1. Summary of Datasets

In our main text, we evaluate the proposed method for few-shot learning tasks on 11 benchmark datasets and domain generalization tasks on ImageNet to its variants (ImageNet-V2, -Sketch, -A and -R). We summarize the dataset information in Table 1.

Specifically, the datasets for the few-shot evaluation are composed of diverse genres, such as recognition of generic objects, fine-grained objects, scenes, textual images and satellite images. The diversity can better verify the effectiveness and robustness of the proposed method. To be consistent with previous works [7,27,28], the “BACKGROUND Google” and “Faces easy” classes are removed in Caltech101 [6]. We also list the used templates [27] for the text-based classifier construction based on the CLIP’s [22] text branch.

For the generalization datasets, ImageNet-V2 and ImageNet-Sketch share the same label space with ImageNet (1000 classes), while the label spaces of ImageNet-A (200 classes) and ImageNet-R (200 classes) are both sub-spaces of the ImageNet label space. The variants of ImageNet contain substantially different data distributions (See Table 1 Description) from ImageNet, which makes them satisfactory domain generalization benchmarks. Following CoOp [28], we choose ImageNet as the source domain data while the variants as the target one.

2. More Experimental Results and Analyses

2.1. Few-Shot Learning

The full numerical results of Figure 3 in the main text are presented in Table 2. Note that the results of Tip-Adapter-F [27] are slightly different from their original paper. The original Tip-Adapter-F tests their models per epoch of training and chooses the best one to report the performance, while other methods such as CoOp test model until the training is done. To make the comparison fair, we re-run the official code of Tip-Adapter-F and test its models at the end of training. Overall, our method achieves the best averaged performance across all shot settings and datasets. In particular, our method reaches the best performance on ImageNet, Caltech101 and StanfordCars for all shot settings and on Flowers102, FGVC Aircraft, SUN397, DTD and EuroSAT for most shot settings. Additionally, despite having limited tunable parameters, the proposed method can always benefit from the expansion of training data, i.e., from 1-shot to 16-shot with the averaged gains from 5.08% to 16.14%. In contrast, Tip-Adapter-F [27] achieves similar performance by linearly increasing the tunable parameters with the number of shots.

2.2. Training Efficiency

Our TaskRes is not only parameter- and data-efficient but also highly efficient in training. As illustrated in Figure 2 in the main text, the high training efficiency is attributed to the absence of additional network modules (as in adapter-style tuning [7]) and the elimination of the need to run the text encoder every time (as in prompt tuning [28]). In particular, the quantitative results show that TaskRes needs merely 11 minutes, much less than 121 minutes used in prompt tuning and 16 minutes in adapter-style tuning, when training models on 4-shot ImageNet with a single GeForce RTX 3090 GPU.

2.3. Ablation Study

Ablation study of TaskRes effectiveness. We present the full results of the ablation study of our TaskRes effectiveness across 11 benchmark datasets in Table 3. Our TaskRes achieves notable improvements over both the regular and enhanced base classifiers across almost all the datasets. When equipping the regular base classifier with our proposed TaskRes, the accuracy of the model is improved by 5.08%, 8.06%, 10.65%, 13.80% and 16.14% for 1-, 2-, 4-, 8- and 16-shot settings, respectively. For the model based on the enhanced base classifier, our method still brings accuracy gains of 3.17%, 4.73%, 5.84%, 3.44% and 2.69% for the above settings, respectively. However, as mentioned in
our method is not very sensitive to scaling factor. Generally, setting \( \alpha \) to 0.5 achieves good performance. However, attaining the most favorable result.

Table 1. Summary of 11 datasets for few-shot learning and 4 target datasets of domain generalization. The 7 selected templates [27] for ImageNet series datasets are “a tip of the [class],” “a bad photo of the [class],” “an origami [class],” “a photo of the large [class],” “[a class] in a video game,” “art of the [class],” and “a photo of the small [class].”

Table 2. Full numerical results of performance comparison on few-shot learning.

Ablation study of scaling factor. We show the full comparison results across 11 datasets in Table 4. Generally, our method is not very sensitive to scaling factor \( \alpha \) when \( \alpha \in [0.3, 1] \), and our TaskRes with even \( \alpha = 0.1 \) can also be a strong performance booster (2.90% accuracy gain). On average, setting \( \alpha \) to 0.5 achieves good performance. However, the best scaling factor \( \alpha \) for various datasets can be different. For instance, a larger \( \alpha \) performs better on Flower102 and EuroSAT, while a smaller one is better for OxfordPets and Food101. We then use a learnable parameter (incorporating a tanh activation) to adaptively determine the value of \( \alpha \). On average, the learned \( \alpha \) attains the most favorable result.

