Down-Sampling Inter-Layer Adapter for Parameter and Computation Efficient Ultra-Fine-Grained Image Recognition

Edwin Arkel Rios[1], Femiloye Oyerinde[2], Min-Chun Hu[3], and Bo-Cheng Lai[1]

[1] National Yang Ming Chiao Tung University, Taiwan
[2] Cohere For AI Community
[3] National Tsing Hua University, Taiwan

Abstract. Ultra-fine-grained image recognition (UFGIR) categorizes objects with extremely small differences between classes, such as distinguishing between cultivars within the same species, as opposed to species-level classification in fine-grained image recognition (FGIR). The difficulty of this task is exacerbated due to the scarcity of samples per category. To tackle these challenges we introduce a novel approach employing down-sampling inter-layer adapters in a parameter-efficient setting, where the backbone parameters are frozen and we only fine-tune a small set of additional modules. By integrating dual-branch down-sampling, we significantly reduce the number of parameters and floating-point operations (FLOPs) required, making our method highly efficient. Comprehensive experiments on ten datasets demonstrate that our approach obtains outstanding accuracy-cost performance, highlighting its potential for practical applications in resource-constrained environments. In particular, our method increases the average accuracy by at least 6.8% compared to other methods in the parameter-efficient setting while requiring at least 123x less trainable parameters compared to current state-of-the-art UFGIR methods and reducing the FLOPs by 30% in average compared to other methods.

Keywords: Vision Transformer · Ultra Fine Grained Visual Categorization · Parameter-Efficient Transfer Learning · Fine-Tuning

1 Introduction

Ultra-fine-grained image recognition (UFGIR) categorizes sub-categories within a macro-category. While conventional FGIR [42] classifies objects usually up to species-level granularity, UFGIR may categorize classes at a finer level, such as cultivars of a plant. It has practical application in various fields such as agriculture [24, 30], medical [31], and industrial [25]. It is a challenging task due to small inter-class differences, large intra-class differences, and low data availability due to the difficulty behind labeling even for human experts [48].

To address this most methods [39, 43–47] utilize coarse image-recognition backbones equipped with additional modules [12, 39] or losses [11, 46] to focus and make better use of discriminative features that encapsulate subtle differences

between fine-grained classes. Specifically, recent works employ Vision Transformers (ViT) [10] since their use of the self-attention mechanism [38], with its global receptive field, allows models to effectively extract and aggregate fine-grained features [18, 34, 36].

However, due to the growing size of state-of-the-art (SotA) ViT backbones, researchers have explored the design of parameter-efficient transfer learning (PETL) methods [3, 16, 20, 37]. Instead of fine-tuning the entire backbone, most of the parameters are frozen, and only specific components are fine-tuned. This allows for reuse of most of the parameters, drastically reducing storage requirements, specially when deploying models across multiple tasks.

PETL methods have shown performance that can match or even surpass specialized FGIR models with full fine-tuning in generic FGIR tasks [3]. However, we observe that in UFGIR tasks PETL methods still lag behind specialized FGIR methods, either in the traditional fine-tune setting or in a novel setting that we coin as parameter-efficient FGIR (PEFGIR), where we only fine-tune the FGIR modules while most of the backbone is frozen. Based on analysis of the ViT features, we observe on Fig. 3 that frozen ViTs suffer from attention collapse [7, 40, 50], a phenomenona where the attention scores across layers become increasingly similar, hindering the feature extraction process.
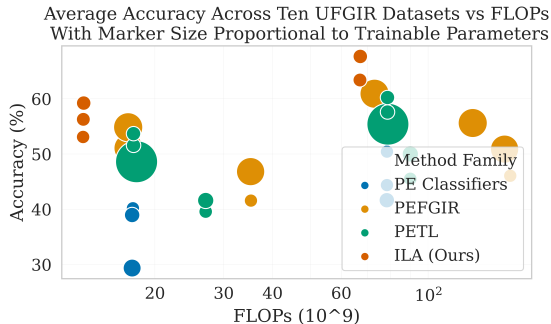

Fig. 1: Average top-1 accuracy (%) across all evaluated datasets vs number of floating-point operations (FLOPs) for different method families, including methods that only fine-tune the classification head, fine-grained image recognition (FGIR) methods in parameter-efficient setting (PEFGIR, only fine-tune the fine-grained discrimination modules) and parameter-efficient transfer learning (PETL) methods. The size of the markers is proportional to the percentage of trainable parameters for each method.

To address this, we propose Intermediate Layer Adapter (ILA), a novel, parameter and computationally efficient method that employs dual spatial downsampling branches as an adapter [16, 20] inserted between transformer layers to aggregate spatial features while preserving fine-grained details. Our proposed approach results in much more diverse attention scores across layers and across a variety of benchmarks we demonstrate its outstanding performance compared to SotA FGIR and PETL methods in terms of accuracy and computational cost, as seen in Fig. 1.

Our contributions are as follows:

1. We propose a novel ILA module to address the attention collapse problem faced by frozen ViTs in UFGIR tasks. The proposed ILA employs dual spatial-down sampling branches to aggregate discriminative features and reduce computational cost.
2. We conduct comprehensive experiments across ten UFGIR datasets comparing more than 15 SotA methods across two image sizes. Through our experiments the proposed ILA obtains outstanding classification performance and enhanced computational efficiency, as measured by the total number of trainable parameters (TTP) and floating-point-operations (FLOPs). In particular, our method increases the average accuracy by at least 6.8% compared to other methods in the parameter-efficient setting while requiring at least 123x less TTPs compared to current SotA UFGIR methods, and reducing the FLOPs by 30% in average compared to other methods.

