Improving Adversarial Robustness via Activation Amplification and Attenuation

Taïga GONÇALVES[1], Yongsong HUANG[1], Tomo MIYAZAKI[1], and Shinichiro OMACHI[1]

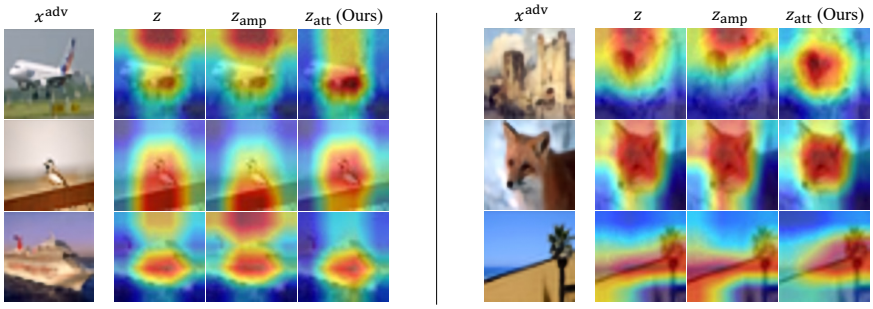
Graduate School of Engineering, Tohoku University, Japan
goncalves.taiga.teo.t6@dc.tohoku.ac.jp, huang.yongsong.c5@tohoku.ac.jp, tomo@tohoku.ac.jp, shinichiro.omachi.b5@tohoku.ac.jp

Abstract. The existence of adversarial attacks is often attributed to the presence of non-robust features in neural networks. While prior defenses reduce their impact via pruning, masking, or feature recalibration, we instead propose to jointly learn to amplify and attenuate these signals through a simple activation scaling mechanism. To this end, we introduce **Activation Amplification and Attenuation (A3)**, a lightweight plug-in module that enhances adversarial robustness with minimal modifications of the activations. A3 dynamically rescales the activations using a learnable mask and a scaling factor derived from the original activation magnitudes. The influence of adversarial perturbations can be amplified or attenuated using the same learnable parameters by simply flipping the sign of the scaling operation. The amplified signals serve as negative references to construct novel contrastive and ranking loss functions. Experimental analysis shows that learning to degrade the predictions in amplification mode simultaneously improves adversarial robustness in attenuation mode. Moreover, A3 relies on only a small number of learnable parameters, with most of its behavior being determined by the scaling mechanism rather than additional network capacity. Extensive experiments demonstrate that integrating A3 into different backbones, datasets, and training methods consistently improves adversarial robustness while introducing negligible computational and memory overhead compared to existing plug-in modules. Code is available at: https://github.com/tgoncalv/A3.

Keywords: Adversarial Attacks · Adversarial Robustness · Non-Robust Features

1 Introduction

The increasing success of Deep Neural Networks (DNNs) across a wide range of applications raises concerns about their security and reliability. In particular, adversarial examples [12, 24] add imperceptible perturbations specifically designed to mislead the models' predictions. To address this vulnerability, numerous works



(a) CIFAR-10 **(b)** CIFAR-100

Fig. 1: Grad-CAM attention maps [26] on adversarial examples. We compute Grad-CAM at *block* 4 of vanilla ResNet-18 (before inserting A3) using the activations z, and compare it with Grad-CAM obtained from A3 in amplification (z_{amp}) and attenuation (z_{att}) modes. The amplification mode tends to emphasize class-irrelevant regions (*e.g.* background sky rather than the airplane/ship), while the attenuation mode reduces the contribution of these regions and focuses more on class-relevant areas. During training, we use z_{amp} as a negative reference to construct novel contrastive and ranking losses, which contributes to the robustness improvements observed in the attenuation mode.

have proposed defense methods to improve the adversarial robustness of DNNs, together with standardized evaluation protocols for assessing their robustness under various types of attacks [5, 6].

Current defense techniques can be broadly divided into two categories. The first group focuses on training strategies [29, 31, 38], which use adversarial examples to introduce new objective functions and regularization terms without modifying the network architecture. The second group directly modifies internal feature representations [3, 18, 30] by inserting lightweight plug-in modules into the backbone, making the network inherently more robust.

In practice, these plug-in modules generally follow the same design principles. **(i) Adaptability:** the module should be compatible with different model architectures and training strategies. **(ii) Efficiency:** the computational and memory overhead should be minimal. **(iii) Effectiveness:** the accuracy on adversarial examples must improve compared to the vanilla backbone. **(iv) Reliability:** the module must remain differentiable and interpretable to avoid reliability issues such as gradient obfuscation [1, 28].

Some plug-in modules compute a mask to identify and suppress non-robust features [2, 36]. More recent methods [11, 18] argued that completely discarding non-robust features is suboptimal, as they may still contain useful information. Therefore, they propose to separate the features into robust and non-robust components, and recalibrate the latter using additional multi-layer perceptrons (MLPs). While effective, these recalibration blocks mainly introduce extra learnable parameters whose contributions to robustness are less interpretable, since

the architecture of these MLPs is fundamentally not much different from the other components of the original backbone.

To this end, we propose **Activation Amplification and Attenuation (A3)**, a lightweight plug-in module designed to strengthen internal feature representations via a simple rescaling mechanism. Our design is inspired by recent advances in Out-of-Distribution (OOD) detection [9,35], which found that appropriately scaling the original activations can significantly improve the separability between In-Distribution (ID) and OOD samples. Since adversarial examples can be viewed as a form of OOD data, we propose a scaling operation specifically designed to improve adversarial robustness. A3 computes a sample-wise scaling factor using a learnable masking function and the distribution of the original activation magnitudes, so that different channels are rescaled with different strengths. This controlled rescaling encourages a clearer separation between useful and less useful predictive signals, thereby focusing robustness improvements on useful cues, as shown in Fig. 1. Moreover, unlike traditional approaches that only suppress or down-weight the influence of non-robust features, A3 supports both amplification and attenuation of these signals using the same learnable parameters by simply flipping the sign of the scaling operation. We then exploit the amplified signals as training-time negative references to define novel contrastive and ranking loss functions. Our contributions can be summarized as follows:

- We propose **A3**, a lightweight plug-in module that can be integrated into different architectures to improve adversarial robustness with minimal computational and memory overhead.
- We design a simple scaling mechanism that can either amplify or attenuate activation patterns to focus robustness improvements on useful predictive signals. We further introduce novel contrastive and ranking loss functions that exploit the amplified activations as negative references to enhance the attenuation mode.
- We show that A3 consistently improves adversarial robustness across models, datasets, and adversarial training methods compared to prior plug-in defense modules. Extensive experiments and ablation studies highlight the importance of both amplification and attenuation modes.

2 Related Work

2.1 Adversarial Training

The most widely used strategy to enhance model robustness is adversarial training (AT) [12, 24], formulated as the following min-max optimization problem:

$$\min_{\theta} \mathbb{E}_{(x,y)\sim D} \left[\max_{\delta \in S} L(f_\theta(x+\delta), y) \right], \tag{1}$$

where f_θ is a neural network with parameters θ, L is a loss function, D is the dataset, and S is the set of allowed perturbations, typically defined as an ℓ_p-norm constraint around the clean input x. The inner maximization aims to find

the optimal perturbation δ that maximizes the training loss under the constraint set S to induce misclassification. The perturbation is generally obtained using gradient-based attacks such as the Fast Gradient Sign Method (FGSM) [12] or the Projected Gradient Descent (PGD) [24]. The outer minimization then updates the model parameters θ to minimize the loss on these adversarial examples.

Many extensions have been proposed to improve the standard AT strategy. TRADES [38] and MART [29] combine both clean and adversarial examples in the outer minimization process to improve model robustness while maintaining high accuracy on clean data. AGR [27] further refines this trade-off by comparing clean and adversarial examples in the gradient space rather than the input space. Other works strengthen the inner maximization to generate more effective adversarial examples that better guide the model to learn robust features. For example, LAS-AT [17] dynamically adjusts the attack strength on a per-sample basis, while CFA [32] adapts attack configurations in a class-wise manner. Finally, Yu *et al*. [21] show that using hard ground-truth labels often lead to robust overfitting and propose Self-Guided Label Smoothing (SGLS), which uses self-distillation to generate soft labels that improve inference-time robustness.

