GRFD (Fig. 1)—a novel GAN-based framework for robust feature disentanglement and adversarial defense. The core of GRFD lies in its adversarial disentanglement architecture, comprising two key components: (1) a Latent Variable Constrained VAE (LVCVAE) generator that explicitly extracts robust features, and (2) a Robust Supervisory Model (RSM) discriminator that guides this process through our proposed Feature Robustness Metric (FRM).

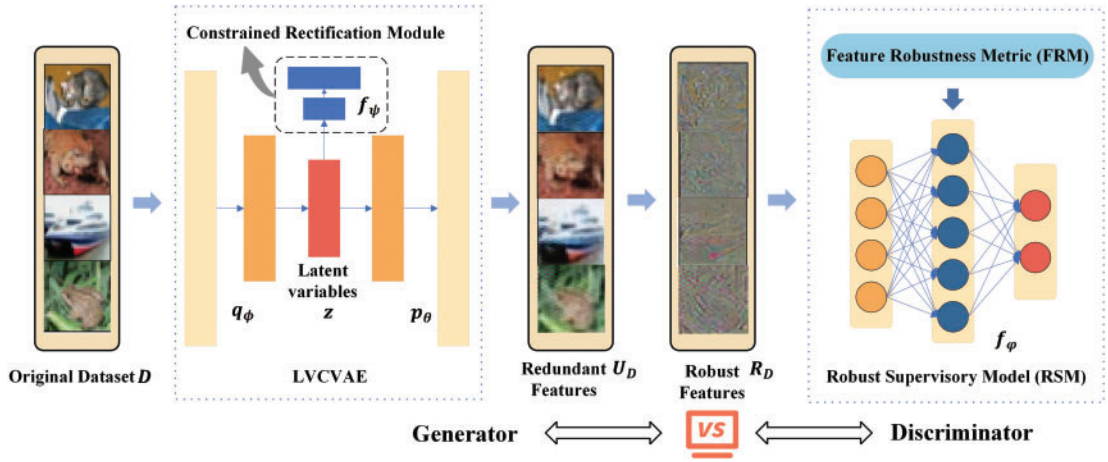

Figure 1: Architecture of our GRFD: a GAN-based robust feature disentanglement framework

During training, GRFD processes the original dataset D through competitive optimization between LVCVAE and RSM, with the complete loss function detailed in Section 3.4. After training, LVCVAE will have the ability to disentangle robust features from the original dataset. In the actual implementation, we set the output of LVCVAE as redundant features U_D. Meanwhile, the robust features R_D are designed as the difference between the original dataset and the redundant features, as shown in Fig. 1. The target model trained on the robust features R_D can successfully resist adversarial attacks and exhibit good robustness.

3.2 Latent Variable Constrained VAE

Preliminary experiments showed that the distribution of latent variables in typical beta-VAE was scattered, as shown in Fig. 2a. This is not in line with its design intent and negatively affects the feature disentanglement process. To address this limitation, we introduce a constrained rectification module that explicitly enforces latent space clustering, yielding our proposed Latent Variable Constrained VAE (LVCVAE). The LVCVAE serves as the generator in GRFD. After training, the latent variables of samples with the same labels are clustered together, as shown in Fig. 2b. Meanwhile, we also provide the ablation analysis in Section 4.3, and LVCVAE is described in detail below.

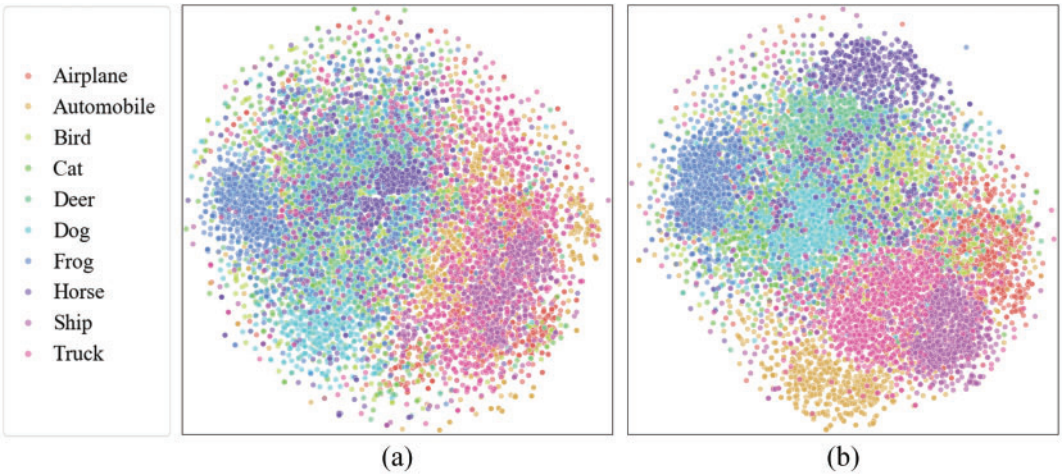

Figure 2: This is the distribution of latent variables, and (**a**) (left) is the case without the constrained rectification module, and (**b**) (right) is the case with the constrained rectification module

