3SAT: A Simple Self-Supervised Adversarial Training Framework

Jiang Fang [*1,2], **Haonan He** [*1,2], **Jiyan Sun** [1], **Jiadong Fu** [1,2], **Zhaorui Guo** [1,2], **Yinlong Liu** [†1,2], **Wei Ma** [1]

[1]Institute of Information Engineering, Chinese Academy of Sciences, Beijing, China
[2]School of Cyber Security, University of Chinese Academy of Sciences, Beijing, China
{fangjiang, hehaonan,sunjiyan, fujiadong, guozhaorui, liuyinlong, mawei}@iie.ac.cn

Abstract

The combination of self-supervised learning and adversarial training (AT) can significantly improve the adversarial robustness of self-supervised models. However, the robustness of self-supervised adversarial training (self-AT) still lags behind that of state-of-the-art (SOTA) supervised AT (sup-AT), even though the performance of current self-supervised learning models has already matched or even surpassed that of SOTA supervised learning models. This issue raises concerns about the secure application of self-supervised learning models. As a result of the incorporation of adversarial training, self-AT becomes a challenging joint optimisation problem. Furthermore, recent studies have shown that the data augmentation methods necessary for constructing positive pairs in self-supervised learning negatively impact the robustness improvement in self-AT. Inspired by this, we propose 3SAT, a simple self-supervised adversarial training framework. 3SAT conducts adversarial training on original, unaugmented samples, reducing the difficulty of optimizing the adversarial training subproblem and fundamentally eliminating the negative impact of data augmentation on robustness improvement. Additionally, 3SAT introduces a dynamic training objective scheduling strategy to address the issue of model training collapse during the joint optimization process when using original samples directly. 3SAT is not only structurally simple and computationally efficient, reducing self-AT training time by half, but it also improves the SOTA self-AT robustness accuracy by 16.19% and standard accuracy by 11.41% under Auto-Attack on the CIFAR-10 dataset. Even more impressively, 3SAT surpasses the SOTA sup-AT method in robust accuracy by a significant margin of 11.25%. This marks the first time that self-AT has outperformed SOTA sup-AT in robustness, indicating that self-AT is a superior method for improving model robustness.

Introduction

Enhancing the robustness of Deep Neural Networks (DNNs) is a critical challenge for their practical deployment. It is well-known that DNNs are vulnerable to small and imperceptible perturbations, which can lead to incorrect predictions with high confidence (Goodfellow, Shlens, and Szegedy 2015; Madry et al. 2017). This vulnerability is particularly concerning in high-risk domains such as autonomous driving (Chen et al. 2021; Sautier et al. 2022) and medical diagnosis (Chaitanya et al. 2020; Peng et al. 2021), where a lack of robustness in neural network models can result in severe threats to life and significant property damage.

Numerous prior studies suggest that Adversarial Training (AT) (Madry et al. 2017) is a commonly employed method to improve the robustness of DNNs. It involves a game between attackers and defenders. The attacker uses carefully crafted samples with adversarial perturbations (known as adversarial examples) to maximize the model training loss. The defender then minimizes the model training loss to improve the model's robustness to adversarial perturbations. Previous studies on adversarial training have mainly focused on supervised learning (Zhang et al. 2019; Wang et al. 2021; Pang et al. 2020). However, despite the fact that self-supervised learning (SSL) has achieved performance comparable to or even surpassing state-of-the-art supervised learning in various tasks (Devlin et al. 2018; Radford et al. 2019; Grill et al. 2020; He et al. 2021) , research on improving the robustness of self-supervised models remains relatively limited.

In SSL , Contrastive Learning (CL) is a commonly used visual self-supervised method. It trains an encoder to project input samples into an embedding space, where *positive sample* pairs (two different augmented versions of the same image) are closer to each other (Grill et al. 2020; Chen and He 2020), while maintaining a greater distance from other randomly selected images (optional) (Chen et al. 2020; He et al. 2020). Contrastive learning relies on positive sample pairs that are semantically consistent but visually diverse. Otherwise, it may lead to representation collapse (He et al. 2021). To achieve this, techniques like random cropping (Takahashi, Matsubara, and Uehara 2019), color distortion (Howard 2013), and Gaussian blur (Gedraite and Hadad 2011) are typically employed to create positive sample pairs.

Considering that encoders trained with contrastive learning can produce transferable visual representations, they are often used as base models for a variety of downstream tasks, including object detection and semantic segmentation . Therefore, having a base model that is both secure and robust, while possessing excellent representational capabilities, is crucial for the safe application of multiple downstream tasks.

Recently, to improve the robustness of encoders, ACL

[*]Equal Contribution
[†]Corresponding Author

(Jiang et al. 2020) integrates adversarial training (AT) with the self-supervised learning (SSL) framework, optimizing both the SSL and AT subproblems simultaneously during encoder pre-training. As shown in Equation 3, the AT objective of ACL is to minimize the contrastive loss between adversarial positive sample pairs corresponding to the positive pairs, making it more challenging to optimize than the SSL objective, which minimizes the contrastive loss between the positive pairs. Since the two subproblems of ACL are intertwined during optimization, solving this joint optimization problem is not straightforward. Building on ACL, DYNACL (Luo, Wang, and Wang 2023) highlights that excessively strong or weak data augmentation methods can impair model robustness. By dynamically adjusting the strength of data augmentation during self-supervised adversarial training (self-AT), DYNACL achieves robustness surpassing that of vanilla supervised adversarial training (sup-AT)(Pang et al. 2020) for the first time.

