

SingLoRA-CLIP: Highly Efficient VLM Adaptation with a Single Parameter Matrix

Nam Nhat Ngo Nguyen, Thu Anh Ngoc Le, Nhi Quynh Phan Le, Nghi Vinh Nguyen, and Hoang Ngoc Tran(✉)

FPT University, CanTho, Vietnam
{namnnnce181939,thulnace180615,nhilpqce180224, nghinvce182108}@fpt.edu.vn, hoang2531992@gmail.com

Abstract. Parameter-Efficient Fine-Tuning (PEFT) is a practical necessity for adapting large Vision-Language Models (VLMs). While Low-Rank Adaptation (LoRA) is a standard approach, its asymmetric two-matrix update (BA) can introduce initialization mismatches that may affect training stability. A recently proposed language-domain variant, SingLoRA, replaces this update with a single symmetric decomposition (AA^T), but its suitability for visual and multimodal adaptation has not yet been established. In this paper, we provide, to the best of our knowledge, the first evaluation of SingLoRA for few-shot adaptation of the CLIP architecture. We benchmark CLIP-SingLoRA on eight image classification datasets and compare it directly against a controlled CLIP-LoRA baseline under the same adaptation setting. Our results show that SingLoRA delivers accuracy that is competitive with standard LoRA while reducing the number of trainable parameters by 50%. These findings suggest that a single-matrix symmetric update can serve as a practical and highly parameter-efficient alternative for few-shot VLM adaptation.

Keywords: Parameter-Efficient Fine-Tuning · Low-Rank Adaptation · Vision-Language Models · Few-Shot Learning

1 Introduction

Vision-Language Models (VLMs) like CLIP [1] learn strong representations from massive image-text datasets [2,3]. Because they align visual and textual spaces through contrastive learning, these models perform very well in zero-shot scenarios. They can recognize new concepts without direct supervision, relying entirely on natural language prompts. Driven by this adaptability, researchers are increasingly interested in fine-tuning VLMs with very little data, a setup known as few-shot learning [4].

Modern VLMs contain billions of parameters. Full fine-tuning is not only computationally expensive but also prone to overfitting when training data

© The Author(s), under exclusive license to Springer Nature Switzerland AG 2027
O. Gervasi et al. (Eds.): ICCSA 2026, LNCS 16768, pp. 251–263, 2027.
https://doi.org/10.1007/978-3-032-30488-9_16

is limited. This challenge has led to Parameter-Efficient Fine-Tuning (PEFT) methods, which adapt large models by optimizing only a small subset of their parameters [5]. Within the VLM literature, research has predominantly focused on prompt-tuning strategies [4,6,7] and on integrating conventional adapter modules [8,9]. Recently, Low-Rank Adaptation (LoRA) [10] has become a standard PEFT technique. LoRA assumes that the weight changes needed to learn a new task have a low "intrinsic rank." Instead of updating the massive original weight matrix, it models the update (ΔW) by multiplying two much smaller matrices (B and A). Zanella and Ben Ayed [11] recently showed that simply applying LoRA to CLIP creates an extremely strong baseline. In fact, it often beats more complex prompt and adapter methods in few-shot tasks.

While empirically successful, standard LoRA introduces a structural flaw. During initialization, matrix A gets random values while matrix B is set to zero. This creates a scale imbalance that can make training unstable [12,13]. To mitigate this issue, Bensaïd et al. [12] introduced **SingLoRA**. Instead of two matrices, SingLoRA uses a single matrix and its transpose to form a symmetric update: $\Delta W = AA^T$. This simple change brings two major benefits: it naturally avoids the scaling conflict, and it cuts the number of trainable parameters exactly in half compared to standard LoRA.

Current literature on SingLoRA is restricted to unimodal language or image generation tasks [12]. Its viability for multimodal, few-shot adaptation remains untested. In particular, transferring SingLoRA to CLIP is not straightforward, because the adaptation must preserve the model's cross-modal alignment while operating in a highly data-scarce regime. In this work, we study whether the single-matrix formulation can remain competitive in that setting by adapting CLIP with SingLoRA and comparing it against standard LoRA and other leading PEFT methods.

Our main contributions are as follows:

- To the best of our knowledge, we provide the first evaluation of SingLoRA for few-shot image classification with a Vision-Language Model.
- We conduct a systematic benchmark on eight diverse datasets across 1-, 4-, and 16-shot settings, including a controlled comparison against a CLIP-LoRA baseline with the same adaptation targets.
- We show that SingLoRA attains competitive performance while using 50% fewer trainable parameters than CLIP-LoRA, making it a strong parameter-efficient alternative for multimodal fine-tuning.