The typical beta-VAE is composed of an encoder with parameters ϕ and a decoder with θ. The encoder fits the posterior distribution $q_\phi(z|x)$ and generates the mean μ and variance σ of gaussian distribution of samples, then by sampling, generates the latent variables z. The decoder fits the data likelihood distribution $p_\theta(x|z)$ to generate a new image based on z. To ensure the latent variable z is clustered according to its class, we improved the typical beta-VAE by applying a constrained rectification module denoted as f_ψ on z. In practice, the constrained rectification module is a fully connected neural network, and the loss constraint $L_{rf} = l\left(f_\psi(z),\ y\right)$ needs to be applied during training, where l denotes the Cross-Entropy loss. By continuously training to minimize the loss between the linear transformation of the latent variable z and the label y, latent variables with the same label can be clustered in the feature space. Therefore, the total training

loss L_{LVCVAE} of LVCVAE is:

$$L_{LVCVAE} = \underbrace{MSE\left(x, x'\right)}_{L_{rc}} + \underbrace{\frac{1}{2}\sum_{i=1}^{d}\left(\mu_{(i)}^2 + \sigma_{(i)}^2 - log\sigma_{(i)}^2 - 1\right)}_{L_{kl}} + \underbrace{l\left(f_{\psi}(z),\ y\right)}_{L_{rf}} \tag{2}$$

Here the first term L_{rc} represents the reconstruction loss, measured by the Mean square Error (MSE) between the generated images x' and the original images x. The second term is the regularization term, quantified by the Kullback-Leibler (KL) divergence between $q_{\phi}\left(z|x\right)$ and the prior standard mixed gaussian distributions $p(z)$, where d denotes the dimension of z. The third term corresponds to our proposed latent variable constraint loss.

According to the information bottleneck theory, the training process of a DNN involves minimizing mutual information. Recent research has shown that image classification relies more on the sparse part of the image. Therefore, in our feature disentanglement task, assuming the intermediate feature is m, our goal is to maximize the mutual information between m and the label y, while minimizing the mutual information between m and the input x. During training, U_D is obtained by minimizing the reconstruction loss ($MSE\left(x, x'\right)$), resulting in high mutual information between x and U_D. To reduce the mutual information between m and x, we set $R_D = D - U_D$, achieving the goal of feature disentanglement. As illustrated in Fig. 3, the remainder of the image R_D precisely meets our requirements and serves as the input to the RSM. Under the supervision of the RSM discriminator, the adversarial training between the LVCVAE and the RSM enables robust feature disentanglement, with the final R_D representing the robust features.

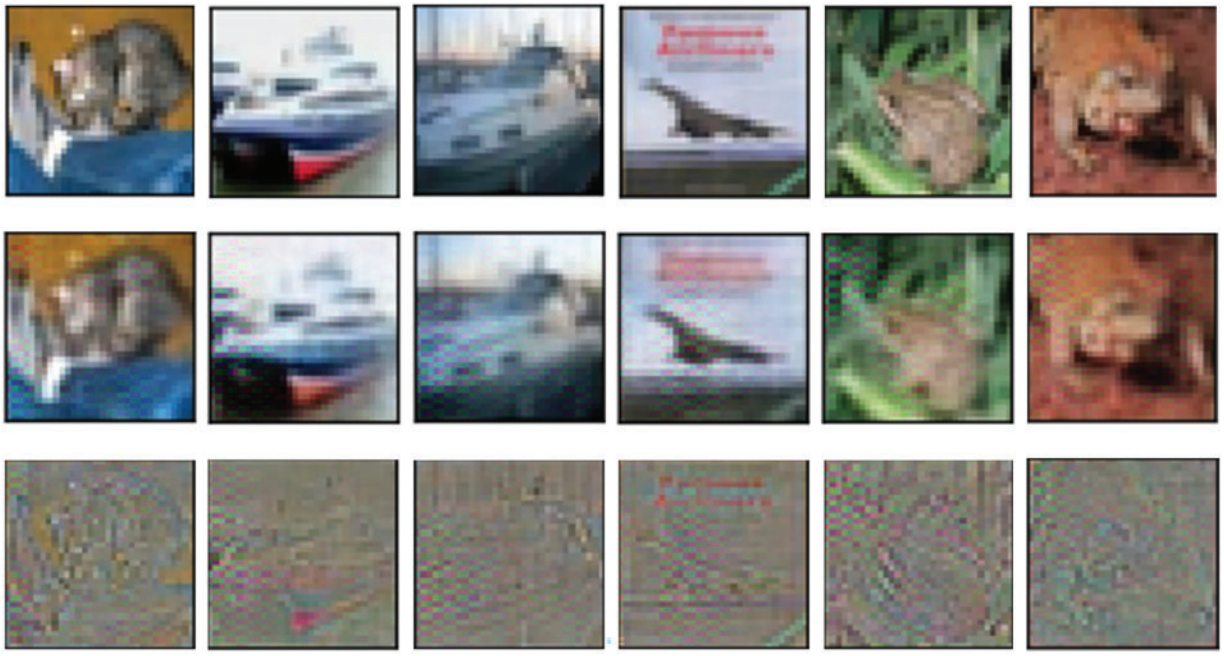

Figure 3: Examples of original images (the first row), their redundant features (the second row) and robust features (the third row) on CIFAR10 dataset

3.3 Robust Supervisory Model

The discriminator in GRFD is our proposed Robust Supervisory Model (RSM) (denoted by f_φ), whose structure is the same as target model and it needs to be trained with LVCVAE simultaneously. The key of RSM is how to design a training loss to quantify the robustness of features R_D, so that LVCVAE can have the ability to disentangle the robust features after joint training with RSM.

Motivated by robust feature hypothesis, the robust features require both the ability of samples classification and the tolerance to perturbations. Therefore, we proposed the Feature Robustness Metric (FRM) L_{frm} to to supervise the disentanglement of robust features. For the former, we measure it using the Cross-Entropy loss between the output of RSM and the labels, denoted as $L_{cl} = l\left(f_\varphi(x_r), y\right)$. For the latter, we conduct an in-depth analysis of the model's robustness mechanism, with details provided below.