2 Related Work

Ultra Fine-Grained Image Recognition UFGIR methods employ backbones pretrained for generic recognition and equip them with modules to select and aggregate discriminative features [12, 39] or employ loss functions and tasks [45, 46] to guide models to more effectively make use of fine-grained features, or both [11, 42]. In the former category, FFVT [39] employs ViT's attention scores to select intermediate low-, medium-, and high-level features that are vital for recognizing small inter-class differences and are aggregated through the last transformer encoder block.

Since UFGIR has an additional challenge due to limited labeled data, research in this direction has been very active in recent years [47, 48]. MaskCOV [46], SPARE [45], and Mix-ViT [44] employ data augmentation and propose self-supervised [4] tasks and losses for the model to learn intrinsic details with limited data. CLE-ViT [43] and CSD [11] employ contrastive learning [8] and the latter also employs self-knowledge distillation [6, 14, 49] to address the challenges faced in UFGIR. However, most of these methods employ ViT backbones with large number of parameters that all need to be stored for deployment and also spend significant resources during training.

Parameter-Efficient Transfer Learning (PETL) techniques aim to fine-tune a small subset of modules while most of the backbone parameters are frozen. These are mostly classified into two: prompt-tuning and adapters. Prompt-tuning [19, 26] incorporates additional task-specific learnable tokens that are appended to the sequence at different stages of the transformer. VQT [37] proposes using the tokens as queries that aggregate layer-wise information and are expedited to the classification layer to incorporate intermediate features into the classification head. However, incorporating additional tokens increases the computational cost of the forward pass, and in the case of VQT, the integration of a large number of input features into the classification head can rapidly increase the number of parameters.

On the other side, adapters were first proposed by Houlsby *et al.* [16] in the natural language processing (NLP) domain and are additional light-weight non-linear modules that are inserted usually inside transformer layers. ConvPass [20] extended this idea for ViTs by incorporating a 2D convolution into an adapter to introduce spatial biases into the design.
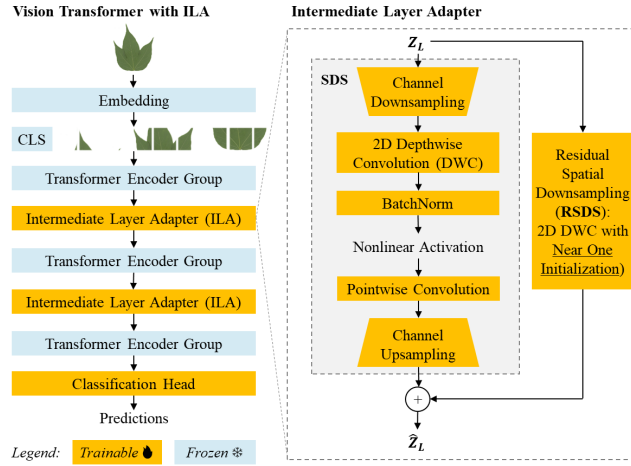
3   Method

Fig. 2: Overview of ViT with our proposed Intermediate Layer Adapter (ILA). Trainable modules are shown in orange while frozen ones are shown in blue. An image is embedded into tokens and forwarded through a series of transformer encoder blocks, which we divide into three groups. After the first two encoder groups the sequence is passed through the ILA. After passing through all the encoder blocks the CLS token is forwarded through a classification head to obtain predictions. In the ILA tokens are forwarded through two spatial downsampling (SDS) branches. In the main SDS branch (highlighted as a grey box) tokens are first downsampled channel-wise and then spatially downsampled through the usage of a 2D depth-wise convolution. The sequence is then forwarded through a BatchNorm layer, a non-linear activation, and a point-wise convolution, before being up-sampled channel-wise. To allow for residual gradient flow we also forward the tokens through a Residual Spatial Downsampling (RSDS) branch implemented as a 2D depth-wise convolution initialized with values near one. Initializing the kernel to values near one allows the RSDS to behave as a learnable identity or pooling function. Then, the outputs of the dual SDS branches are added together and forwarded to the next encoder group.

The overview of our proposed method is shown in Figure 2. Our method is based on a generic Vision Transformer (ViT) [10]. An image is patchified and forwarded through transformer encoder blocks which we divide into three groups, each with 4 blocks. After the first two encoder groups the sequence is passed through the proposed Intermediate Layer Adapter (ILA). After passing through

all the encoder blocks the CLS token is forwarded through a classification head to obtain predictions.

Vision Transformer Encoder Images are patchified using a convolution with kernel size P and flattened into a 1D sequence of D channels with length $N_0 = (h/P) \times (w/P)$, where h and w represent the image width and height. A learnable CLS token [9] is appended to the sequence and learnable positional embeddings are added to encode spatial information. This sequence is passed through a series of transformer encoder blocks each composed of multi-head self-attention (MHSA) and position-wise feed-forward networks (PWFFN) [38]. The output of each block is $\mathbf{z}_l \in \mathbb{R}^{N_l \times D}$ Finally, this output is passed through a LayerNorm [2] and a linear classification layer to obtain predictions.

However, we observe that a frozen ViT encoder applied in UFGIR tasks suffers from attention collapse [40, 50] as shown in Fig. 3. This happens when attention maps across different layers collapse to a single representation and therefore the model is unable to extract meaningful features.
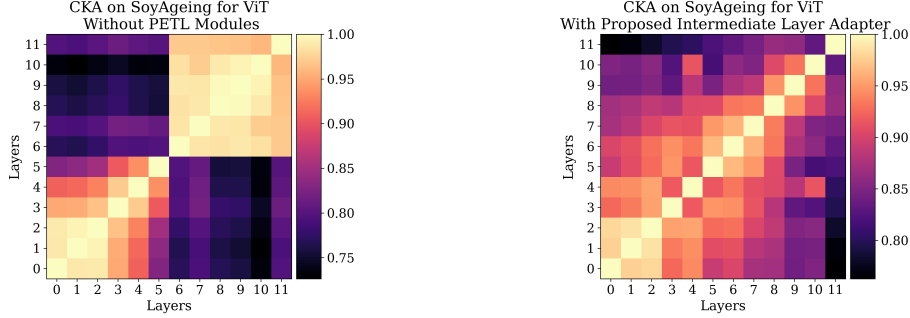

Fig. 3: Centered Kernel Alignment (CKA) similarity [22] between attention layers of a ViT for the vanilla ViT (left) and ours (right). Lighter colors indicates higher similarity.