2 Related Work

2.1 Parameter-Efficient Fine-Tuning (PEFT)

PEFT strategies aim to achieve competitive performance by updating only a small fraction of a model's parameters [5]. These techniques generally fall into a few categories. Selective methods, such as BitFit [14], update specific parameter subsets like biases. Adapter-based approaches insert lightweight trainable

modules into the frozen base model [15]. Prompt-tuning prepends learnable continuous vectors to the input [16,17]. Finally, low-rank adaptation methods model weight updates via low-dimensional matrices, offering a strong balance between performance and computational cost.

2.2 Few-Shot Adaptation of Vision-Language Models

Prompt-tuning is widely used for adapting VLMs with limited data. CoOp [4] introduced learnable context vectors to replace manual text prompts. Building on this, CoCoOp [6] made these prompts instance-conditional to improve generalization. MaPLe [7] extended this approach to the visual encoder for multimodal adaptation, while ProGrad [19] regularized prompt learning to prevent catastrophic forgetting of pretrained knowledge.

In parallel to prompt-tuning, various adapter-based methods have been explored. CLIP-Adapter [8] introduces a simple MLP after the encoders to refine features, while TaskRes [9] learns task-specific residual representations. Interestingly, recent findings from Zanella and Ben Ayed [11] show that directly applying LoRA establishes a baseline that often outperforms these specialized modules. This suggests that low-rank methods are highly effective and directly motivates our investigation into an even more parameter-efficient variant.

2.3 Low-Rank Adaptation and Its Variants

Low-Rank Adaptation (LoRA) [10] has become a foundational technique in PEFT. It operates on the principle that the change in model weights during adaptation, ΔW, has a low intrinsic rank. LoRA materializes this by freezing the pretrained weights W_0 and modeling the update as a trainable, low-rank decomposition $\Delta W = BA$, where the rank r is significantly smaller than the dimensions of W_0. A practical benefit of LoRA is its efficiency at inference time. Since the learned matrix ΔW can be directly added to the pre-trained weights W_0, the model runs without any extra computational overhead.

While LoRA is highly popular, its standard formulation contains an architectural bottleneck. Bensaïd et al. [12] pointed out that initializing matrix A randomly while setting matrix B to zero creates conflicting scales between the two matrices. This imbalance can lead to unstable training curves. As a solution, they proposed **SingLoRA**, a method that drops matrix B entirely. Instead, it relies on a single low-rank matrix A to form a symmetric weight update, defined as $\Delta W = AA^T$. This design avoids the scale imbalance problem and, as a direct benefit, halves the number of trainable parameters for a given rank r. Whether SingLoRA's benefits transfer to the few-shot VLM adaptation setup is the primary question driving this research.

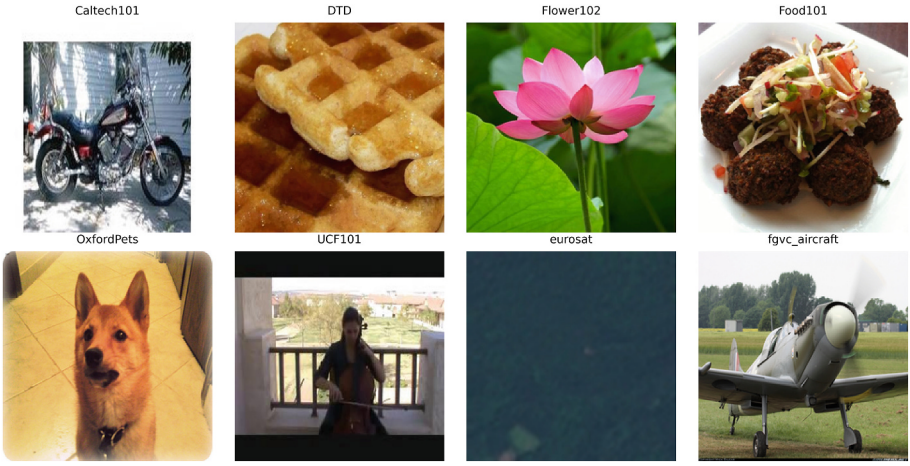

Fig. 1. Visual samples from the eight evaluation benchmarks.