2.2 Robust Feature Manipulation

Another line of work introduces plug-in modules that can be integrated into different architectures to improve the robustness of intermediate features. Xie *et al*. [34] propose a simple feature denoising block based on non-local filtering and a 1×1 convolution to reduce adversarial noise while preserving meaningful signals. FPCM [3] reconfigures the low and high-frequency feature components based on the observation that adversarial perturbations often manifest in the high-frequency domain. EMFF [30] leverages evidential deep learning to estimate feature uncertainty across multiple layers and uses this information to refine the learned representations.

Recent works also focus on identifying non-robust features and suppressing their influence. CAS [2] and CIFS [36] observed that such features are often concentrated in specific channels and proposed modules that identify and suppress these channels on a per-sample basis. Later, Kim *et al*. [18] argued that completely discarding non-robust features is a suboptimal solution as they may still contain useful information. Therefore, they introduced the Feature Separation and Recalibration (FSR) module, which separates robust and non-robust components and recalibrates the latter to reduce adversarial noise while preserving useful cues. FTA2C [11] further refines this recalibration process by enhancing feature alignment between clean and adversarial examples.

2.3 Activation Scaling in OOD Detection

Out-of-distribution (OOD) detection aims to identify inputs that do not belong to the training distribution. Many approaches quantify how likely an input is to be OOD using scoring functions $S(z) : \mathcal{Z} \to \mathbb{R}$ [20, 22, 23] based on the original features or logits. A widely used function is the energy score [23]:

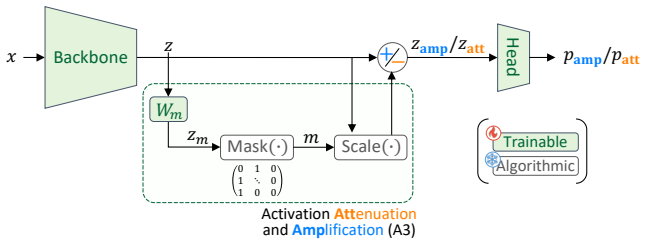


Fig. 2: Overview of the proposed A3 module. A3 is a plug-in module that can be integrated into various models. The activations z are processed using a small learnable projection W_m and a Gumbel-Softmax Mask(·) operator to produce a channel-wise mask. The resulting mask and activation magnitudes are then passed to Scale(·) to compute a rescaling factor used to either amplify (+) or attenuate (−) the activations depending on the sign of the operation. During training, z_{amp} serves as a negative reference while z_{att} serves as a positive reference to define novel contrastive and ranking loss functions. During inference, the final predictions are obtained solely from z_{att}.

$$S_{energy}(z) = -T \cdot \log \sum_{i=1}^{C} e^{z_i/T}, \tag{2}$$

where T is a temperature hyperparameter, C is the number of channels, and z is a C-dimensional vector at the penultimate layer of the model. This function assigns lower energy scores for in-distribution (ID) samples compared to OOD samples, which can then be separated using a simple threshold value.

Recent works have shown that rescaling part of the activations before applying the energy-based scoring can significantly improve the energy gap between ID and OOD data. ASH [9] proposed to use a top-k operation to selectively prune low-magnitude activations while amplifying the remaining ones. Then, SCALE [35] demonstrated that the pruning step yields suboptimal results and that scaling all activations with a carefully designed scaling factor leads to better performance. Later, LTS [10] and AdaSCALE [25] further enhanced the performance of SCALE while preserving the core scaling principle.

3 Method

3.1 Activation Scaling

Preliminaries. OOD detection methods based on activation scaling [9,10,25,35] employ a top-k operation to compute a scaling factor defined as:

$$\text{Scale}_{\text{OOD}}(z, p) = exp\left(\frac{\sum_i z_i}{\sum_{z_i > P_p(z)} z_i}\right), \tag{3}$$

where $P_p(z)$ denotes the p-th percentile of the activations z. These activations are typically extracted from the penultimate layer of the model, after a ReLU

operation. Therefore, all the values z_i are positive and thus $\text{Scale}_{\text{OOD}}(z, p) \geq 1$. Then, the energy score function in Eq. (2) is evaluated by replacing the original activations z with the scaled activations $z \cdot \text{Scale}_{\text{OOD}}(z, p)$. This simple rescaling process results in a larger energy gap between ID and OOD data, thereby facilitating their separation using a threshold.

Although adversarial examples can be viewed as a form of OOD data, several challenges must be addressed to adapt the scaling method for adversarial training. First, the top-k operation used in OOD methods is non-differentiable. In the context of adversarial training, this can increase the risk of gradient obfuscation [1, 28]. This vulnerability was notably highlighted for k-WTA [33], which relied on a top-k operation to suppress non-robust features but was later found to be affected by gradient obfuscation [28]. OOD detection methods are unaffected by this issue because the energy scoring functions are post-hoc methods that are only used during inference, thus do not need to guarantee differentiability. Second, the exponential formulation in Eq. (3) produces excessively large factors, which can cause gradient saturation and numerical instability. These limitations motivate us to retain the intuition behind activation scaling while introducing a differentiable and stable method specifically designed for adversarial robustness.

Proposed Approach. We present an overview of A3 in Fig. 2. To rescale the activations z in a controlled and differentiable manner, we first predict a channel-wise binary mask m using the Gumbel-Softmax trick [16]:

$$\text{Mask}(z_m, \tau_m) = \frac{e^{((\log(\sigma(z_m))+g_1)/\tau_m)}}{e^{((\log(\sigma(z_m))+g_1)/\tau_m)} + e^{((\log(1-\sigma(z_m))+g_2)/\tau_m)}}, \tag{4}$$

where $\sigma(\cdot)$ is the sigmoid function, τ_m is a temperature hyperparameter, and g_1, g_2 are i.i.d. samples drawn from the Gumbel distribution [13]. The vector z_m is obtained by applying Global Average Pooling (GAP) to z across its spatial dimensions and a linear projection using a learnable weight matrix $W_m \in \mathbb{R}^{C \times C}$:

$$z_m = W_m \cdot \text{GAP}(z), \qquad \text{GAP}(z)_c = \frac{1}{HW} \sum_{h=1}^{H} \sum_{w=1}^{W} z_{c,h,w}. \tag{5}$$

The Gumbel-Softmax and linear projection are employed to obtain a learnable masking operation that captures useful predictive patterns in the activations. The generated mask m is then used to compute two statistical values s_1 and s_2 based on the distribution of the original activation magnitudes:

$$a_c = \sum_{h=1}^{H} \sum_{w=1}^{W} |z_{c,h,w}|, \qquad s_1 = \sum_{c=1}^{C} a_c, \qquad s_2 = \sum_{c=1}^{C} a_c \cdot m_c. \tag{6}$$

Here, s_1 represents the overall magnitude of the activations, while s_2 captures the magnitude within the masked activations. These formulations are inspired by the scaling function in Eq. (3), but we replace the non-differentiable top-k selection with a learnable, differentiable mask to prevent the risk of gradient

obfuscation. The learnable projection W_m also allows the masking operation to adapt the scaling function to different models and datasets. Using these two statistics, we compute two scaling factors f_m and f_{1m} as follows:

$$f_m = \frac{\ln(1+s_2)}{\ln(1+s_1)}, \qquad f_{1m} = 1 - f_m. \tag{7}$$

There are several important aspects to note about these scaling factors. First, f_m and f_{1m} are differentiable with respect to the activations z and the mask m, enabling seamless end-to-end optimization. Second, the logarithmic formulation provides more stable gradient behavior compared to the exponential-based scaling function in Eq. (3). In particular, the derivative of the logarithm decays smoothly as input magnitudes increase, which helps mitigate gradient explosion and reduces sensitivity to large activation outliers. This stability is particularly crucial in adversarial training, where adversarial perturbations often induce large activation spikes. We also empirically verify in our ablation study that this formulation is the most effective for our task. We then combine the scaling factors with the mask to compute the following scaling function:

$$\text{Scale}(z, m) = f_m \cdot m + f_{1m} \cdot (1 - m). \tag{8}$$

Finally, we modify the original activations z to obtain either amplified or attenuated signals using the following operation:

$$z_{\text{att}} = z \cdot \big[1 - \text{Scale}(z, m)\big], \qquad z_{\text{amp}} = z \cdot \big[1 + \text{Scale}(z, m)\big]. \tag{9}$$

Unlike the OOD scaling function in Eq. (3), our operation is designed such that the scaling factor $\text{Scale}(z, m)$ is always bounded within $[0, 1]$. In fact, since $s_2 \leq s_1$ by definition, it follows that $f_m \in [0, 1]$ and $f_{1m} \in [0, 1]$. This property ensures that the scaling factors remain bounded, preventing extreme amplification or attenuation of the activations, while also preserving a distribution close to the original activations. During inference, the final predictions are computed solely from the attenuated activations z_{att}. The amplified activations z_{amp} are only used during training to serve as negative references for the contrastive and ranking loss functions described in the following section.