Inter-Layer Adapter Inspired by previous works [29,41] and to encourage the network to focus on distinct areas at different stages of encoding, we explicitly enforce hierarchy in the feature maps by incorporating downsampling in the ILA module. Specifically, we make use of a dual spatial down-sampling (SDS) design. The main branch is composed of a channel down-sampling (CDS), a depth-wise separable convolution where the spatial downsampling takes place, and a channel up-sampling (CUS) module. The design of this branch is similar to the one proposed by Jie *et al*. [20], but we have two important differences: 1) to increase computational efficiency we employ a depth-wise separable convolution [17] for the convolution in main branch, 2) we do not incorporate padding therefore the forward through the convolution reduces the spatial dimension of the inputs. These design changes allow our model to not only reduce the cost, but also increase diversity in the attention maps by explicitly enforcing a hierarchy in the feature maps, which forces the model to focus on different areas at different stages. The output of this branch is denoted as $\mathbf{m}$ and defined as follows:

$$\mathbf{m} = \mathrm{CUS(PWConv(GELU(BN(DWConv(CDS(\mathbf{z}_l))))))} \tag{1}$$

Residual Spatial Downsampling Branch To facilitate the smooth flow of information between layers and mitigate the risk of vanishing gradients within the network [13] we wish to employ a residual connection. However, due to the spatial downsampling operation in Eq. (1), the spatial dimensions of $\mathbf{m}$ are reduced, preventing directly adding $\mathbf{z}_l$. To align their shapes, existing methods typically employ pooling or interpolation operations for downsampling $\mathbf{z}_l$. However, these apply fixed procedures that may discard local features or structural details. Also, they lack the learnability that has allowed neural networks to thrive in recent years [23,38].

Therefore, we propose employing a learnable residual downsampling branch based on depthwise convolutions (DWC) with kernel weights initialized close to 1, which can easily approximate an identity. We term this the Residual Spatial Downsampling (RSDS) branch. The equation for the residual output $r_{d,n}$ for each channel d and each spatial position n of a 1D depthwise convolution (DWC) with D input and output channels and the input having N spatial positions is shown in Eq. (2):

$$r_{d,n} = \sum_{k=0}^{K-1} z_l^{d,n+k} \cdot W_{d,k}; \qquad d = 0, 1, ..., D; \qquad n = 0, 1, ..., N \tag{2}$$

In the case of kernel size $K = 1$ and initializing the values of the kernel weights W to values close to 1 the equation becomes close to an identity function I as shown in Eq. (3):

$$r_{d,n} \approx I(z_l^{d,n}) \approx z_l^{d,n}; \qquad d = 0, 1, ..., D; \qquad n = 0, 1, ..., N \tag{3}$$

However, unlike the identity function, the DWC can adapt its weights to regulate the influence of the original values in the overall output, effectively behaving as a gate [15]. This allows our module to learn different functions based on the gradients. For the case of kernel size different to 1, the proposed DWC with near ones init behaves similarly to a sum-pooling operation. Based on these observations and inspired by how we moved from fixed, manually-designed filters to learnable filters with AlexNet [23], we employ this DWC with near ones initialization to act as a learnable residual.

4   Experiment Methodology

Detailed description for our experiments can be found in the Appendix. We evaluate our method on ten ultra-fine-grained leaves datasets collected by Yu *et al.* [48] where each category represents a cultivar. When applicable, the best values are highlighted in **bold** and the second best are underlined.

We report results using top-1 accuracy (%) and standard deviation of 3 seeds for image size 224 and image size 448. We also report the number of trainable

parameters (TTP) for a group of tasks and the number of floating-point operations (FLOPs). We use Pytorch [32] and Wandb [5] to implement and manage experiments, respectively.

All of our experiments employ the ViT B-16 [10] backbone with patch size 16, number of layers $L = 12$ and hidden dimension size $D = 768$. We propose 3 different variants of ILA which are as follows:

- ILA: the intermediate layer adapters (ILAs) modules with down-sampling are inserted after layer 4 and 8 only.
- ILA$^+$: includes the modules from ILA plus additionally ILAs without down-sampling are inserted at every other layers of the ViT model besides from layer 4 and 8.
- ILA^{++}: includes the modules from ILA plus we additionally incorporate the traditional intra-layer adapters [16] in all layers.

We compare our proposed models against 15 state-of-the-art models in the parameter-efficient setting, grouped into three families based on their characteristics: 1) methods which only fine-tune the classification head, 2) FGIR methods where a module is designed to explicitly select features based on some criteria, and 3) dedicated PETL methods.

Table 1: Top-1 accuracy (%) and total trainable parameters (TTP, in millions for all five tasks) for SotA models on five ultra-FGIR datasets with image size 448. Model* represents all parameters, including the backbone, were fine-tuned.

