ProFeAT: Projected Feature Adversarial Training for Self-Supervised Learning of Robust Representations

Sravanti Addepalli [*1 2] **Priyam Dey** [*1] **R. Venkatesh Babu** [1]

Abstract

The need for abundant labelled data for supervised Adversarial Training (AT) has prompted the use of Self-Supervised Learning (SSL) techniques with AT. The direct application of existing SSL methods to adversarial training has been sub-optimal due to the increased training complexity of combining SSL with AT. A recent approach DeACL (Zhang et al., 2022) mitigates this by utilizing supervision from a standard SSL teacher in a distillation setting, to mimic supervised AT. However, we find that there is still a large performance gap when compared to supervised adversarial training, specifically on larger model capacities. We show that this is a result of mismatch in training objectives of the teacher and student, and propose Projected Feature Adversarial Training (ProFeAT) to bridge this gap. We utilize a projection head in the adversarial training step with appropriate attack and defense losses at the feature and projector, coupled with a combination of weak and strong augmentations for the teacher and student respectively, to improve both clean and robust generalization. Through extensive experiments on several benchmark datasets and models, we demonstrate significant improvements in performance when compared to existing SSL-AT methods, setting a new state-of-the-art. We further report on-par/ improved performance when compared to TRADES, a popular supervised-AT method.

1. Introduction

Self-supervised adversarial training (SSL-AT) aims to learn adversarially robust models without the need of extensive labelling requirements of AT. Existing approaches in this paradigm suffer from both poor robustness performance when linear probed as well as training inefficiency due to multiple attacks generation in the process. A recent work DeACL (Zhang et al., 2022) mitigates this to a large extent by training a robust student using single attack generation under the supervision of a frozen standard SSL pretrained teacher in a distillation setting. Although the performance of this method is on par with supervised AT models on small architectures (ResNet-18), we find that it does not scale to larger settings, such as for models like WideResNet-34-10, which is widely reported in the adversarial ML literature. In this work, we aim to bridge the performance gap between SSL-AT and supervised-AT methods, and improve the scalability of the former to larger model capacities. For this, we consider the SSL-AT distillation setup of DeACL. Note that in contrast to a typical knowledge distillation scenario, the ideal goal for the student in SSL-AT distillation is not to faithfully replicate the teacher, but to leverage weak supervision from the teacher while simultaneously enhancing its adversarial robustness. To achieve this, we propose to impose the distillation loss in a *projection space* (output of the projection layer) instead of *feature space* (output of the feature extractor), while enforcing the smoothness loss in the feature space. We further propose to reuse the pretrained projection layer from the teacher model for improved convergence. For improving the training stability, we introduce an additional regularizer of complementary losses in the respective feature and projection spaces. Finally, contrary to common wisdom in supervised adversarial training, we propose to use *strong augmentations* such as AutoAugment for the student model for better attack diversity, while using *weak augmentations* like pad and crop (PC) for the teacher model in our SSL-AT distillation framework.

We summarize our contributions below:

- We propose Projected Feature Adversarial Training (ProFeAT) - a distillation framework for SSL-AT training, where the projection layer of the standard SSL pretrained teacher is reused for student training. We propose appropriate attack and defense losses for training, coupled with a combination of weak and strong augmentations for the teacher and student respectively.

[*]Equal contribution [1]Vision and AI Lab, Department of Computational and Data Science, IISc Bangalore. [2]Visiting Researcher, Google. Correspondence to: Sravanti Addepalli <sravantia@iisc.ac.in>, Priyam Dey <priyamdey@iisc.ac.in>.

Proceedings of the 41st International Conference on Machine Learning, Vienna, Austria. PMLR 235, 2024.

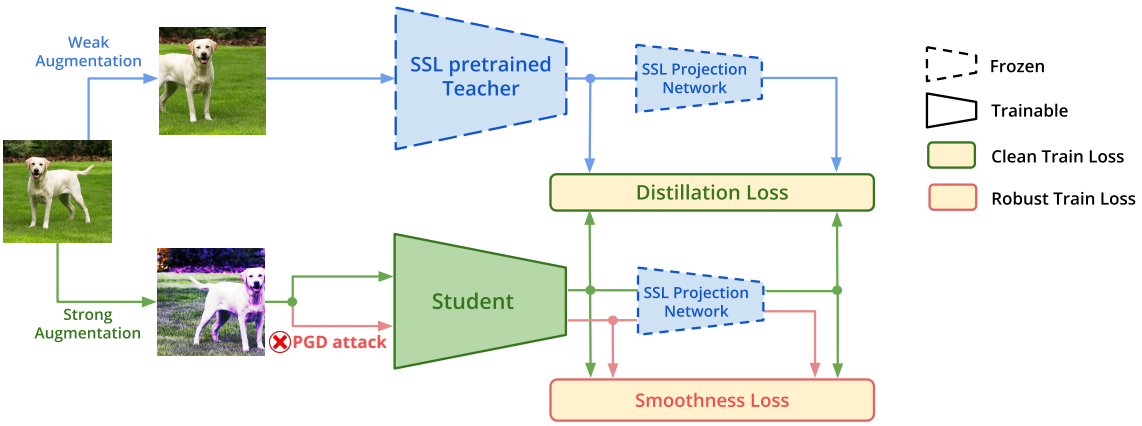

Figure 1. **Proposed approach (ProFeAT):** The student is trained using a distillation loss on clean samples using supervision from an SSL pretrained teacher, and a smoothness loss to enforce adversarial robustness (details of exact loss formulation is presented in Section 3.2). A frozen pretrained projection layer is used at the teacher and student to prevent overfitting to the clean distillation loss. The use of strong augmentations at the student increases attack diversity, while weak augmentations at the teacher reduce the training complexity.

- Towards understanding *why* the projector helps, we first show that the compatibility between the training methodology of the teacher and the ideal goals of the student plays a crucial role in the student model performance in distillation. We further show that the use of a projector can alleviate the negative impact of the inherent misalignment of the above.
- We demonstrate the effectiveness of the proposed approach on the standard benchmark datasets CIFAR-10 and CIFAR-100. We obtain significant gains of $3.5 - 8\%$ in clean accuracy and $\sim 3\%$ in robust accuracy on larger models (WideResNet-34-10), with consistent gains on smaller architectures like ResNet-18 compared to existing baselines, while also outperforming TRADES (Zhang et al., 2019) supervised AT method. Extensive ablation studies are carried to demonstrate the efficacy of the proposed method.

2. Related Works

Self Supervised Learning (SSL): With the abundance of unlabelled data, learning representations through self-supervision has seen major advances in recent years. Contrastive learning based SSL approaches have emerged as a promising direction (Van den Oord et al., 2018; Chen et al., 2020a; He et al., 2020), where different augmentations of a given anchor image form positives, and augmentations of other images in the batch form the negatives. The training objective involves pulling the representations of the positives together, and repelling the representations of negatives.

Self Supervised Adversarial Training: To alleviate the large sample complexity and training cost of adversarial training, there have been several works that have attempted self-supervised learning of adversarially robust representations. Chen et al. (2020b) propose AP-DPE, an ensemble adversarial pretraining framework where several pretext tasks like Jigsaw puzzles (Noroozi & Favaro, 2016), rotation prediction (Gidaris et al., 2018) and Selfie (Trinh et al., 2019) are combined to learn robust representations without task labels. Jiang et al. (2020) propose ACL, that combines the popular contrastive SSL method - SimCLR (Chen et al., 2020a) with adversarial training, using Dual Batch normalization layers for the student model - one for the standard branch and another for the adversarial branch. RoCL (Kim et al., 2020) follows a similar approach to ACL by combining the contrastive objective with adversarial training to learn robust representations. Fan et al. (2021) propose AdvCL, that uses high-frequency components in data as augmentations in contrastive learning, performs attacks on unaugmented images, and uses a pseudo label based loss for training to minimize the cross-task robustness transferability. Luo et al. (2023) study the role of augmentation strength in self-supervised contrastive adversarial training, and propose DynACL, that uses a "strong-to-weak" annealing schedule on augmentations. Additionally, motivated by Kumar et al. (2022), they propose DynACL++ that obtains pseudo-labels via k-means clustering on the clean branch of the DynACL pretrained network, and performs linear-probing (LP) using these pseudo-labels followed by adversarial full-finetuning (AFT) of the backbone. This is a generic strategy that can be integrated with several algorithms including ours.

While most self-supervised adversarial training methods aimed at integrating contrastive learning methods with adversarial training, Zhang et al. (2022) showed that combining the two is a complex optimization problem due to their conflicting requirements. The authors propose Decoupled

Adversarial Contrastive Learning (DeACL), where a teacher model is first trained using existing self-supervised training methods such as SimCLR, and further, a student model is trained to be adversarially robust using supervision from the teacher. While existing methods used ~ 1000 epochs for contrastive adversarial training, the compute requirement for DeACL is much lesser since the first contrastive learning stage does not involve adversarial training, and the second stage is similar in complexity to supervised adversarial training (Ref: Appendix-D). We utilize this distillation framework and obtain significant gains over DeACL, specifically at large model capacities.

