E^{2}VPT: An Effective and Efficient Approach for Visual Prompt Tuning
Supplementary Material

This supplementary contains additional experimental results and discussions of E^{2}VPT, organized as follows:

- §S1 provides per-task results on VTAB-1k and FGVC benchmarks, where the overall results have been provided in the main paper.
- §S2 presents more details and discussion on prepending Q ("query") matrix, and provides per-task results on VTAB-1k $Natural$.
- §S3 discusses our potential development on language-related tasks.
- §S4 gathers additional discussion, and reports recall results on 3 FGVC tasks presented in our paper.
- §S5 provides related license, reproducibility and discusses, social impact, limitations and directions of our future work.

S1. Per-task Results on VTAB-1k and FGVC

S1.1. Per-task Results on ViT-Base

To provide additional information on our results from the paper, we report the average per-task test accuracy (three runs, 24 tasks) on VTAB-1k [34] $Natural$, $Specialized$ and $Structured$, respectively (see Table S1, S2 and S3). FGVC [16] per-task results (5 tasks) are also available in Table S4. Notably, we also include VPT-SHALLOW [16] for reference (*i.e.*, VPT-SHALLOW only introduces 1-st layer visual prompts for lower parameter usage), our method generally get superior performance over VPT-DEEP (VPT) [16] while having competitive parameter usage to VPT-SHALLOW. In conclusion, we show consistently better performance in average over previous parameter-efficient and full fine-tuning methods with low parameter usage.

S1.2. Per-task Results on Swin-Base

We also report the per-task results on VTAB-1k [34] Swin-Base [23] in Table S5, S6 and S7. From these results, we observe similar pattern to §S1.1, which further validates the robustness of our approach.

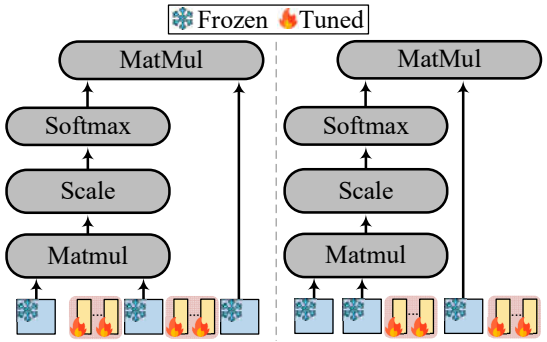

Figure S1. **Prompt locations on self-attention layer**. Two strategies (*i.e.*, "Before" (left) and "After" (right)) are discussed exhaustively in our paper (see Table S14, S15 and S16).

S1.3. Per-task Results on MAE and MoCo v3

We then present the per-task results on VTAB-1k [34] MAE [12] (see Table S8, S9 and S10) and MoCo v3 [5] (see Table S11, S12 and S13), respectively. Our method achieves large performance gains over VPT [16], and reach competitive results to full fine-tuning. The experimental results show generality of our E^{2}VPT on different pretraining objectives.

S1.4. Per-task Results for Prompt Location

We explicitly report the per-task results for different prompt locations on VTAB-1k [34] in Table S14, S15 and S16. Fig. S1 shows the corresponding prompt locations for "before" (left) and "after" (right). We choose "Before" as our baseline method since it obtains slightly better results.

S1.5. Per-task Results for Initialization

We further show the per-task results for initialization on VTAB-1k [34] in Table S17, S18 and S19. As discussed in the main paper, *He initialization* [13] generally provide more stable and preferable performances.

S2. Additional Prompting Strategies

In §3.2, we discuss possible prompting strategies on Q ("queries"), including prepending learnable prompts on Q and V (Strategy A), on Q and K (Strategy B), and on all Q,

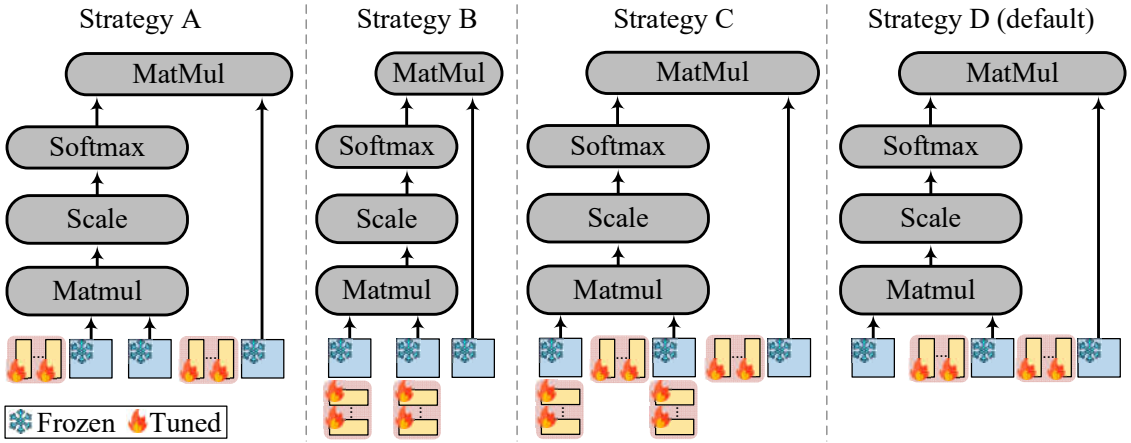

Figure S2. **Four prompting strategies on self-attention layer.** Strategy A, B and C are three additional methods for prepending (See §S2). Strategy D is our default method discussed in our paper. Per-task results on VTAB-1k [34] *Natural* are available in Table S20.

K and V (Strategy C) (See Figure S2). In Table S20, we present per-task results on VTAB-1k [34] *Natural* for these strategies, and compare to our default approach discussed in the main paper, respectively.

Following Eq. 4 in our paper. Strategy A can be simply formulated as:

$$\text{MSA}_A(\cdot) = \text{concat}(\text{softmax}(\frac{Q_h^{'} K_h^{T}}{\sqrt{d}})^T V_h^{'}) \quad (1)$$

where an additional "transpose" operation is required for the dimension alignment.

