Supplementary Materials
Stay Focused is All You Need for Adversarial Robustness

Bingzhi Chen
Beijing Institute of Technology, Zhuhai
Zhuhai, China
chenbingzhi.smile@gmail.com

Ruihan Liu
South China Normal University
Guangzhou, China
liuruihan@m.scnu.edu.cn

Yishu Liu
Harbin Institute of Technology, Shenzhen
Shenzhen, China
liuyishu@stu.hit.edu.cn

Xiaozhao Fang*
Guangdong University of Technology
Guangzhou, China
xzhfang168@126.com

Jiahui Pan
South China Normal University
Guangzhou, China
panjiahui@m.scnu.edu.cn

Guangming Lu*
Harbin Institute of Technology, Shenzhen
Shenzhen, China
luguangm@hit.edu.cn

Zheng Zhang
Harbin Institute of Technology, Shenzhen
Shenzhen, China
darrenzz219@gmail.com

1 Position of StayFocused

The pipeline of our proposed StayFocused framework comprises two essential modules: Spatial-aligned Hypersphere Constraint (SHC), and Channel-adapted Prompting Calibration (CPC). The SHC module is used before CPC. The purpose is to ensure that the model features minimize the hyperspheric distribution distance of the same type of features in the hyperspheric space before dynamically recalibrating the feature response, and at the same time generate angular boundary margins between different categories.

The CPC module is flexible to use on different layers of the DNNs. Therefore, we explore the performance of this method on different layers, taking the experimental results of CIFAR-10 with two-headed StayFocused on ResNet-18 model training as an example, as shown in Table 1. It can be seen from the experimental results that the deeper the layers are, the better the results of our proposed method. The reason for this phenomenon is that CPC can be seen as a feature filtering process, and if used in the first few layers of DNNs, it will make the model lose some detailed features, which will affect the learning of more advanced features by the deeper network, and will eventually lead to a decrease in the accuracy of the model. The deeper layers are more relevant to the final prediction, so using it in the deeper layer will have a significant effect enhancement.

Table 1: Comparison of accuracy (%) as our StayFocused insert CPC module after different layers of ResNet-18.

Layer	Clean	FGSM	PGD-20	PGD-100	C&W
Layer1	59.52	31.47	29.04	28.72	33.47
Layer2	78.09	37.62	34.52	34.05	38.95
Layer3	86.76	45.23	42.68	42.00	40.56
Layer4	88.08	68.08	65.45	64.62	65.26

2 Additional Evaluation on Larger Datasets

Table 2 illustrates the results with ResNet-18 backbone on CIFAR-100 and Tiny-ImageNet. Compared with FSR (CVPR'23) and AGAIN (CVPR'23), our method achieves noticeable improvements across different attack scenarios. The results from these experiments are consistent with our original findings and further corroborate the effectiveness of StayFocused.

Table 2: More evaluation o CIFAR-100 and Tiny-ImageNet.

Methods	CIFAR-100				Tiny-ImageNet			
	Clean	PGD-20	C&W	AA	Clean	PGD-20	C&W	AA
FSR	58.23	25.33	24.54	-	51.77	20.95	19.32	-
AGAIN	63.61	33.69	37.99	27.22	51.13	24.05	20.63	20.20
Ours	**64.11**	**42.74**	**40.13**	**37.28**	**51.84**	**27.27**	**22.29**	**22.43**
Increased	0.50%	9.05%	2.14%	10.06%	0.07%	3.22%	1.66%	2.23%

3 Effect of multi-head training

To assess the effects of multi-head training on StayFocused, we conducted thorough experiments specifically targeting multi-head training in Table 3 and Table 4. Observing the experimental results, it becomes evident that our StayFocused approach, which employs multi-head training, yields significant efficacy. Specifically, when the number of heads is set to four, the model's performance achieves a new state-of-the-art. Furthermore, it's worth noting that an increase in the number of heads correlates with a rise in computational overhead.