3 Methodology

Instead of fine-tuning the entire pretrained CLIP model, we adopt a low-rank adaptation strategy with a small trainable budget. Below, we describe the transition from standard LoRA to the symmetric SingLoRA formulation, followed by its implementation in the CLIP architecture.

3.1 From Asymmetric to Symmetric Low-Rank Adaptation

Standard LoRA [10] modifies a pre-trained weight matrix $W_0 \in \mathbb{R}^{d\times d}$ by learning a low-rank update ΔW. The forward pass becomes $h = W_0 x + \frac{\alpha}{r} BAx$, where $A \in \mathbb{R}^{r\times d}$ and $B \in \mathbb{R}^{d\times r}$ are trainable matrices with a rank $r \ll d$. Because A is random and B is zero at initialization, their scales conflict early in training [12].

To bypass this, we adopt the SingLoRA mechanism [12]. We completely remove matrix B. Instead, the weight update is parameterized using only a single matrix $A \in \mathbb{R}^{d\times r}$ to create a symmetric projection:

$$W' = W_0 + u(t) \cdot \frac{\alpha}{\sqrt{r}} AA^T \tag{1}$$

Equation 1 makes the symmetry of the update explicit. This formulation guarantees matching scales between the projection dimensions right out of the box. Mathematically, dropping B forces a 50% reduction in the parameters required for any given rank r.

To ensure the pretrained VLM features are not abruptly distorted during the initial training steps, the update is scaled by a linear ramp-up function $u(t) = \min(1.0, \frac{t}{T})$, where t is the current step and T is the warm-up duration.

3.2 Applying SingLoRA to CLIP's Attention Layers

Following the successful strategy demonstrated in prior work [11], we apply our adaptation to the Multi-Head Attention (MHA) layers in CLIP's transformer-based encoders. In each MHA block, the adaptation targets the projection matrices for the query (W_q), key (W_k), and value (W_v).

For each of these target layers (e.g., $W_q \in \mathbb{R}^{d\times d}$), we freeze the original weight matrix and introduce a corresponding trainable SingLoRA matrix $A_q \in \mathbb{R}^{d\times r}$. The original forward pass $h_q = W_q x$ is then replaced by its adapted version:

$$h'_q = W_q x + u(t) \cdot \frac{\alpha}{\sqrt{r}} A_q A_q^T x \tag{2}$$

Equation 2 shows how the same symmetric update is instantiated inside the attention module. The same adaptation is applied independently to the key and value projection matrices. By focusing only on the attention weights, we modify the core mechanism of the model with a minimal parameter budget. This implementation directly results in a 50% reduction in trainable parameters compared to applying standard LoRA to the same W_q, W_k, W_v matrices with an identical rank r.

Table 1. Average Top-1 accuracy (%) across eight datasets (ViT-B/16 backbone). **Bold** indicates the best performance, while underlined values denote the second-best result.

Method	1-Shot	4-Shots	16-Shots
Zero-Shot CLIP [1]	65.25	65.25	65.25
CoOp (16) [4]	67.98	75.18	81.23
CoCoOp [6]	68.85	72.33	76.60
TIP-Adapter-F [20]	71.98	76.96	82.06
PLOT++ [21]	71.98	78.48	83.71
TaskRes [9]	71.54	75.23	81.96
ProGrad [19]	70.33	75.54	81.15
CLIP-LoRA (Baseline, r=2)	**73.26**	79.53	**84.52**
CLIP-SingLoRA (Ours, r=2)	<u>73.14</u>	**80.00**	<u>84.39</u>

4 Experiments

We performed a broad empirical evaluation to measure the effectiveness and parameter cost of SingLoRA for adapting Vision-Language Models in a few-shot setting.

4.1 Experimental Setup

To evaluate the generalization of our method across diverse visual domains, we conduct experiments on eight standard image classification benchmarks (visualized in Fig. 1). Rather than exhaustively describing each well-known dataset, we categorize them by their visual challenges:

- **Fine-grained object recognition:** FGVC-Aircraft [22], Food-101 [24], Oxford-Pets [25], and Oxford Flowers 102 [26].
- **General objects and textures:** Caltech-101 [27] and Describable Textures Dataset (DTD) [28].
- **Specialized domains:** EuroSAT [23] for satellite imagery and UCF101 [29] for action recognition. For UCF101, we follow standard practice [11] by extracting the middle frame of each video to form a static image classification task.