Strategy B adds learnable vectors on Q and K as:

$$\text{MSA}_B(\cdot) = \text{concat}(\text{softmax}(\frac{Q_h^{'} K_h^{'T}}{\sqrt{d}}) V_h) \quad (2)$$

We further investigate whether a heavier prompting strategy can result in relatively superior performance by prepending on both Q, K and V in Strategy C:

$$\text{MSA}_C(\cdot) = \text{concat}(\text{softmax}(\frac{Q_h^{'} K_h^{'T}}{\sqrt{d}}) V_h^{'}) \quad (3)$$

All prompting strategies are trained following exactly the same experiment settings mentioned in S§4.1 in our paper. From the results we can observe that:

- Strategy A's performance drops significantly compared to our default key-value prompting strategy (See Figure S2(d))), making it unsuitable for prompting. The reason is that the additional "transpose" operation switch the roles of Q and K in the pretrained backbone, which significantly deteriorates the fine-tuning performance.
- Strategy B obtains slightly lower results with our original key-value prompting (*i.e.*, 79.59% *vs* 80.01%)., which is consistent with our expectation. Notably, Strategy B still outperforms other fine-tuning methods largely, proving the direction of our prompting design on self-attention layer.
- Breaking free from the limitations of parameter usage, in Strategy C, we takes a step further by incorporating learnable vectors into the self-attention layer for all participants. Interestingly, we find additional cost on parameters does not relate to stronger performance (*i.e.*, 79.89% on Strategy C *vs* 80.01% on Strategy D (default)).

Overall, we find all discussed prompting strategies on self-attention layer report inferior (*i.e.*, Strategy A) or competitive (*i.e.*, Strategy B and C, while C requires higher parameter usage) results to our current key-value prompting. Further investigation is needed to find a solution for achieving potential performance gains.

S3. Extension to Language Tasks

As ViT-Base/16 [7] is structurally similar to BERT [6], we test the efficiency of the E^{2}VPT on natural language understanding (NLU) tasks. We include BERT-Large [6] for evaluation. Following [22], we compare full fine-tuning (FULL) [20], Prompt Tuning [20] and P-Tuning v2 [22] on SuperGlue [6] dataset, which is a collection of text classification tasks to test the general language understanding ability. To be specific, the tasks include natural language inference (RTE and CB), coreference resolution (WSC), sentence completion (COPA), word sense disambiguation (WiC), and question answering (MultiRC (Fla), ReCoRD (F1) and BoolQ). In Table S21, we outperform FULL and Prompt Tuning and show competitive results to P-Tuning v2 [22]. Considering E^{2}VPT is designed for visual-related tasks, these results are impressive and suggest future work

for a unified strategy on vision and language tasks under *pretrain-then-finetune* paradigm.

S4. Discussion on Hyperbolic Embeddings

S4.1. Related Works on Hyperbolic Embeddings

The application of hyperbolic embeddings for natural language processing (NLP) tasks is now widespread [24, 25]. Hyperbolic neural networks are introduced as a means of generalizing standard Euclidean operations, allowing for a direct learning of data representations in hyperbolic spaces. [10] further extends conventional linear layers to hyperbolic counterparts, and defines multinomial logistic regression and recurrent neural networks. Multiple studies demonstrate their advantages of hyperbolic embeddings for visual data, especially in the context of few-shot [9, 11, 18] and zero-shot [9, 21] learning. [18] proposes a hybrid architecture that employs most layers in Euclidean space layers, with only the final layers operating in hyperbolic space. Alternatively, [9] focuses on kernelization, which is widely used in Euclidean space, and generalizes it for hyperbolic representations. [21] proposes the direct incorporation of the hierarchical relations for hyperbolic embeddings in application to zero-shot learning.

S4.2. Recall@K Results

In Table S22, we follow [8, 30] and present the corresponding Recall@K metric from our paper. Our method presents relatively higher recall rate when comparing to VPT [16], reflecting a dense clustering in hyperbolic space.

S5. Discussion

S5.1. Asset License and Consent

The majority of VPT [16] is licensed under CC-BY-NC 4.0. Portions of [16] are available under separate license terms: google-research/task_adaptation and huggingface/transformers are licensed under Apache-2.0; Swin-Transformer [23] and ViT-pytorch [7] are licensed under MIT; and MoCo-v3 [5] and MAE [12] are licensed under CC BY 4.0.

S5.2. Reproducibility

E^{2}VPT is implemented in Pytorch [27]. Experiments are conducted on NVIDIA A100-40GB GPUs. To guarantee reproducibility, our full implementation shall be publicly released upon paper acceptance.

S5.3. Social Impact and Limitations

This work introduces E^{2}VPT possessing strong performance gains over several state-of-the-art baselines on two benchmarks, with considerably low parameter usage under *pretrain-then-finetune* paradigm for large-scale models (See §4.2 in our paper). Our approach advances model accuracy, and is valuable in real-world parameter-sensitive applications, e.g., machine learning models on devices and fast adaptation of large-scale models with limited resources. For potential limitations, our method requires two hyperparameters (*i.e.*, length of visual prompts and length of key-value prompts), which needs more combination of lengths on hyperparameters. Though in practice, we find both lengths fall into a relatively narrow range (See §4.1 in our paper), and are sufficient enough to outperform all current methods under *pretrain-then-finetune* paradigm, there is still possible integration [14] of local networks to generate optimal combinations of lengths. This indicates a possible direction for our future research.

S5.4. Future Work

Despite E^{2}VPT systemic effectiveness and efficacy, it also comes with new challenges and unveils some intriguing questions. For example, incorporating a small network into E^{2}VPT to generate optimal combinations of prompt lengths might improve training speed and lead to further performance gains. Another essential future direction deserving of further investigation is the design and analysis of network interpretability [1, 19, 29] and *ad-hoc* explainability [2], which limits current adoption of E^{2}VPT in decision-critical tasks. Furthermore, our experiments in §S2 aim to explore possible performance improvements, yet we have not observed a significant performance gap compared to our current method. Further investigation is needed for potential performance gains. Overall, we believe the results presented in this paper warrant further exploration.