Method	Cotton	SoyAgeing	SoyGene	SoyGlobal	SoyLocal	TTP (10^6)
ViT B-16	39.03	48.61	21.31	24.97	28.72	**1.7**
MPNC [27]	43.89	40.43	20.22	25.79	27.94	4.9
IFA [34]	44.58	56.01	33.09	35.82	29.22	**1.7**
TrFG [12]	51.67	57.54	38.79	45.35	38.06	37.2
FFVT [39]	51.94	69.93	49.97	47.70	42.22	37.2
CAL [33]	44.03	51.16	23.03	34.98	31.61	55.6
RAMS [18]	38.47	50.73	25.83	25.17	29.67	**1.7**
GLSim [35]	45.70	56.58	33.04	39.85	29.95	37.2
VQT [37]	50.97	65.65	36.80	33.99	36.33	106.0
VPT-S [19]	38.19	49.12	27.43	27.37	28.39	<u>2.1</u>
VPT-D [19]	43.19	60.76	38.28	36.83	25.28	6.3
ConvP [20]	48.33	60.18	53.43	45.13	34.22	3.4
ADPT [16]	48.19	73.31	57.04	47.27	36.83	3.3
TrFG* [12]	54.58	72.16	22.38	21.24	40.67	434.3
SIMT* [36]	54.58	34.76	15.46	**70.69**	25.00	497.0
CSD* [11]	**57.92**	**75.39**	**70.82**	56.30	46.17	432.0
ILA$^+$	53.33	68.79	52.65	48.29	<u>46.56</u>	2.6
ILA^{++}	<u>55.42</u>	<u>75.00</u>	<u>62.19</u>	<u>58.14</u>	**50.83**	3.5

5   Results and Discussion

Comparison with State-of-the-Art A summary of aggregated results under
the PE setting are shown in Fig. 1. We observe that not only do the different
versions of ILA achieve the top average accuracies across all tasks, but it is
also parameter and compute efficient. Specifically, ILA^{++} increases accuracy by
6.8% but requires 8% less floating-point operations (FLOPs) and trains 90% less
parameters than FFVT [39], which achieves the second highest average accuracy.

We also report per-dataset accuracies on Tab. 1, including fine-tuned (FT)
FGIR models. While ILA does not obtain the best accuracy when compared to
the best FT FGIR models it achieves a competitive accuracy at a much lower
parameter-cost. Specifically, while the accuracy of ILA^{++} is 1% lower compared
to CSD [11] we remark that our model requires 123x less trainable parameters
compared to CSD. Furthermore, as CSD proposes a self-supervised (SSL) and
knowledge distillation (KD) enhanced training recipe, future work could aim to
combined such SSL and KD recipes with ILA for improved performance.

Table 2: Ablation on the design of the Residual Spatial Downsampling (RSDS) branch
in terms of absolute top-1 accuracy and absolute difference with respect to the baseline.

Model	SoyGlobal		SoyLocal	
	Acc. (%)	Diff. (%)	Acc. (%)	Diff. (%)
Baseline (ViT B-16)	17.88 ± 0.40		28.83 ± 1.04	
No RSDS	2.16 ± 0.89	-15.72	6.56 ± 1.42	-22.27
RSDS: AvgPool	29.07 ± 0.38	11.19	31.22 ± 1.25	2.39
RSDS: Convolution	34.93 ± 1.37	17.05	27.83 ± 5.63	-1.00
RSDS: DWC (Normal Init)	29.16 ± 1.78	11.28	25.22 ± 1.95	3.61
RSDS: DWC (Near Ones)	$\mathbf{43.48 \pm 0.21}$	**25.60**	$\mathbf{41.28 \pm 0.98}$	**12.45**

Ablation on Design of RSDS Results are shown in Tab. 2. We compare the
baseline against a model where the skip connection is forfeited (No RSDS), and
four variations of the RSDS module: based on average pooling, traditional con-
volution, depth-wise convolution with normal init, and our proposed depth-wise
with near ones initialization. It is evident that the usage of RSDS is necessary to
avoid collapse of the network. Furthermore, it is also evident that the proposed
approach is more effective as a residual compared to others.

6   Conclusion

In this paper we propose a novel intermediate layer adapter based on dual-branch
spatial down-sampling for parameter and compute efficient ultra fine-grained
image recognition. The proposed approach increases the diversity in attention
maps and obtains outstanding results in terms of accuracy-cost.

Acknowledgements

References

1. Abnar, S., Zuidema, W.: Quantifying Attention Flow in Transformers (May 2020). https://doi.org/10.48550/arXiv.2005.00928, http://arxiv.org/abs/2005.00928, arXiv:2005.00928 [cs]
2. Ba, J.L., Kiros, J.R., Hinton, G.E.: Layer Normalization (Jul 2016). https://doi.org/10.48550/arXiv.1607.06450, http://arxiv.org/abs/1607.06450, arXiv:1607.06450 [cs, stat]
3. Bai, J., Yuan, L., Xia, S.T., Yan, S., Li, Z., Liu, W.: Improving Vision Transformers by Revisiting High-Frequency Components. In: Avidan, S., Brostow, G., Cissé, M., Farinella, G.M., Hassner, T. (eds.) Computer Vision – ECCV 2022, vol. 13684, pp. 1–18. Springer Nature Switzerland, Cham (2022). https://doi.org/10.1007/978-3-031-20053-3_1, https://link.springer.com/10.1007/978-3-031-20053-3_1, series Title: Lecture Notes in Computer Science
4. Balestriero, R., Ibrahim, M., Sobal, V., Morcos, A., Shekhar, S., Goldstein, T., Bordes, F., Bardes, A., Mialon, G., Tian, Y., Schwarzschild, A., Wilson, A.G., Geiping, J., Garrido, Q., Fernandez, P., Bar, A., Pirsiavash, H., LeCun, Y., Goldblum, M.: A Cookbook of Self-Supervised Learning (Jun 2023). https://doi.org/10.48550/arXiv.2304.12210, http://arxiv.org/abs/2304.12210, arXiv:2304.12210 [cs]
5. Biewald, L.: Experiment Tracking with Weights and Biases (2020), https://www.wandb.com/
6. Caron, M., Touvron, H., Misra, I., Jegou, H., Mairal, J., Bojanowski, P., Joulin, A.: Emerging Properties in Self-Supervised Vision Transformers. In: 2021 IEEE/CVF International Conference on Computer Vision (ICCV). pp. 9630–9640 (Oct 2021). https://doi.org/10.1109/ICCV48922.2021.00951, https://ieeexplore.ieee.org/document/9709990, iSSN: 2380-7504
7. Chen, T., Zhang, Z., Cheng, Y., Awadallah, A., Wang, Z.: The Principle of Diversity: Training Stronger Vision Transformers Calls for Reducing All Levels of Redundancy. In: 2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). pp. 12010–12020. IEEE, New Orleans, LA, USA (Jun 2022). https://doi.org/10.1109/CVPR52688.2022.01171, https://ieeexplore.ieee.org/document/9878548/
8. Chen, T., Kornblith, S., Norouzi, M., Hinton, G.: A Simple Framework for Contrastive Learning of Visual Representations. In: Proceedings of the 37th International Conference on Machine Learning. pp. 1597–1607. PMLR (Nov 2020), https://proceedings.mlr.press/v119/chen20j.html, iSSN: 2640-3498
9. Devlin, J., Chang, M.W., Lee, K., Toutanova, K.: BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. arXiv:1810.04805 [cs] (May 2019), http://arxiv.org/abs/1810.04805, arXiv: 1810.04805

