Supplementary Material: Black Box Few-Shot Adaptation for Vision-Language models

Yassine Ouali[1] Adrian Bulat[1] Brais Matinez[1] Georgios Tzimiropoulos[1,2]
[1]Samsung AI Cambridge [2]Queen Mary University of London
{y.ouali, brais.a}@samsung.com, adrian@adrianbulat.com, g.tzimiropoulos@qmul.ac.uk

1. Additional Ablations

Few-shot Image Classification Paper Fig. 8 shows the obtained results with different numbers of support examples per class, *i.e*. 4, 8 and 16-shot training data per-class. Similar to the results presented in Paper Section 5, LFA either matches (*i.e*. for 4-shot) or outperforms (*i.e*. for 8 and 4-shot) soft-prompting.

Aligning Uni-modal Models: Paper Fig. 13, we reported the results when aligning self-supervised vision models (*i.e*., BYOL, BarlowTwins, and MoCo v3) and the cpt-text encoder accessed via OpenAI's embeddings API for 16-shot per class and with a single center crop. In Tab. 2, we show the obtained average accuracy on the 11 image datasets for 8- and 16-shot per class, and with either a single center crop or five crops. Overall, LFA outperforms kNN and linear probe classifiers, and by a wide margin when the labeled data is scarce (*i.e*., a single crop). Note that another benefit of LFA is the possibility of using data augmentation in the language domain, *e.g*., prompt ensembling, instead of a single standard prompt, *i.e*., "`a photo of a {cls name}`", which can further boost the performances.

2. Experimental Details

Overall, the training procedure of LFA remains as detailed in Paper Section 4.5. After the β-Procrustes initialisation of the mapping $\mathbf{W}$, we refine it using ARerank loss for given number of iteration (*i.e*. to be detailed on a per-dataset basis) using AdamW with a learning rate of 5e-4, a weight decay of 5e-4 and a cosine scheduler that decreases the learning rate to 1e-7 by the end of training. For most experiments, and unless noted otherwise, we add a small amount of Gaussian noise (*i.e*. std of 3.5e-2) and apply dropout (*i.e*. probability of 2.5e-2) to the image embeddings to avoid overfitting and make the training robust to the choice of the number of refinement steps. Next, we will present the different experimental details on a per-setting basis. For each experiment, the class prototypes are generated by inserting the class name in the standard templates [14] *e.g*., "`a photo of a {cls name}`" for image tasks and "`a video frame of a person {action type}`", in order to give a better initialisation of the prototypes and facilitate the image-text alignment. For all few-shot results, we report the average accuracy over 3 runs.

2.1. Standard Few-shot Image Classification

In this setting, we set β based on a 3-fold cross validation on the training set with a 20-70 validation-train split. We select the β with the highest average validation accuracy across the 3-folds and from a set of values in the range [0.0, 1.0] with a step of 0.05. We generate 5 crops for each training image of size 224x224, *i.e*. four corner crops and a central crop, and use their features for training. While the training is robust to the number of refinement steps, we noticed that the better the orthogonal and β-Procrustes initialisation results, and less number of steps needed to obtain the optimal mapping. Tab. 3 shows the number of refinement for each image classification dataset.

2.2. Base-to-New (Zero-Shot) Recognition

For base-to-new experiments, we found that $\beta = 0.9$ performs well on most dataset without requiring a cross-validation step. Similar to the standard few-shot setting, we train with 5 crops and refine the mapping for up to 100 iterations. See Tab. 4 for the number of refinement steps for each dataset.

2.3. Domain Generalisation:

For domain generalisation experiments, we set $\beta = 0.9$ and train with 5 crops for 200 refinement iterations.

2.4. Action Recognition:

For few-shot action recognition experiments, we set $\beta = 0.9$ and train with a single center crop. For UCF101, no dropout is used and the Gaussian noise is reduced to 2.5e-2, and we conduct 300 refinement steps. For HMDB51, conduct 100 refinement steps. When using the whole training set for alignment, we use the same setup as the few-shot set-

Table 1: **Few-shot Classification:** the obtained average Top-1 test acc. on 11 classification datasets with CoOp [14], Linear Probe, and the proposed LFA, with either 4, 8 or 16-shot per class and with RN50 as the visual encoder.