Among the limitations (in main text), we observe a negative transfer on OxfordPets and Food101, similar to CoOp [28]. This negative transfer gap decreases with the number of shots increasing, which suggests that for these two datasets, learning the task-specific information is more difficult than other datasets, so more shots are needed.
More visualization results. We show more results (1-/2-/4-/8-shot settings) of the correlation of learned task residual magnitude and relative transfer difficulty in Figure 1, and the relation between the learned task residual magnitude and the number of shots in Table 5. We have the following observations:

- With more shots, task-specific knowledge can be captured with less variance as the shadows of the lines are shrinking.

2.4. Learned Task Residual

More visualization results. We show more results (1-/2-/4-/8-shot settings) of the correlation of learned task residual magnitude and relative transfer difficulty in Figure 1, and the relation between the learned task residual magnitude and the number of shots in Table 5. We have the following observations:

- The magnitudes of the learned task residuals are positively correlated to the relative transfer difficulty of CLIP for all shot settings, as shown in Figure 1 in this appendix and Figure 4 in the main text. This shows that the proposed task residual can effectively “supplement” the old knowledge according to the task difficulty.

Table 3. Full numerical results of ablation study of our TaskRes effectiveness.

| Setting | Method | ImageNet | Caltech101 | OxfordPets | StanfordCars | Flowers102 | Food101 | FGVC Aircraft | SUN397 | DTD | EuroSAT | UCF101 | Average |
|---------|--------|----------|------------|------------|--------------|------------|---------|---------------|--------|-----|--------|--------|---------|
| 1-shot  | Regular Base | 60.33 | 86.29 | 85.77 | 55.61 | 66.14 | 77.31 | 17.28 | 58.52 | 42.32 | 37.56 | 61.46 | 58.96 |
|         | Regular Base + TaskRes | 61.43 | 88.80 | 83.50 | 58.77 | 78.77 | 74.03 | 21.20 | 61.93 | 59.17 | 61.27 | 64.57 | 64.04 |
|         | Enhanced Base | 61.53 | 88.00 | 86.17 | 57.70 | 66.73 | 77.30 | 19.10 | 62.23 | 43.80 | 44.37 | 65.23 | 61.11 |
|         | Enhanced Base + TaskRes | 61.90 | 88.80 | 83.60 | 59.13 | 79.17 | 74.03 | 21.40 | 62.33 | 50.20 | 61.70 | 64.77 | 64.28 |
| 2-shot  | Regular Base | 60.33 | 86.29 | 85.77 | 55.61 | 66.14 | 77.31 | 17.28 | 58.52 | 42.32 | 37.56 | 61.46 | 58.96 |
|         | Regular Base + TaskRes | 62.17 | 90.13 | 84.43 | 62.77 | 85.63 | 75.30 | 23.07 | 64.33 | 54.53 | 65.77 | 69.10 | 67.02 |
|         | Enhanced Base | 61.87 | 89.37 | 86.93 | 59.75 | 68.23 | 77.53 | 19.87 | 63.83 | 46.53 | 49.5 | 67.63 | 62.82 |
|         | Enhanced Base + TaskRes | 62.63 | 90.27 | 84.63 | 63.70 | 86.57 | 75.17 | 24.13 | 64.97 | 55.13 | 65.83 | 70.00 | 67.55 |
| 4-shot  | Regular Base | 60.33 | 86.29 | 85.77 | 55.61 | 66.14 | 77.31 | 17.28 | 58.52 | 42.32 | 37.56 | 61.46 | 58.96 |
|         | Regular Base + TaskRes | 62.93 | 90.63 | 86.27 | 66.50 | 89.50 | 76.23 | 24.83 | 66.67 | 59.50 | 72.97 | 69.70 | 69.61 |
|         | Enhanced Base | 62.43 | 90.13 | 87.47 | 61.87 | 73.03 | 77.97 | 20.93 | 65.80 | 49.80 | 49.43 | 69.80 | 64.44 |
|         | Enhanced Base + TaskRes | 63.57 | 90.97 | 86.33 | 67.43 | 90.20 | 76.10 | 25.70 | 67.27 | 60.70 | 73.83 | 70.93 | 70.28 |
| 8-shot  | Regular Base | 60.33 | 86.29 | 85.77 | 55.61 | 66.14 | 77.31 | 17.28 | 58.52 | 42.32 | 37.56 | 61.46 | 58.96 |
|         | Regular Base + TaskRes | 64.03 | 92.23 | 87.07 | 70.57 | 94.30 | 76.90 | 29.50 | 68.70 | 64.23 | 78.07 | 74.77 | 72.76 |
|         | Enhanced Base | 63.33 | 91.60 | 88.07 | 66.73 | 87.67 | 78.23 | 23.67 | 68.07 | 59.73 | 67.63 | 74.27 | 69.91 |
|         | Enhanced Base + TaskRes | 64.67 | 92.40 | 87.17 | 71.83 | 94.73 | 76.40 | 31.50 | 68.73 | 64.77 | 79.33 | 75.33 | 73.35 |
| 16-shot | Regular Base | 60.33 | 86.29 | 85.77 | 55.61 | 66.14 | 77.31 | 17.28 | 58.52 | 42.32 | 37.56 | 61.46 | 58.96 |
|         | Regular Base + TaskRes | 64.75 | 92.00 | 88.10 | 74.93 | 96.10 | 78.23 | 33.73 | 70.30 | 67.57 | 82.57 | 76.87 | 75.10 |
|         | Enhanced Base | 64.13 | 92.57 | 89.07 | 71.67 | 92.00 | 78.70 | 27.20 | 70.27 | 64.13 | 76.83 | 77.37 | 73.09 |
|         | Enhanced Base + TaskRes | 65.73 | 93.43 | 87.83 | 76.83 | 96.03 | 77.60 | 36.30 | 70.67 | 67.13 | 84.03 | 77.97 | 75.78 |

