Progressive Adversarial Contrastive Learning: Towards Efficient Data Augmentation in Adversarial Defense

Xiang Gao†, Weiqing Chen, Ying Cui, Xiang Dai, Lican Dai

The Southwest Institute of Electronic Technology, Chengdu 610036, China

Keywords: Adversarial defense; Data augmentation; Deep learning

Citation: Gao X., Chen W.Q., Cui Y., et al.: Progressive Adversarial Contrastive Learning: Towards Efficient Data Augmentation in Adversarial Defense. Data Intelligence, Vol. 7, Art. No.: 2025r36, pp. 1169–1191, 2025. DOI: https://doi.org/10.3724/2096-7004.di.2025.0190

ABSTRACT

In the realm of deep learning, adversarial attacks have surfaced as a significant concern, obstructing the deployment of deep neural networks (DNNs) in security-sensitive sectors. The cost of data annotation, increasing with data volume, has led to the exploration of adversarial training within self-supervised learning frameworks as a prevalent defense solution. Improving adversarial training is essentially a problem of exploring robust and efficient data augmentation. However, the computationally demanding nature of adversarial augmentation within a self-supervised learning framework poses dual challenges concerning performance and efficiency. In this study, we introduce Progressive Adversarial Contrastive Learning (PACL), an innovative defense mechanism against adversarial attacks on neural networks, which incorporates efficient adversarial data augmentation tactics. Specifically, PACL utilizes a sinusoidally enhanced adversarial sample strength growth strategy and has a universal adversarial strength control. This approach mitigates the inefficient computation of adversarial samples during the initial training phase, resulting in data augmentations that more closely mirror potential malicious inputs post-model deployment. Through exhaustive experiments against one supervised and four self-supervised competitive baselines, we empirically substantiate the effectiveness of PACL. Our results demonstrate a 33.90% reduction in computational overhead and an increase in robustness of up to 10.20%.

1. INTRODUCTION

Neural networks have achieved remarkable performance due to the deepening of theoretical research and advancements in computer hardware, often matching or even surpassing human capabilities in various

† Corresponding author: Gao Xiang (E-mail: gaoxiang9156@163.com).

© 2025 Chinese Academy of Sciences. Published under a Creative Commons Attribution 4.0 International (CC BY 4.0) license.

fields such as computer vision, natural language processing, and multi-modal learning [1]. Consequently, the practical applications of deep neural network models have significantly increased. However, in scenarios such as intelligent vehicles [2], face recognition [3], and even military applications [4], the security of these models remains a major concern, making the defense against malicious attacks a persistent research topic.

The concept of adversarial examples, initially discovered and proposed by C. Szegedy et al. [5], uncovers a stealthy and highly successful method to compromise neural networks. Minor perturbations to the original input of deep neural networks can drastically alter the output, enabling gradient-based adversarial attack methods to mislead deep neural networks into generating incorrect predictions. As a result, the deployment of neural networks in security-sensitive applications carries substantial risks, leading to a surge in research on adversarial defense mechanisms such as adversarial training [6-7].

Training a deep neural network is inherently data-driven, and the generation of adversarial examples in adversarial training can be viewed as a dataset augmentation based on the defender's understanding of the attack. Previous adversarial defense mechanisms have proven insufficient to fully protect neural networks from adversarial attacks [8], as the rising cost of data labeling limits the scale and quality of the augmented dataset. Recent research focusing on integrating self-supervised learning frameworks with adversarial training [9] opens a new direction for adversarial defense. However, the computationally demanding nature of adversarial augmentation within a self-supervised learning framework exacerbates challenges related to performance and efficiency, which can be a disaster for training large robust models.

In this paper, our central emphasis lies in augmenting the efficacy of data augmentation in adversarial defense, thereby enhancing the robustness and efficiency of neural networks in these self-supervised contexts. Based on adversarial contrastive learning methods [9], we aim to simultaneously enhance both the performance and efficiency of the defense. Specifically, we use the contrastive loss function as a supervisory signal to generate adversarial samples, thus facilitating self-supervised adversarial training and basic adversarial contrastive learning. We further analyze the phenomenon of redundancy or even the negative impact of adversarial samples during the early stages of training in adversarial contrastive learning. To address the issues brought by this phenomenon, we propose a sinusoidally enhanced adversarial attack strength during different training stages of contrastive training. This leads to the development of an improved adversarial contrastive learning framework, namely, Progressive Adversarial Contrastive Learning (PACL), designed to enhance model robustness and training efficiency.

We validate the effectiveness of PACL through a series of experiments conducted on the mainstream neural network model ResNet and the image classification dataset CIFAR10. The robust accuracy, or the classification accuracy of adversarial examples, serves as the main metric to evaluate the performance of our method. Experimental results show that our proposed method outperforms traditional supervised learning methods under self-supervised conditions, with an increase in robustness of up to 10.20%. Furthermore, compared to existing self-supervised methods, PACL significantly reduces the total training time by 33.90%, thereby enhancing efficiency. This study highlights the potential of the universal PACL

for improving the robustness and efficiency of most deep neural networks, paving the way for their deployment in security-sensitive sectors.

Our contributions can be summarized in three key points: (1) We have identified the issue of redundancy and even negative impact of adversarial examples during the early stages of adversarial contrastive learning training. (2) To address this phenomenon, we propose an adversarial training approach that gradually increases the adversarial intensity, thereby enhancing the model's robustness and training efficiency. (3) We have empirically demonstrated the effectiveness of our method through experiments.

2. RELATED WORKS

2.1 Adversarial training

Adversarial training, serving as a defensive mechanism against adversarial attacks, has been the subject of extensive scrutiny within the machine learning community. Good-fellow et al. [6] were the pioneers of this concept, introducing the Fast Gradient Sign Method (FGSM) for the generation of adversarial examples and subsequently utilizing them for training. This methodology was further refined by Madry et al. [10] with the introduction of Projected Gradient Descent (PGD), a more robust adversarial training technique that exhibited enhanced resistance to adversarial attacks. Zhang et al. [11] introduced TRADES, a method that balances robustness and accuracy by considering the trade-off between them. Despite these advancements, the practical effectiveness of adversarial training in real-world contexts continues to be a vibrant area of research, with persistent endeavors to enhance its scalability and robustness.

2.2 Contrastive adversarial training

Recently, contrastive learning in the realm of self-supervised learning has garnered considerable attention, with its efficacy deriving from its inherent capability as a self-supervised learning method to leverage substantial volumes of unlabeled data [12-15]. Analogous to empirical evidence in natural language processing, an escalation in data volume may culminate in an enhancement of model performance. Nonetheless, given equal amounts of data, contrastive learning techniques such as SimCLR [16] can attain commensurate or even superior performance compared to supervised learning methods. This might be attributed to the capability of contrastive learning to more effectively preserve instance-level features of data, such feature representations can significantly contribute to a broad spectrum of downstream tasks.

Subsequently, a logical conjecture would be that contrastive learning could also bolster the adversarial defense. Robust contrastive learning (RoCL) [9] first presents an adversarial defense methodology that amalgamates contrastive learning and adversarial training. RoCL along with its variants are collectively referred to as adversarial contrastive learning. However, training procedures involving adversarial contrastive learning are computationally demanding, as both contrastive learning and adversarial data augmentation multiplies the total data volume, in addition to the significant extra computation overhead incurred by the adversarial data augmentation process itself. Thus the escalation of the overall training time in adversarial contrastive learning remains a significant problem unsolved.

