UNLEASHING THE POWER OF VISUAL PROMPTING AT THE PIXEL LEVEL

Anonymous authors
Paper under double-blind review

1 IMPLEMENTATION DETAILS

Our implementation is based on Pytorch (Paszke et al., 2019). We use CLIP-B/32, Instagram (Mahajan et al., 2018), and ResNet50 (He et al., 2016) as our pre-trained model, and the batch size is 256, 32, 128, respectively. All visual prompting in our experiments are trained for 1000 epochs. For EVP, we use SGD with a cosine learning rate schedule; the intiial learning rate is 70. The prompting size is 30 pixels by default. To fairly compare with VP, we follow its text prompting setup (Bahng et al., 2022) in CLIP model. Specifically, we use "This is a photo of a [LABEL]" by default for the text prompting. For CLEVR datasets, we use "This is a photo of [LABEL] objects", for DMLab datasets, we use "The distance is [LABEL1], and the reward is [LABEL2]", and for Camelyon17, the text prompting template is "a tissue region [LABEL] tumor".

2 PROMPTING SIZE

The prompting size is defined as $p = \frac{K-k}{2}$, where k is the image size after shirinking, and K is the input size of pre-trained model. Therefore, the number of parameters is $12p(K-p)$, which only depends on p since K is fixed for a given model. In our experiment, the optimal prompting size varies across datasets, as shown in Fig. 1. Since we shrink the original image and pad learnable pixels around it, there shows a tradeoff between the image resolution and the number of parameters. Interestingly, we note that, for datasets with a low resolution (*e.g.*, CIFAR100), the prompting of p=30 achieves the best performance. While for datasets with a high resolution, we note setting p to a small value empirically works the best. For example, we find that p=5 is the best in Food101 dataset which has a resolution of 512×512. The best hyper-parameter is shown in Tab. 1; we note setting prompting size to 30 generally achieve the best overall accuracy on these four datasets.

3 PERFORMANCE UNDER DIFFERENT CORRUPTION CASES

In section 4.3, we see that our EVP outperforms other methods on corruption setting. Here, we list the generalization performance of all methods under various types of corruptions, as shown in Tab. 2 and Tab. 3. Specifically, compared to VP, we note 1) on CIFAR-10-C, EVP yields the largest improvement on constrast (+8.8%) and the smallest improvement on brightness(+3.5%); 2) on CIFAR-100-C, EVP yields the largest improvement on contrast (+11.0%) and the smallest improvement on impulse noise (+1.1%).

4 PERFORMANCE COMPARSION WITH VPT-DEEP

VPT-DEEP is an advanced version of VPT, which additionally introduces learnable tokens at every Transformer layer's input space for enhancing performance. We hereby briefly compare its performance to that of EVP. specifically, we compare the performance of EVP and VPT in three settings: CLIP-model, OOD, and corruption.

Table 1: **The optimal prompting size in our experiments** across 12 datasets on CLIP.

Image size	CIFAR100	CIFAR10	Flowers	Food	EuroSAT	SUN	DMLab	SVHN	Pets	DTD	RESISC	CLEVR
Prompt size (p)	30	30	30	5	30	30	30	30	20	30	30	30

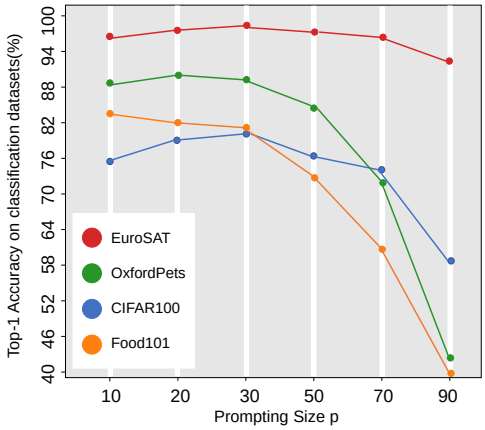

Figure 1: **Ablation on prompting size.** The pre-trained model is CLIP-B/32. We vary the prompting size, which determines the number of parameters, and show the performance on four datasets.

Table 2: Specific performance on CIFAR-10C

Methods	brightness	contrast	defocus_blur	elastic	fog	frost	gaussian_blur	gaussian_noise	glass_blur	impulse_noise
VP	91.9	81.6	87.8	82.0	86.5	84.8	85.4	61.4	60.7	61.9
VPT	87.6	78.0	82.0	73.9	78.8	77.1	79.9	46.8	46.0	55.8
EVP(Ours)	95.4	90.3	93.1	87.7	91.5	89.8	91.6	66.3	69.0	67.8
LP	92.9	85.0	88.6	82.1	87.0	85.6	86.2	56.2	55.1	62.1
FT	94.9	91.6	92.1	84.2	91.2	88.5	91.1	57.1	59.5	68.7
Methods	jpeg_compression	motion_blur	pixelate	saturate	shot_noise	snow	spatter	speckle_noise	zoom_blur	Avg.
VP	74.3	79.7	66.6	88.9	67.5	84.8	87.3	69.1	84.6	78.2
VPT	63.9	72.5	58.1	84.1	56.1	77.9	80.6	59.0	75.9	70.2
EVP(Ours)	80.4	86.3	75.3	93.2	73.6	89.6	91.0	75.2	90.3	84.3
LP	75.1	80.8	77.9	90.5	65.0	86.2	88.6	67.8	85.2	78.8
FT	74.3	86.8	77.3	93.0	66.4	89.0	91.0	69.4	89.0	82.7