However, to the best of our knowledge: 1) As shown in Figure.1, the robustness of current self-AT methods (robust accuracy of 50.60% on CIFAR-10) (Xu et al. 2024) still lags behind the robustness of state-of-the-art supervised adversarial training (sup-AT) methods (robust accuracy of 55.54%) (Sehwag et al. 2021). This phenomenon makes one wonder about the following question:

Should we still rely on supervised adversarial training for security-sensitive applications, even though self-supervised learning models offer better representation transferability?

Additionally, 2) The standard accuracy of self-AT is less than 83%, while the standard accuracy of contrastive learning models without integrated adversarial training generally exceeds 90%. This phenomenon indicates that although AT improves the robustness of the encoder during the joint optimization process of self-AT, it severely compromises the model's representational performance.

The aforementioned two challenges severely undermine the practical use of self-AT models. To address these challenges, we propose 3SAT, a ineSimple ineSelf-ineSupervised ineAdversarial ineTrainin framework. To eliminate the negative impact of data augmentation on model robustness and to simplify the complexity of the joint optimization in self-AT, 3SAT opts to use the original, unaugmented samples and their corresponding adversarial samples as positive pairs for AT. Since the differences between the original samples and their adversarial counterparts with minor perturbations are minimal, the AT subproblem in 3SAT is easier to solve. However, the joint optimization of SSL and AT subproblems can lead to the encoder falling into a trivial solution early in the optimization process, due to the use of positive sample pairs with little variance in AT, resulting in representation collapse. To address this issue, we have innovatively proposed a Dynamic Training Objective Scheduling (DTOS) strategy. DTOS can be considered a decoupling strategy for self-AT joint optimization. In short, in the training process of 3SAT, we first let the self-supervised training part of 3SAT go through a warm-up phase. This phase allows the encoder to acquire a good representation performance through self-supervised learning. Subsequently, we gradually increase the strength of adversarial training to enable the encoder to progressively gain excellent robustness.

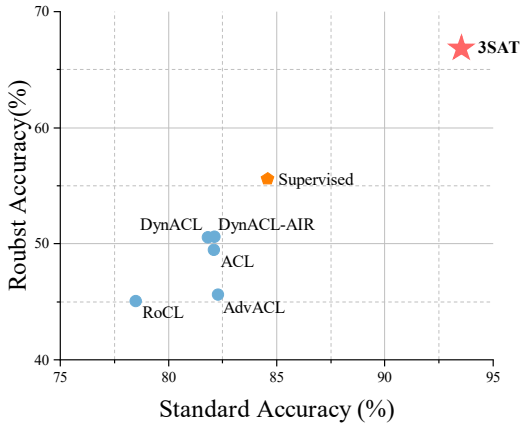


Figure 1: Comparison of standard and robust accuracy between 3SAT and different self-AT methods under AutoAttack. The blue icons represent the robust accuracy and standard accuracy of the leading self-AT methods on the RobustSSL Benchmark. The orange icon represents the performance of the leading sup-AT method (Sehwag et al. 2021) using the ResNet-18 network on RobustBench. 3SAT outperforms sup-AT, self-AT both in terms of standard and robust accuracy.

As a preview of the results, as shown in Figure 1, 3SAT not only significantly surpasses sota self-AT methods in terms of robustness but also exceeds sota sup-AT methods by a substantial margin. Moreover, compared to self-supervised methods, 3SAT incurs almost no loss in encoder representational performance. Even more impressive is that 3SAT not only delivers superior performance but is also computationally efficient. By reducing the computation of adversarial samples from twice to once per adversarial training iteration, as compared to previous self-AT methods, the training time for 3SAT is only half that of SOTA methods (Luo, Wang, and Wang 2023; Jiang et al. 2020) on similarly configured hardware. In summary, our contributions are as follows:

- We innovatively designed the 3SAT framework, a self-AT model that utilizes original samples for adversarial training. 3SAT significantly enhances model robustness while maintaining the superior representational capacity of self-AT models. Additionally, 3SAT reduces the training time required by previous SOTA self-AT training methods by half.
- We proposed a dynamic training objective scheduling strategy to address the training collapse issue caused by directly using original samples for adversarial training.
- Experiments demonstrate that 3SAT surpasses the known SOTA self-AT methods across all evaluation metrics on various datasets. Notably, on CIFAR-10, 3SAT improves the robust accuracy of the sota self-AT method by 16.19% and the standard accuracy by 11.41%. Even more

remarkably, 3SAT's robust accuracy exceeds that of the sota sup-AT method by a significant margin of 11.25%. This marks the first time self-AT has outperformed sup-AT in terms of robustness.

Background and Related Work

We begin this section with a set of notations. We define a labelled dataset $\mathcal{D}_l = \{(x_l, y_l)|x_l \in \mathbb{R}^n, y_l \in [K]\}$, where y_l is the label of sample x_l and K is the number of categories of the samples. Define an unlabelled dataset $\mathcal{D} = \{x|x \in \mathbb{R}^n\}$, where x is an unlabelled sample in $\mathcal{D}$.