3. Proposed Method

In this section, we first motivate the need for a projection layer, and further present the proposed approach ProFeAT.

3.1. Role of Projector in SSL-AT distillation

In this work, we follow the setting proposed by Zhang et al. (2022), where a standard self-supervised pretrained teacher provides supervision for the SSL-AT training of the student model. This is different from a standard distillation setting because the representations of standard and robust models are known to be inherently different (Engstrom et al., 2019). Due to this difference, the ideal goal of the student in the considered distillation setting is not to merely follow the teacher, but to be able to take weak supervision from it while being able to differ considerably. In order to achieve this, we take inspiration from standard SSL literature (Chen et al., 2020a; He et al., 2020; Chen & He, 2021) and propose to utilize a projection layer following the student backbone so as to insulate the impact of the enforced similarity loss from the teacher on the learned representations. Bordes et al. (2023) show that in standard supervised and self-supervised training, a projector is crucial when there is a misalignment between the pretraining and downstream tasks, and aligning them can eliminate the need for the same. Motivated by this, we hypothesize the following for self-supervised distillation:

Student model performance improves by matching the following during distillation:

1. *Training objectives of the teacher and the ideal goals of the student,*
2. *Pretraining and linear probe training objectives of the student.*

The ideal goal of the student depends on the downstream task, which is standard accuracy in standard training, and standard and robust accuracy in case of adversarial training. On the other hand, the training objective of the teacher is to achieve invariance to augmentations of the same image in contrastive learning, and standard accuracy in a supervised training setup. Due to space constraints, we defer the explanation on the intuition behind the hypothesis to Appendix A and provide empirical justification for the same in Appendix E.1.

3.2. ProFeAT: Projected Feature Adversarial Training

We present details on the proposed approach ProFeAT, illustrated in Figure 1. First, a teacher model is trained using a standard SSL training method such as SimCLR (Chen et al., 2020a), whose weights are also later used as an initialization for the student in the distillation stage for better convergence. We now elaborate on the proposed method ProFeAT:

Use of Projection Layer: As discussed in Section 3.1, we use a projection head at the output of the student backbone. As most SSL pretraining methods use similarity-based losses at the output of a projection head for training, we therefore utilize this pretrained projection head for both teacher and student and freeze it during training to prevent convergence to an identity mapping.

Defense loss: We use a combination of clean loss and smoothness loss to enforce adversarial robustness in the student model. Since the clean loss utilizes supervision from the SSL teacher, it is enforced at the outputs of the respective projectors of the teacher and student, as discussed above. Smoothness loss enforces local smoothness in the input loss surface of the student. While the ideal locations for the clean and adversarial losses are the projected and feature spaces respectively, we find that such a loss formulation is difficult to optimize, resulting in either a non-robust model, or collapsed representations. We therefore use complimentary losses as a regularizer in the respective projection and feature spaces, resulting in a combination of losses as shown below (dropping the dependence on x_i for brevity):

$$\mathcal{L}_{pf} = -\sum_i \cos\left(\mathcal{T}_{pf_i}, \mathcal{S}_{pf_i}\right) + \beta \cdot \cos\left(\mathcal{S}_{pf_i}, \tilde{\mathcal{S}}_{pf_i}\right) \quad (1)$$

$$\mathcal{L}_{f} = -\sum_i \cos\left(\mathcal{T}_{f_i}, \mathcal{S}_{f_i}\right) + \beta \cdot \cos\left(\mathcal{S}_{f_i}, \tilde{\mathcal{S}}_{f_i}\right) \quad (2)$$

$$\mathcal{L}_{\text{ProFeAT}} = \frac{1}{2} \cdot \left(\mathcal{L}_{pf} + \mathcal{L}_{f}\right) \quad (3)$$

$$\tilde{x}_i = \underset{\substack{\tilde{x}_i \\ ||\tilde{x}_i - x_i||_\infty \leq \varepsilon}}{\text{argmin}} \cos\left(\mathcal{T}_{pf_i}, \tilde{\mathcal{S}}_{pf_i}\right) + \cos\left(\mathcal{S}_{f_i}, \tilde{\mathcal{S}}_{f_i}\right) \quad (4)$$

Here, L_{pf} and L_f are the defense losses enforced at the projector and feature spaces, respectively. $\mathcal{T}_{pf_i} = \left(\mathcal{T}_p \circ \mathcal{T}_f\right)(x_i)$ is the composition of the projection layer $\mathcal{T}_p$ on the feature backbone $\mathcal{T}_f$ of the teacher $\mathcal{T}$ for a clean input x_i. $\tilde{\mathcal{S}} = \mathcal{S}(\tilde{x})$ where $\tilde{x}$ is the adversarial input and $\mathcal{S}$ represents student representation (other subscript notations for the student are analogous to the teacher). The first term in Equations (1) and (2) represents the **Distilla-**

Table 1. **SOTA comparison:** Standard Linear Probing performance (%) on CIFAR-10 and CIFAR-100 datasets on ResNet-18 and WideResNet-34-10 models. Mean and standard deviation across 3 reruns are reported for DeACL (Zhang et al., 2022) and the proposed approach, ProFeAT. Standard Accuracy (SA), Robust Accuracy against AutoAttack (RA-AA) and PGD-20 (RA-PGD20) reported.

Method	CIFAR-10			CIFAR-100		
	SA	RA-PGD20	RA-AA	SA	RA-PGD20	RA-AA
ResNet-18						
Supervised (TRADES)	83.74	49.35	47.60	59.07	26.22	23.14
AP-DPE	78.30	18.22	16.07	47.91	6.23	4.17
RoCL	79.90	39.54	23.38	49.53	18.79	8.66
ACL	77.88	42.87	39.13	47.51	20.97	16.33
AdvCL	80.85	50.45	42.57	48.34	27.67	19.78
DynACL	77.41	-	45.04	45.73	-	19.25
DynACL++	79.81	-	46.46	52.26	-	20.05
DeACL (Reported)	80.17	53.95	45.31	52.79	30.74	20.34
DeACL (Our Teacher)	80.05$\pm$ 0.29	52.97$\pm$ 0.08	**48.15**$\pm$ 0.05	51.53$\pm$ 0.30	30.92$\pm$ 0.21	21.91$\pm$ 0.13
ProFeAT (**Ours**)	**81.68**$\pm$ 0.23	49.55$\pm$ 0.16	47.02$\pm$ 0.01	**53.47**$\pm$ 0.10	27.95$\pm$ 0.13	**22.61**$\pm$ 0.14
WideResNet-34-10						
Supervised (TRADES)	85.50	54.29	51.59	59.87	28.86	25.72
DynACL++	80.97	48.28	45.50	52.60	23.42	20.58
DeACL	83.83$\pm$ 0.20	57.09$\pm$ 0.06	48.85$\pm$ 0.11	52.92$\pm$ 0.35	32.66$\pm$ 0.08	23.82$\pm$ 0.07
ProFeAT (**Ours**)	**87.62**$\pm$ 0.13	54.50$\pm$ 0.17	**51.95**$\pm$ 0.19	**61.08**$\pm$ 0.18	31.96$\pm$ 0.08	**26.81**$\pm$ 0.11

tion loss (Figure 1), whereas the second term corresponds to the **Smoothness loss** at the respective layers of the student, and is weighted by a hyperparameter β that controls the robustness-accuracy trade-off in the downstream model. The overall loss $\mathcal{L}_{\text{ProFeAT}}$ (Equation (3)) is minimized during training.

Attack generation: As shown in Equation (4), we minimize the cosine similarity between the teacher $\mathcal{T}_{pf}$ and student's adversarial representation $\tilde{\mathcal{S}}_{pf}$ at projection layer for attack generation. Since the feature space is primarily used for enforcing local smoothness in the loss surface of the student, we minimize the cosine similarity between clean $\mathcal{S}_f$ and adversarial $\tilde{\mathcal{S}}_f$ samples of the student at this space.

Augmentations: Strong data augmentations such as AutoAugment (Cubuk et al., 2019), though beneficial for SSL training, are known to deteriorate the performance in supervised-AT (Rice et al., 2020; Gowal et al., 2020). We hypothesize that in SSL-AT training, the *need for better generalization is higher* since the pretraining task is not aligned with the ideal goals of the student, making it crucial to use strong augmentations. However, it is also important to ensure that the training task is not too complex. We thus propose to use a combination of weak and strong augmentations as inputs to the teacher and student respectively, as shown in Figure 1. While the use of weak augmentations at the teacher imparts better supervision to the student, reducing the training complexity, using strong augmentations for the student results in generation of more diverse attacks, thereby improving the robustness.

4. Experiments and Results

Datasets: We compare the performance of the proposed approach ProFeAT with existing methods on the benchmark datasets CIFAR-10 and CIFAR-100 (Krizhevsky et al., 2009), that are commonly used for evaluating the adversarial robustness of models (Croce et al., 2021). Both datasets consist of RGB images of dimension 32×32. CIFAR-10 consists of 50,000 images in the training set and 10,000 images in the test set, with the images being divided equally into 10 classes - airplane, automobile, bird, cat, deer, dog, frog, horse, ship and truck. CIFAR-100 dataset is of the same size as CIFAR-10, with images being divided equally into 100 classes. Due to the larger number of classes, there are only 500 images per class in CIFAR-100, making it a more challenging dataset when compared to CIFAR-10.