3.2 Objective Functions

Cross-Entropy Ranking Loss. To encourage the model to attenuate the influence of adversarial attacks on useful predictive signals, we introduce a ranking loss that enforces the attenuated activations to yield a lower cross-entropy (CE) loss than the amplified activations, using the following formulation:

$$L_{\text{rank}} = \max(\text{CE}(\hat{p}^{\text{adv}}_{\text{att}}, y) - \text{CE}(\hat{p}_{\text{amp}}, y), 0), \tag{10}$$

where $\hat{p}$ denotes the softmax probability output and y is the ground-truth label. This objective serves two primary roles. First, it establishes a competitive optimization process within the module itself, since it simultaneously encourages

lower error for attenuated signals and higher error for amplified signals. Second, the hinge loss formulation helps prevent the model from overfitting to specific adversarial examples. We validate the contributions of both the competitive optimization and the hinge formulation in our ablation study.

Contrastive Logit Loss. Since the attenuation mode aims to reduce the influence of harmful activations, the resulting logits should ideally be invariant to adversarial perturbations. To encourage this behavior, we introduce a contrastive logit loss defined as follows:

$$L_{\text{cl}} = -\log \frac{e^{(sim(l_{\text{att}}^{\text{adv}}, l_{\text{att}})/\tau_{\text{cl}})}}{e^{(sim(l_{\text{att}}^{\text{adv}}, l_{\text{att}})/\tau_{\text{cl}})} + e^{(sim(l_{\text{att}}^{\text{adv}}, l_{\text{amp}})/\tau_{\text{cl}})}}, \tag{11}$$

where $l \in \mathbb{R}^k$ denotes the logit vector for k classes produced by the final classifier head, $sim(\cdot, \cdot)$ is the cosine similarity function, and τ_{cl} is a temperature hyperparameter controlling the sharpness of the similarity distribution. This objective simultaneously minimizes the distance between clean and adversarial logits obtained from the attenuation mode, while also enhancing the separability between the attenuated and amplified logits.

Overall Loss. Our final objective function integrates the standard adversarial training loss with our proposed component losses:

$$L = L_{\text{main}} + \lambda_{\text{rank}} L_{\text{rank}} + \lambda_{\text{cl}} L_{\text{cl}}, \tag{12}$$

where L_{main} is the original training loss of the selected adversarial training strategy (*e.g.*, AT [24], TRADES [38], MART [29]), and λ_{rank}, λ_{cl} are hyperparameters that control the contribution of our proposed ranking and contrastive losses, respectively.

4 Experiments

4.1 Experimental Setup

Training Details. We conduct our experiments on CIFAR-10/100 [19] and Tiny ImageNet [7] datasets using ResNet-18 [15] and WideResNet-34-10 [37] as backbone architectures. We consider three representative adversarial training methods: AT [24], TRADES [38] and MART [29]. For all methods, we employ PGD-10 attack [24] with a maximum perturbation $\epsilon = 8/255$ and step size $\epsilon/4$. We train all models for 100 epochs with a batch size of 128 using SGD optimizer (momentum 0.9, weight decay 5×10^{-4}). The initial learning rate is set to 0.1 with a decay factor of 10 at epochs 75 and 90. These settings are consistent with prior works [11,18,30] to ensure a fair comparison. We insert the A3 module after *block* 4 in ResNet-18 and after *block* 3 in WideResNet-34-10. Further justification for these locations is provided in the supplementary material. Finally, we set the hyperparameters to $\tau_m = 0.1$, $\tau_{\text{cl}} = 10$, $\lambda_{\text{rank}} = 1$ and $\lambda_{\text{cl}} = 5$. The choice of these hyperparameters is discussed in the ablation study.

Table 1: Robust accuracy (%) of **ResNet-18** on CIFAR-10/100 datasets under various attacks. The "clean" column denotes accuracy without attacks. We evaluate the vanilla model, our A3, and other plug-in defenses under different training strategies (AT, TRADES, MART). For each strategy, the best results are in **bold** and the second-best are underlined.

Method	CIFAR-10							CIFAR-100						
ResNet-18	Clean	FGSM	PGD-20	PGD-100	C&W	Ens.	AA	Clean	FGSM	PGD-20	PGD-100	C&W	Ens.	AA
AT (Baseline)	85.04	56.96	49.12	47.51	48.19	46.05	44.28	59.26	29.83	25.17	24.16	24.28	22.85	21.26
+ CAS *(ICLR 2021)* [2]	82.45	56.81	50.02	48.93	50.20	46.97	43.40	58.48	28.92	24.83	23.84	24.82	22.55	20.12
+ CIFS *(ICML 2021)* [36]	80.62	57.34	51.30	49.45	49.06	47.55	43.65	56.64	29.11	25.14	24.21	24.35	22.98	20.41
+ FSR *(CVPR 2023)* [18]	82.78	57.51	51.45	50.05	49.24	47.67	45.92	58.84	29.10	25.24	24.36	24.51	23.05	21.94
+ FPCM *(ICCV 2023)* [3]	85.51	58.07	49.92	48.22	49.44	47.25	46.14	60.05	32.24	28.49	27.45	27.05	25.38	24.25
+ RiFT *(ICCV 2023)* [39]	**85.57**	57.92	50.79	49.19	49.45	47.59	46.21	**60.17**	30.43	26.08	25.34	25.26	23.89	22.75
+ EMFF *(T-PAMI 2025)* [30]	81.82	57.08	51.04	49.63	48.58	47.25	45.84	57.17	31.49	27.79	27.02	26.40	25.22	24.02
+ FTA2C *(NN 2026)* [11]	82.35	57.50	52.03	50.88	48.70	47.14	45.42	56.12	30.75	26.69	26.02	24.85	23.32	21.82
+ A3 (Ours)	84.37	**62.43**	**58.66**	**57.01**	**52.33**	**51.04**	**47.28**	57.50	**33.89**	**32.34**	**31.78**	**27.44**	**26.52**	**24.36**
TRADES (Baseline)	84.02	57.83	51.59	50.48	48.93	48.26	46.98	59.02	31.29	27.44	26.98	24.42	24.07	23.05
+ CAS *(ICLR 2021)* [2]	83.19	57.02	51.24	50.31	49.05	47.99	45.12	57.41	30.22	27.41	26.59	24.90	24.26	22.08
+ CIFS *(ICML 2021)* [36]	81.34	57.40	51.58	50.47	48.95	48.23	45.39	57.68	30.30	27.52	26.94	24.48	23.97	22.19
+ FSR *(CVPR 2023)* [18]	83.46	57.49	51.67	50.78	49.09	48.39	47.50	57.63	30.39	27.57	27.01	24.53	24.13	23.35
+ FPCM *(ICCV 2023)* [3]	83.73	58.41	53.14	52.27	50.61	49.68	48.67	58.98	31.95	29.40	**29.05**	26.03	25.45	24.50
+ RiFT *(ICCV 2023)* [39]	**84.75**	58.48	52.47	51.26	50.01	49.35	48.24	**60.73**	31.74	28.51	27.92	25.42	24.95	23.91
+ EMFF *(T-PAMI 2025)* [30]	83.40	57.96	52.09	50.67	48.91	48.23	47.13	59.35	**32.43**	29.26	28.79	25.68	25.28	24.49
+ FTA2C *(NN 2026)* [11]	83.97	58.19	52.99	51.90	50.27	49.60	48.68	57.88	31.47	28.31	27.91	25.66	24.91	24.00
+ A3 (Ours)	83.77	**58.50**	**55.10**	**54.26**	**51.52**	**50.97**	**49.27**	57.36	31.71	**29.61**	28.81	**26.35**	**25.80**	**24.81**
MART (Baseline)	81.92	57.98	52.12	51.08	48.25	47.56	45.88	57.92	31.36	27.41	26.67	24.96	24.38	22.19
+ CAS *(ICLR 2021)* [2]	80.89	57.73	51.17	49.88	49.13	47.16	43.75	55.24	30.27	26.68	25.89	25.11	23.31	21.12
+ CIFS *(ICML 2021)* [36]	81.01	58.19	52.24	50.67	48.62	47.66	44.06	54.14	30.40	26.82	26.17	24.84	23.53	21.44
+ FSR *(CVPR 2023)* [18]	83.21	58.38	52.50	51.02	48.95	47.78	46.16	56.45	30.55	27.01	26.28	25.27	24.15	23.02
+ FPCM *(ICCV 2023)* [3]	80.92	55.86	49.37	47.92	45.65	44.86	43.46	55.08	28.14	25.50	24.94	22.54	22.19	21.24
+ RiFT *(ICCV 2023)* [39]	**83.53**	58.73	52.87	51.43	48.73	47.93	46.31	56.15	32.32	29.25	28.78	25.96	25.26	23.92
+ EMFF *(T-PAMI 2025)* [30]	82.13	57.87	51.59	50.16	48.14	46.99	45.61	55.87	31.47	28.45	27.88	25.81	25.21	24.14
+ FTA2C *(NN 2026)* [11]	82.16	58.00	52.64	51.37	48.32	47.34	45.69	55.38	30.85	27.45	26.65	24.21	23.27	21.80
+ A3 (Ours)	82.69	**62.83**	**59.05**	**57.05**	**51.70**	**50.86**	**47.29**	**58.86**	**33.01**	**29.79**	**28.70**	**27.28**	**26.01**	**24.24**

