Adversarial Self-Supervised Contrastive Learning

Minseon Kim[1], **Jihoon Tack**[1], **Sung Ju Hwang**[1,2]
KAIST[1], AITRICS[2]
{minseonkim, jihoontack, sjhwang82}@kaist.ac.kr

Abstract

Existing adversarial learning approaches mostly use class labels to generate adversarial samples that lead to incorrect predictions, which are then used to augment the training of the model for improved robustness. While some recent works propose semi-supervised adversarial learning methods that utilize unlabeled data, they still require class labels. However, do we really need class labels *at all*, for adversarially robust training of deep neural networks? In this paper, we propose a novel adversarial attack for unlabeled data, which makes the model confuse the instance-level identities of the perturbed data samples. Further, we present a self-supervised contrastive learning framework to adversarially train a robust neural network without labeled data, which aims to maximize the similarity between a random augmentation of a data sample and its *instance-wise* adversarial perturbation. We validate our method, *Robust Contrastive Learning (RoCL)*, on multiple benchmark datasets, on which it obtains comparable robust accuracy over state-of-the-art supervised adversarial learning methods, and significantly improved robustness against the *black box* and *unseen* types of attacks. Moreover, with further joint fine-tuning with supervised adversarial loss, RoCL obtains even higher robust accuracy over using self-supervised learning alone. Notably, RoCL also demonstrate impressive results in robust transfer learning.

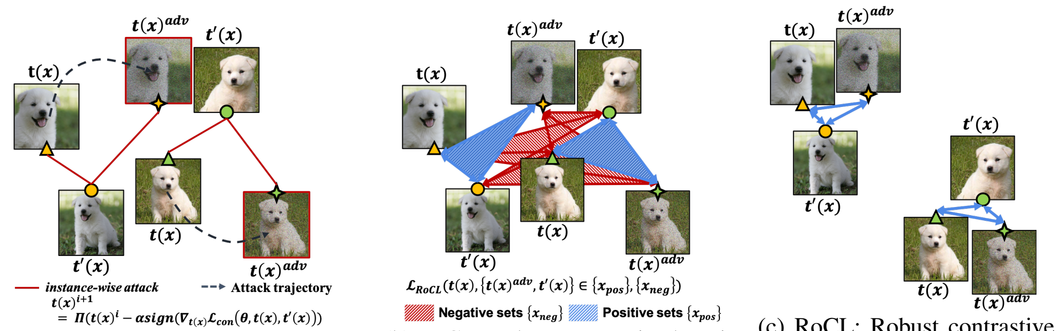
1 Introduction

The vulnerability of neural networks to imperceptibly small perturbations [1] has been a crucial challenge in deploying them to safety-critical applications, such as autonomous driving. Various studies have been proposed to ensure the robustness of the trained networks against adversarial attacks [2, 3, 4], random noise [5], and corruptions [6, 7]. Perhaps the most popular approach to achieve adversarial robustness is adversarial learning, which trains the model with samples perturbed to maximize the loss on the target model. Starting from Fast Gradient Sign Method [8] which apply a perturbation in the gradient direction, to Projected Gradient Descent [9] that maximizes the loss over iterations, and TRADES [2] that trades-off clean accuracy and adversarial robustness, adversarial learning has evolved substantially over the past few years. However, conventional methods with adversarial learning all require *class labels* to generate adversarial attacks.

Recently, self-supervised learning [10, 11, 12, 13, 14], which trains the model on unlabeled data in a supervised manner by utilizing self-generated labels from the data itself, has become popular as means of learning representations for deep neural networks. For example, prediction of the rotation angles [10], and solving randomly generated Jigsaw puzzles [11] are examples of such self-supervised learning methods. Recently, instance-level identity preservation [12, 13] with contrastive learning has shown to be very effective in learning the rich representations for classification. Contrastive self-supervised learning frameworks such as [12, 13, 14] basically aim to maximize the similarity of a sample to its augmentation, while minimizing its similarity to other instances.

In this work, we propose a contrastive self-supervised learning framework to train an adversarially robust neural network *without* any class labels. Our intuition is that we can fool the model by generat-

Preprint. Under review.



(a) Instance-wise adversarial examples (b) RoCL: Robust contrastive learning (c) RoCL: Robust contrastive learning

Figure 1: **Overview of our adversarial contrastive self-supervised learning.** (a) We generate instance-wise adversarial examples from an image transformed using a stochastic augmentation, which makes the model confuse the instance-level identity of the perturbed sample. (b) We then maximize the similarity between each transformed sample and their instance-wise adversaries using contrastive learning. (c) After training, each sample will have significantly reduced adversarial vulnerability in the latent representation space.

ing instance-wise adversarial examples (See Figure 1(a)). Specifically, we generate perturbations on augmentations of the samples to maximize their contrastive loss, such that the instance-level classifier becomes confused about the identities of the perturbed samples. Then, we maximize the similarity between clean samples and their adversarial counterparts using contrastive learning (Figure 1(b)), to obtain representations that suppress distortions caused by adversarial perturbations. This will result in learning representations that are robust against adversarial attacks (Figure 1(c)).

We refer to this novel adversarial self-supervised learning method as *Robust Contrastive Learning (RoCL)*. To the best of our knowledge, this is the first attempt to train robust neural networks *without any labels*, and to generate instance-wise adversarial examples. Recent works on semi-supervised adversarial learning [15, 16] or self-supervised adversarial learning [17] still require labeled instances to generate pseudo-labels on unlabeled instances or class-wise attacks for adversarial training, and thus cannot be considered as fully-unsupervised adversarial learning approaches.

To verify the efficacy of the proposed RoCL, we suggest a robust-linear evaluation for self-supervised adversarial learning and validate our method on benchmark datasets (CIFAR-10 and CIFAR-100) against supervised adversarial learning approaches. The results show that RoCL obtains comparable accuracy to strong supervised adversarial learning methods such as TRADES [2], although it does not use any labels during training. Further, when we extend the method to utilize class labels to fine-tune the network trained on RoCL with class-adversarial loss, we achieve even stronger robustness, *without* losing accuracy when clean samples. Moreover, we verify our rich robust representation with transfer learning which shows impressive performance. In sum, the contributions of this paper are as follows:

- We propose a novel **instance-wise** adversarial perturbation method which does not require any labels, by making the model confuse its instance-level identity.
- We propose a **adversarial self-supervised learning** method to explicitly suppress the vulnerability in the representation space by maximizing the similarity between clean examples and their instance-wise adversarial perturbations.
- Our method obtains **comparable** robustness to supervised adversarial learning approaches without using any class labels on the target attack type, while achieving **significantly better** clean accuracy and robustness on unseen type of attacks and transfer learning.