Contrastive Learning

In encoder pre-training, we sample N samples from $\mathcal{D}$ to form a training mini-batch $\mathcal{X}_{\{N\}}$, and we perform two stochastic data augmentation transformations for each $x \in \mathcal{X}_{\{N\}}$, The sample x after two data augmentations is denoted as $(\tilde{x}, \tilde{x}^+)$, $\tilde{x} \in \tilde{\mathcal{X}}_{\{2N\}}$, and we call $(\tilde{x}, \tilde{x}^+)$ a positive sample pair because it contains the same semantic information. We trained the encoder g and predictor h to predict the output of the momentum encoder g_m. The momentum encoder g_m shares the same network architecture as encoder g, but its parameters are the Exponential Moving Average (EMA) of encoder g's parameters. More precisely, given the encoder parameters θ and a decay rate η, the parameters ξ of the momentum encoder are updated as $\xi = \eta\xi + (1-\eta)\theta$. Euclidean distance is used to measure the difference between the predicted features of the encoder output and the features outputed by the momentum encoder in the feature embedding space

$$\begin{aligned}
&\min_{g,h} \mathcal{L}(g, g_m, h) = \min_{g,h} \mathbb{E}_{x\in\mathcal{D}} \ell_{\mathrm{MSE}}(\tilde{x}, \tilde{x}^+; g, g_m, h), \\
&\ell_{\mathrm{MSE}}(\tilde{x}, \tilde{x}^+; g, g_m, h) = \left\| \frac{g_m(\tilde{x})}{\|g_m(\tilde{x})\|_2} - \frac{h(g(\tilde{x}^+))}{\|h(g(\tilde{x}^+))\|_2} \right\|_2^2,
\end{aligned} \tag{1}$$

where ℓ_{MSE} is Mean Squared Error (MSE) loss. In general, this type of SSL framework, which uses only positive pairs, can more effectively obtain representations with superior generalization performance.

Adversarial Training

sup-AT Supervised adversarial training (sup-AT) (Madry et al. 2017) solves the following max-min optimisation problem:

$$\begin{aligned}
&\delta_{x_l} := \arg\max_{\delta;\|\delta\|\le\epsilon} \ell_{\mathrm{CE}}[f(x_l + \delta), y_l] \\
&\mathcal{L}_{\mathrm{sup-AT}}(f) = \mathbb{E}_{x_l,y} \min_f \ell_{\mathrm{CE}}(f(x_l + \delta_{x_l}), y_l).
\end{aligned} \tag{2}$$

Here, the classifier $f : \mathbb{R}^n \to \mathbb{R}^K$is learned on the adversarial sample $x_l + \delta_{x_l}$, δ_{x_l} is the adversarial perturbation of sample x_l and ϵ denotes the allowed range of adversarial perturbations. ℓ_{CE} is the Cross-Entropy loss. In a supervised setting, utilizing sample labels, we can easily calculate adversarial perturbations using techniques like Fast Gradient Sign Method (FGSM) (Goodfellow, Shlens, and Szegedy 2015) or Projected Gradient Descent (PGD) attack (Madry et al. 2017). Notably, To achieve effective adversarial perturbations, multiple forward and backward iterations are required to compute the sample gradients. Therefore, the computation of adversarial samples is a time-consuming step in adversarial training.

Self-AT Adversarial training is also applicable for enhancing the robustness of self-supervised learning models. self-AT combines self-supervised learning with adversarial training, optimizing both training objectives in a single optimization process. For instance, the training objective of ACL (Jiang et al. 2020) is as follows

$$\begin{aligned}
\mathcal{L}_{\mathrm{self-AT}}(g) = &\min_g \mathbb{E}_{x\in\mathcal{D}}[\ell^{\mathrm{SSL}}(\tilde{x}, \tilde{x}^+; g) \\
&+ \ell^{\mathrm{AT}}(\tilde{x} + \delta, \tilde{x}^+ + \delta^+; g)],
\end{aligned} \tag{3}$$

where ℓ^{SSL} is the training objective of SSL, and ℓ^{AT} is the training objective for Adversarial Training. To achieve better representation performance and robustness, ACL performs adversarial training on positive sample pairs with added adversarial perturbations $(\tilde{x} + \delta, \tilde{x}^+ + \delta^+)$. Consequently, in each adversarial training iteration, ACL requires the computation of two separate adversarial perturbations

$$\begin{aligned}
&\delta := \arg\max_{\delta;\|\delta\|\le\epsilon} \ell(\tilde{x}, \tilde{x} + \delta; g), \\
&\delta^+ := \arg\max_{\delta^+;\|\delta^+\|\le\epsilon} \ell(\tilde{x}^+, \tilde{x}^+ + \delta^+; g).
\end{aligned} \tag{4}$$

ACL utilizes the *Dual Batch Normalization* (BN) technique (Xie et al. 2020). This involves allowing adversarial and clean samples to pass through different BN paths in the encoder network, due to the significant statistical differences between clean and adversarial samples. Building on ACL, DynACL (Luo, Wang, and Wang 2023) posits that both strong and weak data augmentation methods can compromise model robustness. By dynamically adjusting the strength of data augmentation in self-supervised adversarial training, DynACL (Luo, Wang, and Wang 2023) greatly improves the robust accuracy of self-AT. DynACL++ is an enhanced version of DynACL that utilizes pseudo-labels to bridge the gap between pre-training and downstream tasks. Xu et al. further improved the robustness of DynACL by adding Adversarial Invariant Regularization (AIR) to DynACL, called DynACL-AIR (Xu et al. 2024).

Method

Framework Design

The 3SAT structure, as shown in Figure 2, consists of two components: self-supervised representation learning and adversarial training. The left half of 3SAT represents the self-supervised learning part, with details provided in Contrastive Learning section. The right half illustrates the adversarial training part, where the training objective is to minimize the distance between the original samples and their corresponding adversarial samples in the embedding space. The formal expression is as follows

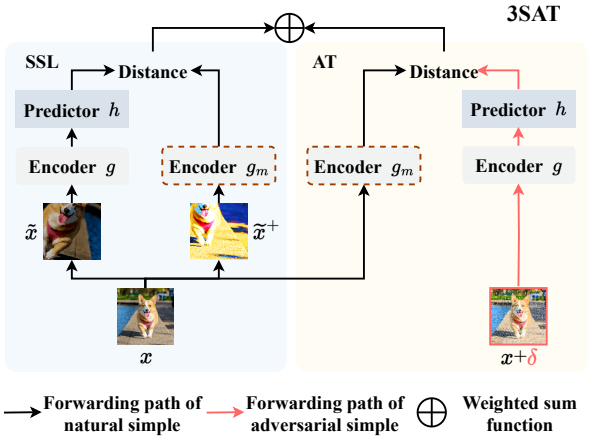


Figure 2: Framework Design. On the left of 3SAT is SSL part. On the right of 3SAT is AT part. 3SAT uses original samples for adversarial training without data augmented samples.

