Learn to Preserve and Diversify: Parameter-Efficient Group with Orthogonal Regularization for Domain Generalization —Supplementary—

Jiajun Hu[1,2], Jian Zhang[1,2], Lei Qi[3,*], Yinghuan Shi[1,2,*], Yang Gao[1,2]

[1] State Key Laboratory for Novel Software Technology, Nanjing University, China
[2] National Institute of Healthcare Data Science, Nanjing University, China
[3] School of Computer Science and Engineering, Key Lab of Computer Network and Information Integration (Ministry of Education), Southeast University, China

A Algorithm

Algorithm 1 Parameter-Efficient Group with Orthogonal Regularization

Require: training data D_{tr}, pre-trained vision transformer $F(\cdot;\theta)$ with B blocks, classification head $H(\cdot;\psi)$, group of LoRA modules $g(\cdot;\phi)$, balancing coefficient α, iteration T

Initialization: Inject $g(\cdot;\phi)$ into $F(\cdot;\theta)$ to get the pre-trained model with group of LoRA modules $G(\cdot;\Phi)$ and freeze the pre-trained model weight
for $t = 1, 2, ..., T$ **do**
  sample a batch (x, y) in D_{tr}
  $\mathcal{L}_{cls} \leftarrow \mathcal{L}_{CE}(H(G(x;\Phi);\psi), y)$ ▷ Eq. (2)
  $\mathcal{L}_{OR} \leftarrow 0$
  for $b = 1, 2, ..., B$ **do**
    $\mathcal{L}_{OR} \leftarrow \mathcal{L}_{OR} + (\mathcal{L}_O(W_b^q) + \mathcal{L}_O(W_b^v))$ ▷ Eq. (8)
  end for
  $\mathcal{L}_{final} \leftarrow \mathcal{L}_{cls} + \alpha\mathcal{L}_{OR}$ ▷ Eq. (9)
  update $g(\cdot;\phi)$, $H(\cdot;\psi)$ to minimize $\mathcal{L}$.
end for
Merge the LoRA group with the pre-trained weight. ▷ Eq. (10)
return G, H

B Evaluation Protocol and Hyperparameters Search

In this section, we provide a detailed description of our evaluation protocol and hyperparameters (HPs) search. In line with prior research in DG, we designate one domain within the dataset as the unseen test domain, while the remaining domains serve as source domains. The final experimental results are obtained by

Table 1: The hyperparameter N used on five DomainBed benchmarks in our experiments.

Hyperparameter	PACS	VLCS	OH	TI	DN
N	2	4	4	4	4

Table 2: Performance comparison with more methods. Leave-one-domain-out accuracy (%) on five DomainBed benchmarks.

Algorithm	PACS	VLCS	OH	TI	DN	Avg
Auto-RGN [14]	90.3±0.5	80.7±0.3	76.7±0.5	48.5±0.6	51.2±0.7	69.5
CoOp [24]	96.1±0.2	80.5±0.6	84.2±0.1	49.4±0.6	59.3±0.1	73.9
UPT [23]	**96.5±0.2**	82.7±0.1	**84.4±0.2**	54.9±0.9	**60.2±0.1**	75.7
PEGO	**96.5±0.1**	**83.2±0.3**	84.2±0.1	**57.3±0.3**	59.3±0.1	**76.1**

averaging the accuracies across all test domains. To maintain consistency with DomainBed [8], 20% of the samples from each source domain are allocated for validation and we adopt the training-domain validation strategy for hyperparameter search and model selection. Furthermore, all experiments are conducted using three different random seeds to ensure the reliability and reproducibility of our experiments.

As for algorithm-agnostic HPs in DomainBed (*e.g.*, learning rate, dropout, weight decay), to reduce the training overhead caused by HPs search, we do not tune any algorithm-agnostic HPs. Specifically, for all the experiments, the learning rate, dropout, and weight decay are fixed to 5e-4, 0, and 0. As regards the algorithm-specific HPs, we fix the rank of LoRA [11] r to 4 and the balance coefficient α to 1e-3 for all the experiments. We only search for the number of LoRA modules N from $\{2, 4, 6\}$. Tab. 1 provides a summary of the searched hyperparameter N on five DomainBed benchmarks in our experiments.

As shown in the ablation experiments of the main body (Sec. 5.1, Pages 12-13), the performance of our method is not sensitive to algorithm-specific HPs. Besides, to save GPU memory, we use half-precision (FP16) during training and inference for all the experiments.