Previous studies have shown a strong relationship between model robustness and the curvature of the loss surface [39], which corresponds to the set of eigenvalues of the hesse matrix H to the loss function. However, in practice, the calculation of H is very resource-consuming, so the finite difference approximation method [40] is usually used to evaluate H in the actual experiment, shown as follows:

$$H = \left(\frac{\partial^2 l}{\partial x_i \partial x_j}\right) \in \mathbb{R}^{d\times d} \approx \frac{\nabla l\left(f(x+hz), y\right) - \nabla l(f(x), y)}{h} for\ h \to 0 \tag{3}$$

where f indicates the DNN model. In addition to the advantage of computational complexity, the input is not continuous. So the perturbation in the input field is more practical and we can calculate a large range of gradient changes in the input field. From another point of view, this also reflects the local linearity of the model.

In feature disentanglement, the robustness metric needs to be taken as a condition for implicit supervision to backpropagate. So the robustness metric needs to be a scalar and the sum of the eigenvalues of hesse matrix L_r can be calculated, as shown below:

$$L_r = \frac{1}{h^2}\mathbb{E}\left\|\nabla l\left(f(x+hz), y\right) - \nabla l(f(x), y)\right\|^2 \tag{4}$$

From the perspective of similarity, this is a L_2-norm similarity. Using this as a regularization term of the loss function for training demonstrated robustness improvements comparable to adversarial training [39]. However, in practice, preliminary experiments found that the effect of feature disentanglement is not good when using the L_2 metric, as it proves less effective for high-dimensional feature spaces. Consequently, we reformulate the local linear gradient similarity using cosine similarity, which better captures angular relationships in high-dimensional vectors. The local linear gradient similarity is transformed into the loss to be incorporated into the model training, and its robustness loss L_{rb} can be expressed as:

$$L_{rb} = -\mathbb{E}_{v\sim\Delta}\left[\frac{\nabla l(f(x), y)\cdot\nabla l(f(x+v), y)}{\|\nabla l(f(x), y)\|\cdot\|\nabla l(f(x+v), y)\|}\right] \tag{5}$$

3.4 Procedure of GRFD

The process of our adversarial defense is illustrated in Algorithm 1, our goal is to obtain a robust target model F_{target} and the key of our GRFD is its training loss:

$$Loss_{all} = \underbrace{\alpha L_{rc} + \beta L_{kl} + \varepsilon L_{rf}}_{LVCVAE} + \underbrace{\gamma L_{cl} + \delta L_{rb}}_{RSM} \tag{6}$$

where α, β, γ, δ, ε are the weights of the corresponding loss, respectively. The larger the α, the more effective the samples reconstruction is. The value of β indicates how well the posterior distribution conforms to the Gaussian distribution. ε is the correlation between the distribution of the latent variables and the labels. γ and δ denote the strength of feature classification and robustness, respectively. This is a multi-loss optimization problem, and the training difficulty is relatively large, in which the selection of these weight factor is particularly important. Since GRFD contains some previously studied modules (beta-VAE), we do not need to adjust the parameters of beta-VAE and use the original values, such as α and β. For the modules proposed in this paper, we conduct a large number of hyperparameter experiments on ε, γ and δ.

Algorithm 1: Process of our adversarial defense method

Data: Original dataset: D, Weight factor: α, β, γ, δ, ε
Result: Robust model: F_{target}
/* I: `Training phase of GRFD` */
repeats
  foreach $(x, y) \in D$ **do**
    $\mu, \sigma = q_\phi(x), z \sim \mathcal{N}(\mu, \sigma)$;
    $L_{rf} = l\left(f_\psi(z), y\right)$;
    $x_u = p_\theta(z), x_r = x - x_u$;
    $L_{rc} = MSE\left(x_u, x\right)$;
    $L_{kl} = 1/2\sum_{i=1}^{d}\left(\mu_{(i)}^2 + \sigma_{(i)}^2 - log\sigma_{(i)}^2 - 1\right)$;
    $U_D = \bigcup\{(x_u, y)\}, R_D = \bigcup\{(x_r, y)\}$;
    $L_{cl} = l\left(f_\varphi(x_r), y\right)$;
    $L_{rb} = -\mathbb{E}_{(x_r,y)\sim R_D}\left[\frac{\nabla l(f_\varphi(x_r), y) \cdot \nabla l(f_\varphi(x_r + v), y)}{\left\|\nabla l(f_\varphi(x_r), y)\right\| \cdot \left\|\nabla l(f_\varphi(x_r + v), y)\right\|}\right], v \sim \mathcal{N}(\mathbf{0}, \mathbf{1})$;
    $Loss_{all} = \alpha L_{rc} + \beta L_{kl} + \varepsilon L_{rf} + \gamma L_{cl} + \delta L_{rb}$;
    Take gradient descent step on $Loss_{all}$ to update the model parameter: $q_\phi, p_\theta, f_\psi, f_\varphi$;
  end
until *converged*;
/* `II:Disentangle the robust features` */
foreach $(x, y) \in D$ **do**
  $\mu, \sigma = q_\phi(x), z \sim \mathcal{N}(\mu, \sigma)$;
  $x_u = p_\theta(z), x_r = x - x_u$;
  $U_D = \bigcup\{(x_u, y)\}, R_D = \bigcup\{(x_r, y)\}$;
end
/* `III:Training phase of robust model` */
repeat
  foreach $(x_r, y) \in R_D$ **do**
    $L = F_{target}(x_r, y)$;
    Take gradient descent step on L to update the parameter: F_{target};
  end
until *converged*;

4 Experiments

In this section, experiments are conducted to verify the effectiveness of GRFD proposed in this paper. It should be noted that the main work of this paper is to conduct the adversarial defense based on the robust feature hypothesis which can explain adversarial attacks to some extent, not just the empirical adversarial defense. The experiments are implemented based on the package TorchAttacks [41] and PyTorch2.0.0.