2 Related Work

Adversarial robustness Obtaining deep neural networks that are robust to adversarial attacks has been an active topic of research since Szegedy et al.[1] first showed their fragility to imperceptible distortions. Goodfellow et al.[8] proposed the fast gradient sign method (FGSM), which perturbs a target sample to its gradient direction, to increase its loss, and also use the generated samples to train the model for improved robustness. Follow-up works [9, 18, 19, 20] proposed iterative variants of the gradient attack with improved adversarial learning frameworks. After these gradient-based attacks have become standard in evaluating the robustness of deep neural networks, many more defenses followed, but Athalye et al. [21] showed that many of them appear robust only because they mask out the gradients, and proposed new types of attacks that circumvent gradient obfuscation. Recent works focus on the vulnerability of the latent representations, hypothesizing them as the main cause of the adversarial vulnerability of deep neural networks. TRADES [2] uses Kullback-Leibler divergence loss between a clean example and its adversarial counterpart to push the decision boundary, to obtain a more robust latent space. Ilyas et al. [22] showed the existence of imperceptible features that help with the prediction of clean examples but are vulnerable to adversarial attacks. On the other hand, instead of defending the adversarial attacks, guarantee the robustness become one of the solutions to the safe model. Li et al.[23], "randomized smoothing" technique has been empirically proposed as certified robustness. Then, Cohen et al. [24], prove the robustness guarantee of randomized smoothing in ℓ_2 norm adversarial attack. Moreover, to improve the performance of randomized smoothing [25] directly attack the smoothed classifier. A common requirement of existing adversarial learning techniques is the availability of class labels, since they are essential in generating adversarial attacks. Recently, semi-supervised adversarial learning [15, 16] approaches have proposed to use unlabeled data and achieved large enhancement in adversarial robustness. Yet, they still require a portion of labeled data, and does not change the class-wise nature of the attack. Contrarily, in this work, we propose instance-wise adversarial attacks that do not require *any* class labels.

Self-supervised learning As acquiring manual annotations on data could be costly, self-supervised learning, which generates supervised learning problems out of unlabeled data and solves for them, is gaining increasingly more popularity. The convention is to train the network to solve a manually-defined (pretext) task for representation learning, which will be later used for a specific supervised learning task (e.g., image classification). Predicting the relative location of the patches of images [26, 27, 11] has shown to be a successful pretext task, which opened the possibility of self-supervised learning. Gidaris et al. [10] propose to learn image features by training deep networks to recognize the 2D rotation angles, which largely outperforms previous self-supervised learning approaches. Corrupting the given images with gray-scaling [28] and random cropping [29], then restoring them to their original condition, has also shown to work well. Recently, leveraging the instance-level identity is becoming a popular paradigm for self-supervised learning due to its generality. Using the contrastive loss between two different transformed images from one identity [12, 13, 30] have shown to be highly effective in learning the rich representations, which achieve comparable performance to fully-supervised models. Moreover, even with the labels, the contrastive loss leverage the performance of the model than using the cross-entropy loss [31].

Self-supervised learning and adversarial robustness Recent works have shown that using unlabeled data could help the model to obtain more robust representations [15]. Moreover, [32] shows that a model trained with self-supervision improves the robustness. Even finetuning the pretrained self-supervised learning helps the robustness [17], and self-supervised adversarial training coupled with the K-Nearest Neighbour classification improves the robustness of KNN [33]. However, to the best of our knowledge, none of these previous works explicitly target for adversarial robustness on unlabeled training. Contrarily, we propose a novel instance-wise attack, which leads the model to predict an incorrect instance for an instance-discrimination problem. This allows the trained model to obtain robustness that is on par or even better than supervised adversarial learning methods.

3 Adversarial Self-Supervised Learning with Instance-Wise Attacks

We now describe how to obtain adversarial robustness in the representations *without* any class labels, using instance-wise attacks and adversarial self-supervised contrastive learning. Before describing ours, we first briefly describe supervised adversarial training and self-supervised contrastive learning.

Adversarial robustness We start with the definition of adversarial attacks under supervised settings. Let us denote the dataset $\mathbb{D} = \{X, Y\}$, where $x \in X$ is training sample and $y \in Y$ is a corresponding label, and a supervised learning model $f_\theta : X \to Y$ where θ is parameters of the model. Given such a dataset and a model, *adversarial attacks* aim towards finding the worst-case examples nearby by searching for the perturbation, which maximizes the loss within a certain radius from the sample (e.g., norm balls). We can define such adversarial attacks as follows:

$$x^{i+1} = \Pi_{B(x,\epsilon)}(x^i - \alpha \texttt{sign}(\nabla_{x^i} \mathcal{L}_{\texttt{CE}}(\theta, x^i, y)) \tag{1}$$

where $B(x, \epsilon)$ is the ℓ_∞ norm-ball around x with radius ϵ, and Π is the projection function for norm-ball. The α is the step size of the attacks and $\texttt{sign}(\cdot)$ returns the sign of the vector. Further, $\mathcal{L}_{\texttt{CE}}$ is the cross-entropy loss for supervised training, and i is the number of attack iterations. This formulation generalizes across different types of gradient attacks. For example, Projected Gradient Descent (PGD) [9] starts from a random point within the $x \pm \epsilon$ and perform i gradient steps, to obtain an attack x^{i+1}.

The simplest and most straightforward way to defend against such adversarial attacks is to minimize the loss of adversarial examples, which is often called *adversarial learning*. The adversarial learning framework proposed by Madry et al.[9] solve the following non-convex outer minimization problem and non-convex inner maximization problem where δ is the perturbation of the adversarial images, and $x + \delta$ is an adversarial example x^{adv}, as follow:

$$\underset{\theta}{\operatorname{argmin}}\, \mathbb{E}_{(x,y)\sim\mathbb{D}}[\max_{\delta \in B(x,\epsilon)} \mathcal{L}_{\texttt{CE}}(\theta, x+\delta, y)] \tag{2}$$

In standard adversarial learning framework, including PGD [9], TRADES [2], and many others, generating such adversarial attacks require to have a class label $y \in Y$. Thus, conventional adversarial attacks are inapplicable to unlabeled data.

Self-supervised contrastive learning The self-supervised contrastive learning framework [12, 13] aims to maximize the agreement between different augmentations of the same instance in the learned latent space while minimizing the agreement between different instances. Let us define some notions and briefly recap the SimCLR. To project the image into a latent space, SimCLR uses an encoder $f_\theta(\cdot)$ network followed by a projector, which is a two-layer multi-layer perceptron (MLP) $g_\pi(\cdot)$ that projects the features into latent vector z. SimCLR uses a stochastic data augmentation t, randomly selected from the family of augmentations $\mathcal{T}$, including random cropping, random color distortion, and random Gaussian blur. Applying any two transformations, $t,\ t' \sim \mathcal{T}$, will yield two samples denoted $t(x)$ and $t'(x)$, that are different in appearance but retains the instance-level identity of the sample. We define $t(x)$'s positive set as $\{x_{\texttt{pos}}\} = t'(x)$ from the same original sample x, while the negative set $\{x_{\texttt{neg}}\}$ as the set of pairs containing the other instances x'. Then, the contrastive loss function $\mathcal{L}_{\texttt{con}}$ can be defined as follows:

$$\mathcal{L}_{\texttt{con},\theta,\pi}(x, \{x_{\texttt{pos}}\}, \{x_{\texttt{neg}}\}) := -\log \frac{\sum_{\{\mathrm{z}_{\texttt{pos}}\}} \exp(\mathrm{sim}(\mathrm{z}, \{\mathrm{z}_{\texttt{pos}}\})/\tau)}{\sum_{\{\mathrm{z}_{\texttt{pos}}\}} \exp(\mathrm{sim}(\mathrm{z}, \{\mathrm{z}_{\texttt{pos}}\})/\tau) + \sum_{\{\mathrm{z}_{\texttt{neg}}\}} \exp(\mathrm{sim}(\mathrm{z}, \mathrm{z}_{\texttt{neg}})/\tau)}, \tag{3}$$

where z, $\{\mathrm{z}_{\texttt{pos}}\}$, and $\{\mathrm{z}_{\texttt{neg}}\}$ are corresponding 128-dimensional latent vectors (z) of x obtained by the encoder and projector $z = p(f_\theta(x))$, $\{x_{\texttt{pos}}\}$, and $\{x_{\texttt{neg}}\}$, respectively. The standard contrastive learning only contains a single sample in the positive set $\{\texttt{pos}\}$, which is $t(x)$. The $\mathrm{sim}(u, v) = u^T v/\|u\|\|v\|$ denote cosine similarity between two vectors and τ is a temperature parameter.