2.3 Strategies to reduce overhead

To reduce the cost in adversarial training, researchers have proposed various methods: Free Adversarial Training (Free-at) [17] utilizes gradient information to generate adversarial examples during model parameter updates, effectively eliminating the overhead associated with traditional methods. You Only Propagate Once (YOPO) [18] optimizes computational costs by only considering the first layer of the network for forward and backward propagation, reducing the propagation volume to one per update since adversary updates are only related to the first layer. Latent Adversarial Training (LAT) [19] explores different layers of the network, including fine-tuning adversarial training models to ensure robustness at the latent level, as latent layers are susceptible to small adversarial perturbations. Cheap Adversarial Training (Cheap-at) [20] suggests using weaker and cheaper adversaries like FGSM combined with random initialization to effectively train robust networks, instead of relying on computationally expensive methods. Fast Adversarial Training (FAT) [21], inspired by the results of Low-at, optimizes the inner maximization problem effectively through random smoothing and proposes a new initialization strategy called backward smoothing, which helps improve model stability and robustness using single-step robust training methods. Additionally, some methods optimize the training process by controlling the strength of adversarial examples, such as controlling the proportion of adversarial examples in a batch of data; controlling the strength of adversarial examples by controlling the step size of a single iteration of generating adversarial examples (Fast Adversarial Training with Adaptive Step Size [22]); controlling the strength of adversarial examples by controlling the maximum value of the norm distance (Instance Adaptive Adversarial Training [23]); controlling the strength of adversarial examples by controlling the number of iterations of generating adversarial examples (Curriculum Adversarial Training [24]); controlling the overall strength of adversarial examples by stopping adversarial training early (Friendly Adversarial Training [25]); controlling the strength of adversarial examples by controlling the weight of the original samples of generating adversarial examples (Misclassification Aware adversarial Training [26]) and so on.

While the aforementioned methods have been effective in reducing the cost of adversarial training, they may not be directly applicable to contrastive adversarial training due to the different nature of the training objectives and processes. Contrastive adversarial training focuses on learning robust feature representations by contrasting examples, which may require different optimization strategies.

In contrast to existing work, this paper presents a novel approach that tailors contrastive learning to the unique characteristics of adversarial training, thereby enhancing model robustness. Recognizing the specific demands of each phase in adversarial contrastive learning, our method introduces a progressive adversarial strength strategy. This strategic adjustment of adversarial intensity not only reduces the computational overhead but also crucially prevents the generation of ineffective or misleading adversarial samples during the initial training phase, which could otherwise hinder model convergence and adversely affect the final performance.

Our approach optimizes for the distinct stages of adversarial contrastive learning, ensuring that the model's exposure to adversarial examples is calibrated to its evolving capabilities. This targeted

optimization enhances the robustness of deep neural networks against adversarial attacks while simultaneously improving training efficiency. By addressing both the computational challenges and the need for accuracy in adversarial contrastive learning, our method offers a comprehensive solution to the complex demands of this training paradigm.

3. PRELIMINARIES

The objective of an adversarial sample attack is to manipulate the input and induce the model into generating a prediction that deviates from the actual label. This manipulation involves introducing subtle alterations to the input, which remain undetectable to the human eye. The attack is executed after the model deployment by exploiting inherent weaknesses within the neural network. The stealthiness of these input-specific changes makes the attack not only difficult to prevent before deployment, but also challenging to detect during the inference phase. Here is the symbolic expression of the definition of adversarial examples:

$$\begin{aligned} \arg\min_{x'} \quad & \mathcal{D}(x,\ x') \\ s.t. \quad & \mathcal{F}_\theta(x') = \hat{y} \neq y \end{aligned} \tag{1}$$

where $\mathcal{D}$ is a distance metric, $\mathcal{F}$ is the model with its parameter θ, $\hat{y}$ and y are the predicted and true labels of the neural network, respectively.

The majority of research on adversarial defenses concentrates on tackling the more perilous white-box attacks [6, 10, 27-28]. Under white-box settings, the attacker can utilize the model's loss function to gain a deeper understanding of how the model predicts the label of an input. Subsequently, the attacker can generate adversarial examples by employing the partial derivatives of the loss function concerning the inputs, and identify the direction of the perturbations that amplify the loss function. The optimization problem in (1) can be reformulated as the ensuing constrained optimization problem:

$$\begin{aligned} \arg\max_{\delta} \quad & \mathcal{L}(\theta,\ x',\ y) \\ s.t. \quad & \mathcal{D}(x,\ x') \leq \epsilon \end{aligned} \tag{2}$$

where θ is the parameter of the model, and $\mathcal{L}$ is a loss function. For example, the cross-entropy loss $\mathcal{L}_{CE}$ is commonly used in classification problems.

To defend against adversarial attacks, adjusting the model's structure or finding more robust parameters are common solutions. As a representative work, adversarial training has developed into many variants [17-18]. The core idea of adversarial training is to add adversarial samples as inputs to the original training process so that the model can optimize based on possible adversarial samples in the training phase, thus making the final model parameters have more robust decision boundaries. The adversarial

training process can be summarized as the following maximization-minimization two-layer optimization problem:

$$\arg \min_{\theta} \mathbb{E}_{(x, y)\sim\mathbb{D}}\left[\max_{x+\delta \in B(x, \epsilon)} \mathcal{L}(\theta, x+\delta, y)\right] \tag{3}$$

where δ is the adversarial perturbation, and $\mathcal{L}$ is the loss function (e.g., cross-entropy).

Describing adversarial training from a higher perspective, the process of adversarial training is a data augmentation of the original input data through the generation of adversarial samples, which adds difficult-to-fit data samples in the neighborhood of the original input [29], making the model have smoother decision boundaries and better generalization when facing unknown inputs. Hence things are similar as long as we can generate adversarial samples within the self-supervised frameworks. Robust Contrastive Learning (RoCL) replaces supervised adversarial sample generation in (3) with a self-supervised approach, making it suitable for the contrastive learning framework. The process for generating adversarial samples in RoCL is illustrated in (4).

$$t(x)^{i+1} = \Pi_{B(t(x), \epsilon)}\left(t(x)^{i} + \alpha \times \operatorname{sign}\left(\nabla_{t(x)^{i}} \mathcal{L}_{contrastive, \theta, \pi}\left(t(x)^{i}, \{t'(x)\}, \{t(x)_{neg}\}\right)\right)\right) \tag{4}$$

where t and t' represent two distinct image transformations used for constructing positive sample pairs, i stands for the current iteration step. ϵ is the perturbation bound, and α is the step size. It is not difficult to discern that this type of attack substitutes the cross-entropy loss function, which requires class labels in the original Projected Gradient Descent (PGD) method, with the contrastive loss function in SimCLR. Viewed from another perspective, the positive and negative sample sets corresponding to each instance can be seen as its labels. The contrastive loss is defined in (5):

$$\begin{aligned} &\mathcal{L}_{contrastive, \theta, \pi}(x, \{x_{pos}\}, \{x_{neg}\}) \\ &= -\log \frac{\sum_{x_{pos}} \exp\left(\operatorname{sim}\left(z, z^{+}\right)/\tau\right)}{\sum_{x_{pos}} \exp\left(\operatorname{sim}\left(z, z^{+}\right)/\tau\right) + \sum_{x_{neg}} \exp\left(\operatorname{sim}\left(z, z^{-}\right)/\tau\right)} \end{aligned} \tag{5}$$

where $\{x_{pos}\}$ and $\{x_{neg}\}$ denote the sets of positive and negative samples that correspond to x, respectively. The variable z represents the feature vector of a sample after it is processed by the feature extraction network f_θ and the projection network $g_{\pi'}$ such that $z = g_\pi(f_\theta(x))$. Here, z^+ and z^- refer to the feature vectors of the corresponding positive and negative samples, respectively. The term sim is a measure of similarity. In the SimCLR model, cosine similarity is used, defined as $\operatorname{sim} = \frac{u^T v}{||u||||v||}$. Finally, τ is a temperature parameter used for adjusting smoothness.