4.1 Setup

Datasets and Models. We conducted experiments on three benchmark datasets: MNIST [42], CIFAR10 [43], CIFAR100 [43], and our target models are Lenet5 [42], WideResNet28(×10), WideResNet28(×10) [44], respectively. The beta-VAEs in LVCVAEs of the three datasets are based on Lenet5, residual block, residual block [45] expectively, and constrained rectification modules are all a linear layer. The architecture of RSM is the same as target model.

Parameter Settings. We set the drop rate in WideResNet28(×10) as 0.3, and the dimension of z in LVCVAE we set as 256, 2024 and 2024 for MNIST, CIFAR10, CIFAR100, respectively. The total training epochs of GRFD on these datasets are all 150. The α in $Loss_{all}$ is set by referring to the classical VAE, when the training epoch is less than 50 α = 10, when it is more than 50 and less than 100 α = 5, and when it is more than 100 and less than 150 α = 1. The β in $Loss_{all}$ is also set by referring to the classical beta-VAE as β = 0.2.

Adversarial Attack Methods. The adversarial attack methods in the experiments are FGSM [8], PGD [21], AA [22], EOTPGD [21], and VMIFGSM [23], which are implemented using the TorchAttacks [41] library. The specific parameters of different attacks are different, and the parameters are shown in Table 1.

Table 1: Parameters of adversarial attacks on MNIST (top part), CIFAR10 and CIFAR100 (bottom part), and the attacks parameters of CIFAR10 and CIFAR100 are the same

Parameters→ attacks↓	Distance measure	Maximum perturbation	Step size	Number of steps
FGSM	L_{inf}	26/255	/	/
PGD	L_{inf}	18/255	8/255	100
AA	L_{inf}	20/255	/	/
EOTPGD	L_{inf}	18/255	8/255	100
VMIFGSM	L_{inf}	18/255	8/255	100
FGSM	L_{inf}	8/255	/	/
PGD	L_{inf}	1/255	1/255	200
AA	L_{inf}	4/255	/	/
EOTPGD	L_{inf}	4/255	2/255	10
VMIFGSM	L_{inf}	4/255	2/255	10

Adversarial Defense Baselines. We compare the effectiveness of our adversarial defense method with other state-of-the-art adversarial defenses: Bit-depth Reduction (B-DR) [31], Feature Squeezing (F.S.) [30], CD-VAE [13], Random Smoothing (R.S.) [7], Gradient Regularization (G.R.) [6], APE-GAN [33] and DiffPure [34]. B-DR and F.S. are methods based on image preprocessing, which can effectively remove adversarial perturbations in images. CD-VAE is also an adversarial defense method based on feature disentanglement. R.S. and G.R. both are certified adversarial defenses. APE-GAN and DiffPure are defense methods based on the reconstruction of images.

Metrics. Both clean and adversarial samples are input into the model, and the experiment result is evaluated by the model prediction accuracy for both types of the data. The metric is simple and intuitive, and is calculated by $Acc = \frac{n(y'=y)}{N}$, where N is the total number of samples, $n(y' = y)$ denotes the number of samples which are correctly predicted.

4.2 Hyperparameter Analysis

In this paper, we have three important parameters: ε, γ and δ. ε represents the weight of constrained rectification module, γ is the weight of classification ability of robust features, and δ indicates the weight of disturbance resistance capability. Next, we conduct experiments on MNIST as an example to determine the value of the three parameters. We employed grid search to identify the optimal hyperparameters that maximize the performance of GRFD. Through preliminary experiments, we found that setting the ranges of ε, γ, and δ to [0.0, 0.7], [0.0, 0.7], and [0.0, 1.4] with step sizes of 0.1, 0.1, and 0.2, respectively, was sufficient to locate a local optimum. Below is the detailed analysis. As shown in Table 2, we find that the value of δ is easy to determine, with the same ε and γ, $\delta = 0.2$ works best, and we set $\delta = 0.2$ in the following experiments. Now that $\delta = 0.2$, we conduct the experiments of ε and γ, and the partial results are shown in Table 2. It is obvious that the clean accuracy and average adversarial accuracy get the maximum value when $\varepsilon = 0.6$, $\gamma = 0.4$ and $\delta = 0.2$. Meanwhile, we get the similar results in CIFAR10 and CIFAR100 datasets, so we set $\varepsilon = 0.6$, $\gamma = 0.4$ and $\delta = 0.2$ in the next experiments.

Table 2: Defense effectiveness under different parameter values on MNIST

Parameters			Clean	Acc of the adversarial attacks					
ε	γ	δ	Acc	FGSM	PGD	AA	EOTPGD	VMIFGSM	Average
Standard model			99.18%	76.63%	58.52%	38.47%	58.8%	54.51%	57.39%
0.6	0.4	0.0	98.51%	61.22%	67.57%	77.73%	67.8%	65.26%	67.92%
0.6	0.4	0.4	98.64%	79.34%	91.5%	89.73%	91.45%	88.39%	88.08%
0.5	0.3	0.2	98.58%	75.75%	90.53%	88.17%	90.37%	87.55%	86.47%
0.5	0.4	0.2	98.32%	74.23%	88.23%	88.07%	87.63%	86.49%	84.93%
0.5	0.5	0.2	98.54%	75.76%	90.9%	89.95%	90.96%	86.2%	86.75%
0.6	0.3	0.2	98.54%	73.8%	85.8%	84.18%	86.35%	81.79%	82.38%
0.6	0.4	0.2	98.82%	88.19%	95.59%	95.59%	95.59%	93.55%	93.69%
0.6	0.5	0.2	98.15%	59.85%	85.49%	83.75%	85.54%	83.62%	79.65%
0.7	0.3	0.2	98.43%	64.1%	86.97%	85.79%	86.99%	83.09%	81.39%
0.7	0.4	0.2	98.79%	88.75%	95.14%	94.21%	95.1%	93.9%	93.42%
0.7	0.5	0.2	98.62%	84.33%	91.67%	90.64%	91.57%	88.49%	89.34%

