Supplementary Materials
R-TPT: Improving Adversarial Robustness of Vision-Language Models through Test-Time Prompt Tuning

Lijun Sheng[1,2], Jian Liang[2,3*], Zilei Wang[1], Ran He[2,3]
[1] University of Science and Technology of China
[2] NLPR & MAIS, Institute of Automation, Chinese Academy of Sciences
[3] University of Chinese Academy of Sciences
slj0728@mail.ustc.edu.cn, liangjian92@gmail.com

1. Algorithm

Here, we provide the pseudocode algorithm of R-TPT to show the process of our proposed defense method clearly.

Algorithm 1 Pseudocode of R-TPT.

Require: Test sample x_t, CLIP model.
▷ Augment x_t via AugMix to obtain views $\{x_i\}_{i=0}^{N}$ and select low-entropy views $\mathcal{B}$.
▷ Update textual prompts via minimizing pointwise entropy of selected views $\mathcal{B}$ via Eq.4.
▷ Obtain the reliability $\{r_i\}_{i=0}^{N}$ of all views via Eq.6.
▷ Obtain the robust prediction by ensembling the predictions $\{p_i\}_{i=0}^{N}$ of all views weighted by the reliability $\{r_i\}_{i=0}^{N}$.

2. Datasets

We provide the content, number of categories and number of images of all datasets involved in the experimental section in Table 1.

Dataset	Description	# Classes	# Test
Caltech101	Object images	100	2,465
Pets	Pet images	37	3,669
Cars	Car images	196	8,041
Flower102	Flower images	102	2,463
Aircraft	Aircraft images	100	3,333
DTD	Describable textures dataset	47	1,692
EuroSAT	Sentinel-2 satellite images	10	8,100
UCF101	Human action images	101	3,783
ImageNet	Object and scene images	1,000	50,000
ImageNet-A	Adversarially filtered images	200	7,500
ImageNet-V2	New test images	1,000	10,000
ImageNet-R	Rendered images	200	30,000
ImageNet-S	Sketch-style images	1,000	50,889

Table 1. Introduction of all datasets involved in experiments.

3. Experimental Results

3.1. Results of Larger Backbone.

We evaluate our method using the CLIP-ViTL/14 model [3] and present the results in Table 2. Our experiments demonstrate that R-TPT outperforms all baseline methods in terms of defense performance, highlighting its robustness even when applied to large-scale backbone architectures. Also, we observe that, in terms of clean adaptation performance, only TPT and C-TPT exhibit positive gains, whereas the remaining methods suffer from negative transfer.

3.2. Results Compared with Training-time defense Methods.

Training-time defense methods [1, 2, 4] typically rely on labeled data and robust pre-trained checkpoints to achieve their performance. To ensure a fair comparison, we have focused our main text on test-time baselines that utilize the same resources as our proposed method. Here, we provide a comprehensive evaluation of training-time methods on fine-grained datasets in Tables 3, 4 to highlight the competitive performance of R-TPT, even in the absence of external data and pre-trained robust checkpoints. It is shown that R-TPT not only remains competitive with training-time methods but also achieves significantly better performance on clean samples. More importantly, R-TPT can further improve the robustness of training-time methods.

References

[1] Lin Li, Haoyan Guan, Jianing Qiu, and Michael Spratling. One prompt word is enough to boost adversarial robustness

*To whom correspondence should be addressed.

Method	Caltech101		Pets		Cars		Flower102		Aircraft		DTD		EuroSAT		UCF101		Avg.	
	Acc.	Rob.	Acc.	Rob.	Acc.	Rob.	Acc.	Rob.	Acc.	Rob.	Acc.	Rob.	Acc.	Rob.	Acc.	Rob.	Acc.	Rob.
CLIP [3]	95.2	0.1	93.1	0.0	76.8	0.0	76.2	0.0	30.0	30.0	52.4	0.0	**55.1**	0.0	73.7	0.0	69.1	3.8
Ensemble	94.9	83.6	93.4	63.5	76.3	40.5	75.0	48.6	31.7	31.7	51.3	31.3	38.7	11.1	71.7	48.3	66.6	44.8
TPT [5]	**95.9**	0.2	93.8	0.0	78.0	0.0	**76.9**	0.0	31.6	31.6	55.1	0.0	51.8	0.0	74.7	0.0	69.7	4.0
C-TPT [6]	95.6	0.1	**94.3**	0.0	77.4	0.0	76.3	0.0	30.4	30.4	**55.4**	0.0	54.0	0.0	**75.1**	0.0	**69.8**	3.8
MTA [7]	95.8	83.1	93.7	64.9	**78.4**	36.6	76.1	44.2	**32.7**	32.7	53.4	27.2	47.8	7.5	74.7	47.5	69.1	43.0
R-TPT	95.7	88.2	93.7	72.9	77.2	49.1	76.2	55.6	31.7	31.7	54.0	38.0	44.3	20.4	74.3	55.6	68.4	51.4

Table 2. Results (%) of various adaptation methods on **fine-grained classification datasets** with pre-trained CLIP-ViT-L/14 ($\epsilon = 4.0$).