References

[1] Alejandro Barredo Arrieta, Natalia Díaz-Rodríguez, Javier Del Ser, Adrien Bennetot, Siham Tabik, Alberto Barbado, Salvador García, Sergio Gil-López, Daniel Molina, Richard Benjamins, et al. Explainable artificial intelligence (xai): Concepts, taxonomies, opportunities and challenges toward responsible ai. *Information Fusion*, 58:82–115, 2020. 3

[2] Michael Biehl, Barbara Hammer, and Thomas Villmann. Prototype-based models in machine learning. *Wiley Interdisciplinary Reviews: Cognitive Science*, 7(2):92–111, 2016. 3

[3] Han Cai, Chuang Gan, Ligeng Zhu, and Song Han. Tinytl: Reduce memory, not parameters for efficient on-device learning. In *NeurIPS*, 2020. 5, 6

[4] Xinlei Chen, Haoqi Fan, Ross Girshick, and Kaiming He. Improved baselines with momentum contrastive learning. In *CVPR*, 2020. 5, 6

[5] Xinlei Chen, Saining Xie, and Kaiming He. An empirical study of training self-supervised vision transformers. In *ICCV*, 2021. 1, 3, 8

[6] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. In *NAACL-HLT*, 2018. 2, 11

[7] 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. 2, 3, 5, 6, 7, 8, 9, 10

[8] Aleksandr Ermolov, Leyla Mirvakhabova, Valentin Khrulkov, Nicu Sebe, and Ivan Oseledets. Hyperbolic vision transformers: Combining improvements in metric learning. In *CVPR*, 2022. 3

[9] Pengfei Fang, Mehrtash Harandi, and Lars Petersson. Kernel methods in hyperbolic spaces. In *CVPR*, 2021. 3

[10] Octavian Ganea, Gary Bécigneul, and Thomas Hofmann. Hyperbolic neural networks. In *NeurIPS*, 2018. 3

[11] Zhi Gao, Yuwei Wu, Yunde Jia, and Mehrtash Harandi. Curvature generation in curved spaces for few-shot learning. In *ICCV*, 2021. 3

[12] Kaiming He, Xinlei Chen, Saining Xie, Yanghao Li, Piotr Dollár, and Ross Girshick. Masked autoencoders are scalable vision learners. In *CVPR*, 2022. 1, 3, 7, 8

[13] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Delving deep into rectifiers: Surpassing human-level performance on imagenet classification. In *ICCV*, 2015. 1

[14] Yun He, Steven Zheng, Yi Tay, Jai Gupta, Yu Du, Vamsi Aribandi, Zhe Zhao, YaGuang Li, Zhao Chen, Donald Metzler, et al. Hyperprompt: Prompt-based task-conditioning of transformers. In *ICML*, 2022. 3

[15] Eugenia Iofinova, Alexandra Peste, Mark Kurtz, and Dan Alistarh. How well do sparse imagenet models transfer? In *CVPR*, 2022. 5, 6, 7, 8, 9, 10

[16] Menglin Jia, Luming Tang, Bor-Chun Chen, Claire Cardie, Serge Belongie, Bharath Hariharan, and Ser-Nam Lim. Visual prompt tuning. In *ECCV*, 2022. 1, 3, 5, 6, 7, 8, 9, 10, 11

[17] Aditya Khosla, Nityananda Jayadevaprakash, Bangpeng Yao, and Fei-Fei Li. Novel dataset for fine-grained image categorization: Stanford dogs. In *CVPR Workshop*, 2011. 11

[18] Valentin Khrulkov, Leyla Mirvakhabova, Evgeniya Ustinova, Ivan Oseledets, and Victor Lempitsky. Hyperbolic image embeddings. In *CVPR*, 2020. 3

[19] Thibault Laugel, Marie-Jeanne Lesot, Christophe Marsala, Xavier Renard, and Marcin Detyniecki. The dangers of post-hoc interpretability: Unjustified counterfactual explanations. In *IJCAI*, 2019. 3

[20] Brian Lester, Rami Al-Rfou, and Noah Constant. The power of scale for parameter-efficient prompt tuning. In *EMNLP*, 2021. 2, 11

[21] Shaoteng Liu, Jingjing Chen, Liangming Pan, Chong-Wah Ngo, Tat-Seng Chua, and Yu-Gang Jiang. Hyperbolic visual embedding learning for zero-shot recognition. In *CVPR*, 2020. 3

[22] Xiao Liu, Kaixuan Ji, Yicheng Fu, Zhengxiao Du, Zhilin Yang, and Jie Tang. P-tuning v2: Prompt tuning can be comparable to fine-tuning universally across scales and tasks. *arXiv preprint arXiv:2110.07602*, 2021. 2, 11

[23] Ze Liu, Yutong Lin, Yue Cao, Han Hu, Yixuan Wei, Zheng Zhang, Stephen Lin, and Baining Guo. Swin transformer: Hierarchical vision transformer using shifted windows. In *ICCV*, 2021. 1, 3, 6, 7

[24] Maximillian Nickel and Douwe Kiela. Poincaré embeddings for learning hierarchical representations. In *NeurIPS*, 2017. 3

[25] Maximillian Nickel and Douwe Kiela. Learning continuous hierarchies in the lorentz model of hyperbolic geometry. In *ICML*, 2018. 3

[26] Maria-Elena Nilsback and Andrew Zisserman. Automated flower classification over a large number of classes. In *Indian Conference on Computer Vision, Graphics & Image Processing*, 2008. 11

[27] Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, Alban Desmaison, Andreas Kopf, Edward Yang, Zachary DeVito, Martin Raison, Alykhan Tejani, Sasank Chilamkurthy, Benoit Steiner, Lu Fang, Junjie Bai, and Soumith Chintala. Pytorch: An imperative style, high-performance deep learning library. In *NeurIPS*, 2019. 3

[28] Sylvestre-Alvise Rebuffi, Hakan Bilen, and Andrea Vedaldi. Learning multiple visual domains with residual adapters. In *NeurIPS*, 2017. 5, 6

[29] Cynthia Rudin. Stop explaining black box machine learning models for high stakes decisions and use interpretable models instead. *Nature Machine Intelligence*, 1(5):206–215, 2019. 3

[30] Yanchao Tan, Carl Yang, Xiangyu Wei, Chaochao Chen, Longfei Li, and Xiaolin Zheng. Enhancing recommendation with automated tag taxonomy construction in hyperbolic space. In *ICDE*, 2022. 3

[31] Catherine Wah, Steve Branson, Peter Welinder, Pietro Perona, and Serge Belongie. The caltech-ucsd birds-200-2011 dataset. 2011. 11

[32] Alex Wang, Yada Pruksachatkun, Nikita Nangia, Amanpreet Singh, Julian Michael, Felix Hill, Omer Levy, and Samuel Bowman. Superglue: A stickier benchmark for general-purpose language understanding systems. In *NeurIPS*, 2019. 11