10. Dosovitskiy, A., Beyer, L., Kolesnikov, A., Weissenborn, D., Zhai, X., Unterthiner, T., Dehghani, M., Minderer, M., Heigold, G., Gelly, S., Uszkoreit, J., Houlsby, N.: An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale. arXiv:2010.11929 [cs] (Oct 2020), `http://arxiv.org/abs/2010.11929`, arXiv: 2010.11929

11. Fang, Z., Jiang, X., Tang, H., Li, Z.: Learning Contrastive Self-Distillation for Ultra-Fine-Grained Visual Categorization Targeting Limited Samples. IEEE Transactions on Circuits and Systems for Video Technology pp. 1–1 (2024). `https://doi.org/10.1109/TCSVT.2024.3370731`, `https://ieeexplore.ieee.org/document/10445701`, conference Name: IEEE Transactions on Circuits and Systems for Video Technology

12. He, J., Chen, J.N., Liu, S., Kortylewski, A., Yang, C., Bai, Y., Wang, C.: TransFG: A Transformer Architecture for Fine-Grained Recognition. In: Proceedings of the First MiniCon Conference (Feb 2022), `https://aaai-2022.virtualchair.net/poster_aaai8475`

13. He, K., Zhang, X., Ren, S., Sun, J.: Deep Residual Learning for Image Recognition. arXiv:1512.03385 [cs] (Dec 2015), `http://arxiv.org/abs/1512.03385`, arXiv: 1512.03385

14. Hinton, G., Vinyals, O., Dean, J.: Distilling the Knowledge in a Neural Network (Mar 2015). `https://doi.org/10.48550/arXiv.1503.02531`, `http://arxiv.org/abs/1503.02531`, arXiv:1503.02531 [cs, stat]

15. Hochreiter, S., Schmidhuber, J.: Long Short-Term Memory. Neural Comput. **9**(8), 1735–1780 (Nov 1997). `https://doi.org/10.1162/neco.1997.9.8.1735`, `https://doi.org/10.1162/neco.1997.9.8.1735`

16. Houlsby, N., Giurgiu, A., Jastrzebski, S., Morrone, B., Laroussilhe, Q.D., Gesmundo, A., Attariyan, M., Gelly, S.: Parameter-Efficient Transfer Learning for NLP. In: Proceedings of the 36th International Conference on Machine Learning. pp. 2790–2799. PMLR (May 2019), `https://proceedings.mlr.press/v97/houlsby19a.html`, iSSN: 2640-3498

17. Howard, A.G., Zhu, M., Chen, B., Kalenichenko, D., Wang, W., Weyand, T., Andreetto, M., Adam, H.: MobileNets: Efficient Convolutional Neural Networks for Mobile Vision Applications (Apr 2017). `https://doi.org/10.48550/arXiv.1704.04861`, `http://arxiv.org/abs/1704.04861`, arXiv:1704.04861 [cs]

18. Hu, Y., Jin, X., Zhang, Y., Hong, H., Zhang, J., He, Y., Xue, H.: RAMS-Trans: Recurrent Attention Multi-scale Transformer for Fine-grained Image Recognition. In: Proceedings of the 29th ACM International Conference on Multimedia. pp. 4239–4248. MM '21, Association for Computing Machinery, New York, NY, USA (Oct 2021). `https://doi.org/10.1145/3474085.3475561`, `https://doi.org/10.1145/3474085.3475561`

19. Jia, M., Tang, L., Chen, B.C., Cardie, C., Belongie, S., Hariharan, B., Lim, S.N.: Visual Prompt Tuning. In: Avidan, S., Brostow, G., Cissé, M., Farinella, G.M., Hassner, T. (eds.) Computer Vision – ECCV 2022, vol. 13693, pp. 709–727. Springer Nature Switzerland, Cham (2022). `https://doi.org/10.1007/978-3-031-19827-4_41`, `https://link.springer.com/10.1007/978-3-031-19827-4_41`, series Title: Lecture Notes in Computer Science

20. Jie, S., Deng, Z.H.: Convolutional Bypasses Are Better Vision Transformer Adapters (Aug 2022). `https://doi.org/10.48550/arXiv.2207.07039`, `http://arxiv.org/abs/2207.07039`, arXiv:2207.07039 [cs]

21. Kong, S., Fowlkes, C.: Low-Rank Bilinear Pooling for Fine-Grained Classification. In: 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR).

pp. 7025–7034. IEEE, Honolulu, HI (Jul 2017). `https://doi.org/10.1109/CVPR.2017.743`, `http://ieeexplore.ieee.org/document/8100226/`