We show that standard contrastive learning, such as SimCLR, is vulnerable to the adversarial attacks as shown in Table 1. To achieve robustness with such self-supervised contrastive learning frameworks, we need a way to adversarially train them, which we will describe in the next subsection.

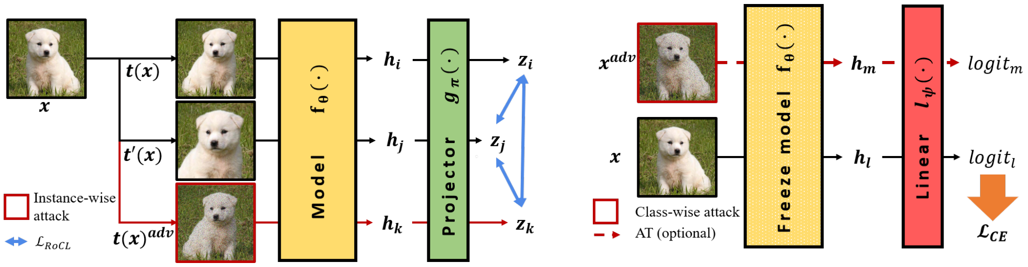
3.1 Adversarial Self-supervised Contrative Learning

We now introduce a simple yet novel and effective approach to adversarially train a self-supervised learning model, using unlabeled data, which we coin as *robust contrastive learning (RoCL)*. RoCL is trained without a class label by using *instance-wise* attacks, which makes the model confuse the instance-level identity of a given sample. Then, we use a contrastive learning framework to maximize the similarity between a transformed example and the instance-wise adversarial example of another transformed example. Algorithm 1 summarizes our robust contrastive learning framework in supplementary B.

Instance-wise adversarial attacks Since class-wise adversarial attacks for existing approaches are inapplicable to the unlabeled case we target, we propose a novel *instance-wise* attack. Specifically, given a sample of an input instance, we generate a perturbation to fool the model by confusing its instance-level *identity*; such that it mistakes it as an another sample. This is done by generating a perturbation that maximizes the self-supervised contrastive loss for discriminating between the instances, as follows:

$$t(x)^{i+1} = \Pi_{B(t(x),\epsilon)}(t(x)^i - \alpha\texttt{sign}(\nabla_{t(x)^i}\mathcal{L}_{\texttt{con},\theta,\pi}(t(x)^i, \{t'(x)\}, \{t(x)_{\texttt{neg}}\})) \tag{4}$$

where $t(x)$ and $t'(x)$ are transformed images with stochastic data augmentations $t, t' \sim \mathcal{T}$, and $t(x)_{\texttt{neg}}$ are the negative instances for $t(x)$, which are examples of other samples x'.



(a) Robust contrastive learning training

(b) Linear evaluation. (Adversarial training optional)

Figure 2: **Adversarial training and evaluation steps for RoCL.** During adversarial training, we maximize the similarity between two differently transformed examples $\{t(x), t'(x)\}$ and their adversarial perturbations $t(x)^{adv}$. After the model is fully trained to obtain robustness, then we evaluate the model on the target classification task by using linear model instead of projector. Here, we could either train the linear classifier only on clean examples, or adversarially train it with class-adversarial examples.

Robust Contrastive Learning (RoCL) We now present a framework to learn robust representation via self-supervised contrastive learning. The adversarial learning objective for an instance-wise attack, following the min-max formulation of [9] could be given as follows:

$$\underset{\theta,\pi}{\operatorname{argmin}}\, \mathbb{E}_{(x)\sim\mathbb{D}}[\max_{\delta\in B(t(x),\epsilon)} \mathcal{L}_{con,\theta,\pi}(t(x)+\delta, \{t'(x)\}, \{t(x)_{\texttt{neg}}\})] \tag{5}$$

where $t(x)+\delta$ is the adversarial image $t(x)^{adv}$ generated by *instance-wise* attacks (eq. 4). Note that we generate the adversarial example of x using a stochastically transformed image $t(x)$, rather than the original image x, which will allow us to generate diverse attack samples. This adversarial learning framework is essentially the same as the supervised adversarial learning framework, except that we train the model to be robust against m-way instance-wise adversarial attacks. Note that the proposed regularization can be interpreted as a denoiser. Since the contrastive objective maximize the similarity between clean samples: $t(x), t'(x)$, and generated adversarial example, $t(x)^{adv}$.

We generate label-free adversarial examples using instance-wise adversarial attacks in equation 4. Then we use the contrastive learning objective to maximize the similarity between clean examples and their instance-wise perturbation. This is done using a simple modification of the contrastive learning objective in Eq. 3, by using the *instance-wise* adversarial examples as additional elements in the positive set. Then we can formulate our *Robust Contrastive Learning* objective as follow:

$$\begin{aligned}&\mathcal{L}_{\texttt{RoCL},\theta,\pi}(t(x), \{t'(x), t(x)^{adv}\}, \{t(x)_{\texttt{neg}}\})\\ &:= -\log \frac{\sum_{\{\mathrm{z}_{\mathrm{pos}}\}} \exp(\mathrm{sim}(\mathrm{z}, \{\mathrm{z}_{\mathrm{pos}}\})/\tau)}{\sum_{\{\mathrm{z}_{\mathrm{pos}}\}} \exp(\mathrm{sim}(\mathrm{z}, \{\mathrm{z}_{\mathrm{pos}}\})/\tau) + \sum_{\{\mathrm{z}_{\mathrm{neg}}\}} \exp(\mathrm{sim}(\mathrm{z}, \{\mathrm{z}_{\mathrm{neg}}\})/\tau)},\end{aligned} \tag{6}$$

where $t(x)^{adv}$ are the adversarial perturbation of an augmented sample $t(x)$, and $t'(x)$ is another stochastic augmentation. The $\{z_{\texttt{pos}}\}$, which is the set of positive samples in the latent feature space, is compose of z' and z^{adv} which are latent vectors of $t'(x)$ and $t(x)^{adv}$ respectively. The $\{z_{\texttt{neg}}\}$ is the set of latent vectors for negative samples in $\{t(x)_{\texttt{neg}}\}$.