4. METHOD

4.1 Motivation

We propose an enhancement approach, grounded in the insight that incorporating adversarial samples during the initial phase of adversarial contrastive learning proves counterproductive rather than advantageous. During conventional adversarial training, the model's early-stage training is inadequate, leading to inaccurate classification of clean samples. Consequently, the gradient of clean samples undergoes significant changes in the early stages, causing the perturbation direction of the same image computed on the model to deviate markedly from that computed on the final model. Adversarial samples generated at this stage fail to represent the potential post-deployment adversarial attacks. Furthermore, as adversarial perturbation involves gradient ascent, the direction of the model parameter update induced by an adversarial sample is likely opposite to the direction caused by the original sample. Consequently, adversarial samples employed during the initial training phase fail to enhance model robustness, and may even degrade the final classification accuracy.

This phenomenon becomes more pronounced in adversarial contrastive learning, attributable to two primary factors. Firstly, contrastive learning introduces multiple augmentations and intricate contrast loss functions, significantly impeding model convergence and leading to the computation of ineffective or even counterproductive adversarial samples. Secondly, the division of adversarial contrastive learning into multiple training phases results in similar ineffective computations of adversarial samples across the corresponding network modules in each phase. Hence, a gradual increase in adversarial sample strength becomes imperative for successful adversarial contrastive learning, enhancing both training efficiency and the final model's robustness. The focus of this paper is on regulating the strength of adversarial samples across three stages of adversarial contrastive learning as shown in Figure 1: training the backbone network, adversarial training of the classifier head, and the final fine-tuning of the model.

4.2 Progressive adversarial data augmentation strategy

Currently, various approaches have been proposed to dynamically adjust adversarial sample strength. For instance, methods like Curriculum Adversarial Training [24] and Friendly Adversarial Training [25] adaptively modify adversarial strength by scrutinizing robust accuracy. Additionally, Fast Adversarial Training with Adaptive Step Size [22] primarily adjusts adversarial sample strength based on the moving average of the gradient norm.

It is essential to note that while certain methods propose control strategies to alter sample strength, their objectives may differ from ours. For instance, Friendly Adversarial Training [25] initially aims to mitigate the overfitting phenomenon during adversarial training on adversarial samples. Moreover, these methods are applied in the conventional adversarial training framework, overlooking the significance of the adversarial sample strength control strategy in adversarial contrastive learning.

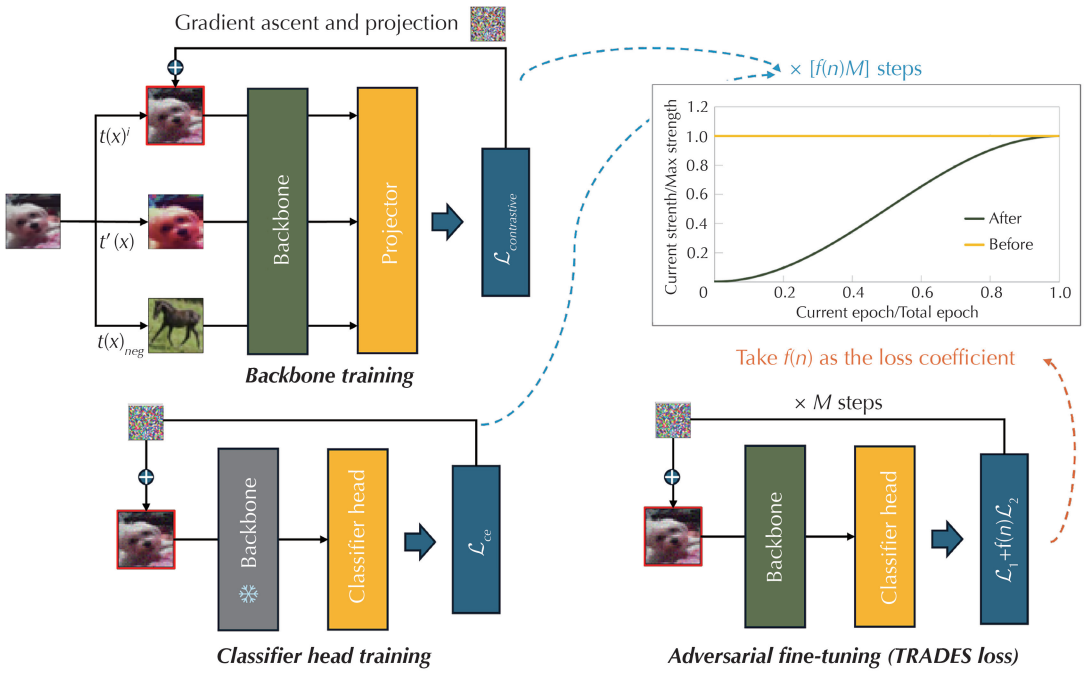


Figure 1. Illustration for the generation of adversarial augmented data in the n^{th} epoch of the each training stage.

Upon analysis of the aforementioned methods, it becomes evident that they entail substantial additional computations, rendering them impractical for controlling adversarial sample strength in adversarial contrastive learning, given the already extensive computational demands of this approach. Furthermore, the control mechanisms in these methods hinge on diverse factors. Certain factors, such as classification accuracy on the validation set, are not applicable in the context of adversarial contrastive learning.

This paper specifies the strength of adversarial examples conditioning on training epochs. In the process of dynamically adjusting adversarial sample strength, various growth curves can be considered, such as quadratic functions, which also exhibit a gradual increase. However, this paper employs the sinusoidal enhancement strategy instead of using a quadratic or other nonlinear growth curves due to several advantages.

The sinusoidal function provides a smooth growth pattern, especially at both the beginning and end of the curve, allowing for a gradual transition of adversarial strength. In comparison, the quadratic function tends to start with a slower growth rate and accelerates significantly toward the later stages of training. This rapid acceleration can lead to an excessive increase in adversarial strength during the final stages, which may cause overfitting and destabilize the model, particularly when the model is nearing convergence.

Additionally, the periodic nature of the sinusoidal function offers natural flexibility in controlling adversarial strength across different training phases. During the earlier stages, the moderate growth of the

sinusoidal curve aligns well with the model's need to gradually adapt to adversarial samples, reducing the negative impact they might have on early training. As the training progresses, the sinusoidal function's smooth convergence ensures that the adversarial strength increases steadily without causing sudden, disruptive changes, which helps maintain the model's stability in later stages.

Moreover, the sinusoidal function is computationally efficient and integrates seamlessly with existing learning rate decay strategies, such as cosine decay, which is widely recognized as an effective way to manage learning rates. The smooth nature of the sinusoidal function complements this strategy, contributing to more coordinated training without increasing computational overhead. In contrast, the quadratic function lacks such flexibility and can introduce additional complexity, which may hinder training efficiency, especially when dealing with large datasets or complex models.