[33] Jason Yosinski, Jeff Clune, Yoshua Bengio, and Hod Lipson. How transferable are features in deep neural networks? In *NeurIPS*, 2014. 5, 6

[34] Xiaohua Zhai, Joan Puigcerver, Alexander Kolesnikov, Pierre Ruyssen, Carlos Riquelme, Mario Lucic, Josip Djolonga, Andre Susano Pinto, Maxim Neumann, Alexey Dosovitskiy, et al. A large-scale study of representation learning with the visual task adaptation benchmark. *arXiv preprint arXiv:1910.04867*, 2019. 1, 2, 5, 6, 7, 8, 9, 10

[35] Jeffrey O Zhang, Alexander Sax, Amir Zamir, Leonidas Guibas, and Jitendra Malik. Side-tuning: a baseline for network adaptation via additive side networks. In *ECCV*, 2020. 5, 6

Table S1. **VTAB-1k [34] *Natural* per-task results for ViT-Base/16 [7]** pretrained on supervised ImageNet-21k. Consistent to our paper, "Number of Wins" in [·] compared to full fine-tuning (Full) [15]. "Number of Wins to VPT" are shown in {·}. "Tuned/Total" is the average percentage of tuned parameters respectively on 24 tasks. The highest accuracy among all approaches except FULL are shown in **bold**. All results are averaged in three runs with different initialization seeds. Same for Table S2-S20. We also report standard deviation error bars for our main results (Table S1, S2, S3 and S4) by calculating each task respectively and averaging across them. Other tables show similar trends on standard deviation error bars.

ViT-Base/16 [7] (85.8M)	VTAB-1k [34] *Natural* [7]							Mean
	CIFAR-100	Caltech101	DTD	Flowers102	Pets	SVHN	Sun397	
FULL [15]	68.9	87.7	64.3	97.2	86.9	87.4	38.8	75.88
LINEAR [15]	63.4	85.0	63.2	97.0	86.3	36.6	51.0	68.93 [1]
PARTIAL-1 [33]	66.8	85.9	62.5	97.3	85.5	37.6	50.6	69.44 [2]
MLP-2 [4]	63.2	84.8	60.5	97.6	85.9	34.1	47.8	67.70 [2]
MLP-3 [4]	63.8	84.7	62.3	97.4	84.7	32.5	49.2	67.80 [2]
MLP-5 [4]	59.3	84.4	59.9	96.1	84.4	30.9	46.8	65.98 [1]
MLP-9 [4]	53.1	80.5	53.9	95.1	82.6	24.4	43.7	61.90 [1]
SIDETUNE [35]	60.7	60.8	53.6	95.5	66.7	34.9	35.3	58.21 [0]
BIAS [28]	72.8	87.0	59.2	97.5	85.3	59.9	51.4	73.30 [3]
ADAPTER-256 [3]	74.1	86.1	63.2	97.7	87.0	34.6	50.8	70.50 [4]
ADAPTER-64 [3]	74.2	85.8	62.7	97.6	87.2	36.3	50.9	70.65 [4]
ADAPTER-8 [3]	74.2	85.7	62.7	97.8	87.2	36.4	50.7	70.67 [4]
VPT-SHALLOW [16]	77.7	86.9	62.6	97.5	87.3	74.5	51.2	76.81 [4]
- Tuned / Total (%)	0.18	0.10	0.04	0.27	0.08	0.19	0.36	0.17
VPT-DEEP [16]	**78.8**	**90.8**	65.8	98.0	88.3	78.1	49.6	78.48 [6]
- Tuned / Total (%)	0.20	0.20	0.15	0.10	0.04	0.54	0.41	0.23
OURS	78.6 ± (0.01)	89.4 ± (0.34)	**67.8** ± (0.53)	**98.2** ± (0.08)	**88.5** ± (0.36)	**85.3** ± (0.33)	**52.3** ± (0.08)	**80.01** ± (0.24) [6]
- Tuned / Total (%)	0.22	0.19	0.12	0.11	0.05	0.24	0.43	0.19
- Pruning (%)	51.3	18.8	55.0	6.3	56.3	15.6	62.5	37.97

Table S2. **VTAB-1k [34] *Specialized* per-task results for ViT-Base/16 [7]** pretrained on supervised ImageNet-21k.

ViT-Base/16 [7] (85.8M)	VTAB-1k [34] *Specialized* (4)				Mean
	Patch Camelyon	EuroSAT	Resisc45	Retinopathy	
FULL [15]	79.7	95.7	84.2	73.9	83.36
LINEAR [15]	78.5	87.5	68.6	**74.0**	77.16 [1]
PARTIAL-1 [33]	78.6	89.8	72.5	73.3	78.53 [0]
MLP-2 [4]	74.3	88.8	67.1	73.2	75.86 [0]
MLP-3 [4]	77.0	88.0	70.2	56.1	72.83 [0]
MLP-5 [4]	73.7	87.2	64.8	71.5	74.31 [0]
MLP-9 [4]	78.5	83.0	60.2	72.3	73.49 [0]
SIDETUNE [35]	58.5	87.7	65.2	61.0	68.12 [0]
BIAS [28]	78.7	91.6	72.9	69.8	78.25 [0]
ADAPTER-256 [3]	76.3	88.0	73.1	70.5	76.98 [0]
ADAPTER-64 [3]	76.3	87.5	73.7	70.9	77.10 [0]
ADAPTER-8 [3]	76.9	89.2	73.5	71.6	77.80 [0]
VPT-SHALLOW [16]	78.2	92.0	75.6	72.9	79.66 [0]
- Tuned / Total (%)	0.01	0.05	0.09	0.01	0.04
VPT-DEEP [16]	81.8	96.1	83.4	68.4	82.43 [2]
- Tuned / Total (%)	1.06	1.07	0.15	0.02	0.57
OURS	**82.5** ± (0.67)	**96.8** ± (0.06)	**84.8** ± (0.56)	73.6 ± (0.04)	**84.43** ± (0.33) [3]
- Tuned / Total (%)	0.20	0.29	0.12	0.07	0.17
- Pruning (%)	65.0	75.0	34.4	47.5	55.48

Table S3. **VTAB-1k [34] *Structured* per-task results for ViT-Base/16 [7]** pretrained on supervised ImageNet-21k.