Linear Evaluation of RoCL With RoCL, we can adversarially train the model without any class labels (Figure 2a). Yet, since the model is trained for instance-wise classification, it cannot be directly used for class-level classification. Thus, existing self-supervised learning models leverage *linear evaluation* [28, 34, 35, 12], which learns a linear layer $l_\psi(\cdot)$ on top of the fixed $f_\theta(\cdot)$ embedding layer (Figure 2b) with clean examples. While RoCL achieves impressive robustness with this standard evaluation (Table 1), to properly evaluate the robustness against a specific type of attack, we propose a new evaluation protocol which we refer to as *robust-linear evaluation (r-LE)*. r-LE trains a *linear* classifier with class-level adversarial examples of specific attack (e.g. ℓ_∞) with the fixed encoder as follows:

$$\underset{\psi}{\operatorname{argmin}}\, \mathbb{E}_{(x,y)\sim\mathbb{D}}[\max_{\delta\in B(x,\epsilon)} \mathcal{L}_{\texttt{CE}}(\psi, x+\delta, y)] \tag{7}$$

where $\mathcal{L}_{\texttt{CE}}$ is the cross-entropy that only optimize parameters of linear model ψ. While we propose r-LE as an evaluation measure, it could be also used as an efficient means of obtaining an adversarially robust network using network pretrained using self-supervised learning.

Table 1: Experimental results with white box attacks on ResNet18 and ResNet50 trained on the CIFAR-10 . r-LE denotes robust linear evaluation, and SCL is the supervised contrastive learning [31] which uses the labels in the contrastive loss. Baselines with * are the models with our data augmentation applied during training. AT denotes the supervised adversarial training[9], and SS denotes the self-supervised loss. tInf is the test inference by the transformed smoothed classifier with 30 iterations. Rot+pretrained is the model [17] which finetunes the network trained with rotation-prediction self-supervised learning. For a fair comparison, we report the single self-supervised model pretrained version with the ResNet50-v2 model. $^+$ is the reported performance of [17]. All models are trained with ℓ_∞; thus the ℓ_∞ is the *seen* adversarial attack and ℓ_2, and ℓ_1 attacks are *unseen*.

Train type	Method	ResNet18							ResNet50						
			seen		*unseen*					*seen*		*unseen*			
			ℓ_∞		ℓ_2		ℓ_1			ℓ_∞		ℓ_2		ℓ_1	
		A_{nat}	ϵ 8/255	16/255	0.25	0.5	7.84	12	A_{nat}	ϵ 8/255	16/255	0.25	0.5	7.84	12
Supervised	$\mathcal{L}_{\text{CE}}$	92.82	0.00	0.00	20.77	12.96	28.47	15.56	93.12	0.00	0.00	13.42	3.44	28.78	13.98
	AT[9]	81.63	44.50	14.47	72.26	**59.26**	**66.74**	55.74	84.03	46.76	17.63	72.98	58.78	65.28	52.45
	TRADES[2]	77.03	**48.01**	**22.55**	**68.07**	57.93	62.93	53.79	82.10	**53.49**	**25.18**	**73.01**	**61.94**	**65.48**	**54.52**
	TRADES*[2]	73.26	42.71	17.71	65.25	56.13	62.89	**55.95**	75.65	46.20	20.96	67.02	57.12	62.46	55.09
	SCL[31]	**94.05**	0.08	0.00	22.17	10.29	38.87	22.58	**95.02**	0.00	0.00	16.72	1.68	39.44	22.59
Semi-Supervised	RST[15]	86.39	56.57	25.98	78.12	67.54	72.24	61.51	-	-	-	-	-	-	-
Self -supervised	SimCLR[12]	**91.25**	0.63	0.08	15.3	2.08	41.49	25.76	**92.69**	0.07	0.00	25.13	3.85	50.17	31.63
	RoCL	83.71	40.27	9.55	66.39	63.82	**79.21**	**76.17**	85.99	43.56	11.38	**70.87**	**67.59**	**82.65**	**80.02**
	RoCL+rLE	80.43	**47.69**	**15.53**	**68.30**	**66.19**	77.31	75.05	80.79	**45.33**	**16.85**	67.14	64.61	77.54	75.76
Self -supervised +finetune	Rot. Pretrained[17]	-	-	-	-	-	-	-	**85.66**$^+$	50.40$^+$	-	-	-	-	-
	RoCL+AT	80.26	40.77	**22.83**	68.64	56.25	65.16	56.07	82.72	50.60	18.83	72.12	**70.03**	**81.02**	**79.22**
	RoCL+TRADES	84.55	43.85	14.29	**73.01**	60.03	68.25	58.04	85.41	45.68	**21.21**	74.06	59.60	65.37	53.54
	RoCL+AT+SS	**91.34**	**49.66**	14.44	70.75	**61.55**	**83.08**	**81.18**	84.67	**52.44**	19.53	**76.61**	66.38	72.76	64.56
Transformation smoothed classifier	TRADES+**tInf**	77.71	52.73	**31.55**	70.25	62.13	66.04	58.43	81.94	57.23	**34.31**	74.46	65.65	68.62	59.66
	RoCL+tInf	**84.11**	52.58	18.55	**81.15**	**76.01**	**82.13**	**80.40**	**86.97**	53.39	18.56	**84.19**	**76.90**	**85.17**	**83.68**
	RoCL+rLE+tInf	81.26	**53.85**	21.22	78.69	73.83	79.74	78.02	80.14	**57.50**	26.28	77.09	73.86	78.20	77.31

Transformation smoothed inference We further propose a simple inference method for robust representation. Previous works [25, 24] proposed *smoothed classifiers*, which obtain smooth decision boundaries for the final classifier by taking an expectation over classifiers with Gaussian noise perturbed samples. This method addresses the problem with sharp classifiers obtained using supervised learning, which may result in misclassification of the points even with small perturbations. Inspired by this, we observe that our objective enforces to assemble all differently transformed images into the adjacent area, and propose a novel *transformation smoothed classifier* for RoCL, which predicts the class c by calculating expectation $\mathbb{E}$ over the transformation $t \sim \mathcal{T}$ for a given input x as follows:

$$S(x) = \underset{c \in Y}{\operatorname{argmax}} \, \mathbb{E}_{t \sim \mathcal{T}}(l(f(t(x))) = c) \tag{8}$$

4 Experimental Results

We verify the efficacy of our RoCL in various settings against both supervised adversarial learning methods, and self-supervised pretraining with adversarial finetuning. We report the results of white-box and black-box attacks in Table 1 and 2, respectively, under ResNet18, ResNet50 [36] trained on CIFAR-10 [37]. We evaluate multiple versions of our model on diverse scenarios : models trained with RoCL only, with self-supervised learning only (RoCL, RoCL+rLE), models that use RoCL for pretraining and perform further standard adversarial training with class-wise attacks (RoCL+AT, RoCL+TRADES, RoCL+AT+SS), and the RoCL with the transformation smoothed classifier. For all baselines and our method, we train with the same attack strength of $\epsilon = 16/255$. For results on CIFAR-100 and details of the evaluation setup, please **see the supplementary C**.

4.1 Main results

Self-supervised adversarial learning To our knowledge, our *RoCL* is the first attempt to achieve robustness in a fully self-supervised learning setting, since existing approaches used self-supervised learning as a pretraining step before supervised adversarial training. We first compare RoCL against SimCLR[12], which is a vanilla self-supervised contrastive learning model. The result shows that the vanilla model is extremely vulnerable to adversarial attacks. However, RoCL achieves high robustness against the target ℓ_∞ attacks, outperforming supervised adversarial training by Madry et al. [9], and obtaining comparable performance to TRADES [2]. This is an impressive result which demonstrates that it is possible to train adversarially robust models without any labeled data. Note that while we used the same number of instances in this experiment, in practice, we can use any number of unlabeled

Table 2: Performance of RoCL against black box attacks on the CIFAR-10. Each column denotes the black box model used to generate the ℓ_∞ adversarial images with ϵ = 8/255 and 16/255, respectively. The row is the target model that is trained with ℓ_∞.