Therefore, the sinusoidal enhancement strategy offers distinct advantages in terms of smoothness, flexibility, and computational efficiency, making it a suitable choice for progressively adjusting adversarial sample strength. This approach effectively balances the need for improved model robustness and efficient training, ensuring the model performs reliably under adversarial conditions. Inspired by this, this paper adopts a sinusoidally enhanced adversarial sample strength growth strategy, as shown in the (6):

$$advStrength = f(n) = \frac{1}{2}M \times \sin\left(\left(\frac{n}{N} - 0.5\right)\pi\right) + \frac{1}{2}M \tag{6}$$

where n is the current training epoch, N is the total number of training epochs, and M is the maximum adversarial strength. Figure 2 shows the curve of the adversarial sample strength corresponding to this function as the training rounds increase.

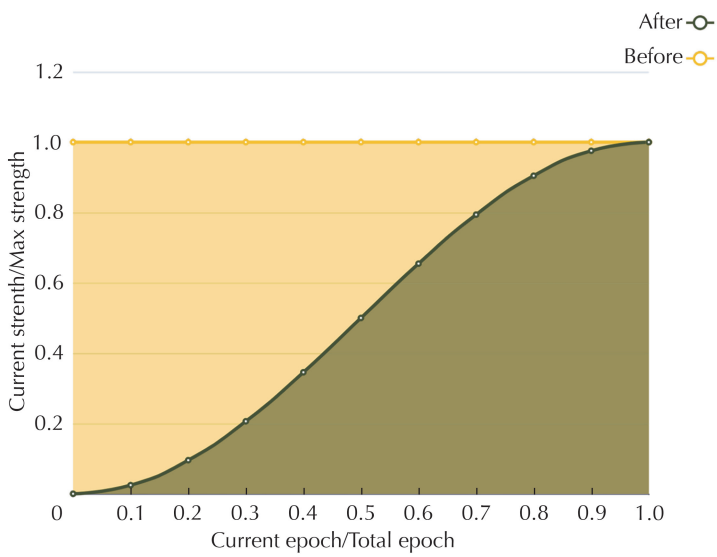


Figure 2. The adversarial sample strength during the training process.

It is noteworthy that when the strength of the adversarial sample exhibits a linear correlation with the time needed for adversarial sample generation, the area beneath this curve and the abscissa axis can serve as a representation of the ongoing training time. Consequently, this enhancement holds particular significance in enhancing the efficiency of adversarial contrastive learning, especially in scenarios involving exceedingly prolonged training durations. An in-depth analysis of this aspect will be presented in the experimental research section.

4.3 Controlling adversarial data augmentation strength

We have defined the importance of strength control and proposed specific numerical adjustment strategies in the above section. However, there are various ways to evaluate the strength of adversarial samples, and different aspects can be considered for controlling their strength. Therefore, this paper also needs to specify how to control the strength of adversarial samples, in order to implement the strategy of adjusting the strength of adversarial samples.

In contrast to the previous works, this paper mainly uses two methods to control the strength of adversarial examples to adapt to the specific needs of different stages of adversarial contrastive learning. Specifically, the training process of PACL is divided into three main stages:

Backbone Network Training. In this stage, the backbone network of the model is trained through self-supervised adversarial training. The adversarial samples introduced at this stage are mainly used to enhance the robustness of the feature extraction part. Since the model has not been fully trained at this stage, introducing high-intensity adversarial samples too early may interfere with the normal learning of the model. Therefore, this method adopts a sine wave enhancement strategy to gradually increase the intensity of adversarial samples, enabling the model to gradually adapt to the interference of adversarial samples and improve the robustness of overall feature extraction. The adversarial intensity in this stage is realized by adjusting the number of iterations for generating adversarial samples, and we modify the following formula from formula (4):

$$t(x)^{i+1} = \Pi_{B(t(x),\,\epsilon)}\left(t(x)^{i} + \alpha \times \operatorname{sign}\left(\nabla_{t(x)^{i}} \mathcal{L}_{\text{contrastive},\,\theta,\,\pi}\left(t(x)^{i},\ \{t'(x)\},\ \{t(x)_{\text{neg}}\}\right)\right)\right), \quad \text{s.t. } i \le f(n) \tag{7}$$

where $t(x)^{i}$ is the adversarial sample at iteration i, resulting from augmenting input x. The gradient $\nabla_{t(x)^{i}} \mathcal{L}_{\text{contrastive},\,\theta,\,\pi}$ pertains to the contrastive loss function, distinguishing positive $\{t'(x)\}$ from negative samples $\{t(x)_{\text{neg}}\}$. The sign function $\operatorname{sign}\left(\nabla_{t(x)^{i}} \mathcal{L}_{\text{contrastive}}\right)$ directs adversarial perturbations, and α modulates the perturbation magnitude. The projection $\Pi_{B(t(x),\,\epsilon)}$ confines updated samples $t(x)^{i+1}$ within an ϵ neighborhood, ensuring perturbations are within an l_{∞} bound. Iteratively, this process crafts adversarial samples to maximize contrastive loss, enhancing model robustness against such perturbations.

Classifier Head Training. After completing the training of the Backbone, the next step is to train the classifier head through adversarial training. This stage usually adopts supervised adversarial training methods to further enhance the robustness of the classification decision part. Similar to the Backbone training stage, we also apply sine wave enhanced adversarial sample intensity in the training of the classifier head to ensure that the adversarial samples introduced in the early stage do not cause too much interference to the model, ultimately enhancing the model's classification ability. The adversarial intensity in this stage is realized by adjusting the number of iterations for generating adversarial samples. The reason for choosing this specific method to adjust the strength is that these two stages are the computationally intensive stages of adversarial contrastive learning. By controlling the number of iterations to generate adversarial examples, the strength of adversarial examples can be gradually increased while significantly reducing the computational cost of adversarial example iteration.

Fine-tuning of the Entire Model. For the last fine-tuning stage of the model, since the model fine-tuning requires fewer training rounds, the specific method of controlling the strength of adversarial examples can no longer consider computational optimization and focus more on the specific effects of strength control. This paper controls the actual strength of adversarial examples in training by controlling the degree of attention to adversarial examples.

Specifically, since the model fine-tuning stage of adversarial contrastive learning usually uses TRADES to achieve better fine-tuning effects, the intensity of attention to adversarial examples can be adjusted by controlling the coefficient of the regularization term representing the model robustness in the loss function used by TRADES for model fine-tuning. That is, controlling the value of $1/\lambda$ to control how much TRADES loss function pays attention to adversarial examples in different stages of training.

$$\arg\min_{\theta} \mathbb{E}\left\{\mathcal{L}_1(f_\theta(\boldsymbol{X}),\ \boldsymbol{Y}) + f(n) \max_{\boldsymbol{X}' \in \mathbb{B}(\boldsymbol{X},\ c)} \mathcal{L}_2\left(f_\theta(\boldsymbol{X}),\ f_\theta\left(\boldsymbol{X}'\right)\right)\right. \tag{8}$$

In the TRADES loss function, the coefficient $1/\lambda$ controls the trade-off between natural and robust accuracy. Here, we replace $1/\lambda$ with the sinusoidally enhanced value $f(n)$ ((6)) to progressively adjust this trade-off. This control method does not directly change the generation process of adversarial examples to affect their adversarial strength but uses TRADES's loss function to achieve precise and continuous strength control, which distinguishes it from existing methods of dynamically controlling the strength of adversarial examples.