Model Architecture: We report the key comparisons with existing methods on two of the commonly considered model architectures in the literature of adversarial robustness (Pang et al., 2021; Zhang et al., 2019; Rice et al., 2020; Croce et al., 2021) - ResNet-18 (He et al., 2016) and WideResNet-34-10 (Zagoruyko & Komodakis, 2016). Although most existing methods for self-supervised adversarial training report results only on ResNet-18 (Zhang et al., 2022; Fan et al., 2021), we additionally consider the WideResNet-34-10 architecture to demonstrate the scalability of the proposed approach to larger model architectures. We perform

the ablation experiments on the CIFAR-100 dataset with WideResNet-34-10 architecture, which is a very challenging setting in self-supervised adversarial training, to be able to better distinguish between different variations adopted during training. We also report our results on the popular transformer-based architecture ViT-B/16 in Table 4 to demonstrate the efficacy of the proposed approach over existing baselines on diverse architecture types.

To evaluate the representations learned after self-supervised adversarial pretraining, we freeze the pretrained backbone, and perform a linear layer training on a downstream labeled dataset consisting of image-label pairs, popularly referred to as linear probing (Kumar et al., 2022). This linear training is done using CE loss on clean samples, unless specified otherwise. We consider the ℓ_∞ based threat model where $||\tilde{x}_i - x_i||_\infty \leq \varepsilon$. The value of ε is set to $8/255$, as is standard in literature (Madry et al., 2018; Zhang et al., 2019). The Robust Accuracy (RA) in the SOTA comparison tables is presented against AutoAttack (Croce & Hein, 2020) (RA-AA) which is widely used as a benchmark for robustness evaluation (Croce et al., 2021). In all other tables, we present robust accuracy against the GAMA attack (Sriramanan et al., 2020) (RA-G) which is known to be competent with AutoAttack, while being significantly faster. We additionally present results against a 20-step PGD attack (Madry et al., 2018) (RA-PGD20), as is standard (Fan et al., 2021; Zhang et al., 2022). Larger gap between PGD-20 and Autoattack/ GAMA occurs when the input loss surface is convoluted, due to the phenomenon of gradient masking (Papernot et al., 2017; Tramèr et al., 2018). Therefore, to compare the robust accuracy between any two defenses, the accuracy against AutoAttack (RA-AA) or GAMA (RA-G) should be considered. The accuracy on clean or natural samples is denoted as SA, which stands for Standard Accuracy.

4.1. Comparison with the state-of-the-art

In Table 1, we present a comparison of the proposed approach ProFeAT with respect to several existing SSL-AT approaches (Chen et al., 2020b; Kim et al., 2020; Jiang et al., 2020; Fan et al., 2021; Zhang et al., 2022) by freezing the backbone and performing linear probing using CE loss on clean samples. To ensure a fair comparison, the same is done for the supervised-AT TRADES model (Zhang et al., 2019) as well. We report results on CIFAR-10 and CIFAR-100 datasets (Krizhevsky et al., 2009), and on ResNet-18 (RN-18) and WideResNet-34-10 (WRN) architectures. The results of existing methods on RN18 are as reported in Zhang et al. (2022). Since DeACL also uses the distillation setting, we reproduce their results using the same teacher as our method, and report the same as "DeACL (Our Teacher)". Since most existing methods do not report results on WRN, we compare our results only with the best performing method (DeACL) and a recent method DynACL (Luo et al., 2023). These results are not reported in the respective papers, hence we run them using the official code.

Table 2. **Transfer Learning:** Standard Linear Probing performance (%) for transfer learning from CIFAR-10 and CIFAR-100 to STL-10 dataset on ResNet-18 and WideResNet-34-10 models.

Method	CIFAR-10 → STL-10			CIFAR-100 → STL-10		
	SA	RA-PGD20	RA-AA	SA	RA-PGD20	RA-AA
ResNet-18						
Supervised	54.70	30.45	22.26	51.11	23.63	19.54
DeACL	60.10	41.40	30.71	50.91	27.76	16.25
ProFeat (**Ours**)	**64.30**	35.50	**30.95**	**52.63**	26.72	**20.55**
WideResNet-34-10						
Supervised	67.15	32.78	30.49	**57.68**	17.49	11.26
DeACL	66.45	39.28	28.43	50.59	27.50	13.49
ProFeat (**Ours**)	**69.88**	35.48	**31.65**	56.68	24.95	**19.46**

The proposed approach ProFeAT obtains competent robustness-accuracy trade-off when compared to the best performing baseline method DeACL on RN-18 architecture. ProFeAT obtains improved clean accuracy ($\sim 2\%$) alongside consistent gains in robust accuracy on CIFAR-100 dataset with RN-18 model. On larger models (WRN), ProFeAT outperforms DeACL by even larger margins, obtaining $\sim 3{-}3.5\%$ gains in both robust and clean accuracy on CIFAR-10, and substantial gain of $\sim 8\%$ in clean accuracy and $\sim 3{-}3.5\%$ gain in robust accuracy for CIFAR-100. Overall, the proposed approach obtains significant gains when compared the DeACL at a similar computational cost (Ref: Appendix D). We also obtain superior results when compared to the supervised AT method TRADES, especially at larger model capacity (WRN). We present results with additional evaluation methods like KNN in Appendix E.2.

Transfer Learning: To evaluate the transferability of the robustness to datasets other than the one pretrained on, we consider the transfer learning setting from CIFAR-10/100 to STL-10 (Coates et al., 2011). We compare the proposed approach with the best baseline DeACL under standard linear probing (LP). As shown in Table 2, when compared to DeACL, the clean accuracy is $\sim 4{-}10\%$ higher on CIFAR-10 and $\sim 1.7{-}6\%$ higher on CIFAR-100. We also obtain $3{-}5\%$ higher robust accuracy when compared to DeACL on CIFAR-100, and higher improvements over TRADES. We also present transfer learning results using lightweight adversarial full finetuning (AFF) in Appendix E.3.

4.2. Ablations

We now present some of the ablation experiments to gain insights into the proposed method, and defer more in-depth ablation results to Appendix E due to space constraints.

Effect of each component on the proposed approach:

We show the impact of each component of the ProFeAT in

Table 3. **Ablation on ProFeAT (CIFAR-100, WRN-34-10):** Performance (%) by enabling different components of the proposed approach. A tick mark in the Projector column means that a frozen pretrained projector is used for the teacher and student, with the defense loss being enforced at the feature and projector as shown in Eq.3. **E1: DeACL (best baseline)**, **E8: ProFeAT (proposed approach)**. E8*:Defense loss applied only at the projector.

Ablation	Projector	Augs	Attack loss	SA	RA-PGD20	RA-G
E1				52.90	32.75	24.66
E2	✓			57.66	31.14	25.04
E3		✓		52.83	35.00	27.13
E4			✓	51.80	31.37	24.77
E5		✓	✓	55.35	35.89	**27.86**
E6	✓		✓	56.57	30.54	25.29
E7	✓	✓		**62.01**	31.62	26.89
E8	✓*	✓	✓	59.65	33.03	26.90
E9	✓	✓	✓	61.05	31.99	27.41

Table-3. Below are the observations based on the results:

- *Projector*: We observe significant gains in clean accuracy ($\sim 5\%$) by introducing the projector along with defense losses at the feature and projection spaces (E1 vs. E2). The importance of the projector is also evident by the fact that removing the projector from the proposed defense results in a large drop (5.7%) in clean accuracy (E9 vs. E5). We observe a substantial improvement of 9.2% in clean accuracy when the projector is introduced in the presence of the proposed augmentation strategy (E3 vs. E7), which is significantly higher than the gains obtained by introducing the same in the baseline DeACL (4.76%, E1 vs. E2).

- *Augmentations*: The proposed augmentation strategy improves robustness across all settings. Introducing the same in the baseline improves its robust accuracy by 2.47% (E1 vs. E3). Moreover, the importance of the proposed strategy is also evident from the fact that in the absence of the same, there is a 4.48% drop in SA and $\sim 2\%$ drop in RA-G (E9 vs. E6).

- *Attack loss*: The impact of the attack loss in feature space can be seen in combination with the proposed augmentations, where we observe an improvement of 2.5% in clean accuracy alongside notable improvements in robust accuracy (E3 vs. E5). However, in presence of projector, the attack results in only marginal robustness gains, possibly because the clean accuracy is already high (E9 vs. E7).

- *Defense loss*: We do not introduce a separate column for defense loss as it is applicable only in the presence of the projector. We show the impact of the proposed defense losses in the last two rows (E8 vs. E9). The proposed defense loss improves the clean accuracy by 1.4% and robust accuracy marginally.