	ResNet18			
Source	8/255		16/255	
Target	AT	TRADES	AT	TRADES
AT [9]	-	**77.48**	-	**63.87**
TRADES [2]	60.73	-	41.87	-
RoCL	**66.76**	77.33	**41.97**	62.98

Table 3: Experimental results of transfer learning on ResNet18 trained on the CIFAR-10 and CIFAR-100 dataset, respectively. We compare with the freezed transfered model in [38]. The model is modified WRN 32-10 [39]. $^+$ is the reported performance of [38].

source	target	Method	A_{nat}	ℓ_∞
CIFAR-100	CIFAR-10	Transfer$^+$[38]	72.05	17.70
		RoCL	**73.93**	**18.62**
CIFAR-10	CIFAR-100	Transfer$^+$[38]	41.59	11.63
		RoCL	**45.84**	**15.33**

Table 4: Attack target image ablation

ℓ_∞ train/ℓ_∞ test	A_{nat}	8/255	16/255
original x	**87.96**	36.6	**11.78**
$t'(x)$	83.71	**40.27**	9.55

Table 5: Attack iteration ablation

ℓ_∞ train/ℓ_∞ test	20	40	100
RoCL	40.27	39.80	39.74

Table 6: Ablation on the attack loss type

ℓ_∞ train/ℓ_∞ test			
		ϵ	
$\mathcal{L}_{\theta,\pi}$	A_{nat}	8/255	16/255
Contrastive	83.71	**40.27**	**9.55**
MSE	**88.35**	40.12	7.88
Cosine similarity	73.49	9.30	0.06
MD	84.40	21.05	1.65

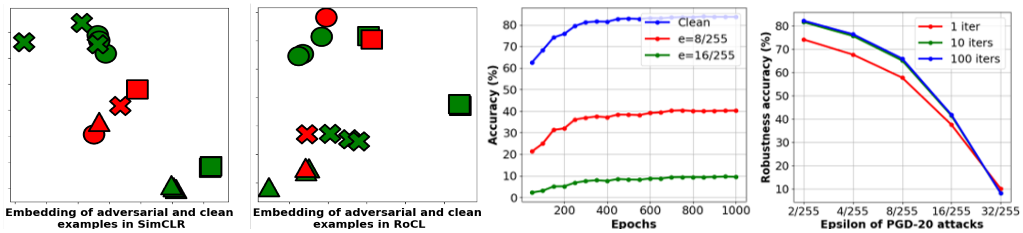
data available to train the model, which may lead to larger performance improvements. To show that this is not the effect of using augmented samples for self-supervised learning, we applied the same set of augmentations for TRADES (TRADES*), but it obtains worse performance over the original TRADES. Moreover, RoCL obtains much better clean accuracy, and significantly higher robustness over the supervised adversarial learning approaches against *unseen* types of attacks (See the results on ℓ_2, ℓ_1 attacks in Table 1), and black box attacks (See Table 2). This makes RoCL more appealing over baselines, and suggests that our method of using instance-wise attacks and suppression of distortion at the latent representation space is a more fundamental solution to ensure robustness against general types of attacks. This point is made more clear in the comparison of RoCL against RoCL with linear evaluation (RoCL+rLE), which trains the linear classifier with class-wiser adversaries. RoCL+rLE improves the robustness against the target ℓ_∞ attacks, but degenerates robustness on unseen types of attacks. Finally, with our proposed transformation smoothed classifier, RoCL obtains even stronger performance on unseen types of attacks (Table 1, the last three rows).

Self-supervised based adversarial fine-tuning Existing works have shown that [40, 17] pretraining the networks with supervised or self-supervised learning improve adversarial robustness. This is also confirmed with our results in Table 1, which show that the models fine-tuned with our method obtain even better robustness and higher clean accuracy over models trained from scratch. We observe that using self-supervised loss during adversarial finetuning further improves robustness. Moreover, our method outperforms on robustness compares to the single self-supervised adversarially pretrained model with adversarially finetuned in the previous work [17].

Comparison to semi-supervised learning Recently, semi-supervised learning[15, 16] have been shown to largely enhance the adversarial robustness of deep networks, by exploiting unlabeled data. However, they eventually require labeled data, to generate pseudo-labels on the unlabeled samples, and to generate class-wise adversaries. Also, they assume the availability of a larger dataset to improve robustness on the target dataset and require extremely large computation resources. Compared to the semi-supervised learning methods, RoCL takes about 1/4 times faster with the same computation resources. Moreover, ours acquires sufficiently high clean accuracy and robustness after 500 epochs (Fig. 3(c)) which takes 25 hours with two RTX 2080 GPUs.

Results on black box attacks We also validate our models in black box attacks setting. We generate the adversarial examples on AT, and TRADES model and test to four different models. As you can see in Table 2, our model is superior than the TRADES [2] at AT black box images. Also, our model even shows comparable performance to AT [9] model in the TRADES black box images.

Transformation smoothed classifier Transformation smoothed classifier can boost the accuracy of not only our models but also in the other models that use the stochastic transformation during training. However, compared to the TRADES in Table 1, we obtain a larger margin in robustness through the transformation smoothed classifier. Intuitively, since we enforce differently transformed samples to



(a) TSNE of SimCLR (b) TSNE of RoCL (c) Learning curve of RoCL (d) Black box robustness

Figure 3: (a,b) Visualizations of the embedding of *instance-wise* adversarial examples and clean examples for SimCLR and RoCL. (c) The learning curve of ResNet18 RoCL. (d) The transformation smoothed classifier performance on AT's black box attack over iteration.

agree, when we operate different transformations on one sample, the latent vectors will be placed in similar representation space. Therefore, we can calculate the transformation ball around the samples which acts as Gaussian ball in [24]. Accordingly, compare to the TRADES, we can obtain smoother classifier and able to acquire better gain in robustness not only white box attacked images, but also in black box images 3(d). As shown in Fig. 3(d), as iteration number increases the robustness also increase. The best part of the transformation smoothed classifier is that we do not have any trade-off in clean accuracy (Table 1).

Transfer learning Unsupervised learning representation has the benefit of applying to the other downstream not only the classification. We demonstrate the effectiveness of our works on transfer learning in Table 3. Surprisingly, our model shows high clean accuracy and high robustness in both CIFAR-100 and CIFAR-10 when the source model is trained with CIFAR-10 and CIFAR-100 respectively, without any other additional loss. Since our method learns rich robust representation, our model shows even better transfer results compare to the fully supervised robust transfer learning[38]. The more experiment results of transfer learning are in **the supplementary A**.

4.2 Ablation study

***Instance-wise* attack verification** We verify our instance-wise attacks on SimCLR (Fig. 3(a)). Our attacks generate the confusing samples as red edge markers which are far apart from the identical instances. Even though the same shapes are placed in adjacent space except for adversarial examples which are all identical samples but differently transformed. However, if we train the model with RoCL (Fig. 3(b)), instance-wise attacks are gathered with transformed images of the same identity.

Attack target image ablation Since our RoCL is an instance discriminative model, we can use any identity for the target for the attack. As shown in Table 5, even when we use the original x for the attacks, our RoCL still shows high natural accuracy along with high robustness. This is because the key to our method is matching the instance-level identity which is not biased with transformation. Therefore, our methods show stable performance with any kind of target which has the same identity.