C Comparisons with More Methods

In this subsection, we conduct a performance comparison between PEGO and more methods, including Auto-RGN [14], CoOp [24], and UPT [23]. Auto-RGN measures the Relative Gradient Norm (RGN) of each transformer layer and sets different learning rates for each layer by its RGN. CoOp and UPT are both Prompt Learning methods that introduce learnable text or visual prompts for fine-tuning. As shown in Tab. 2, our method achieves better average performance

Table 3: Trainable Parameters of Different Methods.

	FT	Adapter [10]	LoRA [11]	VPT [12]	CoOp [24]	UPT [23]	Auto-RGN [14]	PEGO
Parameters	86M	0.16M	0.15M	0.10M	2048	0.57M	86M	0.29M

Table 4: Leave-one-domain-out accuracy (%) of each domain on PACS when using ViT-B/16 pre-trained by CLIP as the backbone.

Algorithm	A	C	P	S	Avg
ERM (FT)	80.5±3.4	86.4±0.6	93.4±1.0	73.2±3.9	83.4±0.4
MIRO [4]	95.6±0.6	96.6±0.2	99.7±0.1	90.7±2.5	95.6±0.6
Adapter [10]	91.8±0.2	93.1±0.4	98.8±0.1	84.4±1.6	92.0±0.5
LoRA [11]	**97.4±0.3**	97.5±0.1	99.7±0.1	89.2±0.4	96.0±0.1
VPT [12]	97.1±0.4	97.8±0.1	**99.9±0.0**	90.1±0.9	96.2±0.3
L^2-SP [17]	93.9±1.0	94.3±0.6	97.8±0.3	83.1±2.3	92.2±0.7
LwF [18]	93.2±1.4	94.2±0.7	98.5±0.2	88.8±0.4	93.6±0.6
LP-FT [13]	89.1±2.8	97.8±0.1	99.8±0.0	89.9±0.2	94.2±0.7
PEGO	97.1±0.1	**98.5±0.2**	99.7±0.1	**90.9±0.2**	**96.5±0.1**

than other methods benefiting from the proposed preserving and diversifying losses.

D Trainable Parameters of Different Methods

The trainable parameters for each dataset are different due to the dimension difference of the classifier. We compare the trainable parameters of all methods on the PACS dataset. As shown in Tab. 3, our method is significantly parameter-efficient compared to FT (0.29M *vs*. 86M).

E Detail Results of Each Domain

In this section, Tabs. 4 to 8 provide the detailed accuracy of algorithms on five DomainBed [8] benchmarks: PACS [16], VLCS [7], OfficeHome [22], TerraIncognita [2] and DomainNet [19]. Since SWAD [3], SMA [1], and GESTUR [15] do not report the detailed results of each domain in their papers, we only present the results of ERM, MIRO [4], Adapter [10], LoRA [11], VPT [12], L^2-SP [17], LP-FT [13], LwF [18] and PEGO.

Table 5: Leave-one-domain-out accuracy (%) of each domain on VLCS when using ViT-B/16 pre-trained by CLIP as the backbone.

Algorithm	C	L	S	V	Avg
ERM (FT)	95.4±0.6	65.6±0.9	72.9±2.2	69.9±2.2	75.9±1.1
MIRO [4]	98.9±0.5	67.1±1.0	81.9±0.4	81.2±0.2	82.3±0.2
Adapter [10]	95.7±0.2	65.9±0.9	79.5±0.7	78.0±0.7	79.8±0.4
LoRA [11]	96.1±0.4	**68.1±0.2**	83.5±0.3	83.1±0.4	82.7±0.0
VPT [12]	96.8±0.5	67.2±0.3	**84.9±0.2**	82.6±0.4	82.9±0.3
LP-FT [13]	94.5±0.3	62.0±0.3	76.4±1.3	77.0±2.9	77.5±0.4
L^2-SP [17]	96.8±0.9	66.2±1.0	78.5±1.6	82.5±0.2	81.0±0.2
LwF [18]	**99.1±0.3**	65.5±1.4	80.4±1.2	82.6±0.2	81.9±0.4
PEGO	96.4±0.1	67.8±0.5	83.3±0.3	**85.2±1.0**	**83.2±0.3**

Table 6: Leave-one-domain-out accuracy (%) of each domain on OfficeHome when using ViT-B/16 pre-trained by CLIP as the backbone.