5. EXPERIMENT

5.1 Settings

The experiments were conducted on machines equipped with NVIDIA RTX 2080Ti GPUs, Intel(R) Xeon(R) Silver 4216 CPU, and 251GB of RAM. All experiments were performed on Ubuntu 18.04.6 operating systems using the PyTorch 1.11.0 framework. The network architecture utilized for the experiments is ResNet18 [30], a widely acclaimed convolutional neural network (CNN) prominent in

various research domains within computer vision. ResNet18 is acknowledged for its effectiveness and has gained popularity in numerous computer vision applications. Concerning the data, we selected CIFAR10 [31] as the dataset, a widely utilized collection of color images for image classification tasks. It is noteworthy that the combination of ResNet and CIFAR10 currently stands as the most prevalent pairing in adversarial attack and defense research, emphasizing its widespread adoption in the field. Additionally, we have employed the MNIST dataset, which serves as another commonly utilized image dataset that has been extensively adopted in recent adversarial training research papers. The data characteristics of MNIST offer a distinct data dimension in adversarial training experiments, further enriching the assessment of model robustness and the validation of training effects. The use of multiple datasets, particularly under the ResNet18 architecture, provides a more comprehensive experimental foundation for the study of adversarial training.

For the adversarial attack used in adversarial contrastive training, we embrace the l_∞ norm constraint paradigm, with the distance threshold ϵ set at 8/255, approximated as 0.0314. Notably, this constraint stands as a benchmark widely employed across existing literature. The l_∞ constraint is equivalent to perturbing the last 3 bits of the standard 8-bit representation in a profound digital image. This modification results in subtle alterations imperceptible to the human eye, aligning with the foundational premise that adversarial samples should be challenging for human detection. Simultaneously, this constraint ensures that the success rate of adversarial attacks approaches near 100%, thus guaranteeing the relative strength of the adversarial samples. Consequently, it is acknowledged as a fitting constraint. Regarding the step size α and the maximum number of iteration rounds k for the iterative generation of adversarial samples, this study employs $\alpha = 0.007$ and $k = 7$, respectively.

For the adversarial attack used in the test, we adopt the standard projected gradient descent (PGD) to generate the adversarial samples under the l_∞ constraint, with $\epsilon = 0.0314$, $\alpha = 0.00314$, $k = 20$. This set of settings is also often labeled as PGD20 other works.

It should be noted that, in the experiments on the MNIST dataset, to ensure the success of adversarial attacks and maintain consistency with existing literature, we referred to recent studies and adopted appropriate parameter settings. For adversarial contrastive training, the training configuration on the MNIST dataset is: epsilon = 0.3, alpha = 0.007, $k = 7$, which ensures that the model effectively handles the generation and processing of adversarial samples during training.

Similarly, to effectively evaluate the robustness of the model during testing, we adopted parameter settings consistent with standard practices in the field. The attack configuration on the MNIST dataset is: epsilon = 0.15, alpha = 0.03, $k = 7$, ensuring the success of the attacks and the reliability of the evaluation results.

To ensure the reproducibility of the experiments, we also provide a complete list of hyperparameters in Table 1. The selection of these parameters is based on preliminary experiments and recommendations from the literature. For instance, we chose $\epsilon = 8/255$ based on its widespread recognition and effectiveness

Table 1. Hyperparameter settings for CIFAR-10 and MNIST datasets.

Parameter	CIFAR-10	MNIST
Lambda (λ)	256	256
Learning rate (lr)	0.1	0.1
Weight decay (decay)	1×10^{-6}	1×10^{-6}
Random seed (seed)	0	0
Batch size (batch_size)	256	256
Number of epochs (epoch)	300	300
Perturbation magnitude (epsilon)	0.0314	0.3
Step size (alpha)	0.007	0.007
Number of iterations (k)	7	7

under the L_∞ norm constraint. The learning rate and batch size were determined through a series of preliminary experiments to ensure stable model training and good performance.

The baseline is categorized into three types: unprotected, representative supervised defense (TRADES), and adversarial contrastive learning variants [32].

5.2 Main result

Figure 3 is an overview of the average robustness improvement of PACL over adversarial contrastive learning using robust accuracy as the evaluation metric. We evaluate the performance on three main stages of adversarial contrastive learning, i.e., the training of the backbone with self-supervised adversarial training, the classifier head training with supervised adversarial training, and the final fine-tuning of the entire network. It can be seen from the results that PACL can continuously and effectively improve the robustness of the model in these three stages, especially in the early training stage, by gradually increasing the intensity of adversarial attacks, the model enhances the ability to distinguish between clean samples and adversarial samples. This progressive strategy not only reduces the vulnerability of the model to counter attacks in the early training stage, but also lays a more stable foundation for the subsequent network fine-tuning stage. These findings indicate that PACL can make full use of the progressive adversarial sample enhancement in the initial training period when building more robust models, which alleviates the problem of slow robustness improvement in traditional adversarial training methods.

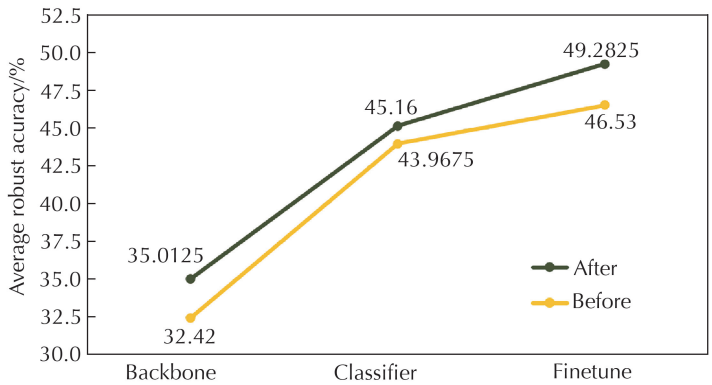


Figure 3. The average robust accuracy of PACL in different stages compared with RoCL.

5.3 Performance on backbone training

Firstly, we verify the effect of using progressive adversarial sample strength in the backbone network training of adversarial contrastive learning, as shown in Table 2 and Table 3. In this set of experiments, clean samples were used to train the classifier head after the backbone network training was completed, which can better reflect whether the robustness of the backbone network has been improved. The difference before and after the improvement lies in whether the training of the backbone network uses the sinusoidally enhanced progressive adversarial sample strength proposed in this paper for adversarial contrastive learning. In this scenario, specifically, the number of iterations increases from 0 to 7 in a sinusoidal curve according to the current training round.

Table 2. Model performance w/o and w/ PACL during the backbone training of adversarial contrastive learning on CIFAR10 using ResNet18.

Base method	w/o PACL			w/PACL		
	CA/%	RA/%	AA/%	CA/%	RA/%	AA/%
No defense	90.44	00.00	45.22	-	-	-
TRADES	77.03	48.01	62.52	-	-	-
RoCL	79.78	28.40	54.09	**81.39** ↑	**33.26** ↑	**57.33** ↑
RoCL (IP)	83.16	35.69	59.43	**84.79** ↑	**39.14** ↑	**61.97** ↑
RoCL (HN)	81.27	28.31	54.79	**81.50** ↑	**33.25** ↑	**57.38** ↑
RoCL (IP & HN)	83.67	37.28	60.48	82.92	34.40	58.66

Table 3. Model performance w/o and w/ PACL during the backbone training of adversarial contrastive learning on MNIST using ResNet18.

