Task Residual for Tuning Vision-Language Models (Appendix)

Tao Yu[1,2*] Zhihe Lu[1*] Xin Jin[1,2] Zhibo Chen[2] Xinchao Wang[1†]
[1]National University of Singapore [2]University of Science and Technology of China
yutao666@mail.ustc.edu.cn, zhihelu@nus.edu.sg, jinxustc@mail.ustc.edu.cn,
chenzhibo@ustc.edu.cn, xinchao@nus.edu.sg

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, textural 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, FGVCAircraft, 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

*Equal contribution.
†Corresponding author.

Name	Number of Classes	Size (Train / Val / Test)	Description	Template
ImageNet [4]	1000	1.28M / - /50000	Recognition of generic objects	Ensemble of 7 selected templates
Caltech101 [6]	100	4128 / 1649 / 2465	Recognition of generic objects	"a photo of a [class]."
OxfordPets [21]	37	2944 / 736 / 3669	Fine-grained classification of pets	"a photo of a [class], a type of pet."
StanfordCars [15]	196	6509 / 1635 / 8041	Fine-grained classification of cars	"a photo of a [class]."
Flowers102 [20]	102	4093 / 1633 / 2463	Fine-grained classification of flowers	"a photo of a [class], a type of flower."
Food101 [2]	101	50500 / 20200 / 30300	Fine-grained classification of foods	"a photo of a [class], a type of food."
FGVCAircraft [19]	100	3334 / 3333 / 3333	Fine-grained classification of aircrafts	"a photo of a [class], a type of aircraft."
SUN397 [26]	397	15880 / 3970 / 19850	Scene classification	"a photo of a [class]."
DTD [3]	47	2820 / 1128 / 1692	Texture classification	"[class] texture."
EuroSAT [9]	10	13500 / 5400 / 8100	Land use & cover classification with satellite images	"a centered satellite photo of [class]."
UCF101 [24]	101	7639 / 1898 / 3783	Action recognition	"a photo of a person doing [class]."
ImageNet-V2 [23]	1000	- / - / 10000	New test data for ImageNet	Ensemble of 7 selected templates
ImageNet-Sketch [25]	1000	- / - / 50889	Sketch-style images of ImageNet classes	Ensemble of 7 selected templates
ImageNet-A [11]	200	- / - / 7500	Natural adversarial examples of 200 ImageNet classes	Ensemble of 7 selected templates
ImageNet-R [10]	200	- / - / 30000	Renditions of 200 ImageNet classes	Ensemble of 7 selected templates

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 "itap of a [class].", "a bad photo of the [class].", "a 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].".

Method	Setting	ImageNet	Caltech101	OxfordPets	StanfordCars	Flowers102	Food101	FGVCAircraft	SUN397	DTD	EuroSAT	UCF101	Average
Zero-Shot CLIP [22]	1-shot	58.18	86.29	85.77	55.61	66.14	**77.31**	17.28	58.52	42.32	37.56	61.46	58.77
CoOp [28]		57.15	87.53	85.89	55.59	68.12	74.32	9.64	60.29	44.39	50.63	61.92	59.59
CLIP-Adapter [7]		61.20	88.60	85.99	55.13	73.49	76.82	17.49	61.30	45.80	61.40	62.20	62.67
Tip-Adapter-F [27]		60.88	**88.80**	**86.04**	56.78	**81.17**	76.22	19.01	61.23	**50.49**	50.34	**66.19**	63.38
Ours		61.43	**88.80**	83.50	58.77	78.77	74.03	21.20	61.93	50.17	61.27	64.57	64.04
Ours*		**61.90**	**88.80**	83.60	**59.13**	79.17	74.03	**21.40**	**62.33**	50.20	**61.70**	64.77	**64.28**
Zero-Shot CLIP [22]	2-shot	58.18	86.29	85.77	55.61	66.14	**77.31**	17.28	58.52	42.32	37.56	61.46	58.77
CoOp [28]		57.81	87.93	82.64	58.28	77.51	72.49	18.68	59.48	45.15	61.50	64.09	62.32
CLIP-Adapter [7]		61.52	89.37	**86.73**	58.74	81.61	77.22	20.10	63.29	51.48	63.90	67.12	65.55
Tip-Adapter-F [27]		61.57	89.61	86.06	61.13	85.40	77.05	21.76	63.19	**55.32**	64.76	68.99	66.80
Ours		62.17	90.13	84.43	62.77	85.63	75.30	23.07	64.33	54.53	65.77	69.10	67.02
Ours*		**62.63**	**90.27**	84.63	**63.70**	**86.57**	75.17	**24.13**	**64.97**	55.13	**65.83**	**70.00**	**67.55**
Zero-Shot CLIP [22]	4-shot	58.18	86.29	85.77	55.61	66.14	77.31	17.28	58.52	42.32	37.56	61.46	58.77
CoOp [28]		59.99	89.55	86.70	62.62	86.20	73.33	21.87	63.47	53.49	70.18	67.03	66.77
CLIP-Adapter [7]		61.84	89.98	**87.46**	62.45	87.17	**77.92**	22.59	65.96	56.86	73.38	69.05	68.61
Tip-Adapter-F [27]		62.62	90.87	86.46	64.86	89.53	77.46	**26.39**	65.88	60.25	69.66	**72.71**	69.70
Ours		62.93	90.63	86.27	66.50	89.50	76.23	24.83	66.67	59.50	72.97	69.70	69.61
Ours*		**63.57**	**90.97**	86.33	**67.43**	**90.20**	76.10	25.70	**67.27**	**60.70**	**73.83**	70.93	**70.28**
Zero-Shot CLIP [22]	8-shot	58.18	86.29	85.77	55.61	66.14	77.31	17.28	58.52	42.32	37.56	61.46	58.77
CoOp [28]		61.56	90.21	85.32	68.43	91.18	71.82	26.13	65.52	59.97	76.73	71.94	69.89
CLIP-Adapter [7]		62.68	91.40	87.65	67.89	91.72	**78.04**	26.25	67.50	61.00	77.93	73.30	71.40
Tip-Adapter-F [27]		64.15	91.70	**88.28**	69.51	91.00	77.90	30.62	**69.23**	62.93	**79.33**	74.76	72.67
Ours		64.03	92.23	87.07	70.57	94.30	76.90	29.50	68.70	64.23	78.07	74.77	72.76
Ours*		**64.67**	**92.40**	87.17	**71.83**	**94.73**	76.40	**31.50**	68.73	**64.77**	**79.33**	**75.33**	**73.35**
Zero-Shot CLIP [22]	16-shot	58.18	86.29	85.77	55.61	66.14	77.31	17.28	58.52	42.32	37.56	61.46	58.77
CoOp [28]		62.95	91.83	87.01	73.36	94.51	74.67	31.26	69.26	63.58	83.53	75.71	73.42
CLIP-Adapter [7]		63.59	92.49	87.84	74.01	93.90	**78.25**	32.10	69.55	65.96	84.43	76.76	74.44
Tip-Adapter-F [27]		65.44	92.63	**88.18**	75.75	94.23	78.11	35.86	**71.00**	66.94	**84.94**	**79.03**	75.65
Ours		64.75	92.90	88.10	74.93	**96.10**	78.23	33.73	70.30	**67.57**	82.57	76.87	75.10
Ours*		**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 2. Full numerical results of performance comparison on few-shot learning.

