SVL-Adapter: Self-Supervised Adapter for Vision-Language Pretrained Models - Supplementary Material

Omiros Pantazis[1]
omiros.pantazis.16@ucl.ac.uk
Gabriel Brostow[1,4]
g.brostow@cs.ucl.ac.uk
Kate E. Jones[1]
kate.e.jones@ucl.ac.uk
Oisin Mac Aodha[2,3]
oisin.macaodha@ed.ac.uk

[1] University College London
[2] University of Edinburgh
[3] Alan Turing Institute
[4] Niantic

A Additional Results

Here we present additional results from experiments and ablation studies that highlight the performance gains and the consistency of SVL-Adapter. We maintain the categorisation of datasets, *i.e*. "Standard" and "Challenging", as defined in the main paper.

A.1 Better self-supervised features improve adaptation performance

In Figs. 3 and 4 of the main paper, we see how the vanilla version of SimCLR [3] can help CLIP adapt especially across challenging downstream tasks. Additionally, to support the assumption that the developments in self-supervised learning are orthogonal to our approach, we exploit the metadata that are typically available for real-world tasks to formulate a self-supervised task that uses more informative positive pairs instead of self-augmentations as suggested in [8]. Specifically, for the four camera trap datasets [1, 2, 8, 10], we perform self-supervised training with context-informed positives and observe significant improvements in both zero- and low-shot classification when the features are exploited by our suggested SVL-Adapter* and SVL-Adapter respectively (Fig. A1). From the results in Fig. A1 we see that the inclusion of context-informed positives instead of self-augmentations during the self-supervised tasks can lead to improvements in both zero-shot and low-shot learning by our proposed SVL-Adapter* and SVL-Adapter approaches. In addition, we show that a simple Triplet loss can replace SimCLR and remain effective on adaptation (see Table A1).

© 2022. The copyright of this document resides with its authors.
It may be distributed unchanged freely in print or electronic forms.

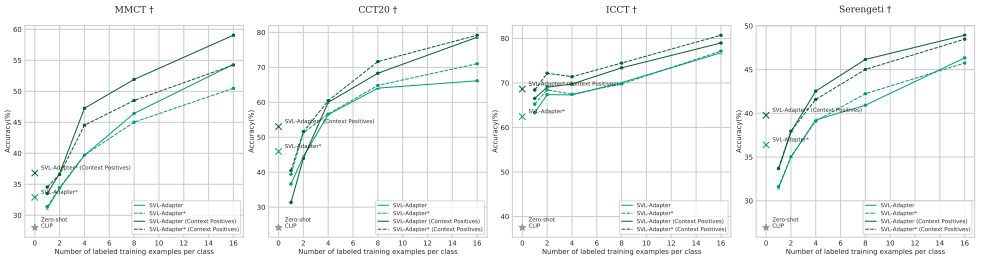

Figure A1: By using a better self-supervised task ("Context Positives" from [8]) to encode features with E_s, we achieve consistent gains in zero- and low-shot classification across the four camera trap datasets for both SVL-Adapter and SVL-Adapter*.

A.2 Better CLIP models translate to better performance

The experiments in the main paper use ResNet50 [6] as the backbone architecture for both CLIP and the self-supervised feature encoder component of SVL-Adapter. Given that the adapter module A_s is expecting the output of the pretrained self-supervised feature encoder E_s we understand than increasing the capacity of its backbone model would make the approach less computationally efficient. However, we experiment with the size of the frozen CLIP Encoder given that it is only used for inference and thus it would not be prohibitive for practitioners, even those with limited resources. Specifically, we replace the ResNet50 CLIP with a ViT/L-14, a large vision transformer [4], *i.e.* CLIP's best performing variant according to [9]. As expected, in Fig. A2 we observe that the improvements in the CLIP backbone are also reflected in the SVL-Adapter combining them with the adapter features both across "Standard" and "Challenging" tasks.

Figure A2: Low-shot evaluation of SVL-Adapter and SVL-Adapter* fused with Zero-shot CLIP with a ResNet50 backbone as in the main paper versus Zero-shot CLIP with a more advanced large visual transformer. We observe significant gains when fusing the logits that come out of the adapter with a more advanced Zero-shot model (Purple) both in "Standard" and "Challenging" tasks.