$$\min_{g,h} \mathcal{L}_{\text{3SAT}}(g, g_m, h) = \min_{g,h} \mathbb{E}_{x\in\mathcal{D}}[\ell^{\text{SSL}}_{\text{MSE}}(\tilde{x}^+, \tilde{x}; g, g_m, h) + s(t) \cdot \ell^{\text{AT}}_{\text{MSE}}(x, x+\delta; g, g_m, h)], \tag{5}$$

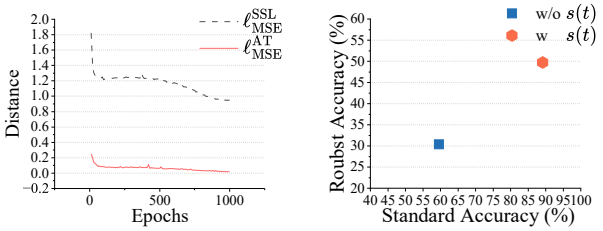
$$\delta := \arg \max_{\delta;\|\delta\|\leq\epsilon} \ell_{\text{MSE}}(x, x+\delta; g, g_m, h), \tag{6}$$

where $s(t)$ is an adversarial training strength dynamic scheduling function, t represents the current training epoch.

On the design of the network structure of the 3SAT encoder, we have abandoned the *Dual Batch Normalization* (BN) architectural design technique that previous work consistently employed, leading to a reduction in the model's parameters. Compared to ACL (Equation 3) (Jiang et al. 2020), 3SAT requires the generation of only one adversarial sample, effectively halving the time needed to produce adversarial samples for adversarial training. Moreover, 3SAT directly employs original samples for adversarial training, fundamentally eliminating the adverse effects that data augmentation in self-supervised learning can have on model robustness. Overall, the 3SAT framework is simple and computationally efficient, establishing it as a simple self-supervised adversarial training framework.

Dynamic Training Objective Scheduling

Considering that aggressive data augmentation methods in self-supervised learning are rarely adopted in downstream tasks, and augmented samples may not be suitable for defending against adversarial attacks applied to original samples in adversarial environments, as well as considering the impact of data augmentation on the robustness of self-AT models and simplifying the optimization difficulty of AT subproblems, We employed direct adversarial training with original samples. Unfortunately, As shown by the blue square dot in Figure.3b, this choice leads to the collapse of training.



(a) Loss Comparison (b) Effects of $s(t)$

Figure 3: (a) When training directly with original samples, there is a significant magnitude difference between $\ell^{\text{SSL}}_{\text{MSE}}$ and $\ell^{\text{AT}}_{\text{MSE}}$. (b) When using $s(t)$ for dynamic scheduling of $\ell^{\text{AT}}_{\text{MSE}}$, the standard and robust accuracy of the model is significantly improved.

We examined the numerical difference between the initial and subsequent terms of the 3SAT training goal as outlined in Equation 5. This difference stems from the fact that the former term $\ell^{\text{SSL}}_{\text{MSE}}$ is responsible for calculating the mean squared error between feature embeddings of positive sample pairs $(\tilde{x}, \tilde{x}^+)$, whereas the latter term $\ell^{\text{AT}}_{\text{MSE}}$ measures the MSE between the embeddings of original samples and their adversarial counterparts $(x, x+\delta)$. From Figure.3a, it can be observed that there exists a significant disparity in magnitude between these two terms. Simply put, for the encoder, compared to the positive sample pair $(\tilde{x}, \tilde{x}^+)$ that constructed using aggressive augmentation methods, the pairs $(x, x+\delta)$ are too similar, because adversarial perturbations visually result in only very small changes to the original sample. This brings about two issues: firstly, the encoder is prone to fall into a trivial solution at the early stage of training leading to a training collapse; secondly, at the early stage of training, when the encoder's performance is still at a low level, the encoder that is trained has already ignored the existence of the adversarial perturbation, which is detrimental to enhancing the robustness of the encoder. To address those issues, we propose to introduce a dynamic scheduling function $s(t)$ to Equation.5 for adjusting the strength of adversarial training. The aim is to encourage the encoder to enhance its ability to represent sample features as much as possible in the early stages of training, followed by gradually increasing the weight of adversarial training to enhance the robustness of the encoder. 3SAT designs the dynamic scheduling function as follows:

$$s(t) = \max\left(0, \frac{t-W}{K-W}\right), \tag{7}$$

where K is the max epochs and W is the number of Warm-up epochs. Intuitively, during the warm-up phase, 3SAT performs only self-supervised learning. After this phase, the dynamic scheduling function of 3SAT gradually increases the strength of adversarial training in a linearly progressive manner with each epoch. The results in Figure.3b demonstrate that integrating a dynamic scheduling function markedly improves both the representational capabilities and robustness of the encoder and avoids training collapse.