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.

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 α 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 α to 0.5 achieves good performance. However, the best scaling factor α for various datasets can be different. For instance, a larger α 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 α. On average, the learned α attains the most favorable result.

Setting	Method	ImageNet	Caltech101	OxfordPets	StanfordCars	Flowers102	Food101	FGVCAircraft	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**	**50.17**	**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	90.33	**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**	**92.90**	**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 3. Full numerical results of ablation study of our TaskRes effectiveness.

α	ImageNet	Caltech101	OxfordPets	StanfordCars	Flowers102	Food101	FGVCAircraft	SUN397	DTD	EuroSAT	UCF101	Average
0	60.33	86.29	**85.77**	55.61	66.14	**77.31**	17.28	58.52	42.32	37.56	61.46	58.96
0.1	60.77	87.43	84.93	59.40	70.27	75.60	19.20	59.63	46.53	54.47	62.27	61.86
0.3	61.37	88.63	84.27	**59.83**	75.47	74.73	20.80	61.83	49.83	60.07	64.53	63.76
0.5	**61.43**	**88.80**	83.50	58.77	78.00	74.03	21.20	**61.93**	**50.17**	61.27	**64.57**	**63.97**
0.7	**61.43**	88.70	82.80	57.80	**78.90**	73.17	**21.23**	61.37	49.57	61.47	63.93	63.67
1	61.23	88.53	81.60	56.23	78.77	71.67	20.83	60.50	49.03	**61.77**	63.07	63.02
Learned	61.33	88.73	84.00	59.47	77.40	74.40	20.63	**61.93**	49.90	60.43	**65.93**	**64.01**

Table 4. Full numerical results of ablation study of scaling factor α 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.

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.

- With shot increasing, the mean and median of the learned task residual magnitudes across 11 datasets tend to increase, which indicates that when more downstream task samples are used, more task-specific knowledge can be explored in our method.