Algorithm	A	C	P	R	Avg
ERM (FT)	59.2±1.3	56.1±0.6	74.8±0.1	75.4±0.8	66.4±0.4
MIRO [4]	80.8±0.1	72.2±0.5	88.6±0.3	88.5±0.2	82.5±0.1
Adapter [10]	67.1±1.2	61.7±0.4	81.5±0.5	81.3±0.6	72.9±0.4
LoRA [11]	83.2±0.2	71.8±0.4	89.1±0.2	**89.5±0.2**	83.4±0.1
VPT [12]	82.9±0.6	71.5±0.6	89.7±0.1	**89.5±0.3**	83.4±0.3
L^2-SP [17]	62.6±1.3	57.1±0.4	76.4±0.8	76.6±0.2	68.2±0.5
LP-FT [13]	64.5±1.4	68.0±0.4	76.7±0.3	79.0±0.2	72.0±0.4
LwF [18]	79.0±1.7	70.4±0.7	86.8±0.3	86.7±0.4	80.7±0.4
PEGO	**83.7±0.3**	**73.3±0.4**	**90.3±0.3**	**89.5±0.3**	**84.2±0.1**

Table 7: Leave-one-domain-out accuracy (%) of each domain on TerraIncognita when using ViT-B/16 pre-trained by CLIP as the backbone.

Algorithm	L100	L38	L43	L46	Avg
ERM (FT)	38.1±0.3	26.7±2.5	41.9±1.3	34.4±1.8	35.3±0.6
MIRO [4]	**65.0±0.6**	46.7±0.7	60.8±1.3	44.9±0.1	54.3±0.3
Adapter [10]	38.8±5.1	44.9±2.0	56.2±0.3	37.8±1.3	44.4±0.8
VPT [12]	55.0±3.9	52.6±1.3	61.3±0.4	47.8±0.4	54.2±0.7
LoRA [11]	54.6±2.4	52.7±1.2	61.2±0.8	**50.5±0.5**	54.8±0.6
LP-FT [13]	42.8±4.2	33.2±3.3	46.7±1.1	33.2±1.1	39.0±1.5
L^2-SP [17]	45.6±5.5	27.2±3.5	49.9±1.3	34.8±0.3	39.4±1.6
LwF [18]	44.4±1.8	34.9±2.6	47.5±1.3	30.9±3.8	39.4±0.6
PEGO	63.2±0.3	**56.4±0.3**	**61.8±1.0**	47.9±0.5	**57.3±0.3**

Table 8: Leave-one-domain-out accuracy (%) of each domain on DomainNet when using ViT-B/16 pre-trained by CLIP as the backbone.

Algorithm	clipart	infograph	painting	quickdraw	real	sketch	Avg
ERM (FT)	68.0±0.1	22.5±0.4	46.5±2.4	18.5±0.6	58.7±1.6	52.5±0.7	44.4±0.5
MIRO [4]	74.9±0.1	37.1±0.2	59.8±0.4	18.7±0.8	72.2±0.1	61.2±0.6	54.0±0.2
Adapter [10]	75.6±0.2	37.6±0.2	63.1±0.2	19.4±0.3	77.2±0.1	64.2±0.3	56.2±0.1
LoRA [11]	76.4±0.1	43.3±0.3	63.6±0.3	**19.5±0.3**	79.2±0.1	66.4±0.1	58.1±0.1
VPT [12]	76.7±0.0	43.1±0.3	66.6±0.1	19.4±0.2	80.3±0.0	67.4±0.1	58.9±0.1
LP-FT [13]	70.9±0.2	26.7±0.3	55.8±0.3	17.1±0.5	66.3±0.4	57.5±0.4	49.1±0.3
L^2-SP [17]	70.6±0.1	28.4±0.3	55.6±0.5	18.3±0.5	68.5±0.4	58.4±0.1	50.0±0.2
LwF [18]	73.2±0.1	30.6±0.3	58.0±0.5	18.6±0.4	69.1±0.2	60.8±0.0	51.7±0.1
PEGO	**76.8±0.1**	**44.6±0.2**	**67.1±0.3**	18.8±0.2	**80.5±0.1**	**67.7±0.1**	**59.3±0.1**

F Limitation

Although our method cannot be easily applied to some traditional convolutional neural networks not containing linear layers (*e.g.*, ResNet [9]), it can be applied to any type of Transformer [21] architecture, similar to LoRA. With the increasing number of Transformer-based architectures being proposed (*e.g.*, ViT [5], ConViT [6], DeiT [20]), our method exhibits a wide range of applications for these networks.