A.3 SSL features alone are not very effective

Even though in the main paper we showed the advantage of using a fusion of adapted self-supervised features and the impressive outputs of Zero-shot CLIP by comparing to existing baselines, we have not compared SVL-Adapter with a variant that only relies on the visual adaptation of the self-supervised features. Here, we compare SVL-Adapter with the adapta-

tion of the SSL features using a similar adapter module but without the blending component in order to understand the necessity of including the CLIP component as described in Eqn. 3 and Sec. 3.2.1 of the main paper. As we see in Fig. A3, SVL-Adapter and SVL-Adapter* outperform Zero-shot CLIP and the independently adapted self-supervised features (SimCLR MLP) on average; thus, a combination, similar to the one proposed with SVL-Adapter in Eqn. 3 in the main paper, should always be preferred as a universal solution.

Figure A3: Here we compare Zero-shot CLIP and adapted self-supervised features (Brown) with our suggested SVL-Adapter variants that essentially are blending the predictions that come out of the two as described in Sec. 3.2.1 of the main paper. We confirm that SVL-Adapter is always better than its constituent parts when used independently. Interestingly, we see the larger benefits of using SVL-Adapter instead of pure self-supervised feature adaptation in the "Standard" datasets.

A.4 Replacing the encoder with ImageNet or CLIP features

To better understand the importance of the features extracted by a self-supervised encoder E_s, we replace them with features extracted by either the CLIP ResNet50 features or an ImageNet pretrained ResNet50 while keeping the rest of the components fixed. In Fig. A4 we see that SVL-Adapter keeps delivering the best performance on average and especially on the challenging tasks when compared with a similar adapter that starts from ImageNet or CLIP ResNet50 features. Perhaps surprisingly, we see that adaptation from ImageNet features is clearly superior to adaptation from the CLIP features when evaluated on the challenging tasks. Despite the large size of the corpus that CLIP is pretrained on, it seems that the quantity of data does not necessarily translate to the visual diversity needed to tackle challenging tasks that potentially fall out of distribution, which could be better addressed with ImageNet pretrained models.

A.5 The impact of λ parameter on SSL and CLIP fusion

As we have seen in the main text, some tasks are easier for vision-language models that are pre-trained on online image and text pairs while some are quite challenging. Thus, a constant fusion parameter λ (see Eqn.3) between adapted self-supervised and Zero-shot CLIP features for all cases would yield sub-optimal results. We run an ablation experiment on SVL-Adapter where we vary λ with values ranging from 0 to 1 and compare with the parameter-free SVL-Adapter* approach proposed in Sec. 3.2.1. Note that a low value of λ here means that the self-supervised features adapted via low-shot learning are taken more into account. After observing Fig. A5, we can see that the "Challenging" tasks (†) need a lower value of λ to perform well on the test set while most of the other datasets reach peak performance with

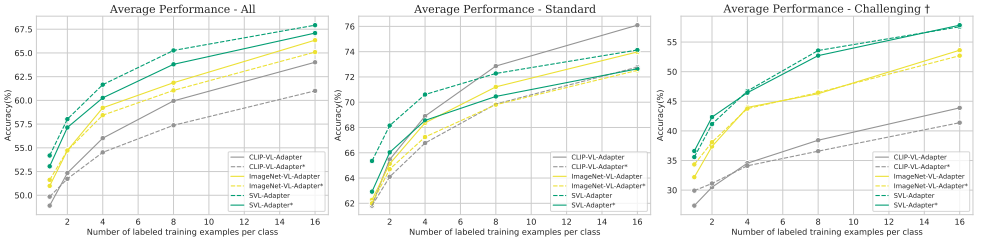

Figure A4: Here we compare the low-shot adaptation of our approach (SVL-Adapter) with two variants that tune a similar adapter on top of supervised ImageNet features (Yellow) and CLIP features (Grey). We see that adapting from self-supervised features keeps giving the best results on average, especially on challenging tasks while we also observe decent performance from the ImageNet-based adapter.

high λ values. In addition, it is shown that the SVL-Adapter* which picks λ based on CLIP's average prediction confidence is typically near the top performing points of the curve.

A.6 Ablations on the components of SVL-Adapter

We perform multiple ablations on the components of SVL-Adapter, varying the pre-training source, CLIP's vision backbone and the underlying self-supervised method. The results can be found on Table A1. Given this table, we confirm that fusing self-supervised features learnt for the task at hand with Zero-shot CLIP gives the best performance. Moreover, replacing either CLIP's backbone with a more advanced architecture or the self-supervised technique with a better one can further boost performance. Indicatively, the average performance of Tip-Adapter-F, the best performing baseline on the four camera trap datasets, is 50.8, which is surpassed by a big margin even by a simple Triplet loss trained in a self-supervised manner in the given dataset.