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

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

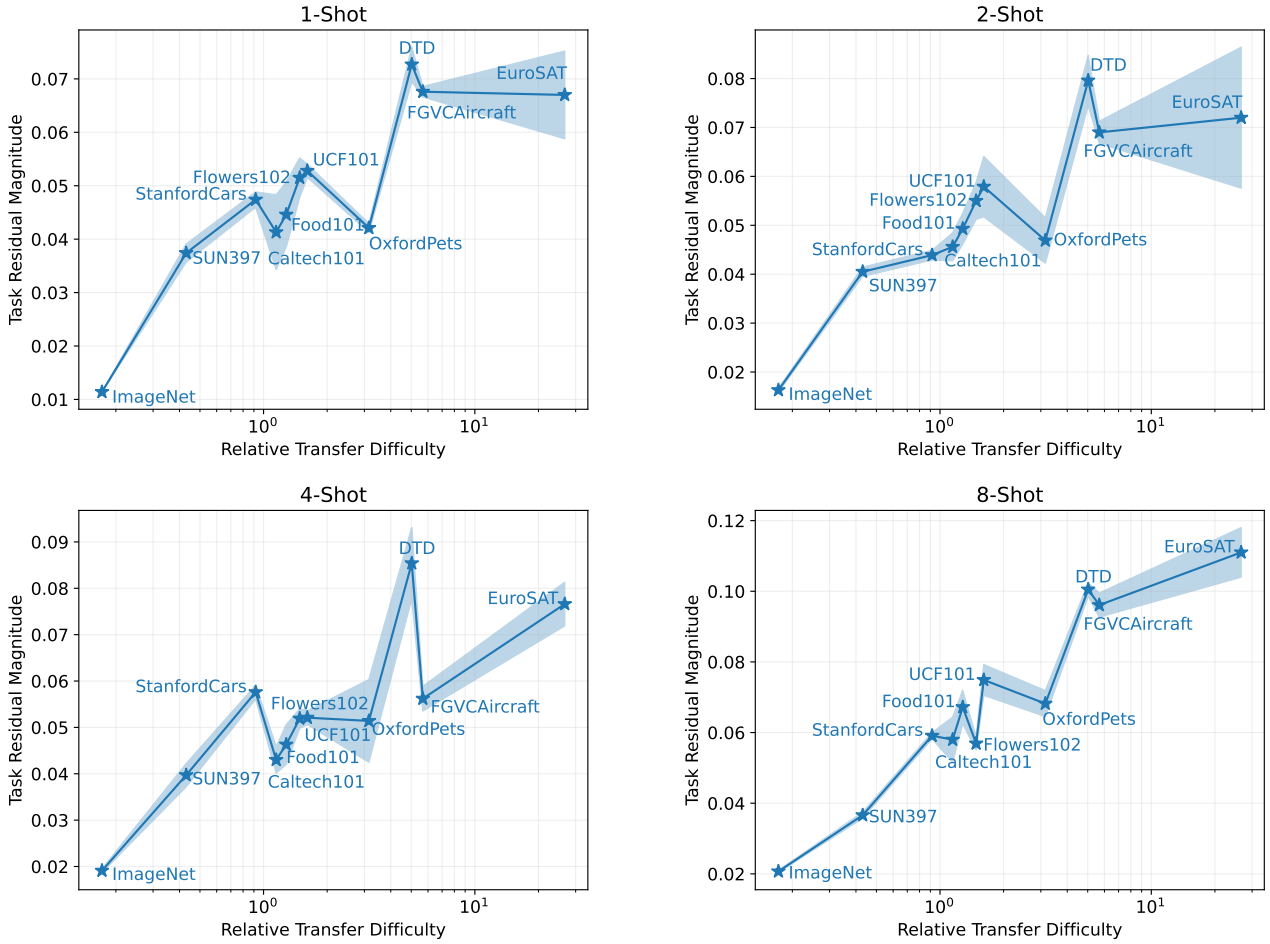

Figure 1. Relation between the magnitude of learned task residuals and the relative transfer difficulty regarding CLIP with 1-/2-/4-/8-shot settings (the 16-shot result is in the main text). The shadow indicates the standard deviation regarding random seeds.

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 $\mathbf{t}$ (*i.e.*, the text-based classifier), CLIP-Adapter first uses two linear layers $\mathbf{W}_1$ and $\mathbf{W}_2$ (incorporating a ReLU activation) to transform $\mathbf{t}$, and then adds the transformed features to the original embeddings $\mathbf{t}$ to obtain a new classifier. The transformation process (or adapter) can be written as

$$\phi(\mathbf{t}) = \text{ReLU}(\mathbf{t}^T\mathbf{W}_1)\mathbf{W}_2. \quad (1)$$

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 $\mathbf{x}$ (*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 ϕ_ω parameterized by ω), we believe that the more simplified design and the much stronger performance exhib-

ited 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 addictive 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. 5

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

[3] Mircea Cimpoi, Subhransu Maji, Iasonas Kokkinos, Sammy Mohamed, and Andrea Vedaldi. Describing textures in the wild. In *CVPR*, pages 3606–3613, 2014. 2

[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. 2

[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. 5

[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. 1, 2

[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. 1, 2, 3, 4

[8] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *CVPR*, pages 770–778, 2016. 5

[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. 2

[10] Dan Hendrycks, Steven Basart, Norman Mu, Saurav Kadavath, Frank Wang, Evan Dorundo, Rahul Desai, Tyler Zhu, Samyak Parajuli, Mike Guo, et al. The many faces of robustness: A critical analysis of out-of-distribution generalization. In *ICCV*, pages 8340–8349, 2021. 2

[11] Dan Hendrycks, Kevin Zhao, Steven Basart, Jacob Steinhardt, and Dawn Song. Natural adversarial examples. In *CVPR*, pages 15262–15271, 2021. 2

[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. 5

[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. 5

[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 *ICCVGIP*, 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