ViT-Base/16 [7] (86.7M)	VTAB-1k [34] *Structured* [8]								Mean
	Clevr/ count	Clevr/ distance	DMLab	KITTI/ distance	dSprites/ location	dSprites/ orientation	SmallNORB/ azimuth	SmallNORB/ elevation	
FULL [15]	56.3	58.6	41.7	65.5	57.5	46.7	25.7	29.1	47.64
LINEAR [15]	34.3	30.6	33.2	55.4	12.5	20.0	9.6	19.2	26.84 [0]
PARTIAL-1 [33]	41.5	34.3	33.9	61.0	31.3	32.8	16.3	22.4	34.17 [0]
MLP-2 [4]	45.2	31.6	31.8	55.7	30.9	24.6	16.6	23.3	32.47 [0]
MLP-3 [4]	47.8	32.8	32.3	58.1	12.9	21.2	15.2	24.8	30.62 [0]
MLP-5 [4]	50.8	32.3	31.5	56.4	7.5	20.8	14.4	20.4	29.23 [0]
MLP-9 [4]	47.5	27.9	28.9	54.0	6.2	17.7	10.8	16.2	26.15 [0]
SIDETUNE [35]	27.6	22.6	31.3	51.7	8.2	14.4	9.8	21.8	23.41 [0]
BIAS [28]	61.5	55.6	32.4	55.9	66.6	40.0	15.7	25.1	44.09 [2]
ADAPTER-256 [3]	45.7	37.4	31.2	53.2	30.3	25.4	13.8	22.1	32.39 [0]
ADAPTER-64 [3]	42.9	39.9	30.4	54.5	31.9	25.6	13.5	21.4	32.51 [0]
ADAPTER-8 [3]	45.2	41.8	31.1	56.4	30.4	24.6	13.2	22.0	33.09 [0]
VPT-SHALLOW [16]	50.5	58.6	40.5	67.1	68.7	36.1	20.2	34.1	46.98 [4]
- Tuned / Total (%)	0.10	0.18	0.09	0.09	0.10	0.10	0.19	0.19	0.13
VPT-DEEP [16]	68.5	60.0	46.5	72.8	73.6	47.9	**32.9**	37.8	54.98 [8]
- Tuned / Total (%)	0.54	2.11	1.07	0.54	0.12	0.55	2.12	2.11	1.14
OURS	**71.7** ± (1.13)	**61.2** ± (0.83)	**47.9** ± (0.05)	**75.8** ± (0.50)	**80.8** ± (0.10)	**48.1** ± (0.53)	31.7 ± (0.29)	**41.9** ± (1.13)	**57.39** ± (0.57) [8]
- Tuned / Total (%)	0.34	0.65	0.44	0.36	0.10	0.38	1.14	0.66	0.51
- Pruning (%)	40.0	68.8	55.0	25.0	26.9	34.4	51.3	62.5	45.49

Table S4. **FGVC [16] per-task results for ViT-Base/16 [7]** pretrained on supervised ImageNet-21k.

ViT-Base/16 [7] (85.8M)	FGVC [16] [5]					Mean
	CUB-200-2011	NAbirds	Oxford Flowers	Stanford Dogs	Stanford Cars	
FULL [15]	87.3	82.7	98.8	89.4	84.5	88.54
LINEAR [15]	85.3	75.9	97.9	86.2	51.3	79.32 [0]
PARTIAL-1 [33]	85.6	77.8	98.2	85.5	66.2	82.63 [0]
MLP-2 [4]	85.7	77.2	98.2	85.4	54.9	80.28 [0]
MLP-3 [4]	85.1	77.3	97.9	84.9	53.8	79.80 [0]
MLP-5 [4]	84.2	76.7	97.6	84.8	50.2	78.71 [0]
MLP-9 [4]	83.2	76.0	96.2	83.7	47.6	77.31 [0]
SIDETUNE [35]	84.7	75.8	96.9	85.8	48.6	78.35 [0]
BIAS [28]	88.4	84.2	98.8	91.2	79.4	88.41 [3]
ADAPTER-256 [3]	87.2	84.3	98.5	89.9	68.6	85.70 [2]
ADAPTER-64 [3]	87.1	84.3	98.5	89.8	68.6	85.67 [2]
ADAPTER-8 [3]	87.3	84.3	98.4	88.8	68.4	85.46 [1]
VPT-SHALLOW [16]	86.7	78.8	98.4	90.7	68.7	84.62 [1]
- Tuned / Total (%)	0.31	0.54	0.23	0.20	0.26	0.31
VPT-DEEP [16]	88.5	84.2	99.0	90.2	**83.6**	89.11 [4]
- Tuned / Total (%)	0.29	1.02	0.14	1.17	2.27	0.98
OURS	**89.1** ± (0.05)	**84.6** ± (0.10)	**99.1** ± (0.07)	**90.5** ± (0.11)	82.8 ± (0.11)	**89.22** ± (0.09) [4]
- Tuned / Total (%)	0.32	0.65	0.15	0.88	1.27	0.65
- Pruning (%)	6.3	40.0	6.3	26.9	50.0	25.9

Table S5. **VTAB-1k [34] *Natural* per-task results for Swin-Base [23]** pretrained on supervised ImageNet-21k.

Swin-Base [23] (86.7M)	VTAB-1k [34] *Natural* (7)							Mean
	CIFAR-100	Caltech101	DTD	Flowers102	Pets	SVHN	Sun397	
FULL [15]	72.2	88.0	71.2	98.3	89.5	**89.4**	45.0	79.10
VPT-SHALLOW [16]	77.7	86.9	62.6	97.5	87.3	74.5	51.2	76.81 [4]
- Tuned / Total (%)	0.18	0.10	0.04	0.27	0.08	0.19	0.36	0.17
VPT-DEEP [16]	79.6	90.8	78.0	99.5	**91.4(3)**	46.4	51.7	78.78 [6]
- Tuned / Total (%)	0.13	0.13	0.07	0.13	0.06	0.70	0.48	0.28
OURS	**82.9**	**92.4**	**78.5**	**99.6**	91.4(1)	82.2	**56.2**	**83.31** [6]
- Tuned / Total (%)	0.27	0.15	0.08	0.15	0.07	0.44	0.49	0.24
- Pruning (%)	35.0	68.8	56.3	37.5	25.0	77.5	25.0	46.44

Table S6. **VTAB-1k [34] *Specialized* per-task results for Swin-Base [23]** pretrained on supervised ImageNet-21k.