Experiment

In this section, we first evaluate the representation performance and robustness of 3SAT on different datasets: CIFAR-10, CIFAR-100 (Krizhevsky 2009), and STL-10 (Coates, Ng, and Lee 2011). These datasets encompass a wide range of class counts and resolutions, both low and high, allowing for a compelling evaluation of the advantages of our method. We compare 3SAT with other baseline methods: RoCL (Kim, Tack, and Hwang 2020), ACL (Jiang et al. 2020), AdvCL (Fan et al. 2021), DynACL (Luo, Wang, and Wang 2023), DynACL-AIR (Xu et al. 2024) and TAROSS (Gowal and Huang 2021). Secondly, we will discuss which designs are necessary and which ones could lead to performance degradation for self-supervised adversarial training.

Pretraining Setings. In model pre-training, we employ the standard ResNet-18 network as the encoder following existing self-AT methods (Kim, Tack, and Hwang 2020; Kim et al. 2022; Luo, Wang, and Wang 2023; Xu et al. 2024). 3SAT[‡] is built upon the BYOL (Grill et al. 2020) training script implemented by solo-learn(da Costa et al. 2022), and we strictly adhere to the settings in solo-learn for all optimizer configurations, augmentations, and projection head structures. We chose 256 as the batch size and performed 1000 epochs of pre-training. On the CIFAR-10 with STL-10 dataset the warm-up parameter W is set to 0, and on the CIFAR-100 dataset the warm-up parameter W is set to 200. To generate adversarial perturbations for adversarial training, we used the ℓ_∞ PGD attack (Madry et al. 2017) and followed all hyperparameters used in DynACL (Luo, Wang, and Wang 2023). To speed up convergence, we only ran 5 steps of PGD in the pre-training stage.

Evaluation Setings. We evaluate the learned representations and their robustness using three evaluation methods: Standard Linear Fine-tuning (SLF), Adversarial Linear Fine-tuning (ALF), and Adversarial Full Fine-tuning (AFF). The first two methods freeze the learned encoder parameters and only adjust the linear classifier. The difference is that SLF uses only natural samples for adjustment, while ALF uses adversarial samples to tune the linear classifier. As for AFF, we adjust the weights of both the pre-trained encoder and the newly added classifier during the fine-tuning process. Under all evaluation methods, we only perform fine-tuning for 25 epochs.

Comparing 3SAT with SOTA

Robustness on Various Datasets In Table 1, we evaluate the robustness of 3SAT and baseline methods on the CIFAR-10, CIFAR-100 and STL-10 datasets. We observe that 3SAT achieve new SOTA robustness and representation performance on all datasets. Specifically, under SLF evaluation method, 3SAT improves the robustness of the previous SOTA Self-AT method by 4.56% ($45.17\% \rightarrow 49.73\%$) on CIFAR-10. On CIFAR-100, 3SAT improves SOTA robustness by 7.59% ($20.45\% \rightarrow 28.04\%$) and on STL-10 robustness is improved by 0.54% ($47.66\% \rightarrow 48.20\%$). 3SAT not only significantly improves the classification accuracy of adversarial examples, but also significantly improves the classification accuracy of natural samples.

Robustness under Different Evaluation methods We have further evaluated the performance of 3SAT under different evaluation methods. As shown in Figure 2, 3SAT demonstrates state-of-the-art robustness and accuracy across all evaluation methods, significantly outperforming previous SOTA methods. In the ALF setting, 3SAT improved the best AA accuracy record by 8.80% ($46.01\% \rightarrow 54.81\%$). At the same time, 3SAT improves the standard accuracy over the SOTA method by more than 11% in both SLF and ALF settings. What's even more amazing is that in the AFF setting, 3SAT achieves an AA accuracy of 66.79%, a remarkable improvement of 16.19% ($50.60\% \rightarrow 66.79\%$) over the previous best self-AT AA accuracy record. Furthermore, this result beats the best sup-AT method using the same Resnet-18 network architecture by an impressive 11.25% ($55.54\% \rightarrow 66.79\%$), suggesting that self-AT may be a superior approach to achieving model robustness. It can also be seen from Figure 1 that the SA accuracy of 3SAT in the AFF setting is over 93%, and is the only method that does not seriously compromise the self-supervised learning representation capability due to integrated adversarial training.

Training Speed We evaluated the total pre-training duration of 3SAT versus other competing self-AT methods on a single RTX3090 GPU. The results, as shown in Table 3, show that 3SAT took the shortest time among all the competing methods, with a training duration of 14.28 hours, which is directly reduced by more than half compared to the 29.4 hours of pre-training of DynACL, which was previously the shortest time-consuming method. These statistics show that 3SAT is both efficient and effective. Compared to other self-AT methods, our research contributes to the mitigation of the greenhouse effect and benefits the environment.

Time Complexity Analysis As shown in Figure 2, 3SAT requires four forward passes and one adversarial sample computation. Assuming the same backbone model is used, the time complexity of a single forward pass is $O(M)$. The computation of an adversarial sample involves K iterations, each requiring one forward and one backward computation. To simplify, we also denote the time complexity of a single backward computation as M. Thus, the overall time complexity of 3SAT is $O((4+2K)\cdot M)$. Similarly, one training step of DynACL (Luo, Wang, and Wang 2023) involves two adversarial sample computations and four forward computations, resulting in a time complexity of $O((4+4K)\cdot M)$. Comparing the two equations, as the number of iterations K increases, DynACL's time complexity is nearly double that of 3SAT. This analysis shows that by reducing one adversarial sample computation, 3SAT effectively halves the time complexity of DynACL, which aligns with our experimental results in Table 3.