Table 4. Full numerical results of ablation study of scaling factor $\alpha$ on 1-shot ImageNet.

| Setting | 1-shot | 2-shot | 4-shot | 8-shot | 16-shot |
|---------|--------|--------|--------|--------|--------|
| Mean    | 0.0124 | 0.0130 | 0.0118 | 0.0200 | 0.0232 |
| Median  | 0.0474 | 0.0493 | 0.0519 | 0.0672 | 0.0638 |

Table 5. Mean and Median of learned task residual magnitudes across 11 datasets.

Does TaskRes effectively preserve the pre-trained boundary? To gain deeper insights to the proposed TaskRes, we compare CoOp [28], CLIP-Adapter [7] and TaskRes regarding the number of “Wrong2Right” (W2R) images (i.e., those initially misclassified but later corrected) and “Right2Wrong” (R2W) images (i.e., those initially correctly classified but later misclassified). The models are trained on 4-shot ImageNet and tested on the complete 50k ImageNet test images. The W2R/R2W results for the three methods are as follows: (CoOp) 4161/4599, (CLIP-Adapter) 3542/2925, and (TaskRes) 3037/1702. This demonstrates that our TaskRes approach is more effective at preserving the pre-trained decision boundaries compared to other methods. Furthermore, we investigate the commonness of the W2R and R2W images and find that these images tend to occur in...
the visual concepts sharing the similar high-level semantics, e.g., upright piano and grand piano.

3. Discussion

Relationship between CLIP-Adapter and our TaskRes.
To make the comparison between CLIP-Adapter [7] and TaskRes clearer, we here focus on CLIP-Adapter performing on the text branch of CLIP. Given the pre-trained text embeddings $t$ (i.e., the text-based classifier), CLIP-Adapter first uses two linear layers $W_1$ and $W_2$ (incorporating a ReLU activation) to transform $t$, and then adds the transformed features to the original embeddings $t$ to obtain a new classifier. The transformation process (or adapter) can be written as

$$\phi(t) = \text{ReLU}(t^T W_1) W_2.$$ (1)

We can observe that the transformation in CLIP-Adapter has no additive bias, which makes the task-specific learning completely dependent on the old features. In contrast, TaskRes introduces a learnable bias $x$ (i.e., task residual) that is not relied on the old features (Eq. 3 in the main text). This allows for more flexibility in learning task-specific knowledge, leading to better performance.