Attack loss ($\mathcal{L}$) Various distance measures can be used to compute the distance between two samples in the representation space. Here, we apply four different distance functions: mean square error (MSE), cosine similarity, Manhattan distance (MD), and contrastive loss. The results in Table 6 shows that contrastive loss is the most effective attack loss, compared to others.

5 Conclusion

In this paper, we tackled a novel problem of learning robust representations without any class labels. We first proposed a *instance-wise attack* to make the model confuse the instance-level identity of a given sample. Then, we proposed a *robust contrastive learning* framework to suppress their adversarial vulnerability by maximizing the similarity between a transformed sample and its instance-wise adversary. Furthermore, we demonstrate an effective transformation smoothed classifier which boosts our performance during the test inference. We validated our method on multiple benchmarks with different neural architectures, on which it obtained comparable robustness to the supervised baselines on the targeted attack without any labels. Notably, RoCL obtained significantly better clean accuracy and better robustness against black box, unseen attacks, and transfer learning, which makes it more appealing as a general defense mechanism. Our work opened a door to more interesting follow-up works on *unsupervised adversarial learning*, which we believe is a more fundamental solution to achieving adversarial robustness with deep neural networks.

Broader Impact

The adversarial example brought alertness to blind deep learning beliefs. Due to the adversarial example, it is necessary to consider the vulnerability of deep learning, and we have been doing various studies to make deep learning more secure. To make a more robust model, we always present new strong attacks which is an additional concern to the research field. However, we can prevent those attacks by using adversarial training. Even though advanced attacks introduce, it is necessary to find a way to deal with those vulnerabilities. In such a paradigm, we should not allow vulnerabilities even for models that learn without labels. We believe that this is the first prominent step toward using safer deep learning in the field of self-supervised learning.

References

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

[2] H. Zhang, Y. Yu, J. Jiao, E. P. Xing, L. E. Ghaoui, and M. I. Jordan, "Theoretically principled trade-off between robustness and accuracy," in *Proceedings of the 36th International Conference on Machine Learning*, 2019.

[3] F. Tramèr and D. Boneh, "Adversarial training and robustness for multiple perturbations," in *Advances in Neural Information Processing Systems*, pp. 5858–5868, 2019.

[4] J. Madaan, Divyam Jin and S. J. Hwang, "Adversarial neural pruning with latent vulnerability suppression," *arXiv preprint arXiv:1908.04355*, 2019.

[5] S. Zheng, Y. Song, T. Leung, and I. Goodfellow, "Improving the robustness of deep neural networks via stability training," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 4480–4488, 2016.

[6] D. Hendrycks and T. Dietterich, "Benchmarking neural network robustness to common corruptions and perturbations," in *International Conference on Learning Representations*, 2019.

[7] D. Yin, R. G. Lopes, J. Shlens, E. D. Cubuk, and J. Gilmer, "A fourier perspective on model robustness in computer vision," in *Advances in Neural Information Processing Systems*, pp. 13255–13265, 2019.

[8] I. J. Goodfellow, J. Shlens, and C. Szegedy, "Explaining and harnessing adversarial examples," in *International Conference on Learning Representations*, 2015.

[9] A. Madry, A. Makelov, L. Schmidt, D. Tsipras, and A. Vladu, "Towards deep learning models resistant to adversarial attacks," in *International Conference on Learning Representations*, 2018.

[10] S. Gidaris, P. Singh, and N. Komodakis, "Unsupervised representation learning by predicting image rotations," in *International Conference on Learning Representations*, 2018.

[11] M. Noroozi and P. Favaro, "Unsupervised learning of visual representations by solving jigsaw puzzles," in *European Conference on Computer Vision*, pp. 69–84, Springer, 2016.

[12] T. Chen, S. Kornblith, M. Norouzi, and G. Hinton, "A simple framework for contrastive learning of visual representations," *arXiv preprint arXiv:2002.05709*, 2020.

[13] K. He, H. Fan, Y. Wu, S. Xie, and R. Girshick, "Momentum contrast for unsupervised visual representation learning," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2020.

[14] Z. Wu, Y. Xiong, S. X. Yu, and D. Lin, "Unsupervised feature learning via non-parametric instance discrimination," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 3733–3742, 2018.

[15] Y. Carmon, A. Raghunathan, L. Schmidt, J. C. Duchi, and P. S. Liang, "Unlabeled data improves adversarial robustness," in *Advances in Neural Information Processing Systems*, pp. 11190–11201, 2019.

[16] R. Stanforth, A. Fawzi, P. Kohli, *et al.*, "Are labels required for improving adversarial robustness?," in *Advances in Neural Information Processing Systems*, 2019.

[17] T. Chen, S. Liu, S. Chang, Y. Cheng, L. Amini, and Z. Wang, "Adversarial robustness: From self-supervised pre-training to fine-tuning," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2020.

[18] S.-M. Moosavi-Dezfooli, A. Fawzi, and P. Frossard, "Deepfool: a simple and accurate method to fool deep neural networks," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 2574–2582, 2016.

[19] A. Kurakin, I. Goodfellow, and S. Bengio, "Adversarial examples in the physical world," *arXiv preprint arXiv:1607.02533*, 2016.

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

[21] A. Athalye, N. Carlini, and D. Wagner, "Obfuscated gradients give a false sense of security: Circumventing defenses to adversarial examples," in *Proceedings of the 35th International Conference on Machine Learning*, 2018.

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

[23] B. Li, C. Chen, W. Wang, and L. Carin, "Certified adversarial robustness with additive noise," in *Advances in Neural Information Processing Systems*, pp. 9459–9469, 2019.

[24] J. Cohen, E. Rosenfeld, and Z. Kolter, "Certified adversarial robustness via randomized smoothing," in *Proceedings of the 36th International Conference on Machine Learning*, pp. 1310–1320, 2019.

[25] H. Salman, J. Li, I. Razenshteyn, P. Zhang, H. Zhang, S. Bubeck, and G. Yang, "Provably robust deep learning via adversarially trained smoothed classifiers," in *Advances in Neural Information Processing Systems*, pp. 11289–11300, 2019.

[26] A. Dosovitskiy, P. Fischer, J. T. Springenberg, M. Riedmiller, and T. Brox, "Discriminative unsupervised feature learning with exemplar convolutional neural networks," *IEEE transactions on pattern analysis and machine intelligence*, vol. 38, no. 9, pp. 1734–1747, 2015.

[27] C. Doersch, A. Gupta, and A. A. Efros, "Unsupervised visual representation learning by context prediction," in *Proceedings of the IEEE International Conference on Computer Vision*, pp. 1422–1430, 2015.

[28] R. Zhang, P. Isola, and A. A. Efros, "Colorful image colorization," in *European conference on computer vision*, pp. 649–666, Springer, 2016.

[29] D. Pathak, P. Krahenbuhl, J. Donahue, T. Darrell, and A. A. Efros, "Context encoders: Feature learning by inpainting," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 2536–2544, 2016.

[30] Y. Tian, C. Sun, B. Poole, D. Krishnan, C. Schmid, and P. Isola, "What makes for good views for contrastive learning," *arXiv preprint arXiv:2005.10243*, 2020.