4 Theoretical and Robust Analysis.

Based on Eq. (11) in main text, the objective function for classification during the adversarial training phase can be formulated as follows [3]:

$$\begin{aligned}\mathcal{L}_{\mathrm{CPC}} &= \mathcal{L}_{\mathrm{CE}}(\sigma(\tilde{z}), y) \\ &= \mathcal{L}_{\mathrm{CE}}\big(\sigma\big(\alpha \cdot zw^{y} + (1-\alpha) \cdot zw'\big), y\big)\end{aligned} \tag{1}$$

where σ denotes the softmax activation function, w' denote the class weights corresponding to candidate classes. By incorporating

*Corresponding authors: Guangming Lu and Xiaozhao Fang.

Table 3: Comparisons of clean accuracy (%) and robust accuracy (%) against various adversarial attacks on the CIFAR-10 dataset.

CIFAR-10		ResNet-18							WideResNet-34-10						
Method	Multi-head	Clean	FGSM	PGD-20	PGD-50	PGD-100	C&W	AA	Clean	FGSM	PGD-20	PGD-50	PGD-100	C&W	AA
StayFocused	Head=2	88.08	68.08	65.45	64.94	64.62	65.26	61.13	87.99	63.34	58.19	57.28	57.24	57.05	57.79
StayFocused	Head=3	$\underline{89.02}$	$\underline{74.72}$	$\underline{74.19}$	$\underline{73.55}$	$\underline{73.39}$	**73.16**	62.10	88.31	68.95	$\underline{63.83}$	$\underline{63.68}$	$\underline{63.67}$	61.66	$\underline{58.10}$
StayFocused	Head=4	**89.80**	**76.87**	**75.81**	**75.59**	**74.94**	$\underline{72.24}$	**67.29**	$\underline{89.28}$	**77.44**	**77.03**	**76.73**	**75.76**	**70.14**	**62.60**

Table 4: Comparisons of clean accuracy (%) and robust accuracy (%) against various adversarial attacks on the SVHN dataset.

SVHN		ResNet-18							WideResNet-34-10						
Method	Multi-head	Clean	FGSM	PGD-20	PGD-50	PGD-100	C&W	AA	Clean	FGSM	PGD-20	PGD-50	PGD-100	C&W	AA
StayFocused	Head=2	93.14	67.79	63.81	62.79	62.54	$\underline{66.03}$	57.72	94.50	70.51	63.27	56.02	54.19	58.34	53.28
StayFocused	Head=3	$\underline{93.40}$	**72.52**	$\underline{67.58}$	$\underline{66.77}$	$\underline{66.57}$	65.99	**58.99**	$\underline{95.06}$	$\underline{79.44}$	$\underline{73.75}$	$\underline{68.21}$	$\underline{65.58}$	$\underline{70.47}$	$\underline{54.76}$
StayFocused	Head=4	**93.54**	$\underline{69.55}$	**68.42**	**67.98**	**67.86**	**68.15**	$\underline{57.89}$	**95.56**	**80.41**	**77.44**	**72.59**	**68.81**	**75.04**	**56.94**

the cross-entropy loss, $\mathcal{L}_{\text{CPC}}$ can be transformed to:

$$\begin{aligned}
\mathcal{L}_{\text{CPC}} &= -\log\left\{\frac{\exp\left[\alpha \cdot zw^{y} + (1-\alpha)\cdot zw'\right]}{\sum_{j=0}^{N}\exp\left(zw^{j}\right)}\right\} \\
&= -\left[\alpha \cdot zw^{y} + (1-\alpha)zw'\right] + \log\left[\sum_{j=0}^{N}\exp\left(zw^{j}\right)\right] \\
&= -\alpha\left(zw^{y} - zw'\right) - zw' + \log\left[\sum_{j=0}^{N}\exp\left(zw^{j}\right)\right] \\
&= -\alpha\left(zw^{y} - zw'\right) + \log\left[\frac{\sum_{j=0}^{N}\exp\left(zw^{j}\right)}{\exp\left(zw'\right)}\right]
\end{aligned} \tag{2}$$