Method	Caltech101		Pets		Cars		Flower102		Aircraft		DTD		EuroSAT		UCF101		Avg.	
	Acc.	Rob.	Acc.	Rob.	Acc.	Rob.	Acc.	Rob.	Acc.	Rob.	Acc.	Rob.	Acc.	Rob.	Acc.	Rob.	Acc.	Rob.
CLIP [3]	85.9	2.6	83.6	0.0	55.7	0.0	**61.7**	0.0	15.7	15.7	40.4	0.8	23.7	0.0	59.0	0.0	53.2	2.4
TeCoA^1 [2]	78.3	78.3	76.0	75.8	22.4	22.3	33.5	33.4	5.8	5.8	26.2	26.0	16.5	16.6	38.4	38.1	37.1	37.0
APT^1[1]	2.9	1.7	31.9	3.8	8.5	0.6	2.6	1.1	0.9	0.9	16.6	7.9	17.0	4.0	11.2	0.9	11.4	2.6
APT^1+TeCoA^1 [1]	82.8	82.8	79.3	79.0	33.9	33.6	42.7	42.6	9.9	9.9	39.2	39.0	**32.9**	32.9	51.5	51.4	46.5	46.4
R-TPT	**86.7**	79.8	**84.6**	74.2	**58.1**	42.9	60.6	51.9	**17.5**	17.5	**41.3**	33.5	21.2	15.9	**59.7**	50.9	**53.7**	45.8

Table 3. Results (%) of training-time defense methods on **fine-grained classification datasets** with pre-trained ResNet50 ($\epsilon = 1.0$).

Method	Caltech101		Pets		Cars		Flower102		Aircraft		DTD		EuroSAT		UCF101		Avg.	
	Acc.	Rob.	Acc.	Rob.	Acc.	Rob.	Acc.	Rob.	Acc.	Rob.	Acc.	Rob.	Acc.	Rob.	Acc.	Rob.	Acc.	Rob.
CLIP [3]	**91.4**	0.2	**85.1**	0.0	60.1	0.0	**64.0**	0.0	18.1	18.1	**43.0**	0.0	**35.8**	0.0	61.6	0.0	**57.4**	2.3
TeCoA^4 [2]	79.3	78.0	66.9	63.7	10.2	9.1	30.8	28.9	6.6	6.6	24.5	24.0	14.5	14.3	34.6	33.4	33.4	32.2
FARE^4 [4]	86.3	85.4	76.7	73.8	39.2	34.4	37.0	34.0	9.5	9.5	28.3	27.3	16.6	16.3	44.2	41.9	42.2	40.3
APT^4 [1]	10.7	0.4	10.0	0.2	1.5	0.1	0.9	0.2	2.6	2.6	9.0	0.1	7.8	6.7	3.7	0.2	5.8	1.3
APT^4+TeCoA^4 [1]	81.4	80.2	66.7	63.9	20.8	18.9	42.5	40.4	5.2	5.2	35.2	33.7	29.3	29.2	40.2	39.4	40.2	38.9
R-TPT	90.6	76.4	84.5	55.8	**63.1**	28.4	62.6	37.6	**19.1**	19.1	42.1	29.1	32.0	5.1	**62.8**	41.0	57.1	36.6

Table 4. Results (%) of training-time defense methods on **fine-grained classification datasets** with pre-trained ViT-B/32 ($\epsilon = 4.0$).

for pre-trained vision-language models. In *Proc. CVPR*, 2024. 1, 2

[2] Chengzhi Mao, Scott Geng, Junfeng Yang, Xin Wang, and Carl Vondrick. Understanding zero-shot adversarial robustness for large-scale models. In *Proc. ICLR*, 2023. 1, 2

[3] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In *Proc. ICML*, 2021. 1, 2

[4] Christian Schlarmann, Naman Deep Singh, Francesco Croce, and Matthias Hein. Robust clip: Unsupervised adversarial fine-tuning of vision embeddings for robust large vision-language models. In *Proc. ICML*, 2024. 1, 2

[5] Manli Shu, Weili Nie, De-An Huang, Zhiding Yu, Tom Goldstein, Anima Anandkumar, and Chaowei Xiao. Test-time prompt tuning for zero-shot generalization in vision-language models. In *Proc. NeurIPS*, 2022. 2

[6] Hee Suk Yoon, Eunseop Yoon, Joshua Tian Jin Tee, Mark Hasegawa-Johnson, Yingzhen Li, and Chang D Yoo. C-tpt: Calibrated test-time prompt tuning for vision-language models via text feature dispersion. In *Proc. ICLR*, 2024. 2

[7] Maxime Zanella and Ismail Ben Ayed. On the test-time zero-shot generalization of vision-language models: Do we really need prompt learning? In *Proc. CVPR*, 2024. 2