Table 4. **Performance across different model architectures:** Standard Linear Probing performance (%) of DeACL (Baseline) and ProFeAT (Ours) across different architectures on CIFAR-100. ViT-B/16 uses Imagenet-1K trained SSL teacher for training, while the teacher in all other cases is trained on the CIFAR-100.

Method	#params (M)	DeACL		ProFeAT (Ours)	
		SA	RA-AA	SA	RA-AA
ResNet-18	11.27	51.53	21.91	**53.47**	**22.61**
ResNet-50	23.50	53.30	23.00	**59.34**	**25.86**
WideResNet-34-10	46.28	52.92	23.82	**61.08**	**26.81**
ViT-B/16	85.79	61.34	17.49	**65.08**	**21.52**

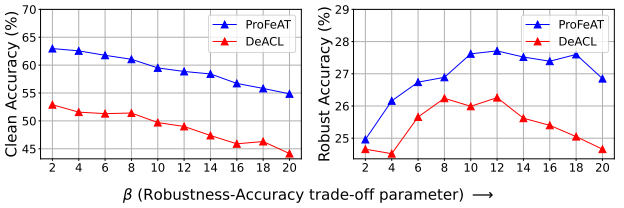

Figure 2. Performance of ProFeAT compared to DeACL (Zhang et al., 2022) across variation in the robustness-accuracy trade-off parameter β on CIFAR-100 dataset with WRN-34-10 model.

Performance across different model architectures: We report performance of the proposed method ProFeAT and the best baseline DeACL on diverse architectures including Vision Transformers (Dosovitskiy et al., 2021) on the CIFAR-100 dataset in Table 4. We note that ProFeAT consistently outperforms the best baseline DeACL across models, with a substantial gain of 4% in robust accuracy for ViT-B/16 and a 8% gain in clean accuracy for WRN architecture.

Robustness-Accuracy trade-off: We present results across variation in the robustness-accuracy trade-off parameter β (Equations (1) and (2)) in Figure 2. The proposed method achieves significantly better robustness and clean accuracy than DeACL across all values of β.

5. Conclusion

In this work, we bridge the performance gap between supervised and self-supervised adversarial training approaches, when scaled to large capacity models. We utilize the distillation setting of (Zhang et al., 2022) where a standard SSL teacher is used to provide supervision to a robust student. Due to the inherent misalignment between the teacher training objective and the ideal goals of the student, we propose to use a projection layer to prevent the network from overfitting to the teacher. We propose appropriate attack and defense losses in the feature and projector spaces alongside the use of weak and strong augmentations for the teacher and student respectively, to improve the attack diversity while maintaining low training complexity. The proposed approach obtains significant gains over existing self-supervised adversarial training methods, especially for large models, demonstrating its scalability.

References

Addepalli, S., Nasery, A., Radhakrishnan, V. B., Netrapalli, P., and Jain, P. Feature reconstruction from outputs can mitigate simplicity bias in neural networks. In *The Eleventh International Conference on Learning Representations*, 2023.

Bordes, F., Balestriero, R., Garrido, Q., Bardes, A., and Vincent, P. Guillotine regularization: Why removing layers is needed to improve generalization in self-supervised learning. *Transactions on Machine Learning Research (TMLR)*, 2023.

Chen, T., Kornblith, S., Norouzi, M., and Hinton, G. A simple framework for contrastive learning of visual representations. In *International Conference on Machine Learning (ICML)*, 2020a.

Chen, T., Liu, S., Chang, S., Cheng, Y., Amini, L., and Wang, Z. Adversarial robustness: From self-supervised pre-training to fine-tuning. In *The IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020b.

Chen, X. and He, K. Exploring simple siamese representation learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2021.

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

Croce, F. and Hein, M. Reliable evaluation of adversarial robustness with an ensemble of diverse parameter-free attacks. In *International Conference on Machine Learning (ICML)*, 2020.

Croce, F., Andriushchenko, M., Sehwag, V., Debenedetti, E., Flammarion, N., Chiang, M., Mittal, P., and Hein, M. Robustbench: a standardized adversarial robustness benchmark, 2021.

Cubuk, E. D., Zoph, B., Mane, D., Vasudevan, V., and Le, Q. V. Autoaugment: Learning augmentation strategies from data. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2019.

Dosovitskiy, A., Beyer, L., Kolesnikov, A., Weissenborn, D., Zhai, X., Unterthiner, T., Dehghani, M., Minderer, M., Heigold, G., Gelly, S., Uszkoreit, J., and Houlsby, N. An image is worth 16x16 words: Transformers for image recognition at scale. In *International Conference on Learning Representations (ICLR)*, 2021.

Engstrom, L., Ilyas, A., Santurkar, S., Tsipras, D., Tran, B., and Madry, A. Adversarial robustness as a prior for learned representations. *arXiv preprint arXiv:1906.00945*, 2019.

Fan, L., Liu, S., Chen, P.-Y., Zhang, G., and Gan, C. When does contrastive learning preserve adversarial robustness from pretraining to finetuning? *Advances in Neural Information Processing Systems (NeurIPS)*, 2021.

Gidaris, S., Singh, P., and Komodakis, N. Unsupervised representation learning by predicting image rotations. *arXiv preprint arXiv:1803.07728*, 2018.

Gowal, S., Qin, C., Uesato, J., Mann, T., and Kohli, P. Uncovering the limits of adversarial training against norm-bounded adversarial examples. *arXiv preprint arXiv:2010.03593*, 2020.

He, K., Zhang, X., Ren, S., and Sun, J. Deep residual learning for image recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016.

He, K., Fan, H., Wu, Y., Xie, S., and Girshick, R. Momentum contrast for unsupervised visual representation learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020.

Hinton, G., Vinyals, O., and Dean, J. Distilling the knowledge in a neural network. *arXiv preprint arXiv:1503.02531*, 2015.

Jiang, Z., Chen, T., Chen, T., and Wang, Z. Robust pre-training by adversarial contrastive learning. *Advances in Neural Information Processing Systems (NeurIPS)*, 2020.

Kim, M., Tack, J., and Hwang, S. J. Adversarial self-supervised contrastive learning. *Advances in Neural Information Processing Systems (NeurIPS)*, 2020.

Krizhevsky, A. et al. Learning multiple layers of features from tiny images. 2009.

Kumar, A., Raghunathan, A., Jones, R., Ma, T., and Liang, P. Fine-tuning can distort pretrained features and underperform out-of-distribution. *arXiv preprint arXiv:2202.10054*, 2022.

Luo, R., Wang, Y., and Wang, Y. Rethinking the effect of data augmentation in adversarial contrastive learning. In *International Conference on Learning Representations (ICLR)*, 2023.

Madry, A., Makelov, A., Schmidt, L., Dimitris, T., and Vladu, A. Towards deep learning models resistant to adversarial attacks. In *International Conference on Learning Representations (ICLR)*, 2018.

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

Pang, T., Yang, X., Dong, Y., Su, H., and Zhu, J. Bag of tricks for adversarial training. *International Conference on Learning Representations (ICLR)*, 2021.

Papernot, N., McDaniel, P., Goodfellow, I., Jha, S., Celik, Z. B., and Swami, A. Practical black-box attacks against machine learning. In *Proceedings of the ACM Asia Conference on Computer and Communications Security (ACM ASIACCS)*, 2017.

Rice, L., Wong, E., and Kolter, J. Z. Overfitting in adversarially robust deep learning. In *International Conference on Machine Learning (ICML)*, 2020.

Sriramanan, G., Addepalli, S., Baburaj, A., and Venkatesh Babu, R. Guided Adversarial Attack for Evaluating and Enhancing Adversarial Defenses. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2020.

Tramèr, F., Kurakin, A., Papernot, N., Goodfellow, I., Boneh, D., and McDaniel, P. Ensemble adversarial training: Attacks and defenses. In *International Conference on Learning Representations (ICLR)*, 2018.

Trinh, T. H., Luong, M.-T., and Le, Q. V. Selfie: Self-supervised pretraining for image embedding. *arXiv preprint arXiv:1906.02940*, 2019.

Van den Oord, A., Li, Y., and Vinyals, O. Representation learning with contrastive predictive coding. *arXiv e-prints*, pp. arXiv–1807, 2018.

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

Zhang, C., Zhang, K., Zhang, C., Niu, A., Feng, J., Yoo, C. D., and Kweon, I. S. Decoupled adversarial contrastive learning for self-supervised adversarial robustness. In *European Conference on Computer Vision (ECCV)*, 2022.

Zhang, H., Yu, Y., Jiao, J., Xing, E., El Ghaoui, L., and Jordan, M. I. Theoretically principled trade-off between robustness and accuracy. In *International Conference on Machine Learning (ICML)*, 2019.