22. Kornblith, S., Norouzi, M., Lee, H., Hinton, G.: Similarity of Neural Network Representations Revisited. In: Proceedings of the 36th International Conference on Machine Learning. pp. 3519–3529. PMLR (May 2019), `https://proceedings.mlr.press/v97/kornblith19a.html`, iSSN: 2640-3498

23. Krizhevsky, A., Sutskever, I., Hinton, G.E.: ImageNet Classification with Deep Convolutional Neural Networks. Advances in Neural Information Processing Systems **25**, 1097–1105 (2012), `https://proceedings.neurips.cc/paper/2012/hash/c399862d3b9d6b76c8436e924a68c45b-Abstract.html`

24. Larese, M.G., Bayá, A.E., Craviotto, R.M., Arango, M.R., Gallo, C., Granitto, P.M.: Multiscale recognition of legume varieties based on leaf venation images. Expert Systems with Applications **41**(10), 4638–4647 (Aug 2014). `https://doi.org/10.1016/j.eswa.2014.01.029`, `https://www.sciencedirect.com/science/article/pii/S0957417414000529`

25. Lehr, J., Sargsyan, A., Pape, M., Philipps, J., Krüger, J.: Automated Optical Inspection Using Anomaly Detection and Unsupervised Defect Clustering. In: 2020 25th IEEE International Conference on Emerging Technologies and Factory Automation (ETFA). vol. 1, pp. 1235–1238 (Sep 2020). `https://doi.org/10.1109/ETFA46521.2020.9212172`, `https://ieeexplore.ieee.org/abstract/document/9212172`, iSSN: 1946-0759

26. Lester, B., Al-Rfou, R., Constant, N.: The Power of Scale for Parameter-Efficient Prompt Tuning. In: Moens, M.F., Huang, X., Specia, L., Yih, S.W.t. (eds.) Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing. pp. 3045–3059. Association for Computational Linguistics, Online and Punta Cana, Dominican Republic (Nov 2021). `https://doi.org/10.18653/v1/2021.emnlp-main.243`, `https://aclanthology.org/2021.emnlp-main.243`

27. Li, P., Xie, J., Wang, Q., Gao, Z.: Towards Faster Training of Global Covariance Pooling Networks by Iterative Matrix Square Root Normalization. In: 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 947–955. IEEE, Salt Lake City, UT (Jun 2018). `https://doi.org/10.1109/CVPR.2018.00105`, `https://ieeexplore.ieee.org/document/8578203/`

28. Li, P., Xie, J., Wang, Q., Zuo, W.: Is Second-Order Information Helpful for Large-Scale Visual Recognition? In: 2017 IEEE International Conference on Computer Vision (ICCV). pp. 2089–2097. IEEE, Venice (Oct 2017). `https://doi.org/10.1109/ICCV.2017.228`, `http://ieeexplore.ieee.org/document/8237490/`

29. Liu, Z., Lin, Y., Cao, Y., Hu, H., Wei, Y., Zhang, Z., Lin, S., Guo, B.: Swin Transformer: Hierarchical Vision Transformer using Shifted Windows. In: 2021 IEEE/CVF International Conference on Computer Vision (ICCV). pp. 9992–10002. IEEE, Montreal, QC, Canada (Oct 2021). `https://doi.org/10.1109/ICCV48922.2021.00986`, `https://ieeexplore.ieee.org/document/9710580/`

30. Mohanty, S.P., Hughes, D.P., Salathé, M.: Using Deep Learning for Image-Based Plant Disease Detection. Frontiers in Plant Science **7** (Sep 2016). `https://doi.org/10.3389/fpls.2016.01419`, `https://www.frontiersin.org/journals/plant-science/articles/10.3389/fpls.2016.01419/full`, publisher: Frontiers

31. Park, W., Ryu, J.: Fine-Grained Self-Supervised Learning with Jigsaw Puzzles for Medical Image Classification (Aug 2023). `https://doi.org/10.48550/arXiv.2308.05770`, `http://arxiv.org/abs/2308.05770`, arXiv:2308.05770 [cs]

32. Paszke, A., Gross, S., Massa, F., Lerer, A., Bradbury, J., Chanan, G., Killeen, T., Lin, Z., Gimelshein, N., Antiga, L., Desmaison, A., Kopf, A., Yang, E., DeVito, Z.,

Raison, M., Tejani, A., Chilamkurthy, S., Steiner, B., Fang, L., Bai, J., Chintala, S.: PyTorch: An Imperative Style, High-Performance Deep Learning Library. In: Wallach, H., Larochelle, H., Beygelzimer, A., Alché-Buc, F.d., Fox, E., Garnett, R. (eds.) Advances in Neural Information Processing Systems 32, pp. 8024–8035. Curran Associates, Inc. (2019), `http://papers.neurips.cc/paper/9015-pytorch-an-imperative-style-high-performance-deep-learning-library.pdf`

33. Rao, Y., Chen, G., Lu, J., Zhou, J.: Counterfactual Attention Learning for Fine-Grained Visual Categorization and Re-Identification. pp. 1025–1034 (2021), `https://openaccess.thecvf.com/content/ICCV2021/html/Rao_Counterfactual_Attention_Learning_for_Fine-Grained_Visual_Categorization_and_Re-Identification_ICCV_2021_paper.html`

34. Rios, E.A., Hu, M.C., Lai, B.C.: Anime Character Recognition using Intermediate Features Aggregation. In: 2022 IEEE International Symposium on Circuits and Systems (ISCAS). pp. 424–428 (May 2022). `https://doi.org/10.1109/ISCAS48785.2022.9937519`, iSSN: 2158-1525

35. Rios, E.A., Hu, M.C., Lai, B.C.: Global-Local Similarity for Efficient Fine-Grained Image Recognition with Vision Transformers (Jul 2024), `http://arxiv.org/abs/2407.12891`, arXiv:2407.12891 [cs]