Evaluation Protocol. We strictly follow the few-shot learning protocol established by CoOp [4]. For each dataset, we randomly sample 1, 4, and 16 shots (images) per class to form the training sets. After fine-tuning, models are evaluated on the original, full test sets. We report Top-1 accuracy for each dataset, alongside the arithmetic mean across all eight datasets to provide an overall view of model performance.

Baselines. We benchmark CLIP-SingLoRA against the frozen Zero-Shot CLIP [1] and several leading PEFT methods. The prompt-tuning baselines are CoOp [4], CoCoOp [6], KgCoOp [18], MaPLe [7], and ProGrad [19]. The adapter-based baselines include CLIP-Adapter [8] and TIP-Adapter-F [20]. We also compare against TaskRes [9] and PLOT++ [21]. For the fairest parameter-efficiency comparison, we also implement a **CLIP-LoRA baseline** [11]. This baseline mirrors our SingLoRA setup exactly. The only difference is that it uses the standard two-matrix update (BA) instead of the symmetric update (AA^T).

Implementation Details. All experiments are built upon the pre-trained CLIP model with a ViT-B/16 visual backbone. For both LoRA and SingLoRA, we inject the low-rank matrices into the query, key, and value projection layers (W_q, W_k, W_v) across all attention blocks in both the vision and text encoders.

To avoid the massive computational cost of grid-searching hyperparameters for every individual dataset, we adopt a single, unified configuration. During our preliminary runs, we noticed that using a scaling factor larger than 1 often destabilized training, likely because the symmetric update overly distorted the pre-trained semantic space. Consequently, we fix the intrinsic rank $r = 2$, the scaling factor $\alpha = 1$, and use a linear warm-up period of $T = 100$ steps for all tasks.

5 Results and Analysis

5.1 Main Results

Our main experimental results for few-shot classification are summarized in Table 1, which reports the average performance across the eight benchmarks. We provide a complete per-dataset breakdown of these figures in the Appendix

Table 2. Ablation of rank (r) and scaling factor (α).

Setting	Method	Config	Avg. Accuracy (%)	Parameters
1-Shot	CLIP-LoRA	(r=2)	73.26	184,320
	CLIP-SingLoRA	(r=2, α=1)	73.14	**92,160**
		(r=4, α=1)	**73.78**	184,320
		(r=4, α=2)	73.45	184,320
		(r=4, α=8)	70.37	184,320
		(r=4, α=16)	68.33	184,320
4-Shots	CLIP-LoRA	(r=2)	79.53	184,320
	CLIP-SingLoRA	(r=2, α=1)	**80.00**	**92,160**
		(r=4, α=1)	79.50	184,320
		(r=4, α=2)	79.07	184,320
		(r=4, α=8)	77.76	184,320

(Table 4). Our main approach, **CLIP-SingLoRA (r=2)**, is benchmarked against the strong CLIP-LoRA baseline.

As reported in Table 1, CLIP-SingLoRA ($r = 2$) remains highly competitive with the CLIP-LoRA baseline across all three shot settings. Most notably, it attains the highest average accuracy in the 4-shot setting (80.00%). In the 1-shot and 16-shot settings, its results remain very close to those of LoRA. Figure 2 shows that this competitive behavior is maintained as the number of shots increases. The key takeaway from these results is the parameter cost: SingLoRA achieves this competitive accuracy while using exactly half the trainable parameters of standard LoRA (as detailed in Table 3). Taken together, these results suggest that a single symmetric projection (AA^T) can be sufficient for effective VLM adaptation in the few-shot regime.

Table 3. Parameter efficiency comparison. Average accuracy from the 4-shot setting is shown for performance reference.

Method	Trainable Parameters	Avg. Accuracy (4-Shots)
CLIP-LoRA (r=2)	184,320	79.53%
CLIP-SingLoRA (r=2)	**92,160 (-50%)**	**80.00%**

5.2 Analysis of Parameter Efficiency

Figure 3 illustrates the trade-off between performance and parameter count. In the 4-shot setting, CLIP-SingLoRA reaches a slightly higher average accuracy