Appendix

A. Intuition behind the proposed Hypotheses

In this section, we justify the intuition behind the hypotheses presented in Section-3.1, which is restated below:

Student model performance improves during distillation by matching the following:

1. *Training objectives of the teacher and the ideal goals of the student,*
2. *Pretraining and linear probe training objectives of the student.*

Hypothesis-1: Consider task-A to be the teacher's training task, and task-B to be the student's downstream task or its ideal goal. The representations in deeper layers (last few layers) of the teacher are more tuned to its training objective, and the early layers contain a lot more information than what is needed for this task (Bordes et al., 2023). Thus, features specific to task-A are dominant or replicated in the final feature layer, and other features that may be relevant to task-B are sparse. When a similarity based distillation loss is enforced on such features, higher importance is given to matching the replicated features, and the sparse features which may be important for task-B are suppressed further in the student (Addepalli et al., 2023). On the other hand, when the student's task matches with the teacher's task, a similarity based distillation loss is very effective in transferring the necessary representations to the student, since they are predominant. Thus, matching the training objective of the teacher with the ideal goals of the student improves downstream performance.

Hypothesis-2: For a given network, aligning the pretraining task with downstream task results in better performance since the matching of tasks ensures that the required features are predominant, and they are easily used by an SVM classifier (or a linear classifier) trained over it (Addepalli et al., 2023). In context of distillation, since the features of the student are trained by enforcing similarity based loss w.r.t. the teacher, we hypothesize that enforcing similarity w.r.t. the teacher is the best way to learn the student classifier as well. To illustrate this, we consider task-A to be the teacher pretraining task, and task-B to be the downstream task or ideal goal of the student. As discussed above, the teacher's features are aligned to task-A and these are transferred effectively to the student. The features related to task-B are suppressed in the teacher and are further suppressed in the student. As the features specific to a given task become more sparse, it is harder for an SVM classifier (or a linear classifier) to rely on that feature, although it important for classification (Addepalli et al., 2023). Thus, training a linear classifier for task-B is more effective on the teacher when compared to the student. The linear classifier of the teacher in effect amplifies the sparse features, allowing the student to learn them more effectively. Thus, training a classifier on the teacher and distilling it to the student is better than training a classifier directly on the student.

B. Mechanism behind Scaling to Larger Datasets

For a sufficiently complex task, a scalable approach results in better performance on larger models given enough data. Although the task complexity of adversarial self-supervised learning is high, the gains in prior approaches are marginal with an increase in model size, while the proposed method results in significantly improved performance on larger capacity models (Table 1). We discuss the key factors that result in better scalability below:

- As discussed in Section 3.1, a mismatch between training objectives of the teacher and ideal goals of the student causes a drop in student performance. This primarily happens because of the overfitting to the teacher training task. As model size increases, the extent of overfitting increases. The use of a projection layer during distillation alleviates the impact of this overfitting and allows the student to retain more generic features that are useful for the downstream robust classification objective. Thus, a projection layer is more important for larger model capacities where the extent of overfitting is higher.
- Secondly, as the model size increases, there is a need for higher amount of training data for achieving better generalization. The proposed method has better data diversity as it enables the use of more complex data augmentations in adversarial training by leveraging supervision from weak augmentations at the teacher.

C. Details on Training and Compute

Training Details: The self-supervised training of the teacher model is performed for 1000 epochs with the SimCLR algorithm (Chen et al., 2020a) similar to prior work (Zhang et al., 2022). We utilize the solo-learn repository [1] for this

[1] https://github.com/vturrisi/solo-learn

Table 5. **Total number of Forward (FP) or Backward (BP) Propagations during training** of the proposed approach when compared to prior works. Distillation based approaches - ProFeAT and DeACL require significantly lesser compute when compared to prior methods, and are only more expensive than supervised adversarial training.

Method	#epochs	#attack steps	#FP or BP for AT	#FP or BP for auxiliary model	Total #FP or BP
Supervised (TRADES)	110	10	1210	0	1210
AP-DPE	450	10	4950	0	4950
RoCL	1000	5	6000	0	6000
ACL	1000	5	12000	0	12000
AdvCL	1000	5	12000	1000	13000
DynACL	1000	5	12000	0	12000
DynACL++	1025	5	12300	0	12300
DeACL	100	5	700	1000	1700
ProFeAT (Ours)	100	5	800	1000	1800

Table 6. **Floating Point Operations per Second (GFLOPS) and latency per epoch during training** of the proposed approach ProFeAT when compared to the baseline DeACL for ResNet-18 and WideResNet-34-10 models. The computational overhead during training is marginal with the addition of the projection layer, and reduces further for larger capacity models.

	ResNet-18			WideResNet-34-10		
	GFLOPS	Time/epoch	#params (M)	GFLOPS	Time/epoch	#params (M)
DeACL	671827	51s	11.27	6339652	4m 50s	46.28
ProFeAT (Ours)	672200	51s	11.76	6340197	4m 50s	46.86
% increase	0.056	0.00	4.35	0.009	0.00	1.25

purpose. For the SimCLR SSL training, we tune and use a learning rate of 1.5 with SGD optimizer, a cosine schedule with warmup, weight decay of $1e{-5}$ and train the backbone for 1000 epochs with other hyperparameters kept as default as in the repository. The self-supervised adversarial training of the feature extractor using the proposed approach is performed for 100 epochs using SGD optimizer with a weight decay of $3e{-4}$, cosine learning rate with 10 epochs of warm-up, and a maximum learning rate of 0.5. We fix the value of β, the robustness-accuracy trade-off parameter (Ref: Equations (1) and (2) in the main paper) to 8 in all our experiments, unless specified otherwise.

Details on Linear Probing: To evaluate the performance of the learned representations, we perform standard linear probing by freezing the adversarially pretrained backbone as discussed in Section 4 of the main paper. We use a class-balanced validation split consisting of 1000 images from the train set and perform early-stopping during training based on the performance on the validation set. The training is performed for 25 epochs with a step learning rate schedule where the maximum learning rate is decayed by a factor of 10 at epoch 15 and 20. The learning rate is chosen amongst the following settings - $\{0.1, 0.05, 0.1, 0.5, 1, 5\}$ with SGD optimizer, and the weight decay is fixed to $2e{-4}$. The same evaluation protocol is used for the best baseline - DeACL (Zhang et al., 2022) as well as the proposed approach, for both in-domain and transfer learning settings.

Compute: The following Nvidia GPUs have been used for performing the experiments reported in this work - V100, A100, and A6000. Each of the experiments are run either on a single GPU, or across 2 GPUs based on the complexity of the run and GPU availability. For 100 epochs of single-precision (FP32) training with a batch size of 256, the proposed approach takes ~ 8 hours and ~ 16GB of GPU memory on a single A100 GPU for WideResNet-34-10 model on CIFAR-100.

D. Computational Complexity

In terms of compute, both the proposed method ProFeAT and DeACL (Zhang et al., 2022) lower the overall computational cost when compared to prior approaches. This is because self-supervised training in general requires larger number of epochs (1000) to converge (Chen et al., 2020a; He et al., 2020) when compared to supervised learning (≤ 100). Prior approaches like RoCL (Kim et al., 2020), ACL (Jiang et al., 2020) and AdvCL (Fan et al., 2021) combine the contrastive training objective of SSL approaches and the adversarial training objective. Thus, these methods require larger number of

Table 7. **Role of projector in self-supervised distillation (CIFAR-100, WRN-34-10):** The drop in accuracy of student $\mathcal{S}$ w.r.t. the teacher $\mathcal{T}$ indicates distillation performance, which improves by matching the training objective of the teacher with ideal goals of the student (S3/ S4 vs. S1), and by using similar losses for pretraining and linear probing (LP) (S2 vs. S1). Using a projector improves performance in case of mismatch in the above (S5 vs. S1). The similarity between teacher and student is significantly higher at the projector space when compared to the feature space in S5.

Exp #	Teacher training	Teacher acc (%)	Projector	LP Loss	Student accuracy after linear probe		$\cos(\mathcal{T},\mathcal{S})$	
					Feature space (%)	Projector space (%)	Feature space	Projector space
S1	Self-supervised	70.85	Absent	CE	64.90	-	0.94	-
S2	Self-supervised	70.85	Absent	$\cos(\mathcal{T},\mathcal{S})$	68.49	-	0.94	-
S3	Supervised	80.86	Absent	CE	80.40	-	0.94	-
S4	Supervised	69.96	Absent	CE	71.73	-	0.98	
S5	Self-supervised	70.85	Present	CE	73.14	64.67	0.19	0.92

Table 8. **Role of projector in self-supervised adversarial distillation (CIFAR-100, WRN-34-10):** Student performance after linear probe at feature space is reported. The drop in standard accuracy (SA) of the student ($\mathcal{S}$) w.r.t. the teacher ($\mathcal{T}$), and the robust accuracy (RA-G) of the student improve by matching the training objective of the teacher with ideal goals of the student (A3 vs. A1), and by using similar losses for pretraining and linear probing (LP) (A2 vs. A1). Using a projector improves performance in case of mismatch in the above (A4 vs. A1).