Baselines. We compare our proposed A3 module against recent plug-in methods that are also compatible with different training methods: CAS [2], CIFS [36], FPCM [3], RiFT [39], FSR [18], FTA2C [11] and EMFF [30]. All baselines are evaluated using the official implementations and recommended hyperparameter configurations provided by the authors.

Evaluation Details. Our evaluation includes clean accuracy (*i.e.*, accuracy without attack) and robust accuracy under the following attacks: FGSM [12], PGD-20, PGD-100 [24] and C&W (with 30 steps under ℓ_∞ norm) [4]. Unless specified otherwise, all attacks are conducted with a maximum perturbation $\epsilon = 8/255$ and a step size of $\epsilon/10$. We also report the ensemble (Ens.) accuracy, defined as the sample-wise worst-case accuracy among the following predictions: Clean, FGSM, PGD-20, PGD-100 and C&W. Finally, we provide results under AutoAttack (AA) [6], which is an ensemble-based attack widely considered as the most reliable evaluation benchmark for adversarial robustness.

4.2 Results

We present the main results on CIFAR-10/100 in Tabs. 1 and 2 and on Tiny ImageNet in Tab. 3. Additional analyses such as confidence statistics and Expected Calibration Error (ECE) [14] are provided in the supplementary material. Unless stated otherwise, we use A3 in attenuation mode at inference time, while the amplification mode is only used during training. Overall, A3 consistently improves adversarial robustness across various models, datasets, and training

Table 2: Robust accuracy (%) of **WideResNet-34-10** on CIFAR-10/100 datasets under various attacks. The "clean" column denotes accuracy without attacks. We evaluate the vanilla model, our A3, and other plug-in defenses under different training strategies (AT, TRADES, MART). For each strategy, the best results are in **bold** and the second-best are underlined.

WideResNet-34-10	CIFAR-10							CIFAR-100						
Method	Clean	FGSM	PGD-20	PGD-100	C&W	Ens.	AA	Clean	FGSM	PGD-20	PGD-100	C&W	Ens.	AA
AT (Baseline)	87.57	60.12	51.58	49.83	51.60	49.27	48.18	62.71	31.59	26.53	25.48	26.55	24.69	23.70
+ CAS *(ICLR 2021)* [2]	86.43	60.16	51.67	50.02	51.43	49.23	46.62	61.98	31.78	26.68	25.98	26.84	24.52	22.37
+ CIFS *(ICML 2021)* [36]	85.96	60.91	51.78	50.18	52.07	49.58	46.59	61.75	31.92	26.84	25.74	26.52	24.70	22.63
+ FSR *(CVPR 2023)* [18]	87.07	61.13	53.81	51.97	52.41	50.65	48.80	61.88	30.92	25.11	23.61	25.64	23.33	22.28
+ FPCM *(ICCV 2023)* [3]	87.13	58.70	50.48	48.54	50.12	48.31	47.36	61.41	30.41	26.07	25.14	26.02	24.56	23.99
+ RiFT *(ICCV 2023)* [39]	**87.85**	60.83	52.32	50.38	52.05	49.84	48.74	**63.00**	31.71	26.65	25.50	26.55	24.94	24.23
+ EMFF *(T-PAMI 2025)* [30]	86.10	62.29	53.92	51.75	51.78	50.38	49.10	60.33	34.22	30.05	29.07	28.59	27.33	26.22
+ FTA2C *(NN 2026)* [11]	86.36	60.44	52.70	51.07	52.22	49.73	48.10	61.07	30.71	25.84	24.53	25.42	23.21	22.07
+ A3 (Ours)	86.64	**63.01**	**58.33**	**56.72**	**55.32**	**53.82**	**51.62**	62.50	**34.64**	**31.22**	**30.23**	**30.51**	**28.98**	**27.68**
TRADES (Baseline)	86.39	61.14	52.69	50.86	52.12	50.25	49.33	61.88	32.50	28.31	27.62	27.04	26.32	25.51
+ CAS *(ICLR 2021)* [2]	86.30	61.62	52.84	51.12	52.81	50.56	47.61	60.74	32.03	28.09	27.53	27.31	26.29	24.49
+ CIFS *(ICML 2021)* [36]	86.08	61.27	52.98	51.27	52.34	50.71	47.96	60.18	32.27	28.34	27.70	27.14	26.11	24.60
+ FSR *(CVPR 2023)* [18]	86.92	62.86	54.24	51.97	52.98	51.10	50.08	59.13	32.70	29.40	28.87	27.43	26.89	26.22
+ FPCM *(ICCV 2023)* [3]	85.83	58.96	51.76	50.12	50.85	49.36	48.54	58.31	31.57	28.23	27.53	26.45	25.80	24.97
+ RiFT *(ICCV 2023)* [39]	**87.29**	61.92	53.25	51.26	52.54	50.58	49.64	62.93	33.44	29.36	28.35	27.84	27.08	26.21
+ EMFF *(T-PAMI 2025)* [30]	86.51	62.41	54.30	52.24	52.01	50.47	49.66	**63.66**	**34.47**	29.78	28.96	27.97	27.22	26.24
+ FTA2C *(NN 2026)* [11]	84.44	62.08	54.38	52.15	52.36	50.95	50.17	57.23	32.57	29.61	28.71	27.32	26.43	26.20
+ A3 (Ours)	85.46	**65.34**	**58.19**	**55.12**	**54.84**	**53.28**	**51.41**	59.78	33.87	**31.45**	**30.61**	**28.26**	**27.66**	**26.46**
MART (Baseline)	86.79	60.78	53.05	51.08	51.00	49.62	48.51	61.39	31.35	26.97	25.85	25.82	24.74	23.96
+ CAS *(ICLR 2021)* [2]	85.75	61.32	53.21	51.22	51.57	50.20	47.27	60.38	30.51	26.53	24.98	25.70	24.32	22.68
+ CIFS *(ICML 2021)* [36]	86.36	61.67	53.44	51.31	51.35	50.39	47.16	59.79	30.57	26.85	25.20	25.66	24.49	22.90
+ FSR *(CVPR 2023)* [18]	86.31	61.60	54.29	52.50	51.75	50.53	49.15	56.64	29.49	24.60	23.53	24.45	22.89	22.16
+ FPCM *(ICCV 2023)* [3]	86.41	61.69	54.14	52.17	52.05	50.87	49.36	60.42	31.77	27.53	26.69	26.71	25.60	24.70
+ RiFT *(ICCV 2023)* [39]	**87.29**	61.92	53.25	51.26	52.54	50.58	49.64	**63.00**	31.71	26.65	25.50	26.55	24.94	24.23
+ EMFF *(T-PAMI 2025)* [30]	85.92	63.11	55.60	54.45	52.51	51.42	49.94	61.04	33.53	29.27	28.33	27.98	26.93	25.88
+ FTA2C *(NN 2026)* [11]	84.93	61.30	54.09	52.29	51.18	50.57	49.18	57.32	29.62	24.83	23.99	24.98	23.49	22.53
+ A3 (Ours)	86.65	**63.61**	**58.36**	**56.00**	**54.54**	**53.32**	**50.59**	60.55	**36.19**	**33.94**	**33.28**	**30.90**	**29.87**	**28.22**

methods. For example, A3 improves the ensemble accuracy (Ens.) of ResNet-18 trained with AT on CIFAR-10 by 4.99% over the vanilla backbone, achieving the strongest performance among the compared plug-in defenses.