CLIP Visual Encoder	Adapted Features	Top-1 Accuracy
-	SimCLR RN50	60.2*
RN50	-	46.5*
RN50	ImageNet RN50	66.3*
RN50	CLIP RN50	64.0*
RN50	SimCLR RN50	**67.9***
ViT-L/14	-	60.1*
ViT-L/14	SimCLR RN50	**74.9***
RN50	SimCLR RN50	60.4‡
RN50	Triplet RN50	57.0‡
RN50	SimCLR RN50 + CP [39]	**66.4**‡

Table A1: Ablation experiments. * are 16-shot Top-1 accuracy for all datasets and ‡ are camera trap datasets only. 'CLIP Visual Encoder' is the zero-shot backbone and 'Adapted Features' are the features used by SVL-Adapter's adapter. (Top) Ablations on the visual representations of SVL-Adapter. (Middle) Impact of different CLIP visual encoders. (Bottom) Impact of different SSL methods.

A.7 Understanding why SSL helps

To try to understand why adaptation from self-supervised features is superior on the challenging tasks, we plot their 2-dimensional embeddings on test images along with the respective projections of CLIP features. We use UMAP [7] for the dimensionality reduction across EuroSAT and MMCT. As we observe from Fig. A6, the embeddings of the self-supervised learning features (right) provide a better separation of the underlying classes in the unseen test set, while the 2-dimension projection of the CLIP extracted features is not as well structured and overlaps in many cases.

A.8 Quantitative results in tabular form

For a more detailed view, we present the quantitative results from Fig. 3 of the main paper in a tabular format (Table A2).

Average Performance - All				
	CoOp	CLIP-Adapter	Tip-Adapter-F	SVL-Adapter
1-shot	48.1	49.7	53.6	**54.2**
2-shot	50.9	53.6	55.2	**58.0**
4-shot	54.4	56.8	58.0	**61.7**
8-shot	59.5	59.9	61.6	**65.3**
16-shot	63.9	64.0	66.2	**67.9**

Average Performance - Standard				
	CoOp	CLIP-Adapter	Tip-Adapter-F	SVL-Adapter
1-shot	59.8	62.9	64.9	**65.4**
2-shot	62.8	66.0	67.2	**68.2**
4-shot	67.4	69.3	70.4	**70.6**
8-shot	70.8	72.3	**73.3**	72.3
16-shot	74.5	75.5	**76.9**	74.1

Average Performance - Challenging †				
	CoOp	CLIP-Adapter	Tip-Adapter-F	SVL-Adapter
1-shot	28.5	27.8	34.7	**35.6**
2-shot	31.2	33.0	35.3	**41.1**
4-shot	32.7	35.9	37.5	**46.7**
8-shot	40.8	39.4	42.1	**53.6**
16-shot	46.3	44.8	48.3	**57.6**

Table A2: Top-1 Accuracy from Fig. 3 of the main paper, but in tabular format. (Top) Averaged across all 16 datasets with the corresponding *zero-shot* results for CLIP and our zero-shot SVL-Adapter* being 46.5% and 52.5% respectively. (Middle) Averaged across the "Standard" datasets with the corresponding *zero-shot* results for CLIP and our zero-shot SVL-Adapter* being 58.4% and 63.1% respectively. (Bottom) Averaged across the "Challenging" datasets with the corresponding *zero-shot* results for CLIP and our zero-shot SVL-Adapter* being 26.7% and 34.8% respectively.

B Implementation Details

B.1 Prompt templates

Across all experiments that involved E_t, *i.e.* the text encoder of CLIP , we used a prompt template for each of the datasets. As mentioned in the main text, for the "Standard" datasets we use the simple prompts suggested by [9] and adopted by the subsequent vision-language adaptation works [5, 10, 12]. With respect to the "Challenging" tasks, we utilized the generic prompt "a photo of a {label}." for MMCT, CCT20, ICCT, Serengeti, and FMoW and "an OCT scan of {label} retina." for the Optical Coherence Tomography (OCT) dataset. To make sure our class names are suitable for the text encoder of CLIP, we transformed some of them in a human-friendly format. For example, in OCT we replaced the disease names "CNV" and "DME" with their full names "Choroidal Neovascularization" and "Diabetic Macular Edema" respectively. In addition, we replace underscores with blanks and separate class names (*e.g.* from "gazelleGrants" to "gazelle grants") where necessary.

B.2 Training details

In addition to the implementation details provided in Sec. 4.1 of the main paper, we present more details about our modeling choices.

Self-supervised pretraining. For self-supervised learning we train for 200 epochs with batch size 256 and learning rate 0.03 and cosine annealing. For optimization we use Stochastic Gradient Descent (SGD) with 0.9 momentum and weight decay 0.0005. On top of the ResNet 50 backbone, we used a projection neural network with 512 hidden size and 128 embedding size. For the image transformations on the 112×112 sized images of the self-supervised task we use cropping with random resizing, random horizontal flipping, color jittering and grayscale conversion. For the context positive mining approach used to generate Fig. A1 we used the metadata and implementation details as suggested by [8].