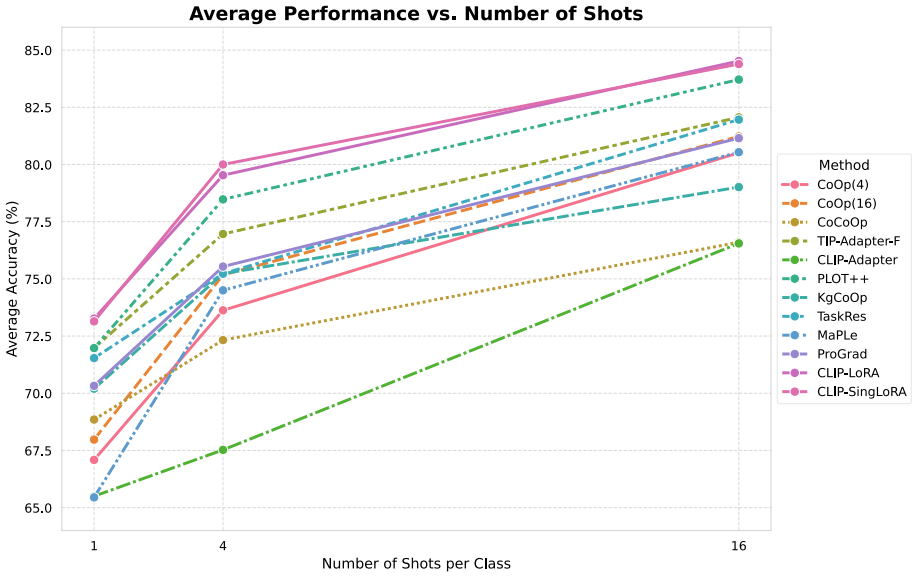

Fig. 2. Average accuracy as a function of the number of shots per class.

(80.00%) than the CLIP-LoRA baseline (79.53%). As detailed in Table 3, this result is obtained using only half the trainable parameters (92,160 vs. 184,320). This comparison indicates that reformulating the weight update as a symmetric matrix can substantially reduce the adaptation budget while preserving competitive performance.

5.3 Ablation Study: Impact of Rank and Scaling Factor

We ablate two key hyper-parameters governing our adaptation module: the intrinsic rank r and the scaling factor α. Table 2 reports the performance variations averaged across all benchmarks. A complete per-dataset breakdown is deferred to Appendix Table 5.

Impact of Rank and Parameter Budget. Table 2 details the impact of the intrinsic rank r. Increasing the rank from $r = 2$ to $r = 4$ improves accuracy in the 1-shot setting (from 73.14% to 73.78%). However, in the 4-shot setting, the lower rank $r = 2$ yields better results (80.00% vs 79.50%). This implies that a lower rank may provide beneficial regularization when more training data is available.

Crucially, configuring SingLoRA with $r = 4$ matches the exact parameter budget of LoRA with $r = 2$ (184,320 parameters). Under this identical budget constraint, SingLoRA outperforms standard LoRA in the 1-shot scenario and

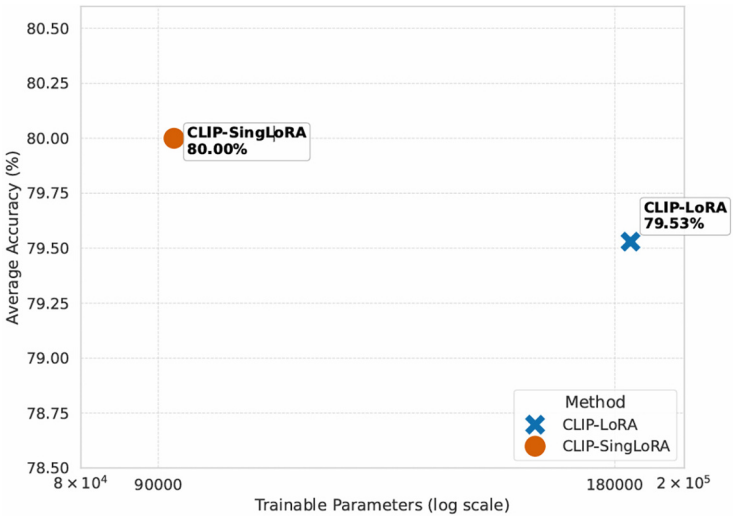

Fig. 3. Performance versus parameter cost in the 4-shot setting.

performs similarly in the 4-shot setting. This indicates that the single-matrix update AA^T remains competitive even when the parameter budget is held constant.

Sensitivity to Scaling Factor α. As shown in Table 2, performance consistently drops as α increases across both few-shot settings. Because the update term ΔW is directly added to the pre-trained weights, a high multiplier can distort the VLM's original semantic space. In our experiments, $\alpha = 1$ provided the most reliable overall behavior.