Swin-Base [23] (86.7M)	VTAB-1k [34] *Specialized* [4]				Mean
	Patch Camelyon	EuroSAT	Resisc45	Retinopathy	
FULL [15]	**86.6**	96.9	**87.7**	73.6	86.21
VPT-SHALLOW [16]	78.2	92.0	75.6	72.9	79.66 [0]
- Tuned / Total (%)	0.01	0.05	0.09	0.01	0.04
VPT-DEEP [16]	80.1	96.2	85.0	72.0	83.33 [0]
- Tuned / Total (%)	0.07	0.13	0.19	0.02	0.10
OURS	83.8	**97.2**	84.8	**74.0**	**84.95** [2]
- Tuned / Total (%)	0.09	0.04	0.20	0.03	0.09
- Pruning (%)	68.8	68.8	93.8	37.5	67.23

Table S7. **VTAB-1k [34] *Structured* per-task results for Swin-Base [23]** pretrained on supervised ImageNet-21k.

Swin-Base [23] (86.7M)	VTAB-1k [34] *Structured* [8]								Mean
	Clevr/ count	Clevr/ distance	DMLab	KITTI/ distance	dSprites/ location	dSprites/ orientation	SmallNORB/ azimuth	SmallNORB/ elevation	
FULL [15]	**75.7**	59.8	**54.6**	78.6	79.4	53.6	**34.6**	**40.9**	59.65
VPT-SHALLOW [16]	50.5	58.6	40.5	67.1	68.7	36.1	20.2	34.1	46.98 [4]
- Tuned / Total (%)	0.10	0.18	0.09	0.09	0.10	0.10	0.19	0.19	0.13
VPT-DEEP [16]	67.6	59.4	50.1	61.3	74.4	50.6	25.7	25.7	51.85 [0]
- Tuned / Total (%)	0.70	0.70	0.14	0.69	0.15	0.09	0.16	0.02	0.38
OURS	74.0	**61.2**	49.5	**81.0**	**80.3**	**50.7**	27.9	34.2	**57.35** [3]
- Tuned / Total (%)	0.70	0.43	0.14	0.51	0.17	0.17	0.16	0.04	0.29
- Pruning (%)	97.5	87.5	35.0	71.9	32.5	92.5	87.5	75.0	72.43

Table S8. **VTAB-1k [34] *Natural* per-task results for ViT-Base/16 [7] pretrained on MAE [12].** Since VPT [16] have considerably lower performance, we do not list the per-task results for simplicity. We instead compare our method to full fine-tuning, and the highest accuracy is shown in **bold**. We post the "Number of Wins" in [·] to full fine-tuning (FULL) [15]. Same for Table S9-S13.

ViT-Base/16 [7] (85.8M)	VTAB-1k [34] *Natural* [7]							Mean
	CIFAR-100	Caltech101	DTD	Flowers102	Pets	SVHN	Sun397	
FULL [15]	24.6	84.2	56.9	**72.7**	**74.4**	**86.6**	15.8	59.31
OURS	**27.0**	**86.2**	**60.4**	71.7	73.7	76.6	**21.1**	**59.52** [4]
- Tuned / Total (%)	0.11	0.11	0.06	0.11	0.05	0.03	0.37	0.12
- Pruning (%)	25.0	37.5	37.5	6.5	25.0	6.3	18.8	22.37

Table S9. **VTAB-1k [34] *Specialized* per-task results for ViT-Base/16 [7] pretrained on MAE [12].**

ViT-Base/16 [7] (85.8M)	VTAB-1k [34] *Specialized* [4]				Mean
	Patch Camelyon	EuroSAT	Resisc45	Retinopathy	
FULL [15]	**81.8**	**94.0**	**72.3**	70.6	**79.68**
OURS	79.5	91.7	66.5	**73.5**	77.80 [1]
- Tuned / Total (%)	0.02	0.03	0.13	0.06	0.06
- Pruning (%)	83.8	18.8	25.0	91.9	54.88

Table S10. **VTAB-1k [34] *Strcutured* per-task results for ViT-Base/16 [7] pretrained on MAE [12].**

ViT-Base/16 [7] (86.7M)	VTAB-1k [34] *Structured* [8]								Mean
	Clevr/ count	Clevr/ distance	DMLab	KITTI/ distance	dSprites/ location	dSprites/ orientation	SmallNORB/ azimuth	SmallNORB/ elevation	
FULL [15]	**67.0**	59.8	**45.2**	75.3	72.5	**47.5**	**30.2**	**33.0**	**53.82**
OURS	41.7	**61.2**	38.7	**76.7**	**81.7**	15.3	14.1	27.8	44.65 [3]
- Tuned / Total (%)	0.04	0.09	0.03	0.19	0.40	0.03	0.03	0.02	0.10
- Pruning (%)	55.0	75.6	82.5	50.0	82.5	18.8	75.0	68.8	63.53

Table S11. **VTAB-1k [34] *Natural* per-task results for ViT-Base/16 [7] pretrained on MOCO [5].**

ViT-Base/16 [7] (85.8M)	VTAB-1k [34] *Natural* [7]							Mean
	CIFAR-100	Caltech101	DTD	Flowers102	Pets	SVHN	Sun397	
FULL [15]	57.6	**91.0**	64.6	**91.6**	79.9	**89.8**	29.1	71.95
OURS	**72.7**	89.1	**69.5**	91.3	**88.5**	82.1	**42.0**	**76.74** [4]
- Tuned / Total (%)	0.16	0.88	0.06	0.11	0.05	0.05	0.37	0.24
- Pruning (%)	43.8	26.9	6.3	6.3	18.8	37.5	50.0	27.09

Table S12. **VTAB-1k [34] *Specialized* per-task results for ViT-Base/16 [7] pretrained on MOCO [5].**

ViT-Base/16 [7] (85.8M)	VTAB-1k [34] *Specialized* [4]				Mean
	Patch Camelyon	EuroSAT	Resisc45	Retinopathy	
FULL [15]	85.1	**96.4**	83.1	**74.2(7)**	84.72
OURS	**95.0**	95.3	**84.5**	74.2(6)	**87.28** [2]
- Tuned / Total (%)	0.02	0.03	0.08	0.03	0.04
- Pruning (%)	56.3	25.0	55.0	70.0	51.58

Table S13. **VTAB-1k [34] *Structured* per-task results for ViT-Base/16 [7] pretrained on MOCO [5].**