Adapter tuning. The downstream task adaptation takes place on top of the frozen features generated by the feature encoders, *e.g.* by the ResNet50 encoder E_s that comes from self-supervised pretraining. We train the adapter module A_s of SVL-Adapter for 50 epochs with batch size of 32. Specifically, this module corresponds to a 2-layer MLP with a 256 hidden layer that receives the frozen features from E_s andf outputs the class logits which are used in the fusion stage described in Eqn.3 of the main paper. For optimization, we utilized Adam with learning rate 0.001 and a cross entropy loss between the logits and the few-shot sample labels or pseudolabels in SVL-Adapter and Zero-shot SVL-Adapter respectively. For the scenario where we use a small validation set to tune the λ parameter (Sec. 3.2.1 of main paper), we use 1 and 2 samples for 1-shot and 2-shot classification respectively and 4 samples for the rest of the labeled 4-, 8-, and 16-shots, similarly with preceding works [5, 10].

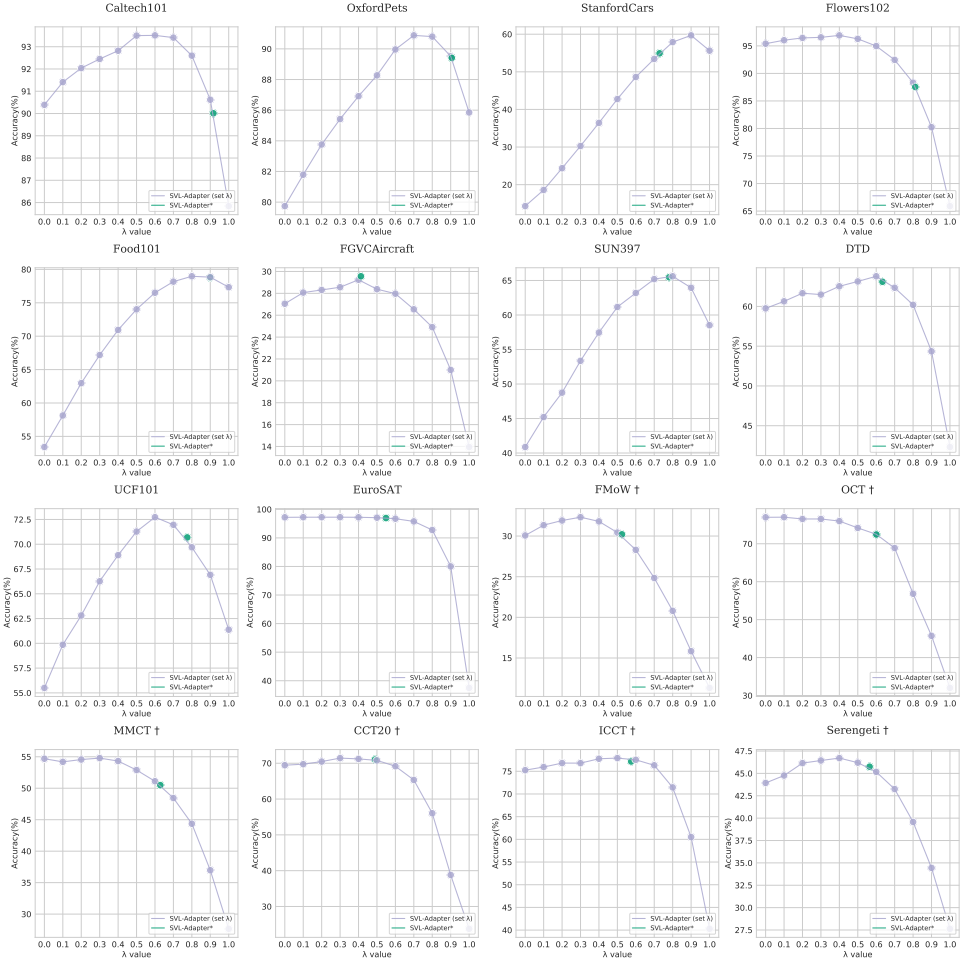

Figure A5: Per-dataset 16-shot, test Top-1 accuracy for the 16 different datasets across various values of the λ parameter along with the proposed SVL-Adapter*. In each case, we report the average of three runs. Datasets marked with † are added by us, and pose a significantly greater challenge to existing methods.