4.3 Ablation Analysis

In order to verify the effectiveness of the distinct modules designed in this paper, a series of ablation experiments are conducted. We select MNIST as an example and the ablation experiments are divided into four cases: Case 1: without any defense, Case 2: Only the LVCVAE without RSM ($\varepsilon = 0.6, \gamma = 0, \delta = 0$), Case 3: the typical beta-VAE and RSM without the constrained rectification module ($\varepsilon = 0, \gamma = 0.4, \delta = 0.2$), Case 4: the whole GRFD framework ($\varepsilon = 0.6, \gamma = 0.4, \delta = 0.2$). The results are shown in Fig. 4a. It can be observed that the RSM (discriminator) is necessary for disentangling robust features (Case 2 vs. Case 4), which also verifies that only supervised models or implicit constraints can achieve feature disentanglement. The

comparison between Case 3 and Case 4 demonstrates that the constrained rectification module in LVCVAE effectively improves both clean accuracy and adversarial accuracy. Compared to the no-defense scenario (Case 0), our GRFD achieves promising adversarial accuracy, though a single module alone (Case 1) may underperform compared to no defense. In conclusion, every module in GRFD is indispensable, and only the integration of all modules enables effective adversarial defense.

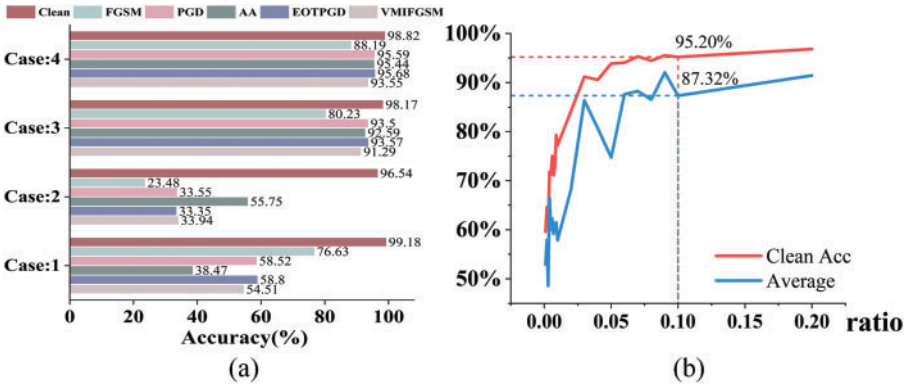

Figure 4: (**a**) is the accuracy of ablation experiments on MNIST. (**b**) illustrates the effect of using different proportions of training data on the disentangling of robust features by GRFD in the MNIST dataset, where "Average" represents the mean adversarial accuracy

4.4 Impact of Data Volume

In real-world, defenders may not have access to the full training dataset as in the lab environment. In practice, they might only possess a small portion of the training data. Therefore, it is worth exploring whether GRFD can still be effective under such limited-data conditions.

To investigate this, we conducted additional experiments using the MNIST dataset to evaluate GRFD's defensive capability with limited data, and the results are shown in Fig. 4b. The results show that even with only 10% of the training data (6000 samples, compared to the full 60,000 in MNIST), GRFD achieves a relatively high clean accuracy, just 3.62% lower than when using the full dataset. Meanwhile, the adversarial accuracy only drops by 6.37% (Full data: Clean Acc—98.82%, Average adversarial acc—93.69%; 10% data: Clean Acc—95.20%, Average adversarial acc—87.32%).

4.5 Defense Compared with Other Methods

We compared the effectiveness of our adversarial defense method with other state-of-the-art adversarial defenses: B-DR, F.S., CD-VAE, R.S., G.R. APE-GAN, and DiffPure on three benchmark datasets and five advanced adversarial attacks, the results are shown in Table 3. Additionally, the "None" method in Table 3 serves as a baseline to better evaluate the trade-off between clean accuracy and adversarial accuracy. In general, The improvement of the robustness of the model is accompanied with the decrease of the accuracy of the model for clean samples. Compared with the other seven typical and state-of-the-art defense methods, our methods achieve higher adversarial accuracy (MNIST: 93.69%, CIFAR10: 77.21%, CIFAR100: 58.91%) with smaller accuracy degradation on clean accuracy (MNIST: 98.82%, CIFAR10: 84.75%, CIFAR100: 63.2%). It should be noted that, on the CIFAR10 and CIFAR100 datasets, while APE-GAN achieves slightly higher clean accuracy than our method (CIFAR10: APE-GAN 88.12% vs. GRFD 84.75%; CIFAR100: APE-GAN 64.94% vs. GRFD 63.2%), GRFD demonstrates a clear advantage in adversarial accuracy (CIFAR10: APE-GAN 72.34% vs. GRFD 77.21%; CIFAR100: APE-GAN 58.61% vs. GRFD 58.91%). Notably, on CIFAR10,

GRFD shows a significant lead under various attacks, including FGSM, PGD, AA, and EOTPGD. Moreover, compared to no defense (None), our method achieves a significant 67% improvement in adversarial accuracy at the cost of only a 10% drop in clean accuracy on CIFAR10, which is an acceptable trade-off.