36. Sun, H., He, X., Peng, Y.: SIM-Trans: Structure Information Modeling Transformer for Fine-grained Visual Categorization. In: Proceedings of the 30th ACM International Conference on Multimedia. pp. 5853–5861. MM '22, Association for Computing Machinery, New York, NY, USA (Oct 2022). `https://doi.org/10.1145/3503161.3548308`, `https://doi.org/10.1145/3503161.3548308`

37. Tu, C.H., Mai, Z., Chao, W.L.: Visual Query Tuning: Towards Effective Usage of Intermediate Representations for Parameter and Memory Efficient Transfer Learning. In: 2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). pp. 7725–7735. IEEE, Vancouver, BC, Canada (Jun 2023). `https://doi.org/10.1109/CVPR52729.2023.00746`, `https://ieeexplore.ieee.org/document/10205336/`

38. Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A.N., Kaiser, L., Polosukhin, I.: Attention is All you Need. In: Advances in Neural Information Processing Systems. vol. 30. Curran Associates, Inc. (2017), `https://papers.nips.cc/paper/2017/hash/3f5ee243547dee91fbd053c1c4a845aa-Abstract.html`

39. Wang, J., Yu, X., Gao, Y.: Feature Fusion Vision Transformer for Fine-Grained Visual Categorization. In: British Machine Vision Conference (BMVC) (Jul 2021), `http://arxiv.org/abs/2107.02341`, arXiv: 2107.02341

40. Wang, P., Zheng, W., Chen, T., Wang, Z.: Anti-Oversmoothing in Deep Vision Transformers via the Fourier Domain Analysis: From Theory to Practice (Mar 2022). `https://doi.org/10.48550/arXiv.2203.05962`, `http://arxiv.org/abs/2203.05962`, arXiv:2203.05962 [cs]

41. Wang, W., Xie, E., Li, X., Fan, D.P., Song, K., Liang, D., Lu, T., Luo, P., Shao, L.: Pyramid Vision Transformer: A Versatile Backbone for Dense Prediction without Convolutions. In: 2021 IEEE/CVF International Conference on Computer Vision (ICCV). pp. 548–558 (Oct 2021). `https://doi.org/10.1109/ICCV48922.2021.00061`, `https://ieeexplore.ieee.org/document/9711179`, iSSN: 2380-7504

42. Wei, X.S., Song, Y.Z., Mac Aodha, O., Wu, J., Peng, Y., Tang, J., Yang, J., Belongie, S.: Fine-Grained Image Analysis with Deep Learning: A Survey. IEEE Transactions on Pattern Analysis and Machine Intelligence pp. 1–1 (2021). `https:`

`//doi.org/10.1109/TPAMI.2021.3126648`, conference Name: IEEE Transactions on Pattern Analysis and Machine Intelligence

43. Yu, X., Wang, J., Gao, Y.: CLE-ViT: Contrastive Learning Encoded Transformer for Ultra-Fine-Grained Visual Categorization. In: Proceedings of the Thirty-Second International Joint Conference on Artificial Intelligence. pp. 4531–4539. International Joint Conferences on Artificial Intelligence Organization, Macau, SAR China (Aug 2023). `https://doi.org/10.24963/ijcai.2023/504`, `https://www.ijcai.org/proceedings/2023/504`

44. Yu, X., Wang, J., Zhao, Y., Gao, Y.: Mix-ViT: Mixing attentive vision transformer for ultra-fine-grained visual categorization. Pattern Recognition **135**, 109131 (Mar 2023). `https://doi.org/10.1016/j.patcog.2022.109131`, `https://www.sciencedirect.com/science/article/pii/S0031320322006112`

45. Yu, X., Zhao, Y., Gao, Y.: SPARE: Self-supervised part erasing for ultra-fine-grained visual categorization. Pattern Recognition **128**, 108691 (Aug 2022). `https://doi.org/10.1016/j.patcog.2022.108691`, `https://www.sciencedirect.com/science/article/pii/S0031320322001728`

46. Yu, X., Zhao, Y., Gao, Y., Xiong, S.: MaskCOV: A random mask covariance network for ultra-fine-grained visual categorization. Pattern Recognition **119**, 108067 (Nov 2021). `https://doi.org/10.1016/j.patcog.2021.108067`, `https://www.sciencedirect.com/science/article/pii/S0031320321002545`

47. Yu, X., Zhao, Y., Gao, Y., Xiong, S., Yuan, X.: Patchy Image Structure Classification Using Multi-Orientation Region Transform. Proceedings of the AAAI Conference on Artificial Intelligence **34**(07), 12741–12748 (Apr 2020). `https://doi.org/10.1609/aaai.v34i07.6968`, `https://ojs.aaai.org/index.php/AAAI/article/view/6968`, number: 07

48. Yu, X., Zhao, Y., Gao, Y., Yuan, X., Xiong, S.: Benchmark Platform for Ultra-Fine-Grained Visual Categorization Beyond Human Performance. In: 2021 IEEE/CVF International Conference on Computer Vision (ICCV). pp. 10265–10275 (Oct 2021). `https://doi.org/10.1109/ICCV48922.2021.01012`, `https://ieeexplore.ieee.org/document/9710088`, iSSN: 2380-7504

49. Zhang, L., Bao, C., Ma, K.: Self-Distillation: Towards Efficient and Compact Neural Networks. IEEE Transactions on Pattern Analysis and Machine Intelligence **44**(08), 4388–4403 (Aug 2022). `https://doi.org/10.1109/TPAMI.2021.3067100`, `https://www.computer.org/csdl/journal/tp/2022/08/09381661/1s4kVUKSRfq`, publisher: IEEE Computer Society

50. Zhou, D., Kang, B., Jin, X., Yang, L., Lian, X., Jiang, Z., Hou, Q., Feng, J.: DeepViT: Towards Deeper Vision Transformer (Apr 2021). `https://doi.org/10.48550/arXiv.2103.11886`, `http://arxiv.org/abs/2103.11886`, arXiv:2103.11886 [cs]

Appendix for Down-Sampling Inter-Layer Adapter for Parameter and Computation Efficient Ultra-Fine-Grained Image Recognition

A   Experiment Methodology

In Tab. 3 we describe the datasets used for our experiments. These are ultra-fine-grained leaves datasets collected by Yu *et al.* [48] where each category represents a confirmed cultivar name attached to the seed obtained from the genetic resource bank.