References

1. Arpit, D., Wang, H., Zhou, Y., Xiong, C.: Ensemble of averages: Improving model selection and boosting performance in domain generalization. In: Conference on Neural Information Processing Systems (NeurIPS). pp. 8265–8277 (2022)
2. Beery, S., Van Horn, G., Perona, P.: Recognition in terra incognita. In: European Conference on Computer Vision (ECCV). pp. 456–473 (2018)
3. Cha, J., Chun, S., Lee, K., Cho, H.C., Park, S., Lee, Y., Park, S.: SWAD: Domain generalization by seeking flat minima. In: Conference on Neural Information Processing Systems (NeurIPS). pp. 22405–22418 (2021)
4. Cha, J., Lee, K., Park, S., Chun, S.: Domain generalization by mutual-information regularization with pre-trained models. In: European Conference on Computer Vision (ECCV). pp. 440–457 (2022)
5. Dosovitskiy, A., Beyer, L., Kolesnikov, A., Weissenborn, D., Zhai, X., Unterthiner, T., Dehghani, M., Minderer, M., Heigold, G., Gelly, S., et al.: An image is worth 16x16 words: Transformers for image recognition at scale. In: International Conference on Learning Representations (ICLR) (2021)
6. d'Ascoli, S., Touvron, H., Leavitt, M.L., Morcos, A.S., Biroli, G., Sagun, L.: ConVit: Improving vision transformers with soft convolutional inductive biases. In: International Conference on Machine Learning (ICML). pp. 2286–2296 (2021)

7. Fang, C., Xu, Y., Rockmore, D.N.: Unbiased metric learning: On the utilization of multiple datasets and web images for softening bias. In: International Conference on Computer Vision (ICCV). pp. 1657–1664 (2013)
8. Gulrajani, I., Lopez-Paz, D.: In search of lost domain generalization. In: International Conference on Learning Representations (ICLR) (2021)
9. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. In: IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). pp. 770–778 (2016)
10. Houlsby, N., Giurgiu, A., Jastrzebski, S., Morrone, B., De Laroussilhe, Q., Gesmundo, A., Attariyan, M., Gelly, S.: Parameter-efficient transfer learning for nlp. In: International Conference on Machine Learning (ICML). pp. 2790–2799 (2019)
11. Hu, E.J., Wallis, P., Allen-Zhu, Z., Li, Y., Wang, S., Wang, L., Chen, W., et al.: LoRA: Low-rank adaptation of large language models. In: International Conference on Learning Representations (ICLR) (2022)
12. Jia, M., Tang, L., Chen, B.C., Cardie, C., Belongie, S., Hariharan, B., Lim, S.N.: Visual prompt tuning. In: European Conference on Computer Vision (ECCV). pp. 709–727 (2022)
13. Kumar, A., Raghunathan, A., Jones, R.M., Ma, T., Liang, P.: Fine-tuning can distort pretrained features and underperform out-of-distribution. In: International Conference on Learning Representations (ICLR) (2022)
14. Lee, Y., Chen, A.S., Tajwar, F., Kumar, A., Yao, H., Liang, P., Finn, C.: Surgical fine-tuning improves adaptation to distribution shifts. In: International Conference on Learning Representations (ICLR) (2023)
15. Lew, B., Son, D., Chang, B.: Gradient estimation for unseen domain risk minimization with pre-trained models. In: International Conference on Computer Vision Workshops (ICCVW). pp. 4436–4446 (2023)
16. Li, D., Yang, Y., Song, Y.Z., Hospedales, T.M.: Deeper, broader and artier domain generalization. In: International Conference on Computer Vision (ICCV). pp. 5542–5550 (2017)
17. LI, X., Grandvalet, Y., Davoine, F.: Explicit inductive bias for transfer learning with convolutional networks. In: International Conference on Machine Learning (ICML). pp. 2825–2834 (2018)
18. Li, Z., Hoiem, D.: Learning without forgetting. IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI) (2017)
19. Peng, X., Bai, Q., Xia, X., Huang, Z., Saenko, K., Wang, B.: Moment matching for multi-source domain adaptation. In: International Conference on Computer Vision (ICCV). pp. 1406–1415 (2019)
20. Touvron, H., Cord, M., Douze, M., Massa, F., Sablayrolles, A., Jégou, H.: Training data-efficient image transformers & distillation through attention. In: International conference on machine learning (ICML). pp. 10347–10357 (2021)
21. Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A.N., Kaiser, Ł., Polosukhin, I.: Attention is all you need. In: Conference on Neural Information Processing Systems (NeurIPS) (2017)
22. Venkateswara, H., Eusebio, J., Chakraborty, S., Panchanathan, S.: Deep hashing network for unsupervised domain adaptation. In: IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). pp. 5018–5027 (2017)
23. Zang, Y., Li, W., Zhou, K., Huang, C., Loy, C.C.: Unified vision and language prompt learning. arXiv:2210.07225 (2022)
24. Zhou, K., Yang, J., Loy, C.C., Liu, Z.: Learning to prompt for vision-language models. International Journal of Computer Vision (IJCV) pp. 2337–2348 (2022)