Although we observe a modest decrease in clean accuracy compared to the vanilla backbone, this is a common trade-off in adversarial training [38, 39]. We attribute this behavior to the relative absence of harmful activations in clean images, which makes the attenuation operation less beneficial for these samples. Nevertheless, A3 maintains a clean accuracy competitive with existing methods while ensuring better robustness against adversarial attacks, which is the primary objective of our approach. Across all three training strategies, A3 achieves the highest AA and Ens. scores without severely compromising clean accuracy.

4.3 Ablation Studies

Amplification, attenuation, or suppression? The A3 module is designed to either amplify or attenuate the influence of adversarial examples by flipping the sign of the scaling operation in Eq. (9). Although we only need the attenuation mode during inference, we compare its performance against the amplification mode in Tab. 4. As expected, the attenuation mode achieves significantly higher accuracy on adversarial examples compared to the amplification mode. Moreover, deactivating the A3 module (*i.e.*, replacing it with an identity function) yields higher results than the amplification mode but remains inferior to the attenuation mode. This is consistent with the intended design of A3, which aims

Table 3: Robust accuracy (%) of **ResNet-18** and **WideResNet-34-10** on Tiny Imagenet dataset under various attacks. The "clean" column denotes accuracy without attacks. We evaluate the vanilla model, our A3, and other plug-in defenses under different training strategies (AT, TRADES, MART). For each strategy, the best results are in **bold** and the second-best are underlined.

Tiny ImageNet	ResNet-18							WideResNet-34-10						
Method	Clean	FGSM	PGD-20	PGD-100	C&W	Ens.	AA	Clean	FGSM	PGD-20	PGD-100	C&W	Ens.	AA
AT (Baseline)	50.33	23.83	20.34	19.91	18.96	17.49	16.42	52.78	26.25	22.84	22.13	21.62	19.93	18.79
+ CAS *(ICLR 2021)* [2]	48.85	24.16	20.56	20.11	19.08	17.30	15.19	51.47	26.58	22.97	22.14	21.87	19.82	17.36
+ CIFS *(ICML 2021)* [36]	48.63	24.52	20.89	20.38	18.46	17.49	15.73	51.72	26.65	23.21	22.59	21.57	20.01	17.09
+ FSR *(CVPR 2023)* [18]	48.16	24.35	20.71	20.28	19.37	17.52	16.60	49.64	26.56	23.40	22.62	21.71	20.10	19.28
+ FPCM *(ICCV 2023)* [3]	45.71	24.80	22.54	**22.30**	19.68	18.53	17.21	54.92	27.70	24.30	23.42	22.45	20.98	19.80
+ RiFT *(ICCV 2023)* [39]	**51.67**	24.44	21.14	20.48	19.28	18.04	16.64	**55.03**	27.20	23.29	22.34	22.13	20.42	19.06
+ EMFF *(T-PAMI 2025)* [30]	48.49	24.16	21.69	21.06	19.45	18.54	17.43	50.12	26.36	23.88	23.31	21.85	20.65	19.45
+ FTA2C *(NN 2026)* [11]	48.62	22.59	20.69	20.50	16.62	16.28	15.30	49.88	26.28	23.14	22.49	21.84	19.97	18.93
+ A3 (Ours)	51.10	**25.46**	**22.97**	22.02	**21.32**	**19.78**	**18.25**	51.92	**27.86**	**25.27**	**24.48**	**23.32**	**21.87**	**20.43**
TRADES (Baseline)	49.95	23.38	20.74	20.46	16.83	16.14	14.80	53.96	27.08	24.09	23.88	19.76	19.38	18.20
+ CAS *(ICLR 2021)* [2]	48.64	23.27	20.93	20.77	16.91	16.09	13.82	52.42	27.64	24.50	24.00	20.03	19.43	17.41
+ CIFS *(ICML 2021)* [36]	47.98	23.55	21.11	20.92	16.76	16.22	13.97	52.19	27.79	24.47	24.12	19.84	19.59	17.28
+ FSR *(CVPR 2023)* [18]	45.77	21.41	19.59	19.37	15.69	15.40	14.47	50.59	27.31	24.84	24.23	20.29	19.82	18.39
+ FPCM *(ICCV 2023)* [3]	49.93	24.29	22.35	22.00	18.29	17.64	16.08	53.13	27.17	25.08	24.83	**21.25**	20.86	19.75
+ RiFT *(ICCV 2023)* [39]	**51.91**	23.92	21.13	20.28	16.82	16.37	13.94	**55.79**	27.53	24.27	23.36	20.20	19.68	18.03
+ EMFF *(T-PAMI 2025)* [30]	50.60	24.09	22.10	21.74	17.37	17.21	16.38	53.45	27.49	25.22	**24.91**	21.11	**20.87**	**19.99**
+ FTA2C *(NN 2026)* [11]	48.62	22.64	20.63	20.46	16.92	16.29	15.32	50.50	27.65	24.63	23.93	20.51	19.87	18.80
+ A3 (Ours)	48.44	**24.95**	**23.27**	**22.86**	**19.33**	**18.90**	**17.73**	52.88	**27.92**	**25.28**	24.89	20.96	20.60	19.07
MART (Baseline)	48.05	24.48	22.21	21.79	19.34	18.23	16.92	52.06	27.93	24.86	24.43	21.73	20.82	19.31
+ CAS *(ICLR 2021)* [2]	46.64	23.75	21.79	20.30	18.86	17.69	15.73	51.88	28.15	24.87	24.26	22.06	20.93	18.63
+ CIFS *(ICML 2021)* [36]	46.46	23.63	21.84	20.38	18.65	17.56	15.44	51.49	28.24	24.99	24.51	21.79	20.72	18.80
+ FSR *(CVPR 2023)* [18]	45.85	23.11	20.86	20.57	17.98	17.33	16.23	50.50	28.03	24.89	24.51	21.77	20.91	19.51
+ FPCM *(ICCV 2023)* [3]	48.45	22.31	19.47	18.96	17.62	16.84	15.88	53.12	28.57	26.06	**25.49**	22.76	21.90	20.37
+ RiFT *(ICCV 2023)* [39]	50.20	25.39	22.76	22.29	19.74	18.81	17.40	**53.61**	28.37	24.90	24.30	22.20	21.18	19.76
+ EMFF *(T-PAMI 2025)* [30]	46.44	24.62	22.93	22.59	19.83	19.28	18.17	51.88	28.25	25.65	25.09	23.24	22.23	**21.35**
+ FTA2C *(NN 2026)* [11]	42.56	22.30	20.56	20.63	20.43	17.15	15.53	50.44	28.25	25.16	24.89	22.30	21.57	20.11
+ A3 (Ours)	**51.03**	**26.36**	**23.89**	**23.34**	**21.78**	**20.67**	**18.96**	53.48	**29.49**	**26.49**	25.28	**23.85**	**22.25**	20.69

Table 4: Robust accuracy (%) of A3 under different modes. We use ResNet-18 trained with AT on CIFAR-10 as the backbone. The attenuation mode corresponds to our default design used in all other experiments. We compare it with the amplification mode as well as two suppression modes that completely zero out either m or $1-m$. The best results are highlighted in **bold**.