Base method	w/o PACL			w/PACL		
	CA/%	RA/%	AA/%	CA/%	RA/%	AA/%
No defense	98.21	0.21	49.26	-	-	-
TRADES	98.92	76.88	87.90	-	-	-
RoCL	95.57	58.88	77.23	**95.59** ↑	47.73	71.66
RoCL (IP)	95.40	41.59	68.50	**95.86** ↑	**42.56** ↑	**69.21** ↑
RoCL (HN)	95.89	62.89	79.39	95.78	58.39	77.09
RoCL (IP & HN)	95.67	55.27	75.47	**96.00** ↑	**57.04** ↑	**76.52** ↑

Analyzing the three accuracy metrics, we find that the use of progressively enhanced adversarial samples for training against the backbone network effectively improves the performance of most of the frameworks, with RoCL and RoCL (HN) having the most significant improvements. Taking RoCL as an example, RA increased from 28.40% to 33.26% after PACL was used, showing that the robustness of the model was effectively improved by dynamically adjusting the strength of the adversarial sample. This improvement was also reflected in the AA, which improved from 54.09% to 57.325%, demonstrating the overall performance of the model when faced with both clean and adversarial samples.

In the RoCL (IP) and RoCL (HN) frameworks, PACL also provides significant robustness improvements. For example, in the RoCL (IP) framework, RA increased from 35.69% to 39.14%, indicating that gradually increasing the strength of the adduction sample through PACL can help the model gradually adapt to more challenging adduction attacks. Improvements in RA are also evident in the RoCL (HN) framework, showing that PACL methods can effectively improve robustness in a variety of adversarial training frameworks. In addition, the introduction of PACL did not have a significant negative effect on the classification ability of clean samples, and on the contrary, in some cases (such as RoCL (IP) and RoCL (HN)), classification accuracy (CA) even showed a small increase. This shows that PACL not only improves the robustness of the model against adversarial samples, but also retains its good performance against clean samples.

5.4 Performance on classifier head training

Secondly, we verify the effect of using progressive adversarial sample strength in the classifier head training of adversarial contrastive learning, as shown in Table 4 and Table 5. The w/o PACL refers to the training classifier head with a uniform number of adversarial sample iterations throughout the training process, which is 10 in this experiment. The w/ PACL refers to the training classifier head for adversarial contrastive learning methods with sinusoidally enhanced asymptotic adversarial sample strength. The range of the strength is from 0 to 10, so the strength of the adversarial samples remains the same at the last epoch of training.

Table 4. Model performance w/o and w/ PACL during the classifier head training of adversarial contrastive learning on CIFAR10 using ResNet18.

Base method	w/o PACL			w/PACL		
	CA/%	RA/%	AA/%	CA/%	RA/%	AA/%
No defense	90.44	00.00	45.22	-	-	-
TRADES	77.03	48.01	62.52	-	-	-
RoCL	78.41	40.60	59.51	78.35	40.51	59.43
RoCL (IP)	80.85	43.72	62.29	77.17	**48.32** ↑	**62.75** ↑
RoCL (HN)	77.50	45.17	61.33	77.29	**45.27** ↑	61.28
RoCL (IP & HN)	81.35	46.38	63.87	81.00	**46.54** ↑	63.77

Table 5. Model performance w/o and w/ PACL during the classifier head training of adversarial contrastive learning on MNIST using ResNet18.

Base method	w/o PACL			w/PACL		
	CA/%	RA/%	AA/%	CA/%	RA/%	AA/%
No defense	98.21	0.21	49.26	-	-	-
TRADES	98.92	76.88	87.90	-	-	-
RoCL	94.32	71.18	82.75	**94.62** ↑	60.36	77.49
RoCL (IP)	94.09	52.37	73.23	**94.54** ↑	**52.59** ↑	**73.57** ↑
RoCL (HN)	94.71	73.63	84.17	**95.14** ↑	71.18	83.16
RoCL (IP & HN)	94.72	67.22	80.97	**95.18** ↑	**69.16** ↑	**82.17** ↑

As can be seen from the results in Table 4 and Table 5, RoCL and its variants outperform non-PACL versions on several measures of robustness accuracy with PACL. For example, in the RoCL (IP) framework, RA with PACL increased from 43.72% to 48.32%, while AA also increased from 62.285% to 62.745%. Although the number of iterations of the adversarial sample was reduced at the beginning of training, this adjustment did not have a negative impact on the final robustness, and on the contrary, the model showed stronger resistance to attack in the final stage. This shows that by gradually increasing the strength of the countersample, excessive disturbance in the early stage of training can be effectively avoided, unnecessary calculation cost can be reduced, and the robustness of the final training results can be ensured.

An important finding was that PACL was not only able to improve robustness in classification head training, it was also able to meet or exceed the effects of classical supervised adversarial training methods

such as TRADES without the need for additional model fine-tuning. For example, in the RoCL (IP) experiment, RA increased from 46.38% to 46.54% which is higher than TRADES.

5.5 Performance on finetuning

Finally, we verify the effect of using progressive adversarial sample strength in model fine-tuning for adversarial contrastive learning, as shown in Table 6. Since the overall number of training epochs in the model fine-tuning stage here is small and does not need to pay too much attention to the time overhead, we choose TRADES as the supervised adversarial finetuning method. The w/o PACL refers to using a unified robust regularization $\lambda = 6$ for the TRADES finetuning. And the w/ PACL refers to using a sinusoidally enhanced λ from 0 to 6 for the TRADES finetuning.

Table 6. Model performance w/o and w/ PACL during the finetuning of adversarial contrastive learning on CIFAR10 using ResNet18.

Base method	w/o PACL			w/PACL		
	CA/%	RA/%	AA/%	CA/%	RA/%	AA/%
No defense	90.44	00.00	45.22	-	-	-
TRADES	77.03	48.01	62.52	-	-	-
RoCL	83.25	43.19	63.22	81.00	**47.60** ↑	**64.30** ↑
RoCL (IP)	81.31	45.99	63.65	**82.90** ↑	**47.18** ↑	**65.04** ↑
RoCL (HN)	81.90	48.37	65.14	80.98	**50.11** ↑	**65.55** ↑
RoCL (IP & HN)	83.94	48.57	66.26	76.66	**52.24** ↑	64.45

Table 7. Model performance w/o and w/ PACL during the finetuning of adversarial contrastive learning on MNIST using ResNet18.

Base method	w/o PACL			w/PACL		
	CA/%	RA/%	AA/%	CA/%	RA/%	AA/%
No defense	98.21	0.21	49.26	-	-	-
TRADES	98.92	76.88	87.90	-	-	-
RoCL	98.92	75.26	87.09	98.73	**79.01** ↑	**88.87** ↑
RoCL (IP)	99.01	59.22	79.12	98.93	**73.35** ↑	**86.14** ↑
RoCL (HN)	98.79	29.63	64.21	**98.85** ↑	**70.05** ↑	**84.45** ↑
RoCL (IP & HN)	98.63	52.38	75.51	**98.92** ↑	**65.86** ↑	**82.39** ↑

The experimental data in Table 6 show that the use of asymptotically enhanced strength of the adversarial samples in the model fine-tuning phase of adversarial contrastive learning can effectively improve robust accuracy. For example, under the RoCL (HN) framework, RA increased from 48.37% to 50.11%, while AA also increased from 65.135% to 65.545%. This means that by gradually increasing the strength of the adversarial sample during the fine-tuning phase, the model can further increase its resistance to attacks while maintaining accuracy against clean samples. In addition, in the RoCL (IP) framework, RA increased from 45.99% to 47.18% and AA increased from 63.65% to 65.04%, further confirming the positive impact of PACL on model robustness and overall performance. It should be noted that there is a general enhancement in the average accuracy after using PACL during the model fine-tuning phase, which indicates that the improvement is not merely enhancing the model robustness by sacrificing the model's fit to the clean samples.