Performance under Semi-supervised Settings Following ACL (Jiang et al. 2020) and DynACL (Luo, Wang, and Wang 2023), we evaluated our proposed 3SAT under a semi-supervised setting. The results are shown in Table 4. We

[‡]Our code is at https://github.com/MengNanFang/3SAT

Pretraining Method	CIFAR10		CIFAR100		STL10	
	AA(%)	SA(%)	AA(%)	SA(%)	AA(%)	SA(%)
RoCL	26.12	77.9	8.72	42.93	26.52	78.19
ACL	37.62	79.32	15.68	45.34	33.24	71.21
AdvCL	37.46	73.23	15.45	37.58	45.26	72.11
TRASSO	36.39	74.06	N/A	N/A	N/A	N/A
DynACL	45.04	77.41	19.25	45.73	46.59	69.67
DynACL-AIR	45.17	78.08	20.45	46.84	47.66	72.3
3SAT (ours)	**49.73**	**89.14**	**28.04**	**57.89**	**48.20**	**83.97**

Table 1: Comparison of supervised and self-supervised adversarial training methods on CIFAR-10, CIFAR-100, and STL-10. SA and AA stand for standard accuracy and robust accuracy under AutoAttack (Croce and Hein 2020).

Pretraining Method	SLF		ALF		AFF	
	AA(%)	SA(%)	AA(%)	SA(%)	AA(%)	SA(%)
sup-AT					55.54	84.59
RoCL	26.12	77.9	8.72	75.62	45.02	78.51
ACL	37.62	79.32	40.91	76.57	49.46	82.11
AdvCL	37.46	73.23	37.28	73.15	48.58	82.31
TAROSS	36.39	74.06	37.25	76.19	N/A	N/A
DynACL	45.04	77.41	45.72	72.87	50.54	81.84
DynACL_AIR	45.17	78.08	46.01	77.42	50.60	82.14
3SAT (ours)	**49.73**	**89.14**	**54.81**	**88.23**	**66.79**	**93.55**

Table 2: Performance comparison on CIFAR-10 with three evaluation methods. SLF, ALF and AFF Standard Linear Fine-tuning, Adversarial Linear Fine-tuning, and Adversarial Full Fine-tuning respectively.

Method	**3SAT**	DynACL	ACL	AdvACL
Duration (h)	**14.28**	29.4	32.7	105.0

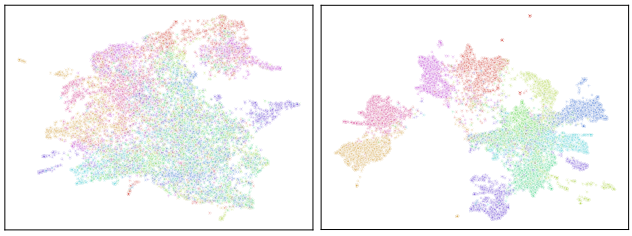
Table 3: Training Speed. 3SAT, due to its structurally simple and computationally efficient framework, as well as the reduction of one-time adversarial perturbation calculations, significantly reduces the time consumption of self-supervised adversarial training.

Pretraining Method	Label Ratio	AFF	
		AA (%)	SA (%)
ACL	1%	45.65	74.76
	10%	45.47	75.14
DynACL++	1%	46.95	76.77
	10%	48.56	78.34
3SAT (ours)	1%	**52.25** (+5.3)	**83.33** (+7.06)
	10%	**60.64** (+12.08)	**90.11** (+11.77)

Table 4: Performance under semi-supervised settings on CIFAR-10.

found that 3SAT demonstrates quite good performance, regardless of whether 1% or 10% of labels were available. Specifically, the robustness of 3SAT with only 1% of labels (52.25%) already surpasses the robustness of SOTA self-AT methods using all labels (50.60%). This result indicates that 3SAT is an exceptionally label-efficient approach.

Learned Representations Visualisation To further demonstrate the robustness of 3SAT, Figure 4 visualizes the representations learned by different self-AT methods on CIFAR-10 using UMAP (McInnes, Healy, and Melville 2018). The results show that the representations learned



(a) DynACL (b) 3SAT (ours)

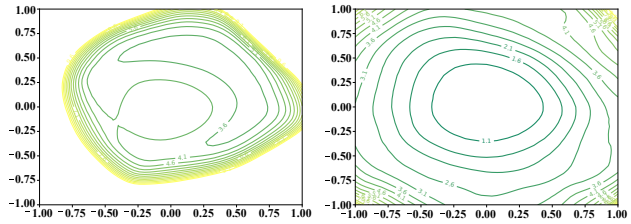
Figure 4: UMAP visualization of representations learned with different self-AT approaches. 3SAT gives a much clearer separation among classes than baseline approaches.

by 3SAT have clearer class boundaries compared to those learned by the baseline. This indicates that 3SAT makes it more difficult for attackers to successfully perturb the images, leading to more robust predictions.

Loss Landscape Previous researches (Prabhu et al. 2019; Li et al. 2018) has shown that a flatter loss landscape often leads to better robust generalization. Inspired by this finding, we visualized the loss landscapes of 3SAT and DYNACL++. As shown in Figures 5a and 5b, the loss landscape of 3SAT is flatter than that of DYNACL++, where the loss changes more rapidly. This also explains why 3SAT exhibits better robustness transferability compared to baseline methods.

Ablation Study for 3SAT

Effects of Warm-up In the Dynamic Training Objective Scheduling function for 3SAT, we designed a warm-up phase for the self-supervised learning part of 3SAT.



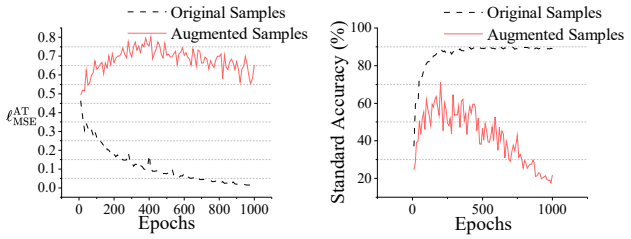
(a) DynACL++ (b) 3SAT (ours)

Figure 5: loss landscape visualization of DYNACL++ and 3SAT. Note that 3SAT enjoys a flatter loss landscape compared with DYNACL++.