5.4 Discussion and Limitations

The empirical picture that emerges from our results is consistent across the main benchmark and the ablation study: SingLoRA can remain competitive with standard LoRA even under a substantially smaller adaptation budget. This is particularly relevant in few-shot VLM adaptation, where the practical objective is often to preserve most of the pretrained model while making only a lightweight task-specific adjustment. From this perspective, our findings suggest that the symmetric single-matrix parameterization is a viable alternative when memory efficiency is a primary concern.

At the same time, our study has several limitations. First, the evaluation is restricted to few-shot image classification with CLIP ViT-B/16, so the conclusions should not yet be generalized to other VLM architectures or multimodal tasks. Second, our efficiency analysis focuses on trainable parameter counts

rather than wall-clock time or peak GPU memory. Future work should therefore examine broader architectures, additional multimodal tasks, and more detailed efficiency measurements.

6 Conclusion

In this paper, we evaluated the feasibility and efficiency of SingLoRA for adapting large Vision-Language Models in data-scarce environments. Previously limited to unimodal language generation, we tested SingLoRA on the multimodal CLIP architecture across eight distinct image classification benchmarks.

Our experimental results show that the traditional, asymmetric two-matrix formulation of LoRA (BA) is not the only effective option for few-shot vision-language adaptation. The single-matrix symmetric decomposition (AA^T) used in SingLoRA achieves accuracy that is consistently competitive with a carefully matched standard LoRA baseline, while strictly halving the trainable parameter budget.

By reducing the number of trainable parameters without materially affecting performance, SingLoRA offers a compelling performance-to-parameter trade-off. We therefore view it as a promising memory-efficient baseline for future research on the parameter-efficient adaptation of large-scale multimodal models.

A Detailed Per-Dataset Results

This appendix provides the detailed per-dataset results for all baseline methods and our proposed CLIP-SingLoRA across the 1, 4, and 16-shot settings. Table 4 presents the Top-1 accuracy (%) for each of the eight benchmark datasets, supplementing the averaged results discussed in the main paper. The best-performing method in each column (per shot setting) is highlighted in **bold**, and the second-best is underlined.

Table 4. Full few-shot classification results (%) on all eight datasets with a ViT-B/16 backbone.

Method	Shots	Aircraft	EuroSAT	Food	Pets	Flowers	Caltech	DTD	UCF	Average
Zero-Shot CLIP [1]	0	24.70	47.50	86.10	89.10	71.40	92.90	43.60	66.70	65.25
1	CoOp (16) [4]	20.80	56.40	84.30	90.20	78.30	92.50	50.10	71.20	67.98
	CoCoOp [6]	28.10	55.40	84.90	91.90	73.40	94.10	52.60	70.40	68.85
	TIP-Adapter-F [20]	28.80	67.80	85.80	90.60	**83.80**	94.00	51.60	73.40	71.98
	CLIP-Adapter [8]	25.20	49.30	86.10	89.00	71.30	92.00	44.20	66.90	65.50
	PLOT++ [21]	28.60	65.40	86.20	91.90	80.50	94.30	**54.60**	74.30	71.98
	KgCoOp [18]	26.80	61.90	86.40	92.10	74.70	94.20	52.70	72.80	70.20
	TaskRes [9]	**31.30**	65.40	84.60	90.20	81.70	93.60	53.80	71.70	71.54
	MaPLe [7]	28.10	29.10	85.40	91.40	74.90	93.60	50.00	71.10	65.45
	ProGrad [19]	28.80	57.00	84.90	91.40	80.90	93.50	52.80	73.30	70.33
	CLIP-LoRA	28.71	72.88	85.08	91.52	83.35	93.67	54.61	**76.29**	**73.26**
	CLIP-SingLoRA	29.10	**73.58**	**85.75**	**92.61**	81.40	93.87	53.37	75.47	73.14
4	CoOp (16) [4]	30.90	69.70	84.50	92.50	92.20	94.50	59.50	77.60	75.18
	CoCoOp [6]	30.60	61.70	86.30	92.70	81.50	94.80	55.70	75.30	72.33
	TIP-Adapter-F [20]	35.70	76.80	86.50	91.90	92.10	94.80	59.80	78.10	76.96
	CLIP-Adapter [8]	27.90	51.20	86.50	90.80	73.10	94.00	46.10	70.60	67.53
	PLOT++ [21]	35.30	83.20	86.50	92.60	92.90	95.10	62.40	79.80	78.48
	KgCoOp [18]	32.20	71.80	86.90	92.60	87.00	95.00	58.70	77.60	75.23
	TaskRes [9]	33.40	74.20	86.00	91.90	85.00	95.00	60.10	76.20	75.23
	MaPLe [7]	30.10	69.90	86.70	93.30	84.90	95.00	59.00	77.10	74.50
	ProGrad [19]	34.10	69.60	85.40	92.10	91.10	94.40	59.70	77.90	75.54
	CLIP-LoRA	**39.15**	87.49	83.72	91.06	93.67	**95.58**	65.19	80.39	79.53
	CLIP-SingLoRA	38.46	**89.53**	84.44	92.45	**93.75**	95.21	**65.01**	**81.15**	**80.00**
16	CoOp (16) [6]	43.20	85.00	84.20	92.00	96.80	95.80	69.70	83.10	81.23
	CoCoOp [6]	33.30	73.60	87.40	93.40	89.10	95.10	63.70	77.20	76.60
	TIP-Adapter-F [20]	44.60	85.90	86.80	92.60	96.20	95.70	70.80	83.90	82.06
	CLIP-Adapter [8]	34.20	71.40	87.10	92.30	92.90	94.90	59.40	80.20	76.55
	PLOT++ [21]	46.70	92.00	87.10	93.60	97.60	96.00	71.40	85.30	83.71
	KgCoOp [18]	36.50	76.20	87.20	93.20	93.40	95.20	68.70	81.70	79.01
	TaskRes [9]	44.90	82.70	86.90	92.40	97.50	95.80	71.50	84.00	81.96
	MaPLe [7]	36.80	87.50	87.40	93.20	94.20	95.40	68.40	81.40	80.54
	ProGrad [19]	43.00	83.60	85.80	92.80	96.60	95.90	68.80	82.70	81.15
	CLIP-LoRA	**54.97**	91.25	84.55	92.53	97.93	96.31	72.28	**86.36**	**84.52**
	CLIP-SingLoRA	52.27	**91.54**	85.24	92.72	**98.05**	**96.63**	**72.40**	86.28	84.39