Mode	A3 Output	Clean	FGSM	PGD-20	PGD-100	C&W	Ens.	AA
Suppression (m)	$z \cdot m$	32.40	20.85	19.65	19.26	16.34	16.11	15.13
Suppression ($1-m$)	$z \cdot (1-m)$	85.04	57.19	51.02	48.27	51.22	48.06	46.55
Amplification/z_{amp}	$z \cdot \big[1 + \text{Scale}(z, m)\big]$	84.16	56.20	50.38	47.81	50.69	47.36	45.92
Deactivated	z	**85.11**	57.78	51.56	48.87	51.62	48.73	46.72
Attenuation/z_{att} **(Ours)**	$z \cdot \big[1 - \text{Scale}(z, m)\big]$	84.37	**62.43**	**58.66**	**57.01**	**52.33**	**51.04**	**47.28**

Table 5: Robust accuracy (%) of A3 under different loss configurations. We use ResNet-18 trained with AT on CIFAR-10 as the backbone. Best results are in **bold**.

Method	Clean	FGSM	PGD-20	PGD-100	C&W	Ens.	AA
ResNet18 + A3 (Ours)	**84.37**	**62.43**	**58.66**	**57.01**	**52.33**	**51.04**	**47.28**
w/o L_{cl}	83.57	61.75	58.02	56.38	51.35	50.24	46.70
w/o L_{rank}	83.49	57.03	51.41	49.03	49.61	48.11	45.41
L_{rank} replaced by Eq. (13a)	84.29	57.85	52.05	49.81	51.54	48.74	46.84
L_{rank} replaced by Eq. (13b)	84.05	58.99	53.50	51.34	51.39	49.13	46.73
L_{rank} replaced by Eq. (13c)	11.19	9.67	9.23	9.17	9.35	8.95	7.80

to enhance or degrade the prediction accuracy depending on the mode. Moreover, the clean accuracy is highest when deactivating the A3 module, which is consistent with our previous analysis indicating that clean images provide fewer opportunities for the attenuation operation to improve robustness.

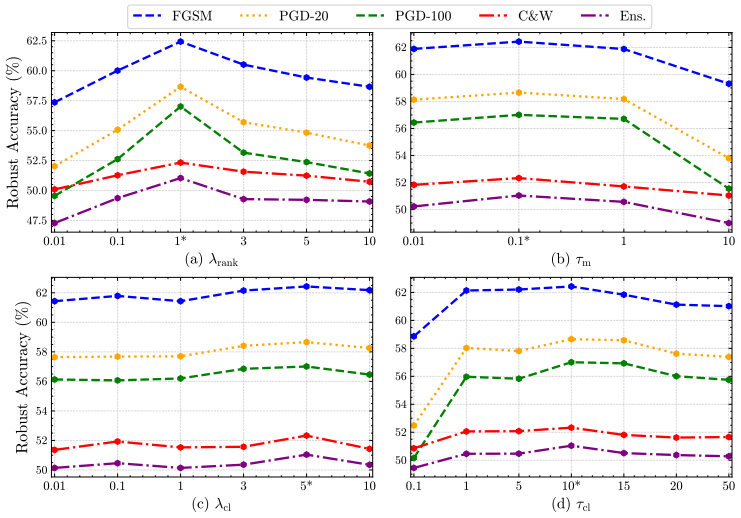


Fig. 3: Ablation study on hyperparameters. We analyze the effect of different hyperparameters in our A3 module on the adversarial robustness against various attacks. We use ResNet-18 trained with AT on CIFAR-10 as the backbone. The '*' indicates the default value used in all other experiments.

We also evaluated a hard suppression strategy by completely masking the activations using either m or $1-m$. Our results in Tab. 4 show that the suppression operations lead to a degradation in performance compared to our attenuation approach. This aligns with the findings in FSR [18], which suggest that simply discarding non-robust features is a suboptimal solution.

Effect of loss components. We measure the individual contribution of L_{rank} and L_{cl} in Eq. (12) by deactivating each of them. Furthermore, we propose to replace L_{rank} with standard cross-entropy losses to validate the benefit of our ranking and hinge loss formulation introduced in Eq. (10). Specifically, we evaluate the following alternative objective functions:

$$L_{\text{CE}}^{\text{att/amp}} = \text{CE}(p_{\text{att}}^{\text{adv}}, y) - \text{CE}(p_{\text{amp}}, y), \tag{13a}$$

$$L_{\text{CE}}^{\text{att}} = \text{CE}(p_{\text{att}}^{\text{adv}}, y), \tag{13b}$$

$$L_{\text{CE}}^{\text{amp}} = -\text{CE}(p_{\text{amp}}, y). \tag{13c}$$

The results in Tab. 5 demonstrate that both L_{rank} and L_{cl} are essential for achieving high adversarial robustness. Moreover, replacing L_{rank} with the unconstrained loss in Eq. (13a) leads to suboptimal performance, indicating that the hinge formulation is necessary to focus the optimization on samples where the error gap between the amplified and attenuated activations is large. Similarly, a performance drop is observed when using Eqs. (13b) and (13c), which removes the dual optimization process between the two scaling modes. These findings

Table 6: Robust accuracy (%) of A3 with different formulations of f_m/f_{1m}. We use ResNet-18 trained with AT on CIFAR-10 as the backbone. The highlighted row corresponds to our default formulation presented in Eq. (7). Best results are in **bold**.

Method	f_m	f_{1m}	Clean	FGSM	PGD-20	PGD-100	C&W	Ens.	AA
linear	$\frac{s_2}{s_1}$	$1-f_m$	84.12	57.94	51.95	49.56	51.22	48.58	46.85
linear (inverted)	$1-f_{1m}$	$\frac{s_2}{s_1}$	84.06	57.64	51.50	49.39	50.97	48.43	46.67
quadratic	$(\frac{s_2}{s_1})^2$	$1-f_m$	84.34	59.70	54.93	52.92	51.27	49.41	46.71
quadratic (inverted)	$1-f_{1m}$	$(\frac{s_2}{s_1})^2$	84.22	57.71	51.75	49.69	51.01	48.80	46.22
exponential	$\exp(1-\frac{s_1}{s_2})$	$1-f_m$	84.33	59.00	52.73	50.96	51.16	48.89	46.86
exponential (inverted)	$1-f_{1m}$	$\exp(1-\frac{s_1}{s_2})$	84.26	57.72	51.86	50.05	50.79	48.57	46.13
log (**Ours**, Eq. (7))	$\frac{\ln(1+s_2)}{\ln(1+s_1)}$	$1-f_m$	**84.37**	**62.43**	**58.66**	**57.01**	**52.33**	**51.04**	**47.28**
log (inverted)	$1-f_{1m}$	$\frac{\ln(1+s_2)}{\ln(1+s_1)}$	84.26	61.94	58.16	56.44	51.61	50.52	46.88

Table 7: Computational efficiency of A3. We compare the number of parameters (M) and FLOPs (G) with and without A3 module. For comparison, we also include the results of FSR [18] and FTA2C [11] which also use feature masking methods.

	ResNet-18		WideResNet-34-10	
Method	# Params (M)	FLOPs (G)	# Params (M)	FLOPs (G)
Vanilla (Baseline)	11.17	0.5567	46.16	6.6647
+ FSR [18]	12.43 (+1.26)	0.5768 (+0.0201)	47.72 (+1.56)	6.7640 (+0.0993)
+ FTA2C [11]	21.88 (+10.71)	0.7278 (+0.1711)	57.20 (+11.04)	7.2360 (+0.6713)
+ A3 **(Ours)**	11.44 **(+0.27)**	0.5588 **(+0.0021)**	46.57 **(+0.41)**	6.6738 **(+0.0091)**

confirm that our hinge-based ranking formulation is crucial for maintaining a stable optimization process and achieving high robust accuracy.

Effect of the hyperparameters. We analyze the impact of the hyperparameters in the A3 module. We empirically found the best configuration with $\tau_m = 0.1$, $\tau_{\text{cl}} = 10$, $\lambda_{\text{rank}} = 1$ and $\lambda_{\text{cl}} = 5$. To justify this choice, we vary each parameter while keeping the others fixed, as illustrated in Fig. 3. Our observations indicate that the module is sensitive to the choice of λ_{rank}, which is expected since this hyperparameter controls the cross-entropy component. We also observe a high sensitivity to λ_{cl} and τ_{cl}, indicating that the contrastive logit loss plays an important role in improving robustness. Finally, the performance remains relatively stable for τ_m when its value is below 1.