Warm-up Epochs	0	200	400	600	800
SA (%)	51.06	57.89	57.87	59.08	60.68
AA (%)	24.96	28.04	29.01	28.86	27.14

Table 5: Standard and robust accuracy under Auto-Attack of 3SAT pre-trained models on the CIFAR-100 dataset with different warm-up epoches selected.

Our experiments demonstrate that for complex datasets like CIFAR-100, initially allowing the self-supervised learning part to learn data representations independently is crucial for improving the model's standard accuracy and robust accuracy. From Table 5, we observe that after 200 epochs of warm-up, 3SAT's standard accuracy on the CIFAR-100 dataset improved by 6.83% $(51.06\% \rightarrow 57.89\%)$, and its robust accuracy improved by 3.08% $(24.96\% \rightarrow 28.04\%)$. We also note that 3SAT is not highly sensitive to the warm-up, the standard and robust accuracy of 3SAT do not vary significantly when the warm-up exceeds 200 epochs. It should be emphasized that even without warm-up, 3SAT's performance surpasses that of SOTA methods.



(a) Adversarial Training Loss (b) Model Accuracy

Figure 6: (a) The use of augmented samples leads to high adversarial training losses. (b) Failure of adversarial training leads to failure of self-supervised training of the model. This problem does not occur when using the original samples.

Augmented or Original Samples? 3SAT selects the objective of adversarial training as narrowing the embedding distance between the original samples and their corresponding adversarial samples $(x, x+\delta)$, rather than between the augmented samples and their corresponding adversarial samples $(\hat{x}+\delta, \hat{x}^+)$. This choice is based on the following reasons: 1) Augmented samples may not be suitable for defending against adversarial attacks on the original samples; 2) Preliminary experiments, as shown in Figure 6, showed that narrowing the distance between as shown in Figure 6 leads to excessively high loss, causing self-supervised adversarial training to fail. In Self-AT, because SSL employs aggressive augmentation methods, narrowing the embedding distance between $(\hat{x}+\delta, \hat{x}^+)$ is already a challenging task, and attempting to further narrow the distance between $(\hat{x}+\delta, \hat{x}^+)$ exacerbates the difficulty of solving this joint optimization problem. ACL (Jiang et al. 2020) employs a dual-BN network structure to separate and simplify the joint optimization problem, but this design inevitably increases model redundancy, reducing training efficiency. The approach of 3SAT, which directly performs adversarial training on the original samples, can be seen as another way of decoupling the two joint optimization problems.

BN Path	SA(%)	AA(%)
Nartual Path	85.68	15.61
Adversarial Path	72.37	38.42

Table 6: Comparison of standard fine-tuning (SLF) performance of 3SAT for different BN paths.

Disadvantages of the Dual BN Paths Studies (Xie et al. 2020) and (Jiang et al. 2020) show that adversarial examples have distinct statistical characteristics compared to natural samples, and mixing them in Batch Normalization (BN) layers can hurt both robustness and accuracy. To address this, dual BN paths have been proposed to separate the statistics of adversarial and natural samples. However, this approach has two drawbacks: 1) Modern architectures like transformers and ViT do not use BN layers, limiting dual BN to models like ResNet. 2) The dual BN design weakens the model's representational power, offering only marginal robustness improvements. As shown in Table 6, the natural sample path has high representation but low robustness, while the adversarial sample path has poor representation with limited robustness gain. The trade-off between representational performance and robustness presents an irreconcilable issue for models utilizing dual BN path. Study (Zhang et al. 2022) argues that the statistical differences between natural and adversarial samples are minimal, rendering dual BN unnecessary. By discarding dual BN, 3SAT significantly improves both representational performance and robustness.

Conclusion

In this paper, we propose 3SAT. In contrast to previous methods, 3SAT performs adversarial training directly on original samples, thus avoiding the negative impact of aggressive data augmentation techniques used in self-supervised learning on improving model robustness through adversarial training. 3SAT has improved the robust accuracy of the state-of-the-art self-AT method by 16.19% and standard accuracy by 11.41%. Additionally, 3SAT's robust accuracy exceeds that of the self-AT method by a significant margin of 11.25%. This is also the first time that the self-AT model robustness exceeds the SOTA sup-AT model robustness.

Acknowledgements

This work was supported in part by the Climbing Program of Institute of Information Engineering, Chinese Academy of Sciences under Grant E3Z0031.

References

Chaitanya, K.; Erdil, E.; Karani, N.; and Konukoglu, E. 2020. Contrastive learning of global and local features for medical image segmentation with limited annotations. *Advances in neural information processing systems*, 33: 12546–12558.

Chen, K.; Hong, L.; Xu, H.; Li, Z.; and Yeung, D.-Y. 2021. Multisiam: Self-supervised multi-instance siamese representation learning for autonomous driving. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 7546–7554.

Chen, T.; Kornblith, S.; Norouzi, M.; and Hinton, G. 2020. A Simple Framework for Contrastive Learning of Visual Representations. arxiv:2002.05709.

Chen, X.; and He, K. 2020. Exploring Simple Siamese Representation Learning. *arXiv:2011.10566 [cs]*.

Coates, A.; Ng, A.; and Lee, H. 2011. An analysis of single-layer networks in unsupervised feature learning. In *Proceedings of the fourteenth international conference on artificial intelligence and statistics*, 215–223. JMLR Workshop and Conference Proceedings.