Table 3: Comparison of adversarial defense effectiveness between our GRFD defense method and other state-of-the-art defense methods on MNIST (top part), CIFAR10 (middle part) and CIFAR100 (bottom part). "None" indicates no defense method is applied

Datasets↓	Metrics↓	None	B-DR	F.S.	CD-VAE	R.S.	G.R.	APE-GAN	DiffPure	GRFD
MNIST	Clean Acc	99.18%	97.23%	97.85%	97.28%	98.7%	97.60%	97.45%	98.10%	98.82%
	FGSM	76.63%	78.39%	80.12%	30.78%	83.43%	82.50%	84.52%	85.30%	88.19%
	PGD	58.52%	65.26%	70.45%	38.18%	93.28%	90.20%	89.16%	92.80%	95.59%
	AA	38.47%	69.41%	75.30%	59.82%	89.56%	88.90%	92.38%	93.50%	95.44%
	EOTPGD	58.8%	72.35%	78.20%	38.02%	91.2%	90.10%	94.23%	94.00%	95.68%
	VMIFGSM	54.51%	64.12%	70.80%	37.85%	93.62%	92.30%	94.58%	93.20%	93.55%
	Average	57.39%	69.91%	74.97%	40.93%	90.22%	88.80%	90.97%	91.76%	93.69%
CIFAR10	Clean Acc	94.76%	86.41%	87.20%	87.67%	85.65%	86.80%	88.12%	87.50%	84.75%
	FGSM	35.85%	52.98%	60.30%	65.47%	69.96%	68.40%	70.56%	72.10%	78.81%
	PGD	9.36%	43.57%	50.25%	59.06%	62.86%	61.80%	71.23%	70.50%	78.98%
	AA	8.19%	41.85%	55.60%	60.33%	68.59%	67.30%	73.42%	74.80%	79.17%
	EOTPGD	0.21%	45.46%	50.10%	62.56%	58.78%	60.20%	72.26%	73.50%	75.97%
	VMIFGSM	0.21%	44.82%	55.80%	63.42%	65.42%	64.90%	74.22%	74.58%	73.12%
	Average	10.76%	45.74%	54.41%	62.17%	65.12%	64.52%	72.34%	73.10%	77.21%
CIFAR100	Clean Acc	73.66%	60.23%	62.50%	64.11%	64.23%	63.80%	64.94%	64.20%	63.2%
	FGSM	17.43%	36.54%	45.30%	46.98%	52.31%	50.80%	59.12%	58.50%	58.18%
	PGD	5.23%	28.75%	40.20%	52.19%	49.45%	48.90%	57.88%	57.20%	59.74%
	AA	4.89%	26.86%	42.10%	49.34%	47.62%	47.30%	59.85%	59.50%	60.45%
	EOTPGD	0.14%	42.19%	45.80%	45.97%	56.86%	55.20%	58.83%	57.60%	57.41%
	VMIFGSM	0.15%	38.44%	44.90%	44.23%	54.13%	53.80%	57.36%	56.90%	58.78%
	Average	5.57%	34.56%	43.66%	47.74%	52.07%	51.20%	58.61%	57.94%	58.91%

4.6 An In-Depth Analysis of Clean and Adversarial Accuracy in GRFD

When using the GRFD method for adversarial defense, a slight drop in clean accuracy is often sacrificed in exchange for a significant improvement in adversarial accuracy. We have conducted a deeper analysis of the reasons behind this accuracy decline, which lies in the extraction of robust features. According to the robust feature hypothesis, DNN models rely on features useful for classification (useful features) when classifying samples, where these features are associated with minimizing the Cross-Entropy between labels and samples. However, a significant portion of these useful features consists of non-robust features, which are the primary cause of adversarial vulnerability. During the process of disentangling robust features, in addition to minimizing the Cross-Entropy between labels and samples, there is also a robust term—local linear gradient similarity. This leads to the removal of some non-robust features, even if they are useful for classification. As a result, adversarial accuracy increases while clean accuracy decreases.

5 Conclusion

To mitigate the vulnerability of DNNs to adversarial attacks, motivated by the feature hypothesis that the non-robust features is the reason for adversarial attacks, we propose a GAN-based robust feature disentanglement framework GRFD. GRFD is composed of a generator which adopt our improved latent Variable Constrained VAE (LVCVAE), and a discriminator which is carefully designed with a Robust Supervision Model (RSM). For LVCVAE, we add a constraint rectification module to the classical beta-VAE, which successfully solves the problem of scattered distribution of latent variables. Meanwhile, the structure

RSM is the same as target model, and the key of RSM is our proposed Feature Robustness Metric (FRM). It combines the classification ability and the resistance to perturbations to measure the robustness of the features, and it is used as part of the loss in the GRFD training process. Extensive experiments validate our effectiveness relative to other state-of-the-art adversarial defense methods.

Acknowledgement: The authors are thankful to researchers in Key Laboratory of Cyberspace Security, Ministry of Education of China for the helpful discussion.

Funding Statement: This work was funded by the National Natural Science Foundation of China Project "Research on Intelligent Detection Techniques of Encrypted Malicious Traffic for Large-Scale Networks" (Grant No. 62176264).

Author Contributions: The authors confirm contribution to the paper as follows: Conceptualization, Baolin Li and Tao Hu; methodology, Baolin Li; software, Baolin Li and Xinlei Liu; validation, Baolin Li and Tao Hu; formal analysis, Baolin Li; investigation, Baolin Li and Jichao Xie; resources, Baolin Li; data curation, Baolin Li; writing—original draft preparation, Baolin Li; writing—review and editing, Baolin Li and Tao Hu; visualization, Baolin Li; supervision, Baolin Li; project administration, Jichao Xie; funding acquisition, Peng Yi. All authors reviewed the results and approved the final version of the manuscript.