[31] P. Khosla, P. Teterwak, C. Wang, A. Sarna, Y. Tian, P. Isola, A. Maschinot, C. Liu, and D. Krishnan, "Supervised contrastive learning," *arXiv preprint arXiv:2004.11362*, 2020.

[32] D. Hendrycks, M. Mazeika, S. Kadavath, and D. Song, "Using self-supervised learning can improve model robustness and uncertainty," in *Advances in Neural Information Processing Systems*, pp. 15637–15648, 2019.

[33] K. Chen, H. Zhou, Y. Chen, X. Mao, Y. Li, Y. He, H. Xue, W. Zhang, and N. Yu, "Self-supervised adversarial training," *arXiv preprint arXiv:1911.06470*, 2019.

[34] P. Bachman, R. D. Hjelm, and W. Buchwalter, "Learning representations by maximizing mutual information across views," in *Advances in Neural Information Processing Systems*, pp. 15509–15519, 2019.

[35] A. Kolesnikov, X. Zhai, and L. Beyer, "Revisiting self-supervised visual representation learning," *CoRR*, vol. abs/1901.09005, 2019.

[36] 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*, pp. 770–778, 2016.

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

[38] A. Shafahi, P. Saadatpanah, C. Zhu, A. Ghiasi, C. Studer, D. Jacobs, and T. Goldstein, "Adversarially robust transfer learning," in *International Conference on Learning Representations*, 2020.

[39] S. Zagoruyko and N. Komodakis, "Wide residual networks," *arXiv preprint arXiv:1605.07146*, 2016.

[40] D. Hendrycks, K. Lee, and M. Mazeika, "Using pre-training can improve model robustness and uncertainty," in *Proceedings of the 36th International Conference on Machine Learning*, 2019.

Supplementary
Adversarial Self-Supervised Contrastive Learning

Organization The supplementary file is organized as follows. In section A, we describe the experimental details, including the descriptions of the datasets and the evaluation process. We then provide an algorithm which summarizes our RoCL in section B. Then, we further report the RoCL results on both **CIFAR 10** and **CIFAR 100** against PGD attacks and CW attacks in Section C. Finally, perform ablation studies of our RoCL in section D.

A Experimental Setup

A.1 Training detail and dataset

CIFAR 10 The CIFAR 10 [37] dataset consists of 60,000 RGB images of size 32×32, from ten general object classes. The dataset consists of 5,000 training images and 1,000 test images for each class. We use ResNet18 and ResNet50 [36] for this dataset without any linear layers. For the projector network we set the output dimension as 128. For adversarial training parameters, we set the perturbation $\epsilon = 0.0314$, step size $\alpha = 0.007$, and number of iteration $K = 7$. We train the model with the batch size $B = 256$, $\lambda = 1/512$ for 1000 epochs.

CIFAR 100 The CIFAR 100 [37] dataset consists of 60,000 RGB images of size 32×32, from 100 general object classes. The dataset consists of 500 training images and 100 test images for each class. For this exeperiments on this dataset, we use ResNet18 [36] without any linear layers, and use the projector with 128 output dimensions. We set the perturbation $\epsilon = 0.0314$, step size $\alpha = 0.007$, and the number of iteration $K = 7$. We train the model with batch size $B = 256$, $\lambda = 1/512$, and train the model for 1000 epochs.

A.2 Evaluation

Linear evaluation setup In the linear evaluation phase, we train the linear layer ψ on the top of the frozen encoder f. We train the linear layer for 150 epochs with the learning rate of 0.2. The learning rate is dropped by a factor of 10 at 30, 50, 100 epoch of the training progress. We use stochastic gradient descent (SGD) optimizer with a momentum of 0.9, weight decay of $5e$-4, and train the linear layer with the cross-entropy (CE) loss.

Robust linear evaluation setup For robust linear evaluation, we train the linear layer π on the top of the frozen encoder f_θ, as done with linear evaluation. We train the linear layer for 150 epochs with an learning rate of 0.02. The learning rate scheduling and the optimizer setup is the same with the setup for linear evaluation. We use the project gradient descent (PGD) attack to generate class-wise adversarial examples. We perform ℓ_∞ attack with epsilon $\epsilon = 0.0314$ and the step size $\alpha = 0.007$ for 10 steps.

Robustness evaluation setup For evaluation of adversarial robustness, we use white-box project gradient descent (PGD) attack. We evaluate under PGD attacks with 20, 40, 100 steps. We set $\ell_\infty, \ell_2, \ell_1$ attacks with $\epsilon = 0.0314, 0.072$ for ℓ_∞, $\epsilon = 0.25, 0.5$ for ℓ_2, and $\epsilon = 7.84, 12$ for ℓ_1 for testing CIFAR 10 and CIFAR 100.

Transfer learning setup We first briefly describe robust transfer learning and our experiments in its experimental setting. Shafahi et al. [38] suggest that an adversarially trained model can be transferred to another model to improve upon its robustness. They used modified WRN 32-10 to train the fully supervised adversarial model. Moreover, they initialize the student network with an adversarially trained teacher network and utilize the distillation loss and cross-entropy loss to train the student network's linear layer on the top of the encoder layer. We follow the experimental settings of Shafahi et al. [38], and train only the linear layer with cross-entropy loss. However, we did not use the distillation loss in order to evaluate the robustness of the encoder trained with our RoCL only (ResNet18). We train the linear model with CIFAR 100 on top of the frozen encoder, which is trained on CIFAR 10. We also train the linear layer with CIFAR 10 on top of the frozen encoder, which is trained on CIFAR 100. We train the linear layer for 100 epochs with a learning rate of 0.2. We use stochastic gradient descent (SGD) for optimization.

B Algorithm of RoCL

We present the algorithm for RoCL in Algorithm 1. During training, we generate the instance-wise adversarial examples using contrastive loss and then train the model using two differently transformed images and their instance-wise adversarial perturbations. We also include a regularization term that is defined as a contrastive loss between the adversarial examples and clean transformed examples.

Algorithm 1 Robust Contrastive Learning (RoCL)

```
Input: Dataset $\mathbb{D}$, parameter of model $\theta$, model $f$, parameter of projector $\pi$, projector $p$, constant $\lambda$
    for all iter $\in$ number of training iteration do
        for all $x \in$ minibatch $B = \{x_1, \ldots, x_m\}$ do
            Generate adversarial examples from transformed inputs          ▷ instance-wise attacks
            $t(x)^{i+1} = \Pi_{B(t(x),\epsilon)}(t(x)^i - \alpha \texttt{sign}(\nabla_{t(x)^i} \mathcal{L}_{\texttt{con},\theta,\pi}(t(x)^i, \{t'(x)\}, t(x)_{\texttt{neg}})))$
        end for
        $\mathcal{L}_{total} = \frac{1}{N} \sum_{i=1}^{N} [\mathcal{L}_{\texttt{RoCL},\theta,\pi}(t(x)_i, \{t'(x)_i, t(x)_i^{adv}\}, \{t(x)_{\texttt{neg}}\})$
                $+ \lambda \mathcal{L}_{\texttt{con},\theta,\pi}(t(x)_i^{adv}, \{t'(x)_i\}, \{t(x)_{\texttt{neg}}\})]$          ▷ Contrastive loss
        Optimize the weight $\theta$, $\pi$ over $\mathcal{L}_{total}$
    end for
```

C Results of CIFAR 10 and CIFAR 100

While we only report the performance of RoCL on CIFAR 10 in the main paper as the baselines we mainly compare against only experimented on this dataset, we further report the performance of RoCL on CIFAR 100 as well (Table 7) and performance against CW attacks [20] (Table 8). We observe that RoCL consistently achieves comparable performance to that of the supervised adversarial learning methods, even on the CIFAR 100 dataset. Moreover, when employing the robust linear evaluation, RoCL acquires better robustness over the standard linear evaluation. Finally, the transformation smoothed classifier further boosts the performance of RoCL on both datasets.