Croce, F.; and Hein, M. 2020. Reliable Evaluation of Adversarial Robustness with an Ensemble of Diverse Parameter-Free Attacks. arxiv:2003.01690.

da Costa, V. G. T.; Fini, E.; Nabi, M.; Sebe, N.; and Ricci, E. 2022. solo-learn: A Library of Self-supervised Methods for Visual Representation Learning. *Journal of Machine Learning Research*, 23(56): 1–6.

Devlin, J.; Chang, M.-W.; Lee, K.; and Toutanova, K. 2018. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*.

Fan, L.; Liu, S.; Chen, P.-Y.; Zhang, G.; and Gan, C. 2021. When Does Contrastive Learning Preserve Adversarial Robustness from Pretraining to Finetuning? arxiv:2111.01124.

Gedraite, E. S.; and Hadad, M. 2011. Investigation on the effect of a Gaussian Blur in image filtering and segmentation. In *Proceedings ELMAR-2011*, 393–396. IEEE.

Goodfellow, I. J.; Shlens, J.; and Szegedy, C. 2015. Explaining and Harnessing Adversarial Examples. arxiv:1412.6572.

Gowal, S.; and Huang, P.-S. 2021. SELF-SUPERVISED ADVERSARIAL ROBUSTNESS FOR THE LOW-LABEL, HIGH-DATA REGIME.

Grill, J.-B.; Strub, F.; Altché, F.; Tallec, C.; Richemond, P. H.; Buchatskaya, E.; Doersch, C.; Pires, B. A.; Guo, Z. D.; Azar, M. G.; Piot, B.; Kavukcuoglu, K.; Munos, R.; and Valko, M. 2020. Bootstrap Your Own Latent: A New Approach to Self-Supervised Learning. *arXiv:2006.07733 [cs, stat]*.

He, K.; Chen, X.; Xie, S.; Li, Y.; Dollár, P.; and Girshick, R. 2021. Masked Autoencoders Are Scalable Vision Learners. arxiv:2111.06377.

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

Howard, A. G. 2013. Some improvements on deep convolutional neural network based image classification. *arXiv preprint arXiv:1312.5402*.

Jiang, Z.; Chen, T.; Chen, T.; and Wang, Z. 2020. Robust Pre-Training by Adversarial Contrastive Learning. In *Advances in Neural Information Processing Systems*, volume 33, 16199–16210. Curran Associates, Inc.

Kim, M.; Ha, H.; Son, S.; and Hwang, S. J. 2022. Targeted Adversarial Self-Supervised Learning. arxiv:2210.10482.

Kim, M.; Tack, J.; and Hwang, S. J. 2020. Adversarial Self-Supervised Contrastive Learning.

Krizhevsky, A. 2009. Learning multiple layers of features from tiny images. Technical report.

Li, H.; Xu, Z.; Taylor, G.; Studer, C.; and Goldstein, T. 2018. Visualizing the loss landscape of neural nets. *Advances in neural information processing systems*, 31.

Luo, R.; Wang, Y.; and Wang, Y. 2023. Rethinking the Effect of Data Augmentation in Adversarial Contrastive Learning. https://arxiv.org/abs/2303.01289v2.

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

McInnes, L.; Healy, J.; and Melville, J. 2018. Umap: Uniform manifold approximation and projection for dimension reduction. *arXiv preprint arXiv:1802.03426*.

Pang, T.; Yang, X.; Dong, Y.; Su, H.; and Zhu, J. 2020. Bag of tricks for adversarial training. *arXiv preprint arXiv:2010.00467*.

Peng, J.; Wang, P.; Desrosiers, C.; and Pedersoli, M. 2021. Self-paced contrastive learning for semi-supervised medical image segmentation with meta-labels. *Advances in Neural Information Processing Systems*, 34: 16686–16699.

Prabhu, V. U.; Yap, D. A.; Xu, J.; and Whaley, J. 2019. Understanding adversarial robustness through loss landscape geometries. *arXiv preprint arXiv:1907.09061*.

Radford, A.; Wu, J.; Child, R.; Luan, D.; Amodei, D.; Sutskever, I.; et al. 2019. Language models are unsupervised multitask learners. *OpenAI blog*, 1(8): 9.

Sautier, C.; Puy, G.; Gidaris, S.; Boulch, A.; Bursuc, A.; and Marlet, R. 2022. Image-to-lidar self-supervised distillation for autonomous driving data. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 9891–9901.

Sehwag, V.; Mahloujifar, S.; Handina, T.; Dai, S.; Xiang, C.; Chiang, M.; and Mittal, P. 2021. Robust learning meets generative models: Can proxy distributions improve adversarial robustness? *arXiv preprint arXiv:2104.09425*.

Takahashi, R.; Matsubara, T.; and Uehara, K. 2019. Data augmentation using random image cropping and patching for deep CNNs. *IEEE Transactions on Circuits and Systems for Video Technology*, 30(9): 2917–2931.

Wang, Y.; Ma, X.; Bailey, J.; Yi, J.; Zhou, B.; and Gu, Q. 2021. On the convergence and robustness of adversarial training. *arXiv preprint arXiv:2112.08304*.

Xie, C.; Tan, M.; Gong, B.; Wang, J.; Yuille, A.; and Le, Q. V. 2020. Adversarial Examples Improve Image Recognition. arxiv:1911.09665.

Xu, X.; Zhang, J.; Liu, F.; Sugiyama, M.; and Kankanhalli, M. S. 2024. Enhancing adversarial contrastive learning via adversarial invariant regularization. *Advances in Neural Information Processing Systems*, 36.

Zhang, C.; Zhang, K.; Zhang, C.; Niu, A.; Yoo, C. D.; and Kweon, I. S. 2022. A Closer Look at Dual Batch Normalization and Two-domain Hypothesis In Adversarial Training With Hybrid Samples.

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