B Detailed Ablation Study Results

Table 5 provides the detailed per-dataset results for our ablation study on the intrinsic rank (r) and the scaling factor (α), supplementing the summarized analysis in the main paper.

Table 5. Full per-dataset results for the ablation study on CLIP-SingLoRA configurations.

Method	Config (r, α)	Shots	Aircraft	EuroSAT	Food	Pets	Flowers	Caltech	DTD	UCF	Average	Parameters
CLIP-SingLoRA	(r=2, α=1)	**1**	29.10	73.58	85.75	92.61	81.40	93.87	53.37	75.47	73.14	92,160
	(r=4, α=1)		30.12	73.74	85.03	92.70	82.62	94.08	55.38	76.58	73.78	184,320
	(r=4, α=2)		29.82	72.23	84.49	91.82	83.96	93.83	55.20	76.26	73.45	184,320
	(r=4, α=8)		25.71	63.19	81.53	88.17	84.25	93.87	53.49	72.77	70.37	184,320
CLIP-SingLoRA	(r=2, α=1)	**4**	38.46	89.53	84.44	92.45	93.75	95.21	65.01	81.15	80.00	92,160
	(r=4, α=1)		38.13	88.31	84.06	91.66	93.26	95.46	64.66	80.49	79.50	184,320
	(r=4, α=2)		36.78	88.52	82.98	90.46	94.03	95.62	64.13	80.04	79.07	184,320
	(r=4, α=8)		35.82	88.67	80.40	86.45	94.36	94.89	64.18	77.29	77.76	184,320