To further analyze, we extend CLIP-Adapter to two linear transformation versions: linear adapters with and without learnable bias. Experimental results on 4-shot ImageNet show that linear adapters, both with and without bias (Acc.: 60.93% and 60.90%, respectively), underperform the original nonlinear adapter (Acc.: 61.27%), while the nonlinear adapter is outperformed by our TaskRes (Acc.: 62.93%). This indicates that the key for success is not the use of linear or nonlinear adapters, but the utilization of the prior-independent learnable parameters, i.e., the learnable parameters decoupled from the pre-trained features.

Lastly, although TaskRes could theoretically be considered as a special case of general adapter-style tuning (with adapter $\phi_\omega$ parameterized by $\omega$), we believe that the more simplified design and the much stronger performance exhibit-
imated by TaskRes have the potential to inspire the community.

TaskRes versus Tip-Adapter(-F). Tip-Adapter [27], one of the state-of-the-art methods, has a training-free version (i.e., Tip-Adapter) and an enhanced version Tip-Adapter-F which requires training. Our TaskRes has the following differences from Tip-Adapter(-F). (i) Different perspectives: Tip-Adapter(-F) is designed to adjust the classification results (i.e., logits) produced by the pre-trained classifier via feature retrieval/matching in the training set, while TaskRes performs on the weights of the classifier by tuning a prior-independent parameters (i.e., task residual) added to the pre-trained classifier. Despite the various perspectives, Tip-Adapter(-F) and our TaskRes are theoretically complementary. (ii) Different scalability: The number of tunable parameters of Tip-Adapter-F linearly increases with shot number while ours does not increase, which makes TaskRes more scalable than Tip-Adapter-F. While (training-free) Tip-Adapter does not need tunable parameters, the inference of an image requires all training sample features. Besides, the performance of Tip-Adapter largely underperforms Tip-Adapter-F and TaskRes.

Difference between prompt tuning in GLIP and our TaskRes. The prompt tuning in GLIP [16] performs on the intermediate features $P^0$, which are the outputs of the text encoder and the inputs for subsequent neural networks (NNs) such as BERT layers [14]. During tuning, GLIP omits the text encoder, removing the need to run it at every training step, which is similar to our TaskRes. However, the $P^0$ is subsequently fed into the following NNs, and updating $P^0$ still requires running the NNs (both forward and backward) each time. As a result, GLIP’s prompt tuning tends to follow a prompt tuning style.

For which types of tasks does TaskRes yield greater improvements? Our TaskRes achieves more significant improvement on tasks where more specialized/expertise knowledge is needed, e.g., EuroSAT, DTD and Flowers102. With 1-shot data, TaskRes improves those tasks by 7.85% $\sim$ 23.71%. With 16-shot data, the improvements are enlarged to 25.25% $\sim$ 45.01%. This is because our TaskRes can effectively learn task-specific knowledge.

4. Broader Impact

In this work, we conduct experiments and perform analyses based on CLIP [22]. However, our proposed concept of learning additive residual weights for efficient transfer learning is generic and can be adopted to a wider range of vision-language models, such as ALIGN [13], Perceiver IO [12], Flamingo [1], and others. Furthermore, this concept can potentially be extended to tuning vision [5, 8, 18] or language [14, 17] models.

References

[1] Jean-Baptiste Alayrac, Jeff Donahue, Pauline Luc, Antoine Miech, Iain Barr, Yana Hasson, Karel Lenc, Arthur Mensch, Katie Millican, Malcolm Reynolds, et al. Flamingo: a visual language model for few-shot learning. arXiv preprint arXiv:2204.14198, 2022.

[2] Lukas Bossard, Matthieu Guillaumin, and Luc Van Gool. Food-101–mining discriminative components with random forests. In ECCV, pages 446–461. Springer, 2014.

[3] Mircea Cimpoi, Subhransu Maji, Iasonas Kokkinos, Sammy Mohamed, and Andrea Vedaldi. Describing textures in the wild. In CVPR, pages 3606–3613, 2014.

[4] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In CVPR, pages 248–255. Ieee, 2009.

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

[6] Li Fei-Fei, Rob Fergus, and Pietro Perona. Learning generative visual models from few training examples: An incremental bayesian approach tested on 101 object categories. In CVPRW, pages 178–178. Ieee, 2004.

[7] 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.

[8] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In CVPR, pages 770–778, 2016.

[9] Patrick Helber, Benjamin Bischke, Andreas Dengel, and Damian Borth. Eurosat: A novel dataset and deep learning benchmark for land use and land cover classification. IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing, 12(7):2217–2226, 2019.