ViT-Base/16 [7] (86.7M)	VTAB-1k [34] *Structured* [8]								Mean
	Clevr/ count	Clevr/ distance	DMLab	KITTI/ distance	dSprites/ location	dSprites/ orientation	SmallNORB/ azimuth	SmallNORB/ elevation	
FULL [15]	55.2	56.9	44.6	77.9	63.8	**49.0**	**31.5**	36.9	51.98
OURS	**59.7**	**62.7**	**45.7**	**79.8**	**80.6**	48.8	23.3	**38.7**	**54.91** [6]
- Tuned / Total (%)	0.02	0.14	0.02	0.19	0.10	0.08	0.04	0.07	0.08
- Pruning (%)	68.8	78.1	18.8	56.3	37.5	6.3	6.3	6.3	34.8

Table S14. **Prompt location per-task results on VTAB-1k [34] *Natural*** for ViT-Base/16 [7] pretrained on supervised ImageNet-21k. See §S1.4, same for Table S15 and S16.

ViT-Base/16 [7] (85.8M)	VTAB-1k [34] *Natural* [7]							Mean
	CIFAR-100	Caltech101	DTD	Flowers102	Pets	SVHN	Sun397	
FULL [15]	57.6	91.0	64.6	91.6	79.9	89.8	29.1	71.95
VPT-SHALLOW [16]	77.7	86.9	62.6	97.5	87.3	74.5	51.2	76.81 [4]
- Tuned / Total (%)	0.18	0.10	0.04	0.27	0.08	0.19	0.36	0.17
VPT-DEEP [16]	78.8	90.8(3)	65.8	98.0	88.3	78.1	49.6	78.48 [6]
- Tuned / Total (%)	0.20	0.20	0.15	0.10	0.04	0.54	0.41	0.23
OURS-After	**79.3**	**90.8(4)**	**69.4**	**98.3**	**88.6**	**86.0**	**52.3(1)**	**80.67** [6]
- Tuned / Total (%)	0.13	0.16	0.13	0.16	0.31	0.05	0.43	0.20
- Pruning (%)	83.8	43.2	32.5	6.3	50.0	52.1	55.0	46.12
OURS-Before (default)	78.6	89.4	67.8	98.2	88.5	85.3	52.2(7)	80.01 [6]
- Tuned / Total (%)	0.22	0.19	0.12	0.11	0.05	0.24	0.43	0.19
- Pruning (%)	51.3	18.8	55.0	6.3	56.3	15.6	62.5	37.97

Table S15. **Prompt location per-task results on VTAB-1k [34] *Specialized*** for ViT-Base/16 [7] pretrained on supervised ImageNet-21k.

ViT-Base/16 [7] (85.8M)	VTAB-1k [34] *Specialized* [4]				Mean
	Patch Camelyon	EuroSAT	Resisc45	Retinopathy	
FULL [15]	85.1	96.4	83.1	74.3	84.72
VPT-SHALLOW [16]	78.2	92.0	75.6	72.9	79.66 [0]
- Tuned / Total (%)	0.01	0.05	0.09	0.01	0.04
VPT-DEEP [16]	81.8	96.1	83.4	68.4	82.43 [2]
- Tuned / Total (%)	1.06	1.07	0.15	0.02	0.57
OURS-After	82.2	96.7	84.1	**74.2**	84.30 [3]
- Tuned / Total (%)	0.13	0.11	0.06	0.02	0.08
- Pruning (%)	82.5	56.3	91.3	75.0	76.28
OURS-Before (default)	**82.5**	**96.8**	**84.8**	73.6	**84.43** [3]
- Tuned / Total (%)	0.20	0.29	0.12	0.07	0.17
- Pruning (%)	65.0	75.0	34.4	47.5	55.48

Table S16. **Prompt location per-task results on VTAB-1k [34] *Structured*** for ViT-Base/16 [7] pretrained on supervised ImageNet-21k.

ViT-Base/16 [7] (86.7M)	VTAB-1k [34] *Structured* [8]								Mean
	Clevr/ count	Clevr/ distance	DMLab	KITTI/ distance	dSprites/ location	dSprites/ orientation	SmallNORB/ azimuth	SmallNORB/ elevation	
FULL [15]	55.2	56.9	44.6	77.9	63.8	49.0	31.5	36.9	51.98
VPT-SHALLOW [16]	50.5	58.6	40.5	67.1	68.7	36.1	20.2	34.1	46.98 [4]
- Tuned / Total (%)	0.10	0.18	0.09	0.09	0.10	0.10	0.19	0.19	0.13
VPT-DEEP [16]	68.5	60.0	46.5	72.8	73.6	47.9	32.9	37.8	54.98 [8]
- Tuned / Total (%)	0.54	2.11	1.07	0.54	0.12	0.55	2.12	2.11	1.14
OURS-After	70.6	58.8	46.9	**77.5**	**81.4**	**48.6**	30.5	39.8	56.76 [8]
- Tuned / Total (%)	0.38	0.24	0.73	0.24	0.12	0.27	0.62	0.54	0.39
- Pruning (%)	26.9	91.9	26.9	50.0	15.6	43.8	81.3	70.0	50.80
OURS-Before (default)	**71.7**	**61.2**	**47.9**	75.8	80.8	48.1	**31.7**	**41.9**	**57.39** [8]
- Tuned / Total (%)	0.34	0.65	0.44	0.36	0.10	0.38	1.14	0.66	0.51
- Pruning (%)	40.0	68.8	55.0	25.0	26.9	34.4	51.3	62.5	45.49

Table S17. **Initialization per-task results on VTAB-1k [34] *Natural*** for ViT-Base/16 [7] pretrained on supervised ImageNet-21k. See §S1.5, same for Table S18 and S19.

ViT-Base/16 [7] (85.8M)	VTAB-1k [34] *Natural* [7]							Mean
	CIFAR-100	Caltech101	DTD	Flowers102	Pets	SVHN	Sun397	
OURS-*Turnc. Norm.*	**79.9**	**89.9**	66.0	98.1	88.3	84.8	51.4	79.77 [6]
- Tuned / Total (%)	0.13	0.14	0.06	0.15	0.05	0.18	0.37	0.15
- Pruning (%)	75.6	62.5	91.9	18.8	25.0	78.1	68.8	60.10
OURS-*He* (default)	78.6	89.4	**67.8**	**98.2**	**88.5**	**85.3**	**52.3**	**80.01** [6]
- Tuned / Total (%)	0.22	0.19	0.12	0.11	0.05	0.24	0.43	0.19
- Pruning (%)	51.3	18.8	55.0	6.3	56.3	15.6	62.5	37.97

Table S18. **Initialization per-task results on VTAB-1k [34] *Specialized*** for ViT-Base/16 [7] pretrained on supervised ImageNet-21k.