where N is the number of classification categories. Suppose $h = zw^{y} - zw'$, Eq. (2) can be reformulated as:

$$\begin{aligned}
\mathcal{L}_{\text{CPC}} &= -\alpha h + \\
&\log\left\{\frac{\exp\left(zw'\right)\sum_{j=0}^{N}\left[\exp\left(zw^{j} - zw'\right)\right]}{\exp\left(zw'\right)}\right\} \\
&= -\alpha h + \log\left[\sum_{j\neq y}^{N-1}\exp\left(zw^{j} - zw'\right) + \exp\left(h\right)\right].
\end{aligned} \tag{3}$$

Theoretically, under ideal classification conditions,

$$\nabla_{h}\mathcal{L} = -\alpha + \frac{\exp(h)}{c + \exp\left(h\right)} = 0. \tag{4}$$

Consequently, we can infer the following:

$$zw^{y} - zw' = \log\left[\frac{\alpha \cdot \sum_{j\neq y}^{N-1}\exp\left(zw^{j} - zw^{k}\right)}{1-\alpha}\right]. \tag{5}$$

The following properties can be obtained from Eq. (5):

1) When α approaches 1, the model exclusively learns features associated with the true label and $zw^{y} - zw' = \infty$. This signifies that the trained model will indefinitely amplify the difference in logits between the predicted label and other labels. Excessive logit differences can result in overly confident predictions and a deficiency in adaptive capability [2]. Furthermore, from the perspective of activation features, when considering a dataset that includes N categories, to achieve $zw^{y} - zw' = \infty$, there will be $w^{y} - w' = \infty$. Multiple ineffective values approaching infinity will be present when satisfying the conditions mentioned for w. In the inference stage, there will be a substantial reduction in the number of activated features that effectively contribute to calculating the predicted probabilities. This hampers the model's ability to incorporate knowledge from diverse activated features, thereby impacting its generalization capability.

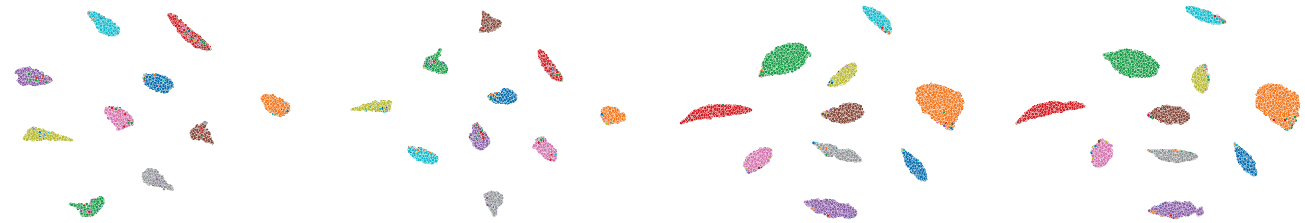
2) When $\alpha \in [0.5, 1)$, $zw^{y} - zw'$ tends to a constant C with respect to α. The logit difference between the true class and the predicted Top-K classes does not diverge to infinity. Our method trains the model to focus not only on the activated features of the predicted class but also on the activated features of relevant classes. There will be more valid values in w available for inference, which benefits the model's generalization capability.

5 Algorithm of StayFocused.

Our StayFocused method is summarized in Algorithm 1.