Availability of Data and Materials: The data that support the findings of this study are openly available in [Github] at [https://github.com/brother2cat/GRFD] (accessed on 23 July 2025). More data of this study that is not on [https://github.com/brother2cat/GRFD] (accessed on 23 July 2025) is available from the Corresponding Author, [Tao Hu], upon reasonable request.

Ethics Approval: Not applicable.

Conflicts of Interest: The authors declare no conflicts of interest to report regarding the present study.

References

1. Sun Z, Feng C, Patras I, Tzimiropoulos G. LAFS: landmark-based facial self-supervised learning for face recognition. In: 2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR); 2024 Jun 16–22; Seattle, WA, USA. p. 1639–49.
2. Hu Y, Yang J, Chen L, Li K, Sima C, Zhu X, et al. Planning-oriented autonomous driving. In: 2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR); 2023 Jun 17–24; Vancouver, BC, Canada. p. 17853–62.
3. Lee C, Lee SH, Kim C.S. MFP: making full use of probability maps for interactive image segmentation. In: 2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR); 2024 Jun 16–22; Seattle, WA, USA. p. 4051–9.
4. Szegedy C, Zaremba W, Sutskever I, Bruna J, Erhan D, Goodfellow I, et al. Intriguing properties of neural networks. arXiv:1312.6199. 2014.
5. Khan Z, Fu Y. Consistency and uncertainty: identifying unreliable responses from black-box vision-language models for selective visual question answering. In: 2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR); 2024 Jun 16–22; Seattle, WA, USA. p. 10854–63.
6. Liu D, Wu LY, Li B, Boussaid F, Bennamoun M, Xie X, et al. Jacobian norm with selective input gradient regularization for interpretable adversarial defense. Pattern Recognit. 2024;145:109902. doi:10.1016/j.patcog.2023.109902.
7. Cohen J, Rosenfeld E, Kolter JZ. Certified adversarial robustness via randomized smoothing. In: Proceedings of the 36th International Conference on Machine Learning (ICML); 2019 Jun 9–15; Long Beach, CA, USA. p. 1310–20.
8. Goodfellow IJ, Shlens J, Szegedy C. Explaining and harnessing adversarial examples. arXiv:1412.6572. 2015.
9. Fawzi A, Fawzi O, Frossard P. Fundamental limits on adversarial robustness. In: Proceedings of the ICML, 2015 Workshop on Deep Learning; 2015 Jul 6–11; Lille, France. 55 p.

10. Tabacof P, Valle E. Exploring the space of adversarial images. In: 2016 International Joint Conference on Neural Networks (IJCNN); 2016 Jul 24–29; Vancouver, BC, Canada. p. 426–33.
11. Cubuk ED, Zoph B, Schoenholz SS, Le QV. Intriguing properties of adversarial examples. arXiv:1711.02846. 2018.
12. Ilyas A, Santurkar S, Tsipras D, Engstrom L, Tran B, Madry A. Adversarial examples are not bugs, they are features. In: Advances in Neural Information Processing Systems 33 (NeurIPS 2019); 2019 Dec 8–14; Vancouver, BC, Canada. p. 125–36.
13. Yang K, Zhou T, Zhang Y, Tian X, Tao D. Class-disentanglement and applications in adversarial detection and defense. In: Advances in Neural Information Processing Systems 35 (NeurIPS 2021); 2021 Dec 6-14; Online. p. 16051–63.
14. Locatello F, Bauer S, Lucic M, Raetsch G, Gelly S, Schölkopf B, et al. Challenging common assumptions in the unsupervised learning of disentangled representations. In: Proceedings of the 36th International Conference on Machine Learning (ICML); 2019 Jun 9–15; Long Beach, CA, USA. Vol. 97, p. 4114–24.
15. Goodfellow I, Pouget-Abadie J, Mirza M, Xu B, Warde-Farley D, Ozair S, et al. Generative adversarial networks. Commun ACM. 2020;63(11):139–44. doi:10.1145/3422622.
16. Xu L, Zeng X, Li W, Xie Y. BH2I-GAN: bidirectional hash_code-to-image translation using multi-generative multi-adversarial nets. Pattern Recognit. 2023;133(4):109010. doi:10.1016/j.patcog.2022.109010.
17. Higgins I, Matthey L, Pal A, Burgess CP, Glorot X, Botvinick MM, et al. beta-VAE: learning basic visual concepts with a constrained variational framework. In: The 5th International Conference on Learning Representations (ICLR); 2017 Apr 24–26; Toulon, France. p. 1–22.
18. Kurakin A, Goodfellow IJ, Bengio S. Adversarial examples in the physical world. In: The 5th International Conference on Learning Representations (ICLR); 2017 Apr 24–26; Toulon, France. p. 1–14.
19. Tramèr F, Kurakin A, Papernot N, Goodfellow IJ, Boneh D, McDaniel PD. Ensemble adversarial training: attacks and defenses. In: The 6th International Conference on Learning Representations (ICLR); 2018 Apr 30–May 3; Vancouver, BC, Canada. p. 1–20.
20. Madry A, Makelov A, Schmidt L, Tsipras D, Vladu A. Towards deep learning models resistant to adversarial attacks. In: The 6th International Conference on Learning Representations (ICLR); 2018 Apr–May 3; Vancouver, BC, Canada. p. 1–23.
21. Athalye A, Engstrom L, Ilyas A, Kwok K. Synthesizing robust adversarial examples. In: Dy JG, Krause A, editors. Proceedings of the 35th International Conference on Machine Learning (ICML); Westminster, UK: PMLR; 2018. Vol. 80, p. 284–93.
22. Croce F, Hein M. Reliable evaluation of adversarial robustness with an ensemble of diverse parameter-free attacks. In: Proceedings of The 37th International Conference on Machine Learning (ICML); 2020 Jul 13–18; Online. Vol. 119, p. 2206–16.
23. Wang X, He K. Enhancing the transferability of adversarial attacks through variance tuning. In: 2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR); 2021 Jun 20–25; Nashville, TN, USA. p. 1924–33.
24. Wang H, Li G, Liu X, Lin L. A hamiltonian monte carlo method for probabilistic adversarial attack and learning. IEEE Trans Pattern Anal Mach Intell. 2022;44(4):1725–37. doi:10.1109/tpami.2020.3032061.
25. Carlini N, Wagner DA. Towards evaluating the robustness of neural networks. In: 2017 IEEE Symposium on Security and Privacy (SP); 2017 May 22–26; San Jose, CA, USA. p. 39–57.
26. Su L, Chen J, Jiang P. Generating adversarial examples for white-box attacks based on GAN. In: 2023 China Automation Congress (CAC); 2023 Nov 17–19; Chongqing, China. p. 910–5. doi:10.1109/cac59555.2023.10450537.
27. Pomponi J, Scardapane S, Uncini A. Pixle: a fast and effective black-box attack based on rearranging pixels. In: 2022 International Joint Conference on Neural Networks (IJCNN); 2022 Jul 18–23; Padua, Italy. p. 1–7.
28. Andriushchenko M, Croce F, Flammarion N, Hein M. A query-efficient black-box adversarial attack via random search. In: Computer Vision—ECCV 2020—16th European Conference. Cham, Switzerland: Springer; 2020. p. 484–501.
29. Zhang H, Wang J. Defense against adversarial attacks using feature scattering-based adversarial training. In: Wallach HM, Larochelle H, Beygelzimer A, d'Alché-Buc F, Fox EB, Garnett R, editors. Advances in Neural Information Processing Systems 33 (NeurIPS 2019); Westminster, UK: PMLR; 2019. p. 1829–39.