Table 3: Dataset Statistics

Datasets	Classes	Train Images	Test Images
Cotton	80	240	240
SoyAgeing	198	4950	4950
SoyAgeingR1	198	990	990
SoyAgeingR3	198	990	990
SoyAgeingR4	198	990	990
SoyAgeingR5	198	990	990
SoyAgeingR6	198	990	990
SoyGene	1110	12763	11143
SoyGlobal	1938	5814	5814
SoyLocal	200	600	600

We conduct our experiments in two stages: first we conduct a learning rate search, $LR \in (0.3, 0.1, 0.03, 0.01, 0.003)$ based on subsets from the train data to select the best learning rate. We select the LR with highest accuracy in the validation subset. Then, in the second stage we use the LR from the first stage to train the model on the full training data and evaluate on the test set with 3 seeds. We use the Stochastic Gradient Descent (SGD) optimizer with momentum 0.9, batch size 8, cosine learning scheduler with 500 steps warm-up and we train all models for 50 epochs with automatic mixed-precision.

For data preprocessing, we resize our images to a square of size 300×300 or 600×600 and then crop a random square during training (or a center crop during inference) of size 224×224 or 448×448. All images are horizontally flipped and normalized based on standard ImageNet *mean* and *std* values.

We report results using top-1 accuracy (percentage) and standard deviation of the 3 seeds for image size 224 and image size 448. We also report computational cost based on a server with an RTX 3090 GPU. We report the number of trainable parameters (TTP) for a group of tasks, and the number of floating-point operations (FLOPs). We use Pytorch [32] to implement our experiments and Wandb [5] for experiment managing.

When applicable, the best values are highlighted in **bold** and the second best are underlined.

All of our experiments employ the ViT B-16 [10] backbone with patch size 16, number of layers $L = 12$ and hidden dimension size $D = 768$. We propose 3 different variants of ILA which are as follows:

- ILA: the intermediate layer adapters (ILAs) modules with down-sampling are inserted after layer 4 and 8 only.
- ILA$^+$: includes the modules from ILA plus additionally ILAs without down-sampling are inserted at every other layers of the ViT model besides from layer 4 and 8.
- ILA^{++}: includes the modules from ILA plus we additionally incorporate the traditional intra-layer adapters [16] in all layers.

We compare our proposed models against 15 state-of-the-art models in the parameter-efficient, grouped into three families based on their characteristics. The first includes methods which only fine-tune the classification head. This includes the following:

- Linear Classifier (Baseline): the most simple PE method which keeps all backbone parameters frozen and only fine-tunes the classification head.
- Low-Rank Bilinear Pooling (LR-BLP) [21]: employs a low-rank projection to reduce the dimensionality of the bilinearly pooled features.
- Matrix Power Normalized Covariance (MPN-Cov) [27,28]: applies covariance pooling of high-level features to lessen instabilities of bilinear pooling.
- Intermediate Features Aggregation (IFA) [34]: selects the CLS tokens from intermediate layers and forwards them through a small MLP to first aggregate cross-layer features before outputting classification predictions.

The second category includes FGIR methods where a module is designed to explicitly select features based on some criteria, along with a possible module to aggregate these selected discriminative features. We evaluate these models in the parameter-efficient setting (PEFGIR) where only a small percentage of modules are fine-tuned. It is composed of:

- CAL [33]: employs counter-factuality to train a bilinear attention module which is used for both pooling features and to generate augmented versions of the input images (crops and masked). The fine-tuned components include the attention module and the bilinear attention pooling classification head.
- TransFG [12]: selects features from the previous-to-last layer based on head-wise attention rollout [1], a matrix-multiplication based aggregation of attention scores across layers. We fine-tune the last transformer encoder block where the feature aggregation happens and the linear classification head.
- FFVT [39]: selects and aggregates intermediate features based on layer-wise attention. Same as the previous one: the last transformer encoder block and the classification head.

– RAMS-T [18]: crops the image for data augmentation based on attention rollout [18]. Fine-tunes only the classification head.
– GLSim [35]: computes the similarity between global and local representations of an image to select crops. Fine-tunes an aggregator transformer encoder block and the classification head.

The third category includes dedicated PETL methods as follows:

– VPT-Shallow (VPT-Sh) [19]: appends learnable prompts to the sequence at the start of the transformer.
– VPT-Deep [19]: appends learnable prompts to the sequence before each transformer block and then removes them after each block.
– Visual Query Tuning (VQT) [37]: appends learnable prompts to be used as queries only (not keys or values) to the sequence prior to the MHSA module of each layer. These prompts are expedited towards the classification head where they are concatenated into a single large dimensional linear layer.
– Adapter [16]: incorporates a small MLP inserted after the MHSA and PWFFN of each transformer encoder block.
– ConvPass [20]: similar to the previous it incorporates a small MLP inserted inside the transformer block, but this MLP incorporates a 3×3 convolution in between the channel downsampling and upsampling of the adapter.

We integrate all the previous methods into our experiment framework to ensure consistent training and evaluation for a fair comparison. Asides from these, we also compare against results previously published in the UFGIR literature, in particular the results published by Fang *et al.* [11] which include:

– SIM-Trans [36]: incorporates structure information and multi-level feature contrastive learning.
– CSDNet [11]: incorporates contrastive learning and self-distillation for learning with limited samples.