Table 3: Specific performance on CIFAR-100C

Adaptation	brightness	contrast	defocus_blur	elastic	fog	frost	gaussian_blur	gaussian_noise	glass_blur	impulse_noise
VP	71.3	55.5	65.7	57.9	61.5	58.0	62.2	30.6	27.6	37.4
VPT	72.8	62.8	67.9	57.9	64.6	58.7	65.4	30.3	25.8	41.0
EVP(Ours)	77.7	66.5	73.2	63.5	69.1	64.2	70.2	36.3	31.8	38.5
LP	75.5	63.1	69.8	60.0	66.2	63.4	66.4	31.1	30.8	39.4
FT	80.7	73.1	75.7	62.9	72.9	66.6	73.6	34.8	31.9	46.1
Adaptation	jpeg_compression	motion_blur	pixelate	saturate	shot_noise	snow	spatter	speckle_noise	zoom_blur	Avg.
VP	47.4	55.2	46.1	61.7	37.7	59.3	63.5	38.6	61.1	52.5
VPT	45.2	57.8	45.0	64.3	38.0	60.5	64.9	39.4	63.4	54.0
EVP(Ours)	52.8	63.5	51.4	69.1	43.8	65.4	68.6	44.6	68.0	58.6
LP	49.5	61.4	57.2	66.8	39.7	64.7	67.8	42.4	65.8	56.9
FT	48.3	66.5	52.6	73.5	42.9	69.1	72.9	44.3	70.9	61.1

Table 4: Performance comparison across 12 datasets with CLIP. We note EVP outperforms VPT-deep with fewer tunable parameters. The results where EVP outperforms VPT-deep are highlight in **bold**.

Adaptation	Tunable params (M)	CIFAR100	CIFAR10	Flowers	Food	EuroSAT	SUN	DMLab	SVHN	Pets	DTD	RESISC	CLEVR	Avg.
VPT-D	0.092	78.3	96.1	84.4	85.6	97.4	70.2	57.7	90.1	92.5	70.1	90.6	69.7	81.9
EVP(Ours)	0.062	**81.2**	**96.6**	82.3	84.1	**97.6**	**71.0**	**62.3**	**90.5**	90.0	68.4	89.7	**75.9**	82.5

Table 5: Robustness comparison on **out-of-distribution** and **corruption** datasets. Left: out-of-distribution datasets. Right: corruption datasets. We can observe that EVP achieves much stronger robustness on both out-of-distribution setting and corruption setting.

Model	Adaptation	iwildcam	camelyon17	fmow	Avg.
CLIP	VPT-D	62.7	93.6	39.3	65.2
CLIP	Ours	64.9	**95.1**	40.2	**66.7**

Model	Adaptation	CIFAR100-C	CIFAR10-C	Avg.
CLIP	VPT-D	56.3	82.6	69.5
CLIP	Ours	58.6	**84.3**	71.5

4.1 Performance on CLIP-model

In this section, based on CLIP model, we conducted a comparative analysis of the performance of EVP and VPT-deep on 12 classification dataset. The results are shown in Tab. 4. We can see that EVP demonstrates an average performance improvement of 0.6% over VPT-deep(82.5% *v.s.* 81.9%), with only 0.062 Million tunable parameters, which is 0.03 million fewer than VPT-deep.

4.2 Performance on Robustness

Following the setting in main text, we test the robustness of EVP and VPT-deep to distribution shift and common image corruption. Tab. 5 presents the specific conparative results. In both the OOD (+1.5%) and corruption settings (+2.0%), EVP achieves superior performance consistently compared to VPT-deep, which demonstrates the robustness of EVP.

References

Hyojin Bahng, Ali Jahanian, Swami Sankaranarayanan, and Phillip Isola. Exploring Visual Prompts for Adapting Large-Scale Models. In *arxiv*, 2022.

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *CVPR*, 2016.

Dhruv Mahajan, Ross Girshick, Vignesh Ramanathan, Kaiming He, Manohar Paluri, Yixuan Li, Ashwin Bharambe, and Laurens Van Der Maaten. Exploring the limits of weakly supervised pretraining. In *ECCV*, 2018.

Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, et al. Pytorch: An imperative style, high-performance deep learning library. In *NeurIPS*, 2019.