[10] Dan Hendrycks, Steven Basart, Norman Mu, Saurav Kada-vath, Frank Wang, Evan Dorundo, Rahul Desai, Tyler Zhu, Samyak Parajuli, Mike Guo, et al. The many faces of robustness: A critical analysis of out-of-distribution generalization. In ICCV, pages 8340–8349, 2021.

[11] Dan Hendrycks, Kevin Zhao, Steven Basart, Jacob Steinhardt, and Dawn Song. Natural adversarial examples. In CVPR, pages 15262–15271, 2021.

[12] Andrew Jaegle, Sebastian Borgeaud, Jean-Baptiste Alayrac, Carl Doersch, Catalin Ionescu, David Ding, Skanda Koppula, Daniel Zoran, Andrew Brock, Evan Shelhamer, Olivier J Henaff, Matthew Botvinick, Andrew Zisserman, Oriol Vinyals, and Joao Carreira. Perceiver IO: A general architecture for structured inputs & outputs. In ICLR, 2022.

[13] Chao Jia, Yinfei Yang, Ye Xia, Yi-Ting Chen, Zarana Parekh, Hieu Pham, Quoc Le, Yun-Hsuan Sung, Zhen Li, and Tom Duerig. Scaling up visual and vision-language representation learning with noisy text supervision. In ICML, pages 4904–4916. PMLR, 2021.
[14] Jacob Devlin Ming-Wei Chang Kenton and Lee Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. In NAACL-HLT, pages 4171–4186, 2019. 5
[15] Jonathan Krause, Michael Stark, Jia Deng, and Li Fei-Fei. 3d object representations for fine-grained categorization. In ICCVW, pages 554–561, 2013. 2
[16] Liunian Harold Li, Pengchuan Zhang, Haotian Zhang, Jianwei Yang, Chunyuan Li, Yiwu Zhong, Lijuan Wang, Lu Yuan, Lei Zhang, Jenq-Neng Hwang, et al. Grounded language-image pre-training. In CVPR, pages 10965–10975, 2022. 5
[17] Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. Roberta: A robustly optimized bert pretraining approach. arXiv preprint arXiv:1907.11692, 2019. 5
[18] 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 CVPR, pages 10012–10022, 2021. 5
[19] Subhransu Maji, Esa Rahtu, Juho Kannala, Matthew Blaschko, and Andrea Vedaldi. Fine-grained visual classification of aircraft. arXiv preprint arXiv:1306.5151, 2013. 2
[20] Maria-Elena Nilsback and Andrew Zisserman. Automated flower classification over a large number of classes. In ICVGIP, pages 722–729. IEEE, 2008. 2
[21] Omkar M Parkhi, Andrea Vedaldi, Andrew Zisserman, and CV Jawahar. Cats and dogs. In CVPR, pages 3498–3505. IEEE, 2012. 2
[22] 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, pages 8748–8763. PMLR, 2021. 1, 2, 5
[23] Benjamin Recht, Rebecca Roelofs, Ludwig Schmidt, and Vaishaal Shankar. Do imagenet classifiers generalize to imagenet? In ICML, pages 5389–5400. PMLR, 2019. 2
[24] Khurram Soomro, Amir Roshan Zamir, and Mubarak Shah. Ucf101: A dataset of 101 human actions classes from videos in the wild. arXiv preprint arXiv:1212.0402, 2012. 2
[25] Haohan Wang, Songwei Ge, Zachary Lipton, and Eric P Xing. Learning robust global representations by penalizing local predictive power. NeurIPS, 32, 2019. 2
[26] Jianxiong Xiao, James Hays, Krista A Ehinger, Aude Oliva, and Antonio Torralba. Sun database: Large-scale scene recognition from abbey to zoo. In 2010 IEEE computer society conference on computer vision and pattern recognition, pages 3485–3492. IEEE, 2010. 2
[27] Renrui Zhang, Zhang Wei, Rongyao Fang, Peng Gao, Kunchang Li, Jifeng Dai, Yu Qiao, and Hongsheng Li. Tip-adapter: Training-free adaption of clip for few-shot classification. In ECCV, 2022. 1, 2, 5
[28] Kaiyang Zhou, Jingkang Yang, Chen Change Loy, and Ziwei Liu. Learning to prompt for vision-language models. IJCV, 130(9):2337–2348, 2022. 1, 2, 3