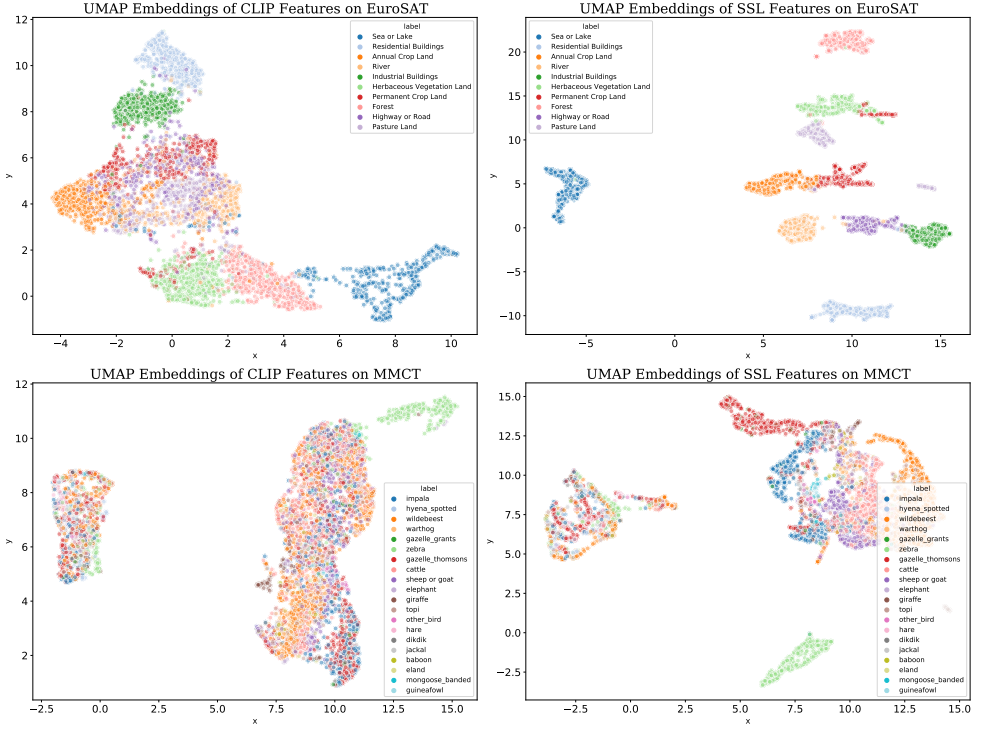

Figure A6: 2-dimensional UMAP embeddings of test image features coming from CLIP (Left) or Self-Supervised (Right) ResNet50 feature extractors. For both EuroSAT (top) and MMCT (bottom) we observe that the self-supervised learning features correspond to a better separation between the dataset classes.

References

[1] Island Conservation Camera Traps. http://lila.science/datasets/island-conservation-camera-traps.

[2] Sara Beery, Grant Van Horn, and Pietro Perona. Recognition in terra incognita. In *ECCV*, 2018.

[3] Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. A simple framework for contrastive learning of visual representations. In *ICML*, 2020.

[4] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, et al. An image is worth 16x16 words: Transformers for image recognition at scale. In *ICLR*, 2021.

[5] Peng Gao, Shijie Geng, Renrui Zhang, Teli Ma, Rongyao Fang, Yongfeng Zhang, Hongsheng Li, and Yu Qiao. Clip-adapter: Better vision-language models with feature adapters. *arXiv preprint arXiv:2110.04544*, 2021.

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

[7] Leland McInnes, John Healy, and James Melville. Umap: Uniform manifold approximation and projection for dimension reduction. *arXiv preprint arXiv:1802.03426*, 2018.

[8] Omiros Pantazis, Gabriel J. Brostow, Kate E. Jones, and Oisin Mac Aodha. Focus on the positives: Self-supervised learning for biodiversity monitoring. In *ICCV*, 2021.

[9] 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 *ICML*, 2021.

[10] Alexandra Swanson, Margaret Kosmala, Chris Lintott, Robert Simpson, Arfon Smith, and Craig Packer. Snapshot serengeti, high-frequency annotated camera trap images of 40 mammalian species in an african savanna. *Scientific data*, 2(1), 2015.

[11] Renrui Zhang, Rongyao Fang, Peng Gao, Wei Zhang, Kunchang Li, Jifeng Dai, Yu Qiao, and Hongsheng Li. Tip-adapter: Training-free clip-adapter for better vision-language modeling. In *ECCV*, 2022.

[12] Kaiyang Zhou, Jingkang Yang, Chen Change Loy, and Ziwei Liu. Learning to prompt for vision-language models. *IJCV*, 2022.