5.6 Efficiency

In addition to the improvement in model robustness, the improvement also significantly improves the training efficiency of adversarial contrastive learning. Theoretically, disregarding the difference between the iteration time of the adversarial samples and the normal time of updating the gradient, as well as the need to round the number of iterations, for the first two phases of the three phases with the highest percentage of time consumption, the improvement has saved half of the additional reverse gradient computation due to the adversarial training, and the experimental data also confirm the efficiency improvement, as shown in the training time consumption in Figure 4. It is not difficult to find that while the model robustness has been significantly improved, the improvement proposed in this paper significantly improves the training elapsed time problem of adversarial contrastive learning. The training time is reduced by 33.33% for the backbone network and 42.11% for the classifier head. Overall, the total time until the completion of model fine-tuning is reduced by 33.90%. Further comparisons with Adversarial Contrastive Learning (AdvCL) are provided in the Appendix (Table 9).

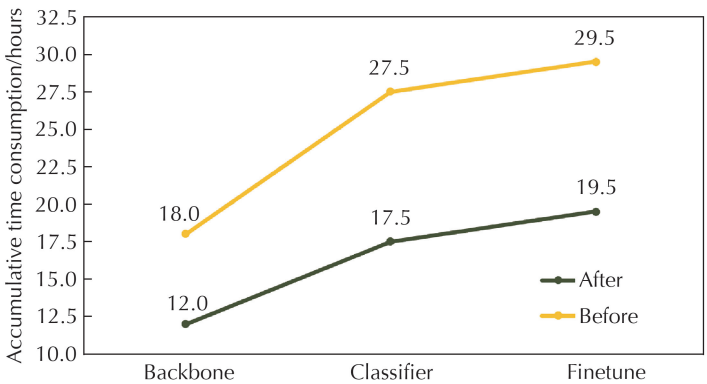


Figure 4. Accumulative time consumption of PACL in different stages compared with RoCL.

5.7 Effectiveness across different neural network architectures

In this paper, we not only validated the effectiveness of PACL on the ResNet framework but also further applied it to a newer neural network model, MobileNetV2, to assess the applicability and versatility of PACL across different model architectures. MobileNetV2, introduced in recent years as an efficient convolutional neural network model, employs depthwise separable convolution and inverted residual structures, significantly optimizing the model's computational efficiency and parameter utilization. By integrating PACL into the adversarial training of MobileNetV2, we found that this method could effectively enhance the model's robustness, especially under adversarial attack scenarios.

The experimental results in Table 8 indicate that MobileNetV2 with PACL demonstrated a significant improvement in robustness accuracy (RA) under various types of adversarial attacks compared to the baseline model without PACL. This outcome suggests that PACL not only excels on traditional large networks like ResNet but also possesses strong robustness enhancement capabilities in newer model architectures such as MobileNetV2. It is evident that PACL is well-suited for different types of neural networks, offering a more universal and effective solution for adversarial training.

Table 8. Model performance w/o and w/ PACL of adversarial contrastive learning on CIFAR10 using MobileNetV2.

Base method	w/o PACL			w/PACL		
	CA/%	RA/%	AA/%	CA/%	RA/%	AA/%
No defense	88.49	00.00	44.25	-	-	-
TRADES	79.42	39.19	59.31	-	-	-
RoCL	81.31	42.46	61.89	80.95	**45.34** ↑	**63.15** ↑
RoCL (IP)	81.72	42.23	61.98	81.24	**45.08** ↑	**63.16** ↑
RoCL (HN)	81.48	41.88	61.68	81.40	**45.61** ↑	**63.51** ↑
RoCL (IP & HN)	81.90	42.43	62.17	80.63	**45.26** ↑	**62.95** ↑

5.8 Limitations

In the previous section, Progressively Adversarial Contrastive Learning (PACL) has shown progressively better performance in adversarial training due to its progressively enhanced adversarial sample strength strategy, which both increases model robustness and maintains good classification of clean samples. However, the pros and cons of this approach may vary in different scenarios.

While PACL effectively improves robustness, its application in specific tasks requires a careful balance between robustness and standard classification accuracy For example, in the fine-tuning phase of Table 6, although PACL improved the robustness accuracy (RA), in some frameworks, such as RoCL (IP & HN), the classification accuracy (CA) of the model decreased. This suggests that in some scenarios, PACL may

be more inclined to improve the robustness of the model, however, this robustness improvement may be accompanied by a slight sacrifice in the ability to classify clean samples. Therefore, in scenarios where classification accuracy is extremely high, such as medical image classification or financial fraud detection, PACL may face a trade-off between increased robustness and decreased classification accuracy.

6. CONCLUSION

PACL is a promising defense mechanism for neural networks against adversarial attacks. Our experimental results demonstrate its performance and efficiency.

For future work, it is necessary to optimize the running time of the model. Existing security-sensitive applications and real-time sensitive applications intersect highly. Although PACL has achieved a significant efficiency improvement in the model training stage, it does not focus on the running time of the model after deployment. It is necessary to optimize the underlying aspects of the model, such as parameter accuracy and input accuracy, to reduce damage to real-time performance while introducing defense mechanisms.

Striking the right balance between robustness and classification accuracy is also crucial, especially in high-stakes applications like medical imaging or financial fraud detection. Future studies could develop adaptive strategies that dynamically balance these two aspects based on the specific needs of the application.

Additionally, as adversarial attack strategies continue to evolve, one significant challenge will be the potential adaptation of attacks to counter our progressive adversarial strength defense, which gradually increases the intensity of attacks. Future research will need to address how these adaptive attacks could bypass or undermine the incremental defense mechanisms, ensuring that our approach remains robust against increasingly sophisticated adversarial techniques.

In conclusion, PACL offers significant improvements in defending neural networks against adversarial attacks. By focusing on optimizing runtime, refining the balance between robustness and accuracy, and staying ahead of adversarial attack evolution, PACL can be further developed into an even more potent and widely applicable defense mechanism. This will help ensure the security and reliability of neural networks in the face of ever-changing adversarial challenges.

DATA AVAILABILITY STATEMENT

The CIFAR10 dataset used in this study is publicly available under the original license provided by the dataset authors and can be accessed at https://www.cs.toronto.edu/~kriz/cifar.html. The MNIST dataset is publicly available and can be obtained through widely used machine-learning libraries such as PyTorch or TensorFlow. Researchers using these datasets should follow the terms of use specified by the original dataset providers

AUTHOR CONTRIBUTIONS

This research was conducted by the first author, Xiang Gao (gaoxiang9156@163.com). The entire process, from conceptualization to data analysis and manuscript preparation, was carried out by Xiang Gao with the assistance of Weiqing Chen, Ying Cui, Xiang Dai, and Lican Dai. The detailed contributions are: Xiang Gao, Conceptualization; Xiang Gao, Methodology; Xiang Gao, Writing-Original Draft; Weiqing Chen, Data Curation; Weiqing Chen, Formal Analysis; Weiqing Chen, Implement; Ying Cui, Funding Acquisition; Xiang Dai, Investigation; Xiang Dai, Writing-Review; Lican Dai, Project Administration; Lican Dai, Visualization.