Exp #	Teacher training	Teacher accuracy		Projector	LP Loss	Student accuracy		$\cos(\mathcal{T},\mathcal{S})$
		SA (%)	RA-G (%)			SA (%)	RA-G (%)	
A1	Self-supervised (standard training)	70.85	0	Absent	CE	50.71	24.63	0.78
A2	Self-supervised (standard training)	70.85	0	Absent	$\cos(\mathcal{T},\mathcal{S})$	54.48	23.20	0.78
A3	Supervised (TRADES adversarial training)	59.88	25.89	Absent	CE	54.86	27.17	0.94
A4	Self-supervised (standard training)	70.85	0	Present	CE	57.51	24.10	0.18

training epochs (1000) for the adversarial training task, which is already computationally expensive due to the requirement of generating multi-step attacks during training. ProFeAT and DeACL use a SSL teacher for training and thus, the adversarial training is more similar to supervised training, requiring only 100 epochs. In Table 5, we present the approximate number of forward and backward propagations for each algorithm, considering both pretraining of the auxiliary network used and the training of the main network. It can be noted that the distillation based approaches - ProFeAT and DeACL require significantly lesser compute when compared to prior methods, and are only more expensive than supervised adversarial training. In Table 6, we present the FLOPS required during training for the proposed approach and DeACL. One can observe that there is a negligible increase in FLOPS compared to the baseline approach.

E. Additional Results

E.1. Empirical justification of our hypothesis on self-supervised distillation

We now empirically justify the hypothesis proposed in Section 3.1 by considering several distillation settings involving standard and adversarial, supervised and self-supervised trained teacher models in Tables 7 and 8. The results are presented on CIFAR-100 with WideResNet-34-10 architecture for both teacher and student. The standard self-supervised model is trained using SimCLR (Chen et al., 2020a). Contrary to a typical knowledge distillation setting where a cross-entropy loss is also used (Hinton et al., 2015), all the experiments presented involve the use of only self-supervised losses for distillation (cosine similarity between representations), and labels are used only during linear probing. Adversarial self-supervised distillation in Table 8 is performed using a combination of distillation loss on natural samples and smoothness loss on adversarial samples as shown in Equation (2). A randomly initialized trainable projector is used at the output of student backbone in S5 of Table 7 and A4 of Table 8. Here, the training loss is considered in the projected space of the student $\mathcal{S}_p$ rather than the feature space $\mathcal{S}_f$.

1. Matching the training objectives of teacher with the ideal goals of the student: We first consider the standard training of a student model, using either a self-supervised or supervised teacher in Table 7. In the absence of a projector, the drop in student accuracy w.r.t. the respective teacher accuracy is 6% with a self-supervised teacher (S1), and $< 0.5\%$ with a supervised teacher (S3). To ensure that our observations are not a result of the 10% difference in teacher accuracy between S1 and S3, we present results and similar observations with a supervised sub-optimally trained teacher in S4. Thus, a supervised teacher is significantly better than a self-supervised teacher for distilling representations specific to a given task,

Table 9. **Additional evaluation on pretrained models:** Performance (%) of DeACL (best baseline) and ProFeAT (Ours) on CIFAR-10 and CIFAR-100 datasets with WideResNet-34-10 architecture. The model is first pretrained using the respective self-supervised adversarial training algorithm, and further we compute the standard accuracy (SA) and robust accuracy against GAMA (RA-G) using several methods such as standard linear probing (LP), training a 2 layer MLP head (MLP), and performing KNN in the feature space (k=10). The proposed method achieves improvements over the baseline across all evaluation methods.

Method	LP Eval		MLP Eval		KNN Eval	
	SA	RA-G	SA	RA-G	SA	RA-G
CIFAR-10						
DeACL	83.60	49.62	85.66	48.74	87.00	54.58
ProFeAT	**87.44**	**52.24**	**89.37**	**50.00**	**87.38**	**55.77**
CIFAR-100						
DeACL	52.90	24.66	55.05	22.04	56.82	31.26
ProFeAT	**61.05**	**27.41**	**63.81**	**26.10**	**58.09**	**32.26**

justifying the above hypothesis. We next consider adversarial training of a student, using either a standard self-supervised teacher, or a supervised adversarially trained teacher (TRADES) in Table 8. Since the TRADES model is more aligned with the ideal goals of the student, despite its lower clean accuracy, the clean and robust accuracy of the student are better than those obtained using a standard self-supervised model as a teacher (A3 vs. A1). This further justifies the first hypothesis.

2. Matching the pretraining and linear probe training objectives of the student: To align pretraining with linear probing, we perform linear probing on the teacher model, and further train the student by maximizing the cosine similarity between the logits of the teacher and student. This boosts the student accuracy by 3.6%, in Table 7 (S2 vs. S1) and by 3.8% in Table 8 (A2 vs. A1).

The projector isolates the representations of the student from the training loss, as indicated by the lower similarity between the student and teacher at feature space when compared to that at the projector (in S5 and A4), and prevents overfitting of the student to the teacher training objective. This makes the student robust to the misalignment between the teacher training objective and ideal goals of the student, and also to the mismatch in student pretraining and linear probing objectives, thereby improving student performance, as seen in Table 7 (S5 vs. S1) and Table 8 (A4 vs. A1).

E.2. Addition evaluation on self-supervised trained models

We compare the performance of DeACL (best baseline) and ProFeAT (Ours) on CIFAR-10 and CIFAR-100 datasets with WideResNet-34-10 architecture in Table 9. The model is first pretrained using the respective self-supervised adversarial training algorithm, and further we compute the standard accuracy (SA) and robust accuracy against GAMA (RA-G) using several methods such as standard linear probing (LP), training a 2 layer MLP head (MLP), and performing KNN in the feature space (k=10). We note that the proposed method achieves improvements over the baseline across all evaluation methods. Since the training of classifier head in LP and MLP is done using standard training and not adversarial training, the robust accuracy reduces as the number of layers increases (from linear to 2-layers), and the standard accuracy improves. The standard accuracy of KNN is better than the standard accuracy of LP for the baseline, indicating that the representations are not linearly separable. Whereas, as is standard, for the proposed approach, LP standard accuracy is higher than that obtained using KNN. The adversarial attack used for evaluating the robust accuracy using KNN is generated using GAMA attack on a linear classifier. The attack is suboptimal since it is not generated by using the evaluation process (KNN), and thus the robust accuracy against such an attack is higher.

E.3. Transfer Learning with Adversarial Full-Finetuning

We present transfer learning results using lightweight adversarial full finetuning (AFF) to STL-10 and Caltech-101, in Table 10. Caltech-101 contains 101 object classes and 1 background class, with 2416 samples in the train set and 6728 samples in the test set. The number of samples per class range from 17 to 30, and thus this is a suitable dataset to highlight the practical importance of adversarial self-supervised pretrained representations for low-data regime. Towards this, a base robustly pretrained model is finetuned using the TRADES adversarial training for 25 epochs. We present results for WideResNet-34-10 models that are pretrained on CIFAR-10/100 respectively. We note that the proposed method

Table 10. **Transfer Learning to STL-10 and Caltech-101:** Transfer learning performance (%) with adversarial full finetuning (AFF) using TRADES (Zhang et al., 2019) algorithm for 25 epochs, from CIFAR-10 and CIFAR-100 to STL-10 and Caltech-101 datasets on WideResNet-34-10 architecture. The proposed method outperforms both DeACL and the supervised trained model. Standard Accuracy (SA), robustness against PGD-20 (RA-PGD20) and GAMA (RA-G) are reported.

Method	SA	RA-PGD20	RA-G	SA	RA-PGD20	RA-G
	CIFAR-10 → STL10			CIFAR-100 → STL10		
Supervised (TRADES)	64.58	39.83	32.78	64.22	34.20	31.01
DeACL	61.65	31.88	28.34	60.89	33.06	30.00
ProFeAT	**74.12**	40.15	**36.04**	**68.77**	35.35	**31.23**
	CIFAR-10 → Caltech-101			CIFAR-100 → Caltech-101		
Supervised (TRADES)	62.46	40.77	39.40	**64.97**	42.95	41.02
DeACL	62.65	41.39	39.18	61.01	40.56	39.09
ProFeAT	**66.11**	45.29	**42.12**	64.16	42.95	**41.25**

Table 11. **Ablation on Projector training configurations (CIFAR-100, WRN-34-10):** Performance (%) using variations in projector (proj.) initialization (init.) and trainability. SA: Standard Accuracy, RA-G: Robust accuracy against GAMA, RA-PGD20: Robust Accuracy against PGD-20 attack.

Ablation	Student proj.	Proj. init. (Student)	Teacher proj	Proj. init. (Teacher)	SA	RA-PGD20	RA-G
AP1	Absent	-	Absent	-	55.35	35.89	**27.86**
AP2	Trainable	Random	Absent	-	**63.07**	32.05	26.57
AP3	Frozen	Pretrained	Absent	-	40.43	27.51	22.23
AP4	Trainable	Pretrained	Absent	-	62.89	31.97	26.57
AP5	Trainable	Random (common)	Trainable	Random (common)	53.43	35.58	27.23
AP6	Trainable	Pretrained (common)	Trainable	Pretrained (common)	54.60	36.10	27.41
AP7	Trainable	Pretrained	Frozen	Pretrained	58.18	35.26	27.73
Ours	Frozen	Pretrained	Frozen	Pretrained	61.05	31.99	27.41

outperforms DeACL by a large margin. Further, we note that by using merely 25 epochs of AFF, the proposed method achieves improvements of around 4% on CIFAR-10 and 11% on CIFAR-100 when compared to the linear probing accuracy presented in Table 2, highlighting the practical utility of the proposed method. The AFF performance of the proposed approach is better than that of a supervised TRADES pretrained model as well.