ViT-Base/16 [7] (85.8M)	VTAB-1k [34] *Specialized* [4]				Mean
	Patch Camelyon	EuroSAT	Resisc45	Retinopathy	
OURS-*Turnc. Norm.*	82.1	96.6	84.6	**73.9**	84.30 [3]
- Tuned / Total (%)	0.14	0.20	0.10	0.02	0.12
- Pruning (%)	67.5	25.0	50.0	56.3	49.7
OURS-*He* (default)	**82.5**	**96.8**	**84.8**	73.6	**84.43** [3]
- Tuned / Total (%)	0.20	0.29	0.12	0.07	0.17
- Pruning (%)	65.0	75.0	34.4	47.5	55.48

Table S19. **Initialization per-task results on VTAB-1k [34] *Structured*** for ViT-Base/16 [7] pretrained on supervised ImageNet-21k.

ViT-Base/16 [7] (86.7M)	VTAB-1k [34] *Structured* [8]								Mean
	Clevr/ count	Clevr/ distance	DMLab	KITTI/ distance	dSprites/ location	dSprites/ orientation	SmallNORB/ azimuth	SmallNORB/ elevation	
OURS-*Turnc. Norm.*	70.1	58.8	47.2	75.5	80.6	**48.6**	30.2	39.9	56.36 [8]
- Tuned / Total (%)	0.27	0.64	0.28	0.32	0.07	0.23	0.33	0.54	0.34
- Pruning (%)	53.1	70.0	80.0	62.5	62.5	62.5	90.0	70.0	68.83
OURS-*He* (default)	**71.7**	**61.2**	**47.9**	**75.8**	**80.8**	48.1	**31.7**	**41.9**	**57.39** [8]
- Tuned / Total (%)	0.34	0.65	0.44	0.36	0.10	0.38	1.14	0.66	0.51
- Pruning (%)	40.0	68.8	55.0	25.0	26.9	34.4	51.3	62.5	45.49

Table S20. **Key-value prompts vs additional prompting strategies on self-attention layer.** We provide per-task results under *pretrain-then-finetune* paradigm for VTAB-1k [34] *Natural* with a ViT-Base/16 [7] pretrained on supervised ImageNet-21k. See §S2.

ViT-Base/16 [7] (85.8M)	VTAB-1k [34] *Natural* [7]							Mean
	CIFAR-100	Caltech101	DTD	Flowers102	Pets	SVHN	Sun397	
FULL [15]	68.9	87.7	64.3	97.2	86.9	87.4	38.8	75.88
VPT-SHALLOW [16]	77.7	86.9	62.6	97.5	87.3	74.5	51.2	76.81 [4]
- Tuned / Total (%)	0.18	0.10	0.04	0.27	0.08	0.19	0.36	0.17
VPT-DEEP [16]	78.8	**90.8**	65.8	98.0	88.3	78.1	49.6	78.48 [6]
- Tuned / Total (%)	0.20	0.20	0.15	0.10	0.04	0.54	0.41	0.23
OURS (Strategy A)	23.7	64.7	55.4	75.7	43.9	71.9	17.6	50.41 [0]
- Tuned / Total (%)	0.17	0.16	0.09	0.14	0.13	0.20	0.40	0.18
- Pruning (%)	35.0	43.8	67.5	25.0	25.0	35.0	40.0	38.76
OURS (Strategy B)	79.0	89.7	**68.7**	98.0	87.6	82.4	51.7	79.59 [6]
- Tuned / Total (%)	0.29	0.17	0.25	0.13	0.21	0.31	0.40	0.25
- Pruning (%)	50.0	60.0	25.0	25.0	85.0	42.8	70.0	51.11
OURS (Strategy C)	**79.7**	89.6	68.5	97.9	87.8	83.3	**52.4**	79.89 [6]
- Tuned / Total (%)	0.50	0.51	0.15	0.19	0.45	0.51	0.47	0.35
- Pruning (%)	65.0	60.0	75.0	18.8	55.0	62.5	37.5	53.4
OURS (Strategy D (default))	78.6	89.4	67.8	**98.2**	**88.5**	**85.3**	52.3	**80.01** [6]
- Tuned / Total (%)	0.22	0.19	0.12	0.11	0.05	0.24	0.43	0.19
- Pruning (%)	51.3	18.8	55.0	6.3	56.3	15.6	62.5	37.97

Table S21. **Per-task results for SuperGLUE development set [32] with a pretrained BERT-Large [6].** The highest accuracy among all approaches except FULL [20] are shown in **bold**. See §S3.

BERT-Large [6] (335M)	SuperGLUE [6] [8]								Mean
	BoolQ	CB	COPA	MultiRC (Fla)	ReCoRD (F1)	RTE	WiC	WSC	
FULL [20]	77.7	94.6	69.0	70.5	70.6	70.4	74.9	68.3	74.50
Prompt Tuning [20]	67.2	80.4	55.0	59.6	44.2	53.5	63.0	64.4	60.91
P-Tuning v2 [22]	73.1	**94.6**	73.0	**70.6**	**72.8**	78.3	**75.1**	**68.3**	**75.73**
OURS	**74.4**	80.4	**77.0**	65.8	71.9	**78.7**	74.3	67.3	73.73

Table S22. **Recall@K metric for 3 FGVC tasks** (*i.e.*, FGVC CUB-200-2011 [31], Oxford Flowers [26] and Stanford Dogs [17]) for 128-dimensional embeddings. The highest recall is shown in **bold**. See §S4.2.

Method	CUB-200-2011 [31]				Oxford Flowers [26]				Stanford Dogs [17]			
	1	2	4	8	1	2	4	8	1	2	4	8
VPT [16]	71.8	80.2	89.5	95.8	98.7	98.9	99.2	99.3	79.4	86.1	91.2	95.3
OURS	74.7	82.8	89.2	95.8	99.0	99.1	99.4	99.5	86.0	90.1	92.8	95.3