Choice of the scaling factors. We empirically verify the effectiveness of the factors f_m and f_{1m} in Eq. (7) by evaluating several alternative formulations, including linear, quadratic, and exponential functions. We also evaluate the configurations where the roles of f_m and f_{1m} are inverted. Detailed mathematical formulations are provided in Tab. 6. It is important to note that these functions are constrained to the range $[0, 1]$ to preserve the intended behavior of the scaling operations in Eq. (9). The results in Tab. 6 show that our proposed logarithmic formulation significantly outperforms the other formulations.

Computational Efficiency. We analyze the computational overhead introduced by A3 in Tab. 7. We report the number of parameters and FLOPs for ResNet-18 and WideResNet-34-10. The FLOPs are calculated using an input size of $3 \times 32 \times 32$ (CIFAR-10) with a batch size of 1. The results show that A3 introduces negligible overhead in terms of both parameters and FLOPs compared to other masking-based methods, while achieving higher robust accuracy.

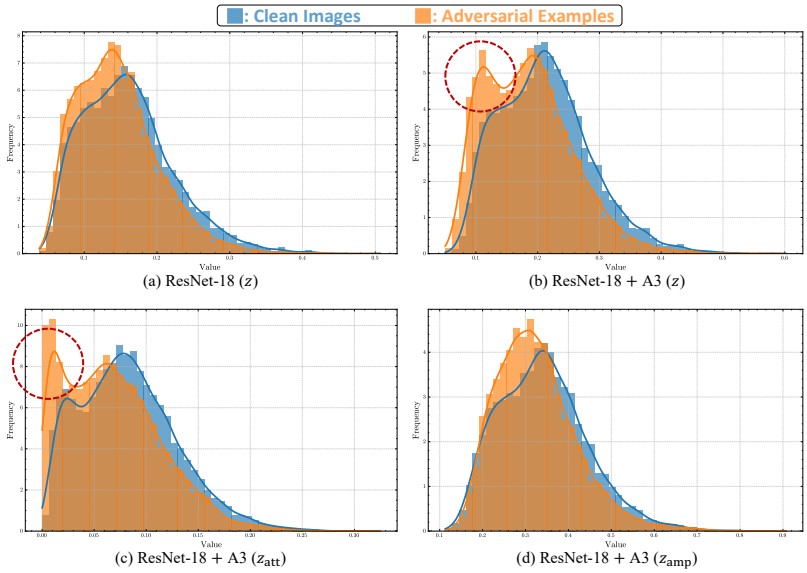


Fig. 4: Distribution of the activations under different modes. We extract z from ResNet-18 at *block* 4 and visualize the distribution of average activation amplitudes for clean and adversarial examples. We compare four settings: (a) ResNet-18 trained without A3, (b) ResNet-18 trained with A3 (before scaling), (c) A3 after attenuation (z_{att}), and (d) A3 after amplification (z_{amp}). While clean and adversarial examples produce similar distributions in (a), the presence of A3 in (b) induces a clear distribution shift for adversarial examples (encircled in red). This shift is further accentuated in (c) via attenuation, thereby enhancing the separability between clean and adversarial examples. In contrast, the amplification mode in (d) increases the activation magnitudes for both inputs, resulting in overlapping distributions similar to (a).

4.4 Activation Distribution Analysis

To further understand how our A3 module improves adversarial robustness, we analyze the distribution of the activations before and after applying A3 to the backbone. Figure 4 presents the histogram of the averaged activation amplitudes collected from ResNet-18. In Fig. 4 (a), we show the activations from a vanilla ResNet-18 after *block* 4, *i.e.*, where A3 is usually inserted. Figure 4 (b) shows the activations from the same location in a model trained with A3. Interestingly, a distribution shift is already visible, indicating that the presence of the A3 module influences the learned representations during training. Specifically, the distribution in Fig. 4 (b) shows two distinct peaks, suggesting that the network has already learned to separate the activations into two groups. The left peak (encircled in red) only appears with adversarial examples. We hypothesize that these activations correspond to activation patterns induced by adversarial perturbations, which are effectively suppressed by A3.

We show the activations frequency distribution after applying the A3 module in attenuation mode (z_{att}) in Fig. 4 (c). Here, the left peak originally observed in Fig. 4 (b) is effectively suppressed for adversarial examples, bringing it close to zero. The effect is less pronounced for clean images, which is consistent with our previous analysis in Tab. 1 that clean accuracy remains relatively unaffected while adversarial robustness improves.

In contrast, the amplification mode (z_{amp}) in Fig. 4 (d) shows substantial overlap between the clean and adversarial distributions, similar to the vanilla model in Fig. 4 (a). This indicates that while the attenuation mode enhances separability between clean and adversarial examples, the amplification mode intentionally reduces this separability, leading to the performance degradation observed in Tab. 4.

5 Conclusion

In this work, we introduced **Activation Amplification and Attenuation (A3)**, a lightweight plug-in module designed to improve adversarial robustness through controlled rescaling operations on the intermediate activations. A3 computes a scalar factor based on the original activation magnitudes to selectively amplify or attenuate intermediate activation patterns. The key idea is to use these amplified and attenuated activations during training to jointly learn robustness enhancement and prediction degradation. This process is optimized using novel ranking and contrastive loss functions that leverage the amplified activations as a negative reference. Extensive experiments demonstrate that A3 consistently improves adversarial robustness across different models, datasets, and training strategies while introducing negligible computational and memory overhead. By offering an efficient feature-based defense with few trainable parameters, A3 provides a promising direction for building more robust and interpretable models.

Acknowledgements

This work was supported by JSPS KAKENHI (Grant Numbers JP24K22315 and JP26K21240) and JST BOOST (Grant Number JPMJBS2421).

References

1. Athalye, A., Carlini, N., Wagner, D.: Obfuscated Gradients Give a False Sense of Security: Circumventing Defenses to Adversarial Examples. In: International Conference on Machine Learning. pp. 274–283. PMLR (2018), https://proceedings.mlr.press/v80/athalye18a.html
2. Bai, Y., Zeng, Y., Jiang, Y., Xia, S.T., Ma, X., Wang, Y.: Improving Adversarial Robustness via Channel-wise Activation Suppressing. In: International Conference on Learning Representations (2021), https://openreview.net/forum?id=zQTezqCCtNx