E.4. Training configuration of the Projector

We present ablations using different configurations of the projection layer in Table 11. As discussed in Appendix E.1, we observe a large boost in clean accuracy when a random (or pretrained) trainable projection layer is introduced to the student (AP2/ AP4 vs. AP1). While the use of pretrained frozen projection head only for the student degrades performance considerably (AP3), the use of the same for both teacher and student (Ours) yields a optimal robustness-accuracy trade-off across all variations. The use of a common trainable projection head for both teacher and student results in collapsed representations at the projector output (AP5, AP6), yielding results similar to the case where projector is not used for both teacher and student (AP1). This issue is overcome when the pretrained projector is trainable only for the student (AP7).

E.5. Architecture of the Projector

In the proposed approach, we use the following 2-layer MLP projection head for both self-supervised pretraining of the teacher and adversarial training of the student: (1) **ResNet-18: 512-512-256**, and (2) **WideResNet-34-10: 640-640-256**. In Table 12, we present results using different configurations and architectures of the projector. Firstly, the use of a linear projector (APA2) is similar to the case where projector is not used for student training (APA1), with $\sim 21\%$ drop in clean accuracy of the student with respect to the teacher. This improves to $12 - 17\%$ when a non-linear projector is introduced (APA3-APA6 and Ours). The use of a 2-layer MLP (Ours) is marginally better than the use of a 3-layer MLP (APA3) in terms of clean accuracy of the student. The accuracy of the student is stable across different architectures of the projector (Ours, APA4, APA5). However, the use of a bottleneck architecture (APA6) results in a higher drop in clean accuracy of the student.

Table 12. **Ablation on Projector Configuration and Architecture (CIFAR-100, WRN-34-10):** Performance (%) obtained by varying the projector configuration (config.) and architecture (arch.). A non-linear projector effectively reduces the gap in clean accuracy between the teacher and student. A bottleneck architecture for the projector is worse than other variants. SA: Standard Accuracy, RA-PGD20: Robust Accuracy against PGD-20 attack, RA-G: Robust Accuracy against GAMA.

Ablation	Projector config.	Projector arch.	Teacher SA	Student SA	Drop in SA	%Drop in SA	RA-PGD20	RA-G
APA1	No projector	-	70.85	55.35	15.50	21.88	35.89	**27.86**
APA2	Linear layer	640-256	68.08	53.35	14.73	21.64	35.57	27.47
Ours	2 Layer MLP	640-640-256	70.85	61.05	9.80	13.83	31.99	27.41
APA3	3 Layer MLP	640-640-640-256	70.71	60.37	10.34	14.62	31.44	27.37
APA4	2 Layer MLP	640-640-640	69.88	61.24	8.64	12.36	31.88	27.36
APA5	2 Layer MLP	640-2048-640	**70.96**	**61.76**	9.20	12.97	29.53	26.66
APA6	2 Layer MLP	640-256-640	69.37	57.87	11.50	16.58	34.53	27.56

Ablation	Teacher	Student	SA	RA-PGD20	RA-G
AG1	PC	PC	56.57	30.54	25.29
AG2	AuAu	AuAu	60.76	31.83	27.21
AG3	PC1	PC2	56.95	30.94	25.39
AG4	AuAu1	AuAu2	59.51	32.44	**28.15**
AG5	AuAu	PC	57.28	31.23	26.14
Ours	PC	AuAu	**61.05**	31.99	27.41

Table 13. **Ablation on Augmentations used (CIFAR-100, WRN-34-10):** Performance (%) using different augmentations for the teacher and student. (PC: Pad+Crop, AuAu: AutoAugment). Standard Accuracy (SA) and Robust accuracy against GAMA (RA-G), PGD-20 (RA-PGD20) reported.

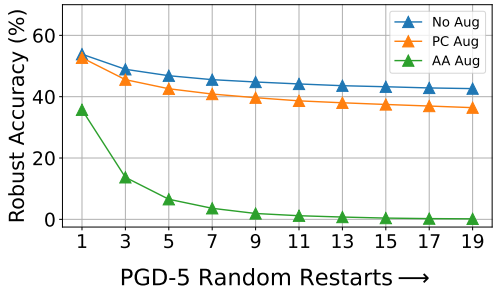

Figure 3. Robust accuracy of a supervised TRADES model across random restarts of PGD 5-step attack (CIFAR-100, WRN-34-10).

E.6. Augmentations for the Student and Teacher

We present ablation experiments to understand the impact of different augmentations used for the teacher and student separately in Table 13. The base method (AG1) uses common Pad and Crop (PC) augmentation for both teacher and student. By using more complex augmentations —AutoAugment followed by Pad and Crop (denoted as AuAu in the table), there is a significant improvement in both clean and robust accuracy. By using separate augmentations for the teacher and student, there is an improvement in the case of PC (AG3), but a drop in clean accuracy accompanied by better robustness in case of AuAu. Finally by using a mix of both AuAu and PC at the student and teacher respectively (Ours), we obtain improvements in both clean and robust accuracy, since the former improves attack diversity (shown in Figure 3), while the latter makes the training task easier.

Table 14. **Ablation on Attack Loss (CIFAR-100, WRN-34-10):** Performance (%) with variations in attack loss at feature (feat.) and projector (proj.). While the proposed defense is stable across several variations in the attack loss, minimizing a combination of both losses $\cos(\mathcal{T},\mathcal{S})$ and $\cos(\mathcal{S},\mathcal{S})$ gives the best robustness-accuracy trade-off. SA: Standard Accuracy, RA-PGD20: Robust Accuracy against PGD-20 attack, RA-G: Robust Accuracy against GAMA.

Ablation	Attack @ feat.	Attack @ proj.	SA	RA-PGD20	RA-G
AT1	$\cos(\mathcal{T},\mathcal{S})$	$\cos(\mathcal{T},\mathcal{S})$	60.84	31.41	26.78
AT2	$\cos(\mathcal{S},\mathcal{S})$	$\cos(\mathcal{S},\mathcal{S})$	61.30	31.86	26.75
AT3	$\cos(\mathcal{T},\mathcal{S})$	$\cos(\mathcal{S},\mathcal{S})$	60.69	32.34	**27.44**
AT4	$\cos(\mathcal{S},\mathcal{S})$	-	61.62	31.69	26.62
AT5	-	$\cos(\mathcal{S},\mathcal{S})$	61.09	31.78	27.00
AT6	$\cos(\mathcal{T},\mathcal{S})$	-	**62.01**	31.62	26.89
AT7	-	$\cos(\mathcal{T},\mathcal{S})$	61.18	31.43	27.24
Ours	$\cos(\mathcal{S},\mathcal{S})$	$\cos(\mathcal{T},\mathcal{S})$	61.05	31.99	27.41

Table 15. **Ablation on Defense Loss (CIFAR-100, WRN-34-10):** Performance (%) with variations in training loss at feature (feat.) and projector (proj.). "clean" denotes the cosine similarity between representations of teacher and student on clean samples. "adv" denotes the cosine similarity between representations of the corresponding clean and adversarial samples either at the output of student $(\mathcal{S}, \mathcal{S})$ or between the teacher and student $(\mathcal{T}, \mathcal{S})$. SA: Standard Accuracy, RA-G: Robust accuracy against GAMA, RA-PGD20: Robust Accuracy against PGD-20 attack.

Ablation	Loss @ feat.	Loss @ proj.	SA	RA-PGD20	RA-G
AD1	clean + adv$(\mathcal{S}, \mathcal{S})$	-	55.35	35.89	**27.86**
AD2	-	clean + adv$(\mathcal{S}, \mathcal{S})$	59.65	33.03	26.90
AD3	clean + adv$(\mathcal{S}, \mathcal{S})$	clean	61.69	31.34	26.40
AD4	clean + adv$(\mathcal{S}, \mathcal{S})$	adv$(\mathcal{S}, \mathcal{S})$	49.59	31.79	25.35
AD5	adv$(\mathcal{S}, \mathcal{S})$	clean	59.72	3.77	1.38
AD6	adv$(\mathcal{S}, \mathcal{S})$	clean + adv$(\mathcal{S}, \mathcal{S})$	59.22	34.08	26.50
AD7	clean	clean + adv$(\mathcal{S}, \mathcal{S})$	62.24	30.55	25.97
AD8	clean + adv$(\mathcal{S}, \mathcal{S})$	clean + adv$(\mathcal{T}, \mathcal{S})$	63.85	29.97	23.91
AD9	clean + adv$(\mathcal{T}, \mathcal{S})$	clean + adv$(\mathcal{T}, \mathcal{S})$	**65.34**	27.75	22.40
Ours	clean + adv$(\mathcal{S}, \mathcal{S})$	clean + adv$(\mathcal{S}, \mathcal{S})$	61.05	31.99	27.41

E.7. Attack loss

For performing adversarial training using the proposed approach, attacks are generated by minimizing a combination of cosine similarity based losses as shown in Equation (4) of the main paper. This includes an unsupervised loss at the feature representations of the student and another loss between the representations of the teacher and student at the projector. As shown in Table 14, we obtain a better robustness-accuracy trade-off by using a combination of both losses rather than by using only one of the two losses, due to better diversity and strength of attack. These results also demonstrate that the proposed method is not very sensitive to different choices of attack losses.