N-shot	Method	Pets	Flowers102	Aircraft	DTD	EuroSAT	Cars	Food101	SUN397	Caltech101	UCF101	ImageNet	Avg.	Δ
	CLIP RN50	85.77	66.14	17.28	42.32	37.56	55.61	77.31	58.52	86.29	61.46	58.18	58.77	
4	Linear Probe	56.35	84.80	23.57	50.06	68.27	48.42	55.15	54.59	84.34	62.23	41.29	57.19	
	CoOp	**86.06**	86.52	22.02	52.72	**70.93**	**61.62**	**72.64**	63.67	88.53	67.06	**59.96**	66.52	
	LFA	82.21	**88.28**	**24.15**	**54.51**	68.76	60.69	71.81	**65.50**	**88.88**	**69.25**	58.36	**66.58**	**+0.06**
8	Linear Probe	65.94	92.00	29.55	56.56	**76.93**	60.82	63.82	62.17	87.78	69.64	49.55	64.98	
	CoOp	83.58	91.81	28.18	59.14	77.65	67.32	72.04	65.64	90.33	72.74	**62.04**	70.04	
	LFA	**84.93**	**92.62**	**30.17**	**60.54**	77.36	**67.60**	**74.47**	**68.54**	**91.33**	**73.33**	61.36	**71.10**	**+1.06**
16	Linear Probe	76.42	**94.95**	**36.39**	63.97	82.76	70.08	70.17	67.15	90.63	73.72	55.87	71.10	
	CoOp	86.16	94.80	32.29	63.16	83.55	73.27	74.46	69.12	91.62	75.29	63.08	73.35	
	LFA	**86.75**	94.56	35.86	**66.35**	**84.13**	**73.58**	**76.32**	**71.32**	**92.68**	**77.00**	**63.65**	**74.75**	**+1.40**

Table 2: **Aligning Disjoint Modalities:** we show the obtained average Top-1 acc. on 11 image classification datasets for 8- and 16-shot per class and with either a single center crop or five crops as data augmentation.

		BYOL				BarlowTwins				MoCo v3			
N-shot	Crops	kNN	Lin. Probe	LFA	Δ	kNN	Lin. Probe	LFA	Δ	kNN	Lin. Probe	LFA	Δ
16	1	50.61	56.33	**64.26**	**+7.93**	50.77	56.19	**64.45**	**+8.26**	54.64	59.99	**66.90**	**+6.91**
16	5	51.69	61.24	**64.48**	**+3.24**	51.64	61.49	**64.91**	**+3.41**	55.7	64.79	**67.23**	**+2.44**
8	1	44.27	51.09	**58.15**	**+7.05**	43.86	50.70	**58.08**	**+7.37**	48.04	54.77	**60.93**	**+6.15**
8	5	47.07	55.69	**58.31**	**+2.62**	47.02	55.71	**58.52**	**+2.81**	51.11	59.16	**61.00**	**+1.84**

ting, but we conduct 500 refinement steps for UCF101 and 300 for HMDB51.

Table 3: **Refinement Steps for Standard Few-shot Classification:** we specify the number of refinement steps on a per-dataset basis.

Datasets	Nbr. refinement steps
Cars	2000
Caltech101, DTD, Aircraft	1000
EuroSAT, Food101, ImageNet, UCF101	200
Flowers, SUN397	100
Pets	30

Table 4: **Refinement Steps for Base-to-New (Zero-Shot) Recognition:** we specify the number of refinement steps on a per-dataset basis.