30. Xu W, Evans D, Qi Y. Feature squeezing: detecting adversarial examples in deep neural networks. In: 25th Annual Network and Distributed System Security Symposium (NDSS); 2018 Feb 18–21; San Diego, CA, USA. p. 1–15.
31. Guo C, Rana M, Cissé M, van der Maaten L. Countering adversarial images using input transformations. In: The 6th International Conference on Learning Representations (ICLR); 2018 Apr 30–May 3; Vancouver, BC, Canada. p. 1–12.
32. Liao F, Liang M, Dong Y, Pang T, Hu X, Zhu J. Defense against adversarial attacks using high-level representation guided denoiser. In: 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR); 2018 Jun 18–23; Salt Lake City, UT, USA. p. 1778–87.
33. Jin G, Shen S, Zhang D, Dai F, Zhang Y. APE-GAN: adversarial perturbation elimination with GAN. In: 2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP); 2019 May 12–17; Brighton, UK. p. 3842–6.
34. Nie W, Guo B, Huang Y, Xiao C, Vahdat A, Anandkumar A. Diffusion models for adversarial purification. In: Proceedings of the 39th International Conference on Machine Learning (ICML); 2022 Jul 17–23; Baltimore, MD, USA. Vol. 162, p. 16805–27.
35. Papernot N, McDaniel PD, Wu X, Jha S, Swami A. Distillation as a defense to adversarial perturbations against deep neural networks. In: 2016 IEEE Symposium on Security and Privacy (SP); 2016 May 22–26; San Jose, CA, USA. p. 582–97.
36. Bortolussi L, Carbone G, Laurenti L, Patane A, Sanguinetti G, Wicker M. On the robustness of bayesian neural networks to adversarial attacks. IEEE Trans Neural Netw Learn Syst. 2025;36(4):6679–92. doi:10.1109/tnnls.2024.3386642.
37. Peng S, Xu W, Cornelius C, Hull M, Li K, Duggal R, et al. Robust principles: architectural design principles for adversarially robust CNNs. In: The 34th British Machine Vision Conference 2023 (BMVC); 2023 Nov 20–24; Aberdeen, UK. p. 739–40.
38. Yang Z, Li L, Xu X, Zuo S, Chen Q, Zhou P, et al. TRS: transferability reduced ensemble via promoting gradient diversity and model smoothness. In: Advances in Neural Information Processing Systems 35 (NeurIPS 2021); Westminster, UK: PMLR; 2021. p. 17642–55.
39. Moosavi-Dezfooli S, Fawzi A, Uesato J, Frossard P. Robustness via curvature regularization, and vice versa. In: 2019 IEEE Conference on Computer Vision and Pattern Recognition (CVPR); 2019 Jun 15–20; Long Beach, CA, USA. p. 9078–86.
40. Zhang C, Benz P, Imtiaz T, Kweon IS. Understanding adversarial examples from the mutual influence of images and perturbations. In: 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR); 2020 Jun 13–19; Seattle, WA, USA. p. 14509–18.
41. Kim H. Torchattacks: a pytorch repository for adversarial attacks. arXiv:2010.01950. 2020.
42. Lecun Y, Bottou L, Bengio Y, Haffner P. Gradient-based learning applied to document recognition. Proc IEEE. 1998;86(11):2278–324. doi:10.1109/5.726791.
43. Krizhevsky A, Hinton G. Learning multiple layers of features from tiny images. In: Technical report. Toronto, ON, Canada: University of Toronto; 2009.
44. Zagoruyko S, Komodakis N. Wide residual networks. In: Proceedings of the British Machine Vision Conference (BMVC); 2016 Sep 19–22; York, UK. p. 87.1–12.
45. He K, Zhang X, Ren S, Sun J. Deep residual learning for image recognition. In: 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR); 2016 Jun 27–30; Las Vegas, NV, USA. p. 770–8.