Table 7: Experimental results with white box attacks on ResNet18 trained on the CIFAR 10 and CIFAR 100 dataset. r-LE denotes robust linear evaluation. AT denotes the supervised adversarial training[9]. All models are trained with ℓ_∞; thus the ℓ_∞ is the *seen* adversarial attack and ℓ_2, and ℓ_1 attacks are *unseen*.

Train type	Method	CIFAR10							CIFAR100						
			seen		*unseen*					*seen*		*unseen*			
			ℓ_∞		ℓ_2		ℓ_1			ℓ_∞		ℓ_2		ℓ_1	
		A_{nat}	ϵ 8/255	16/255	0.25	0.5	7.84	12	A_{nat}	ϵ 8/255	16/255	0.25	0.5	7.84	12
Supervised	$\mathcal{L}_{\texttt{CE}}$	**92.82**	0.00	0.00	20.77	12.96	28.47	15.56	**71.35**	0.00	0.00	6.54	2.31	11.14	5.86
	AT[9]	81.63	44.50	14.47	**72.26**	**59.26**	**66.74**	**55.74**	53.97	**20.09**	**6.19**	43.08	32.29	40.43	33.18
	TRADES[2]	77.03	**48.01**	**22.55**	68.07	57.93	62.93	53.79	56.63	17.94	4.29	**44.82**	**33.76**	**43.70**	**37.00**
Self -supervised	SimCLR[12]	**91.25**	0.63	0.08	15.3	2.08	41.49	25.76	**57.46**	0.04	0.02	6.58	0.7	19.27	12.1
	RoCL	83.71	40.27	9.55	66.39	63.82	**79.21**	**76.17**	56.13	19.31	4.30	38.65	35.94	**50.21**	**46.67**
	RoCL + rLE	80.43	**47.69**	**15.53**	**68.30**	**66.19**	77.31	75.05	51.82	**26.27**	**8.94**	**41.59**	**39.86**	49.00	46.91
Transformation smoothed classifier	**RoCL+tInf**	84.11	52.58	18.55	81.15	76.01	82.13	80.40	57.29	26.31	8.16	53.75	46.31	54.53	52.06

Table 8: Experimental results with white box CW attacks [20] on ResNet18 trained on the CIFAR 10. r-LE denotes robust linear evaluation. All models are trained with ℓ_∞

		CIFAR10		CIFAR100	
Train type	Method	A_{nat}	CW	A_{nat}	CW
Self -supervised	**RoCL**	83.71	77.35	56.13	44.57
	RoCL+rLE	80.43	76.15	51.82	44.77

D Ablation

In this section, we report the results of several ablation studies of our RoCL model. For all experiments, we train the backbone network with 500 epochs and train the linear layer with 100 epochs, which yield models with sufficiently high clean accuracy and robustness. We first examine the effects of the target image when generating the instance-wise adversarial examples. Along with instance-wise attacks, the regularization term in algorithm 1 can also affect the final performance of the model. To examine lambda's effect on the transformed images, we set lambda as $\lambda = 1/128$ for CIFAR 10 and $\lambda = 1/256$ for CIFAR 100. We also examine the effects of lambda λ on the CIFAR 10 dataset.

D.1 Adversarial contrastive learning

We examine the effect of the transformation function on the instance-wise attack and the regularization. For each input instance x, we generated three transformed images $t(x), t'(x)$, and $t(x)^{adv}$ and use them as the positive set. The results in Table 9 demonstrate that using any transformed images from the same identity for instance-wise attacks is equally effective. In contrast, for regularization, using images transformed with a different transformation function from the one used to generate attack helps obtain improved clean accuracy and robustness.

Instance-wise attack To generate instance-wise attacks, we can decide which identity we will use for instance-wise attack. Since the original transformed image $t(x)$ and image transformed with another transformation $t'(x)$ have the same identity, we can use both of them in instance-wise attacks. To find the optimal perturbation that maximizes the contrastive loss between adversarial examples and same identity images, we vary **X** in the following equation:

$$t(x)^{i+1} = \Pi_{B(t(x),\epsilon)}(t(x)^i - \alpha \texttt{sign}(\nabla_{t(x)^i} \mathcal{L}_{\texttt{con},\theta,\pi}(t(x)^i, \{\mathbf{X}\}, t(x)_{\texttt{neg}}))) \tag{9}$$

where **X** is either $t'(x)$ and $t(x)$.

Regularization To regularize the learning, we can calculate the contrastive loss between adversarial examples and clean samples with the same instance-level identity. We vary **Y** in the regularization term to examine which identity is the most effective, as follows:

$$\lambda \mathcal{L}_{\texttt{con},\theta,\pi}(t(x)_i^{adv}, \{\mathbf{Y}\}, \{t(x)_{\texttt{neg}}\}) \tag{10}$$

where **Y** can be $t'(x)$ and $t(x)$.

Table 9: Experimental results with white box attacks on ResNet18 trained on the CIFAR 10 and CIFAR 100 dataset. All models are trained with ℓ_∞.

	instance-wise attack (**X**)		Regularization (**Y**)		CIFAR 10		CIFAR 100	
Method	$t'(x)$	$t(x)$	$t'(x)$	$t(x)$	A_{nat}	ℓ_∞	A_{nat}	ℓ_∞
RoCL	✓	-	✓	-	**82.79**	**36.71**	**55.64**	**17.56**
	✓	-	-	✓	81.47	29.97	53.84	14.18
	-	✓	✓	-	82.43	34.93	55.61	17.42
	-	✓	-	✓	81.96	30.99	53.76	14.74

D.2 Lambda λ and batch size B

We observe that λ, which controls the amount of regularization in the robust contrastive loss, and the batch size for calculating the contrastive loss, are two important hyperparameters for our robust contrastive learning framework. We examine the effect of two hyperparameters in Table 10, and Table 11. We observe that the optimal lambda λ is different for each batch size B.

Table 10: lambda λ ablation experimental results with white box attacks on ResNet18 trained on the CIFAR 10 dataset. All models are trained with ℓ_∞.

			ℓ_∞	
CIFAR 10	λ	A_{nat}	8/255	16/255
RoCL	1/16	82.05	35.12	8.05
	1/32	82.25	36.02	**8.68**
	1/64	**83.00**	36.26	8.19
	1/128	82.79	36.71	8.34
	1/256	82.12	**38.05**	8.52
	1/512	82.68	37.24	8.53

Table 11: Ablation study of the batch size B, for the white box attacks on ResNet18 trained on the CIFAR 10 dataset. All models are trained with ℓ_∞ attacks.

				ℓ_∞	
CIFAR 10	B	λ	A_{nat}	8/255	16/255
RoCL	128	1/128	82.70	37.13	8.98
	128	1/256	82.90	36.86	8.89
	256	1/256	82.12	38.05	8.52
	512	1/256	81.48	34.98	7.42