Datasets	Nbr. refinement steps
Caltech101, DTD, UCF101, ImageNet	100
EuroSAT, Food101, Flowers, Cars, SUN397	50
Caltech101	40
Pets	10

2.5. Aligning Disjoint Modalities

For the alignment of the features of uni-modal models, we use 3 self-supervised RN50 visual encoders: BYOL, BarlowTwins and MoCo v3, in addition to the cpt-text encoder as our language encoder. After extracting the features, we first conduct a Gaussian random projection implemented using `scikit-learn` [6] to reduce the dimensionality of the visual features from 2048 to 1536 to match those of the text encoder. Then we proceed as in the standard few-shot classification setup, by first finding β with cross-validation, initializing the mapping with β-Procrustes, then refining it for 700-800 iterations using 5 crops.

In terms of the kNN and linear probe baselines, we follow the practical recommendations of [9], and train the classifiers on the ℓ_2 normalized and forzen visual features (*i.e.* the original 2048-d features). For the kNN classification, we use 16 neighbors for training, as for the linear probe, we follow [9] and use the multinomial logistic regression implementations of `scikit-learn`, and train with an ℓ_2 penalty and the LBFGS solver. Similar to LFA, all baselines were trained with 5 crops.

References

[1] Joao Carreira and Andrew Zisserman. Quo vadis, action recognition? a new model and the kinetics dataset. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2017.

[2] Kensho Hara, Hirokatsu Kataoka, and Yutaka Satoh. Can spatiotemporal 3d cnns retrace the history of 2d cnns and imagenet? In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018.

[3] Dan Hendrycks, Steven Basart, Norman Mu, Saurav Kadavath, Frank Wang, Evan Dorundo, Rahul Desai, Tyler Zhu, Samyak Parajuli, Mike Guo, et al. The many faces of robustness: A critical analysis of out-of-distribution generalization. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 8340–8349, 2021.

[4] Dan Hendrycks, Kevin Zhao, Steven Basart, Jacob Steinhardt, and Dawn Song. Natural adversarial examples. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 15262–15271, 2021.

[5] Chen Ju, Tengda Han, Kunhao Zheng, Ya Zhang, and Weidi Xie. Prompting visual-language models for efficient video understanding. In *Computer Vision–ECCV 2022: 17th European Conference, Tel Aviv, Israel, October 23–27, 2022, Proceedings, Part XXXV*, pages 105–124. Springer, 2022.

[6] Fabian Pedregosa, Gaël Varoquaux, Alexandre Gramfort, Vincent Michel, Bertrand Thirion, Olivier Grisel, Mathieu Blondel, Peter Prettenhofer, Ron Weiss, Vincent Dubourg, et al. Scikit-learn: Machine learning in python. *the Journal of machine Learning research*, 12:2825–2830, 2011. 2

[7] 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 *International conference on machine learning*, pages 8748–8763. PMLR, 2021.

[8] Benjamin Recht, Rebecca Roelofs, Ludwig Schmidt, and Vaishaal Shankar. Do imagenet classifiers generalize to imagenet? In *International conference on machine learning*, pages 5389–5400. PMLR, 2019.

[9] Yonglong Tian, Yue Wang, Dilip Krishnan, Joshua B Tenenbaum, and Phillip Isola. Rethinking few-shot image classification: a good embedding is all you need? In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XIV 16*, pages 266–282. Springer, 2020. 2

[10] Du Tran, Heng Wang, Lorenzo Torresani, Jamie Ray, Yann LeCun, and Manohar Paluri. A closer look at spatiotemporal convolutions for action recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018.

[11] Haohan Wang, Songwei Ge, Zachary Lipton, and Eric P Xing. Learning robust global representations by penalizing local predictive power. *Advances in Neural Information Processing Systems*, 32, 2019.

[12] Saining Xie, Chen Sun, Jonathan Huang, Zhuowen Tu, and Kevin Murphy. Rethinking spatiotemporal feature learning for video understanding. In *Proceedings of the European Conference on Computer Vision*, 2018.

[13] Kaiyang Zhou, Jingkang Yang, Chen Change Loy, and Ziwei Liu. Conditional prompt learning for vision-language models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 16816–16825, 2022.

[14] Kaiyang Zhou, Jingkang Yang, Chen Change Loy, and Ziwei Liu. Learning to prompt for vision-language models. *International Journal of Computer Vision*, 130(9):2337–2348, 2022. 1, 2