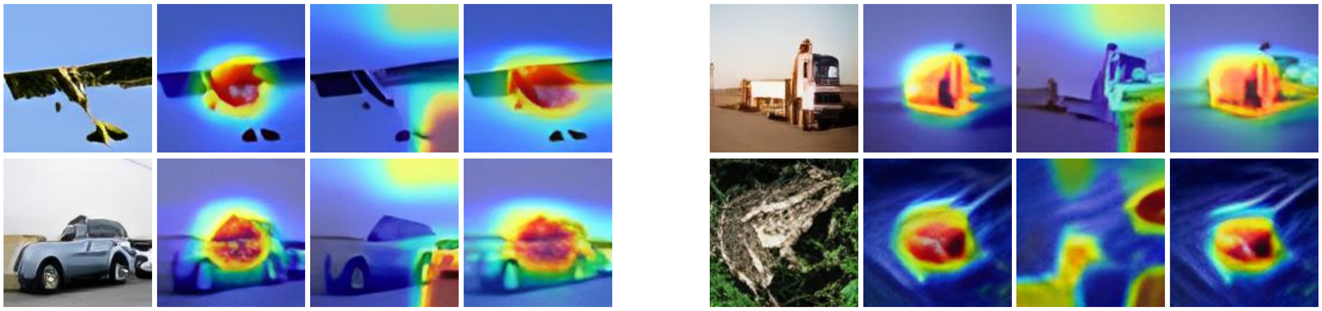
6 Additional Visualization Results

To further verify the effectiveness of StayFocused, we use Grad-cam [1] to supplement the visualization experiment of the feature representation distribution of clean and adversarial samples. Figure 1 illustrates the outcomes achieved through StayFocused training applied to the SVHN datasets across ResNet-18 and WideResNet-34-10 architectures. As illustrated in Figure 2, our approach consistently focuses its attention on the most relevant regions corresponding to the ground-truth labels. From the experimental results, it can be seen that our method can better classify clusters, which are relatively compact within the same category and relatively separated within different categories.



(a) CIFAR-10, Ours, Clean (b) CIFAR-10, Ours, Adversarial (c) SVHN, Ours, Clean (d) SVHN, Ours, Adversarial

Figure 1: T-SNE Visualization of the discriminative features learned by our StayFocused method on WideResNet-34-10.



(a) Origin (b) Clean (c) Adversarial (d) Ours (a) Origin (b) Clean (c) Adversarial (d) Ours

Figure 2: Illustration of activation map visualization.

Algorithm 1 AT with StayFocused

```
Input: Initialized adversarial encoder $\mathcal{F}$, clean encoder $\mathcal{F}_\phi$,
    Minibatch image $\mathcal{D}$, multi-head number $n$.
Output: Robust encoder $\mathcal{F}$.
for $(x, y)$ in $\mathcal{D}$ do
    Obtain adversarial examples $x'$ via PGD-10;
    for $i$ in Range $(0, n)$ do
        if $i \geq 1$ then
            Stage1: Compute $\mathcal{L}_{\mathrm{KL}}(f_\phi(x), f_\phi(\mathrm{Mask}(x)))$;
        end if
        Stage2: Spatial-aligned constraint with hypersphere contrastive learning;
        Compute $\mathcal{L}_{\mathrm{SHC}}$ with Eq. (10) in main text;
        Stage3: Channel-wise recalibration with top-$K$ prompts;
        Compute $\mathcal{L}_{\mathrm{CPC}}$ with Eq. (13) in main text;
    end for
    Cross-optimize parameters of $\mathcal{F}$ and $\mathcal{F}_\phi$ with Eq. (18) in main text.
end for
```

References

[1] Ramprasaath R Selvaraju, Michael Cogswell, Abhishek Das, Ramakrishna Vedantam, Devi Parikh, and Dhruv Batra. 2017. Grad-cam: Visual explanations from deep networks via gradient-based localization. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*. 618–626.

[2] Christian Szegedy, Vincent Vanhoucke, Sergey Ioffe, Jon Shlens, and Zbigniew Wojna. 2016. Rethinking the inception architecture for computer vision. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 2818–2826.

[3] Shenglin Yin, Kelu Yao, Sheng Shi, Yangzhou Du, and Zhen Xiao. 2023. AGAIN: Adversarial Training With Attribution Span Enlargement and Hybrid Feature Fusion. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 20544–20553.