3. Bu, Q., Huang, D., Cui, H.: Towards Building More Robust Models with Frequency Bias. In: IEEE/CVF International Conference on Computer Vision. pp. 4379–4388 (2023). https://doi.org/10.1109/ICCV51070.2023.00406
4. Carlini, N., Wagner, D.: Towards Evaluating the Robustness of Neural Networks. In: IEEE Symposium on Security and Privacy. pp. 39–57 (2017). https://doi.org/10.1109/SP.2017.49
5. Croce, F., Andriushchenko, M., Sehwag, V., Debenedetti, E., Flammarion, N., Chiang, M., Mittal, P., Hein, M.: RobustBench: A standardized adversarial robustness benchmark. In: Advances in Neural Information Processing Systems (2021), https://openreview.net/forum?id=SSKZPJCt7B
6. Croce, F., Hein, M.: Reliable evaluation of adversarial robustness with an ensemble of diverse parameter-free attacks. In: International Conference on Machine Learning. pp. 2206–2216. PMLR (2020), https://proceedings.mlr.press/v119/croce20b.html
7. Deng, J., Dong, W., Socher, R., Li, L.J., Li, K., Fei-Fei, L.: ImageNet: A Large-Scale Hierarchical Image Database. In: IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 248–255 (2009). https://doi.org/10.1109/CVPR.2009.5206848
8. Dhillon, G.S., Azizzadenesheli, K., Lipton, Z.C., Bernstein, J., Kossaifi, J., Khanna, A., Anandkumar, A.: Stochastic Activation Pruning for Robust Adversarial Defense. In: International Conference on Learning Representations (2018), https://openreview.net/forum?id=H1uR4GZRZ
9. Djurisic, A., Bozanic, N., Ashok, A., Liu, R.: Extremely Simple Activation Shaping for Out-of-Distribution Detection. In: International Conference on Learning Representations (2023), https://openreview.net/forum?id=ndYXTEL6cZz
10. Djurisic, A., Liu, R., Nikolic, M.: Logit scaling for out-of-distribution detection. Machine Vision and Applications **36**(5), 108 (2025). https://doi.org/10.1007/s00138-025-01730-8
11. Gao, Z., Liu, C., Shi, Y., Guo, X., Xu, J., Zhang, H., Shi, L.: FTA2C: Achieving Superior Trade-off between Accuracy and Robustness in Adversarial Training. Neural Networks **194**, 108176 (2026). https://doi.org/10.1016/j.neunet.2025.108176
12. Goodfellow, I.J., Shlens, J., Szegedy, C.: Explaining and Harnessing Adversarial Examples. In: International Conference on Learning Representations (2015), https://arxiv.org/abs/1412.6572
13. Gumbel, E.J.: Statistical Theory of Extreme Values and Some Practical Applications. A Series of Lectures. Tech. Rep. PB175818, National Bureau of Standards, Washington, D. C. Applied Mathematics Div. (1954), https://ntrl.ntis.gov/NTRL/dashboard/searchResults/titleDetail/PB175818.xhtml
14. Guo, C., Pleiss, G., Sun, Y., Weinberger, K.Q.: On Calibration of Modern Neural Networks. In: International Conference on Machine Learning. pp. 1321–1330. PMLR (2017), https://proceedings.mlr.press/v70/guo17a.html
15. He, K., Zhang, X., Ren, S., Sun, J.: Deep Residual Learning for Image Recognition. In: IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 770–778 (2016). https://doi.org/10.1109/CVPR.2016.90
16. Jang, E., Gu, S., Poole, B.: Categorical Reparameterization with Gumbel-Softmax. In: International Conference on Learning Representations (2017), https://openreview.net/forum?id=rkE3y85ee
17. Jia, X., Zhang, Y., Wu, B., Ma, K., Wang, J., Cao, X.: LAS-AT: Adversarial Training with Learnable Attack Strategy. In: IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 13388–13398 (2022). https://doi.org/10.1109/CVPR52688.2022.01304

18. Kim, W.J., Cho, Y., Jung, J., Yoon, S.E.: Feature Separation and Recalibration for Adversarial Robustness. In: IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 8183–8192 (2023). https://doi.org/10.1109/CVPR52729.2023.00791
19. Krizhevsky, A.: Learning Multiple Layers of Features from Tiny Images. Technical Report, University of Toronto (2009), https://cave.cs.toronto.edu/kriz/learning-features-2009-TR.pdf
20. Lee, K., Lee, K., Lee, H., Shin, J.: A Simple Unified Framework for Detecting Out-of-Distribution Samples and Adversarial Attacks. In: Advances in Neural Information Processing Systems. vol. 31. Curran Associates, Inc. (2018), https://papers.nips.cc/paper_files/paper/2018/hash/abdeb6f575ac5c6676b747bca8d09cc2-Abstract.html
21. Li, Z., Yu, D., Wei, L., Jin, C., Zhang, Y., Chan, S.: Soften to Defend: Towards Adversarial Robustness via Self-Guided Label Refinement. In: IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 24776–24785 (2024). https://doi.org/10.1109/CVPR52733.2024.02340
22. Liang, S., Li, Y., Srikant, R.: Enhancing The Reliability of Out-of-distribution Image Detection in Neural Networks. In: International Conference on Learning Representations (2018), https://openreview.net/forum?id=H1VGkIxRZ
23. Liu, W., Wang, X., Owens, J.D., Li, Y.: Energy-based Out-of-distribution Detection. In: Advances in Neural Information Processing Systems. vol. 33, pp. 21464–21475 (2020), https://proceedings.neurips.cc/paper/2020/hash/f5496252609c43eb8a3d147ab9b9c006-Abstract.html
24. Madry, A., Makelov, A., Schmidt, L., Tsipras, D., Vladu, A.: Towards Deep Learning Models Resistant to Adversarial Attacks. In: International Conference on Learning Representations (2018), https://openreview.net/forum?id=rJzIBfZAb
25. Regmi, S.: AdaSCALE: Adaptive Scaling for OOD Detection. arXiv preprint arXiv:2503.08023 (2025)
26. Selvaraju, R.R., Cogswell, M., Das, A., Vedantam, R., Parikh, D., Batra, D.: Grad-CAM: Visual Explanations from Deep Networks via Gradient-based Localization. Int J Comput Vis **128**(2), 336–359 (2017). https://doi.org/10.1007/s11263-019-01228-7
27. Tong, H., Zhang, X., Jin, Y., Lou, J., Wu, K., Chen, X.: Balancing Generalization and Robustness in Adversarial Training via Steering through Clean and Adversarial Gradient Directions. In: Proceedings of the 32nd ACM International Conference on Multimedia. pp. 1014–1023. ACM (2024). https://doi.org/10.1145/3664647.3680963
28. Tramèr, F., Carlini, N., Brendel, W., Madry, A.: On Adaptive Attacks to Adversarial Example Defenses. In: Advances in Neural Information Processing Systems. vol. 33, pp. 1633–1645 (2020), https://proceedings.neurips.cc/paper/2020/hash/11f38f8ecd71867b42433548d1078e38-Abstract.html
29. Wang, Y., Zou, D., Yi, J., Bailey, J., Ma, X., Gu, Q.: Improving Adversarial Robustness Requires Revisiting Misclassified Examples. In: International Conference on Learning Representations (2020), https://openreview.net/forum?id=rklOg6EFwS
30. Wang, Z., Xu, X., Zhu, L., Bin, Y., Wang, G., Yang, Y., Shen, H.T.: Evidence-Based Multi-Feature Fusion for Adversarial Robustness. IEEE Transactions on Pattern Analysis and Machine Intelligence **47**(10), 8923–8937 (2025). https://doi.org/10.1109/TPAMI.2025.3582518

31. Waseda, F., Chang, C.C., Echizen, I.: Rethinking Invariance Regularization in Adversarial Training to Improve Robustness-Accuracy Trade-off. In: International Conference on Learning Representations (2025), https://openreview.net/forum?id=M9SKazbVkJ
32. Wei, Z., Wang, Y., Guo, Y., Wang, Y.: CFA: Class-Wise Calibrated Fair Adversarial Training. In: IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 8193–8201 (2023). https://doi.org/10.1109/CVPR52729.2023.00792
33. Xiao, C., Zhong, P., Zheng, C.: Enhancing Adversarial Defense by k-Winners-Take-All. In: International Conference on Learning Representations (2020), https://openreview.net/forum?id=Pr86Lt1nOU
34. Xie, C., Wu, Y.: Feature Denoising for Improving Adversarial Robustness. In: IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 501–509 (2019). https://doi.org/10.1109/CVPR.2019.00059
35. Xu, K., Chen, R., Franchi, G., Yao, A.: Scaling for Training Time and Post-hoc Out-of-distribution Detection Enhancement. In: International Conference on Learning Representations (2024), https://openreview.net/forum?id=RDSTjtnqCg
36. Yan, H., Zhang, J., Niu, G., Feng, J., Tan, V.Y.F., Sugiyama, M.: CIFS: Improving Adversarial Robustness of CNNs via Channel-wise Importance-based Feature Selection. In: International Conference on Machine Learning. pp. 11693–11703. PMLR (2021), https://proceedings.mlr.press/v139/yan21e.html
37. Zagoruyko, S., Komodakis, N.: Wide Residual Networks. In: British Machine Vision Conference. pp. 87.1–87.12. British Machine Vision Association (2016). https://doi.org/10.5244/C.30.87
38. Zhang, H., Yu, Y., Jiao, J., Xing, E.P., Ghaoui, L.E., Jordan, M.I.: Theoretically Principled Trade-off between Robustness and Accuracy. In: International Conference on Machine Learning. pp. 7472–7482. PMLR (2019), https://proceedings.mlr.press/v97/zhang19p.html
39. Zhu, K., Wang, J., Hu, X., Xie, X., Yang, G.: Improving Generalization of Adversarial Training via Robust Critical Fine-Tuning. In: IEEE/CVF International Conference on Computer Vision. pp. 4401–4411 (2023). https://doi.org/10.1109/ICCV51070.2023.00408