REFERENCES

[1] J. R. Rabunal and J. Dorado, *Artificial neural networks in real-life applications*. IGI Global, 2006.

[2] X. Wang and C. Wang, "A resource allocation method using federated deep reinforcement learning in vehicular networks," *Telecommunication Engineering*, 2024.

[3] Y. Wang, Z. Hu, X. Tu, L. Jianhua, J. Tao, X. Jiangjun, Y. Zihao, and D. Jinhua, "Facial feature point stability detection based on improved sbr algorithm," *Telecommunication Engineering*, 2023.

[4] B. Yu, C. Wenkai, X. Baorong, Z. Lei, Z. Lianyu, and M. Wentao, "Infrared ship detection by combining target extraction and deep learning," *Telecommunication Engineering*, 2023.

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

[6] I. J. Goodfellow, J. Shlens, and C. Szegedy, "Explaining and harnessing adversarial examples," *arXiv:1412.6572*, 2014.

[7] J. Wang, C. Wang, Q. Lin, C. Luo, C. Wu, and J. Li, "Adversarial attacks and defenses in deep learning for image recognition: A survey," *Neurocomputing*, 2022.

[8] K. Ren, T. Zheng, Z. Qin, and X. Liu, "Adversarial attacks and defenses in deep learning," *Engineering*, vol. 6, no. 3, pp. 346–360, 2020.

[9] M. Kim, J. Tack, and S. J. Hwang, "Adversarial self-supervised contrastive learning," *Advances in Neural Information Processing Systems*, vol. 33, pp. 2983–2994, 2020.

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

[11] H. Zhang, Y. Yu, J. Jiao, E. Xing, L. El Ghaoui, and M. Jordan, "Theoretically principled trade-off between robustness and accuracy," in *International conference on machine learning*. PMLR, 2019, pp. 7472–7482.

[12] Z. Wu, Y. Xiong, S. X. Yu, and D. Lin, "Unsupervised feature learning via nonparametric instance discrimination," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2018, pp. 3733–3742.

[13] M. Ye, X. Zhang, P. C. Yuen, and S.-F. Chang, "Unsupervised embedding learning via invariant and spreading instance feature," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2019, pp. 6210–6219.

[14] Y. Tian, D. Krishnan, and P. Isola, "Contrastive multiview coding," in *Computer Vision-ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XI16*. Springer, 2020, pp. 776–794.

[15] K. He, H. Fan, Y. Wu, S. Xie, and R. Girshick, "Momentum contrast for unsupervised visual representation learning," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2020, pp. 9729–9738.

[16] T. Chen, S. Kornblith, M. Norouzi, and G. Hinton, "A simple framework for contrastive learning of visual representations," in *International conference on machine learning*. PMLR, 2020, pp. 1597–1607.

[17] A. Shafahi, M. Najibi, M. A. Ghiasi, Z. Xu, J. Dickerson, C. Studer, L. S. Davis, G. Taylor, and T. Goldstein, "Adversarial training for free!" *Advances in Neural Information Processing Systems*, vol. 32, 2019.

[18] D. Zhang, T. Zhang, Y. Lu, Z. Zhu, and B. Dong, "You only propagate once: Accelerating adversarial training via maximal principle," *Advances in Neural Information Processing Systems*, vol. 32, 2019.

[19] N. Kumari, M. Singh, A. Sinha, H. Machiraju, B. Krishnamurthy, and V. N. Balasubramanian, "Harnessing the vulnerability of latent layers in adversarially trained models," in *Proceedings of the 28th International Joint Conference on Artificial Intelligence*, 2019, pp. 2779–2785.

[20] E. Wong, L. Rice, and J. Z. Kolter, "Fast is better than free: Revisiting adversarial training," in *International Conference on Learning Representations*, 2019.

[21] J. Chen, Y. Cheng, Z. Gan, Q. Gu, and J. Liu, "Efficient robust training via backward smoothing," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 36, 2022, pp. 6222–6230.

[22] Z. Huang, Y. Fan, C. Liu, W. Zhang, Y. Zhang, M. Salzmann, S. Süsstrunk, and J. Wang, "Fast adversarial training with adaptive step size," *arXiv:2206.02417*, 2022.

[23] Y. Balaji, T. Goldstein, and J. Hoffman, "Instance adaptive adversarial training: Improved accuracy tradeoffs in neural nets," *arXiv:1910.08051*, 2019.

[24] Q.-Z. Cai, M. Du, C. Liu, and D. Song, "Curriculum adversarial training," *arXiv:1805.04807*, 2018.

[25] J. Zhang, X. Xu, B. Han, G. Niu, L. Cui, M. Sugiyama, and M. Kankanhalli, "Attacks which do not kill training make adversarial learning stronger," in *International conference on machine learning*. PMLR, 2020, pp. 11278–11287.

[26] Y. Wang, D. Zou, J. Yi, J. Bailey, X. Ma, and Q. Gu, "Improving adversarial robustness requires revisiting misclassified examples," in *International Conference on Learning Representations*, 2020.

[27] A. Kurakin, I. J. Goodfellow, and S. Bengio, "Adversarial examples in the physical world," in *Artificial intelligence safety and security*. Chapman and Hall/CRC, 2018, pp. 99–112.

[28] N. Carlini and D. Wagner, "Towards evaluating the robustness of neural networks," in *2017 ieee symposium on security and privacy (sp)*, 2017, pp. 39–57.

[29] L. Li and M. Spratling, "Data augmentation alone can improve adversarial training," *arXiv:2301.09879*, 2023.

[30] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 770–778.

[31] A. Krizhevsky, G. Hinton *et al.*, "Learning multiple layers of features from tiny images," 2009.

[32] Q. Yu, J. Lou, X. Zhan, Q. Li, W. Zuo, Y. Liu, and J. Liu, "Adversarial contrastive learning via asymmetric infonce," in *Computer Vision-ECCV 2022: 17th European Conference, Tel Aviv, Israel, October 23–27, 2022, Proceedings, Part V*. Springer, 2022, pp. 53–69.

APPENDIX

To further evaluate the effectiveness of PACL, we conducted supplementary experiments comparing its performance with AdvCL on the CIFAR-10 dataset using the ResNet-18 architecture. The experimental results show that PACL not only reduced computational costs but also enhanced model robustness and performance on CIFAR-10.

Table 9. Model performance of AdvCL w/o PACL and w/PACL on CIFAR10 using ResNet18.

AdvCL Stage	w/o PACL			w/PACL		
	CA	RA	AA	CA	RA	AA
Stage 1	17.58	11.74	14.66	16.52	**11.80** ↑	**14.16**
Stage 2	15.27	12.73	14.00	**16.19** ↑	**13.25** ↑	**14.72** ↑
Stage 3	62.24	35.26	48.75	**68.51** ↑	**41.50** ↑	**55.01** ↑
Time consumption		25.13h			**17.32h** ↓	