E.8. Defense Loss

We present ablation experiments across variations in defense loss at the feature space and the projection head in Table 15. In the proposed approach (Ours), we introduce a combination of clean and robust losses at both feature and projector layers, as shown in Equation (3). By introducing the loss only at the features (AD1), there is a considerable drop in clean accuracy as seen earlier, which can be recovered by introducing the clean loss at the projection layer (AD3). Instead, when only the robust loss is introduced at the projection layer (AD4), there is a large drop in clean accuracy confirming that the need for projection layer is mainly enforcing the clean loss. When the combined loss is enforced only at the projection head (AD2), the accuracy is close to that of the proposed approach, with marginally lower clean and robust accuracy. Enforcing only adversarial loss in the feature space, and only clean loss in the projector space is a hard optimization problem, and this results in a non-robust model (AD5). As shown in Table 16, even by increasing β in AD5, we do not obtain a robust model, rather, there is a representation collapse. Thus, as discussed in Section 3, it is important to introduce the adversarial loss as a regularizer in the projector space as well (AD6). Enforcing only one of the two losses at the feature space (AD6 and AD7) also results in either inferior clean accuracy or robustness. Finally from AD8 and AD9 we note that the robustness loss is better when implemented as a smoothness constraint on the representations of the student, rather than by matching representations between the teacher and student. Overall, the proposed approach (Ours) results in the best robustness-accuracy trade-off.

E.9. Weighting of the Defense Loss at the feature and projector

In the proposed approach, the defense losses are equally weighted between the feature and projector layers as shown in Equation (3). In Figure 4, we present results by varying the weighting λ between the defense losses at the feature ($\mathcal{L}_f$) and projector ($\mathcal{L}_{fp}$) layers: $\mathcal{L}_{\text{ProFeAT}} = \lambda \cdot \mathcal{L}_f + (1 - \lambda) \cdot \mathcal{L}_{fp}$, where $\mathcal{L}_{fp}$ and $\mathcal{L}_f$ represent the overall defense losses at the projector and feature respectively (Equations (1) and (2)). It can be noted that the two extreme cases of $\lambda = 0$ and $\lambda = 1$ result in a drop in clean accuracy, with a larger drop in the case where the loss is enforced only at the feature layer. The robust accuracy shows lesser variation across different values of λ. Thus, the overall performance is stable over the range $\lambda \in [0.25, 0.75]$, making the default setting of $\lambda = 0.5$ a suitable option.

β	Standard Acc. (SA)	Robust Acc. (RA-G)
1	**67.34**	0.46
5	51.99	0.71
10	31.34	7.81
50	11.59	2.55
100	8.23	**2.61**

Table 16. **Failure of AD5 defense loss in Table 15:** Using clean and adversarial loss exclusively at projector and feature space respectively results in an unstable optimization problem. As shown above, a lower value of β results in a non-robust model, while higher β results in representational collapse.

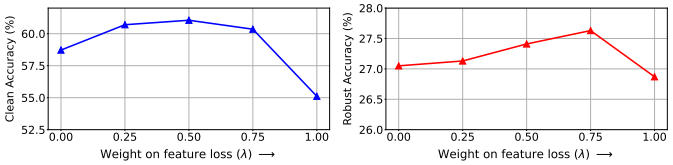

Figure 4. Performance (%) of the proposed approach ProFeAT by varying the weight between the defense losses at the feature and projector: $\mathcal{L}_{\text{ProFeAT}} = \lambda \cdot \mathcal{L}_f + (1-\lambda) \cdot \mathcal{L}_{fp}$, where $\mathcal{L}_{fp}$ and $\mathcal{L}_f$ represent the overall defense losses at the projector and feature respectively (Equations (1) and (2) of the main paper). The performance is stable across the range $\lambda \in [0.25, 0.75]$.

Table 17. **Ablation on the number of training epochs for the teacher SSL model (CIFAR-100, WRN-34-10):** Performance (%) obtained by varying the number of epochs for which the standard self-supervised teacher model is pretrained. Improvements in accuracy of the teacher result in corresponding gains in both standard and robust accuracy of the student. SA: Standard Accuracy, RA-PGD20: Robust Accuracy against PGD-20 attack, RA-G: Robust Accuracy against GAMA.

#epochs of PT	Teacher SA	Student SA	Drop in SA	%Drop in SA	RA-PGD20	RA-G
100	55.73	49.37	6.36	11.41	25.23	20.86
200	65.43	56.16	9.27	14.17	28.67	24.15
500	69.27	59.62	9.65	13.93	31.46	26.75
1000	**70.85**	**61.05**	9.80	13.83	31.99	**27.41**

E.10. Accuracy of the self-supervised teacher model

The self-supervised teacher model is obtained using 1000 epochs of SimCLR (Chen et al., 2020a) training in all our experiments. We now study the impact of training the teacher model for lesser number of epochs. As shown in Table 17, as the number of teacher training epochs reduces, there is a drop in the accuracy of the teacher, resulting in a corresponding drop in the clean and robust accuracy of the student model. Thus, the performance of the teacher is crucial for training a better student model.

E.11. Self-supervised training algorithm of the teacher

In the proposed approach, the teacher is trained using the popular self-supervised training algorithm SimCLR (Chen et al., 2020a), similar to prior works (Zhang et al., 2022). In this section, we study the impact of using different algorithms for the self-supervised training of the teacher and present results in Table 18. In order to ensure consistency across different SSL methods, we use a *random trainable* projector (2-layer MLP with both hidden and output dimensions of 640) for

Table 18. **Ablation on the algorithm used for training the self-supervised teacher model (CIFAR-100, WRN-34-10):** Performance (%) of the proposed approach by varying the pretraining algorithm of the teacher model. A random trainable projector is used for training the student model, to maintain uniformity in projector architecture across all methods. SA: Standard Accuracy, RA-PGD20: Robust Accuracy against PGD-20 attack, RA-G: Robust Accuracy against GAMA.

Method (Teacher training)	Teacher SA	Student SA	RA-PGD20	RA-G
SimCLR	67.98	62.20	31.31	26.13
SimCLR (tuned)	70.85	63.07	32.05	26.57
BYOL	**72.97**	63.19	31.63	**26.82**
Barlow Twins	67.74	60.69	29.46	24.48
SimSiam	68.60	63.46	32.06	26.69
MoCoV3	72.48	**65.57**	32.22	26.65
DINO	68.75	60.61	30.16	24.80

Table 19. **Ablation on number of attack steps used for adversarial training (CIFAR-100, WRN-34-10):** Performance (%) using lesser number of attack steps (2 steps) when compared to the standard case (5 steps) during adversarial training. Clean/ Standard Accuracy (SA) and robust accuracy against GAMA (RA-G) and AutoAttack(RA-AA) are reported. The proposed approach is stable at lower attack steps as well, while being better than both TRADES (Zhang et al., 2019) and DeACL (Zhang et al., 2022).

# attack steps	Supervised (TRADES)			DeACL			ProFeAT (Ours)		
	SA	RA-G	RA-AA	SA	RA-G	RA-AA	SA	RA-G	RA-AA
2	**60.80**	24.49	23.99	51.00	24.89	23.45	60.43	26.90	**26.23**
5	**61.05**	25.87	25.77	52.90	24.66	23.92	**61.05**	27.41	**26.89**

training the student and do not employ any projection head for the pretrained frozen teacher. While the default teacher trained using SimCLR was finetuned across hyperparameters, we utilize the default hyperparameters from the solo-learn[2] Github repository for this table, and thus present SimCLR also without tuning for a fair comparison. For uniformity, we report all results with $\beta = 8$ (the robustness-accuracy trade-off parameter). From Table 18, we note that in most cases, the clean accuracy of the student increases as the accuracy of the teacher improves, while the robust accuracy does not change much. We note that this table merely shows that the proposed approach can be effectively integrated with several base self-supervised learning algorithms for the teacher model. However, it does not present a fair comparison across different SSL pretraining algorithms, since the ranking on the final performance of the student would change if the pretraining SSL algorithms were used with appropriate hyperparameter tuning.

E.12. Improving the efficiency of the self-supervised adversarial training

Similar to prior works (Zhang et al., 2022), the proposed approach uses 5-step PGD based optimization for attack generation during adversarial training. In Table-19, we present results with lesser optimization steps (2 steps). The proposed approach is stable and obtains similar results even by using 2-step attack. Even in this case, the clean and robust accuracy of the proposed approach is significantly better than the baseline approach DeACL (Zhang et al., 2022), and also outperforms the supervised TRADES model (Zhang et al., 2019).

[2] https://github.com/vturrisi/solo-learn