References

1. Radford, A., et al.: Learning transferable visual models from natural language supervision. In: Proceedings of the 38th International Conference on Machine Learning, pp. 8748–8763 (2021)
2. Jia, C., et al.: Scaling up visual and vision-language representation learning with noisy text supervision. In: Proceedings of the 38th International Conference on Machine Learning, pp. 4904–4916 (2021)
3. Zhai, X., et al.: LiT: Zero-shot transfer with locked-image text tuning. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 18123–18133 (2022)
4. Zhou, K., Yang, J., Loy, C.C., Liu, Z.: Learning to prompt for vision-language models. Int. J. Comput. Vision **130**(9), 2337–2348 (2022)
5. Lialin, V., Deshpande, V., Rumshisky, A.: Scaling down to scale up: a guide to parameter-efficient fine-tuning. CoRR abs/2303.15647 (2023)
6. Zhou, K., Yang, J., Loy, C.C., Liu, Z.: Conditional prompt learning for vision language models. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 16816–16825 (2022)
7. Khattak, M.U., Rasheed, H., Maaz, M., Khan, S., Khan, F.S.: MaPLe: multimodal prompt learning. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 19113–19122 (2023)
8. Gao, P., et al.: CLIP-Adapter: better vision-language models with feature adapters. Int. J. Comput. Vision **132**(2), 581–595 (2024)
9. Yu, T., Lu, Z., Jin, X., Chen, Z., Wang, X.: Task residual for tuning vision-language models. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 10899–10909 (2023)
10. Hu, E.J., et al.: LoRA: Low-rank adaptation of large language models. In: International Conference on Learning Representations (2022)
11. Zanella, M., Ben Ayed, I.: Low-rank few-shot adaptation of vision-language models. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops, pp. 1593–1603 (2024)
12. David, B., Noam, R., Roy, V., Daniel, B., Ron, K.: SingLoRA: low rank adaptation using a single matrix. CoRR abs/2507.05566 (2025)
13. Hayou, S., Ghosh, N., Yu, B.: LoRA+: efficient low rank adaptation of large models. In: Proceedings of the 41st International Conference on Machine Learning, pp. 17783–17806 (2024)

14. Ben Zaken, E., Goldberg, Y., Ravfogel, S.: BitFit: simple parameter-efficient fine-tuning for transformer-based masked language-models. In: Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers), pp. 1–9 (2022)
15. Houlsby, N., et al.: Parameter-efficient transfer learning for NLP. In: Proceedings of the 36th International Conference on Machine Learning, pp. 2790–2799 (2019)
16. Lester, B., Al-Rfou, R., Constant, N.: The power of scale for parameter-efficient prompt tuning. In: Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing, pp. 3045–3059 (2021)
17. Li, X.L., Liang, P.: Prefix-tuning: optimizing continuous prompts for generation. In: Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers), pp. 4582–4597 (2021)
18. Yao, H., Zhang, R., Xu, C.: Visual-language prompt tuning with knowledge-guided context optimization. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 6757–6767 (2023)
19. Zhu, B., Niu, Y., Han, Y., Wu, Y., Zhang, H.: Prompt-aligned gradient for prompt tuning. In: Proceedings of the IEEE/CVF International Conference on Computer Vision, pp. 15659–15669 (2023)
20. Zhang, R., et al.: TIPAdapter: training-free adaption of CLIP for few-shot classification. In: European Conference on Computer Vision, pp. 493–510. Springer, Cham (2022)
21. Chen, G., Yao, W., Song, X., Li, X., Rao, Y., Zhang, K.: PLOT: prompt learning with optimal transport for vision-language models. In: International Conference on Learning Representations (2023)
22. Maji, S., Rahtu, E., Kannala, J., Blaschko, M., Vedaldi, A.: Fine-grained visual classification of aircraft. CoRR abs/1306.5151 (2013)
23. Helber, P., Bischke, B., Dengel, A., Borth, D.: EuroSAT: a novel dataset and deep learning benchmark for land use and land cover classification. IEEE J. Sel. Top. Appl. Earth Obser. Rem. Sens. **12**(7), 2217–2226 (2019)
24. Bossard, L., Guillaumin, M., Van Gool, L.: Food-101–mining discriminative components with random forests. In: European Conference on Computer Vision, pp. 446–461. Springer, Cham (2014)
25. Parkhi, O.M., Vedaldi, A., Zisserman, A., Jawahar, C.V.: Cats and dogs. In: 2012 IEEE Conference on Computer Vision and Pattern Recognition, pp. 3498–3505 (2012)
26. Nilsback, M.E., Zisserman, A.: Automated flower classification over a large number of classes. In: 2008 Sixth Indian Conference on Computer Vision, Graphics & Image Processing, pp. 722–729 (2008)
27. Fei-Fei, L., Fergus, R., Perona, P.: Learning generative visual models from few training examples: an incremental Bayesian approach tested on 101 object categories. In: 2004 Conference on Computer Vision and Pattern Recognition Workshop, p. 178 (2004)
28. Cimpoi, M., Maji, S., Kokkinos, I., Mohamed, S., Vedaldi, A.: Describing textures in the wild. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 3606–3613 (2014)
29. Soomro, K., Zamir, A.R., Shah, M.: UCF101: a dataset of 101 human actions classes from videos in the wild. CoRR abs/1212.0402 (2012)