Random Amalgamation of Adapters for Flatter Loss Landscapes: Towards Class-Incremental Learning with Better Stability

Yao Deng [2,3*], **Xiang Xiang** [1,2,3*†], **Jiaqi Gui** [2,3]

[1] School of Computer Science and Technology, Huazhong University of Science and Technology, China
[2] School of Artificial Intelligence and Automation, Huazhong University of Science and Technology, China
[3] HUST AI and Visual Learning Lab (HAIV Lab), Huazhong University of Science and Technology (HUST), China

Abstract

Class-incremental learning (CIL) enables models to continuously learn from streaming data while mitigating catastrophic forgetting of prior knowledge. Our research reveals that the CIL performance of pre-trained models (PTMs) varies significantly across different datasets, a phenomenon underexplored in existing studies. Through visualization, we observe that flatter loss landscapes correlate with superior CIL performance. This insight motivates us to enhance PTMs' CIL capability by promoting loss landscapes' flatness. Initially, we propose independently optimizing multiple adapter branches to equip PTMs with diverse learnable parameters, thereby improving stability during parameter updates. However, given computational and memory constraints, the number of adapters a PTM can accommodate is limited. To address this, we introduce a training strategy with randomized adapter amalgamation (RAA), compelling the model to maintain low loss across a broader and more continuous parameter space, significantly enhancing flatness. Furthermore, we refine existing sharpness-aware minimization techniques to further optimize the loss landscapes. Our extensive experiments and visualization results validate the efficacy of the method, resulting in the state-of-the-art (SOTA) performance.

Code — https://github.com/HAIV-Lab/RAA

Introduction

Class-incremental learning (CIL) aims to enable models to continuously learn in the incoming data stream while retaining learned information. Numerous methods have been proposed to mitigate the challenge of catastrophic forgetting within the realm of CIL, with the shared objective of developing a model with stability and plasticity. These approaches (Rebuffi et al. 2017; Wang et al. 2022a; Aljundi et al. 2018; Kirkpatrick et al. 2017; Mallya and Lazebnik 2018; Mallya, Davis, and Lazebnik 2018) are dedicated to improving adaptation to novel data streams while maintaining proficient performance in both recent and historical data in a harmonious manner. However, these methods do not take into account leveraging the generalization of large-scale pre-trained models (PTM) (Alexey 2020), which have achieved revolutionary success in the training of deep neural networks.

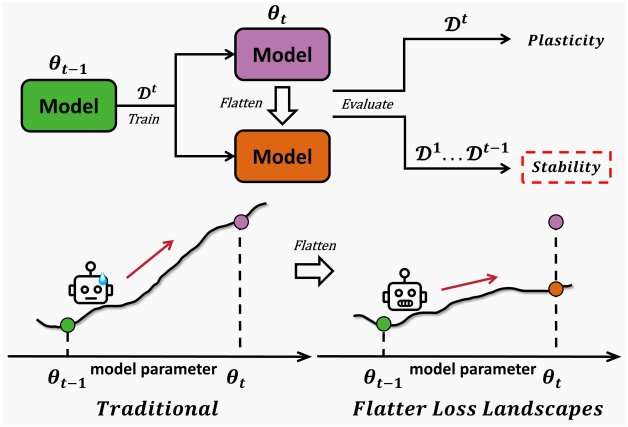


Figure 1: Relationship between loss landscapes and the stability. Models are able to retain lower loss when switching to the following sessions with flatter loss landscapes.

Recent PTM-based CIL methods are being actively explored and have shown significant performance gains (Zhou et al. 2024a). The enhancement can be attributed to the generalization imparted by models pre-trained on extensive datasets. Existing approaches primarily leverage the generalization through efficient parameter tuning methods, yet the updates to learnable parameters across different sessions still lead to forgetting of the old data distribution. While the stability-plasticity dilemma remains a pivotal research focus in CIL, existing evaluation metrics, primarily relying on per-session accuracy degradation, fail to provide direct and comprehensive quantification of this fundamental trade-off capability. Moreover, our investigations reveal that the CIL performance of different PTMs exhibits significant variability across diverse datasets, a phenomenon that remains unexplored and inadequately explained in current studies.

In this paper, we conduct experiments on various datasets with different PTMs and find that the performance is inconsistent across different datasets. To explain this phenomenon, inspired by (Foret et al. 2020), improving the generalization of neural networks, we sought to explore the cor-

*Equal contribution; co-first author.
†Correspondence to xex@hust.edu.cn; also w/ Peng Cheng Lab.

relation between the CIL performance with flatness of the loss landscapes. In this context, flatness denotes the capacity to maintain low loss as the model parameters oscillate around the optimal point. As parameters update during CIL, flat loss landscapes guarantee the preservation of low loss value while transitioning across sessions (as illustrated in Fig. 1), which reflects the PTMs' stability. By visualizing the loss landscapes, we ascertain that irrespective of the dataset used, a model's CIL performance and the flatness of its loss landscapes exhibit a direct and significant relationship.

The most straightforward way in adapter tuning involves independently optimizing parameters for multiple adapters, enabling the original PTM to accommodate a broader range of additional parameters. This allows the incrementally updated model to maintain low loss on previous data. However, while increasing the number of adapters enhances generalization capability, practical constraints on computational and memory resources prevent unlimited expansion of adapter quantities. Furthermore, this approach merely optimizes discrete, independent adapters without guaranteeing continuous low-loss regions across the parameter space.

To address the limitations, we propose a novel training strategy that randomly combines parameters from multiple adapters during optimization, enabling the PTM to achieve lower loss when employing these hybrid adapters. By sampling completely random combination weights at each training step, the model is required to perform gradient descent on distinct adapter configurations, thereby forcing the PTM to adapt to a broader spectrum of parameter variations. Moreover, previous works on flatness optimization are also evaluated and show improved CIL performance. We follow (Foret et al. 2020) to utilize and enhance sharpness-aware minimization (SAM+) to optimize the parameters in a manner that ensures the regions surrounding the model parameters consistently exhibit low loss values, as opposed to solely identifying an optimal parameter. The experiments and visualization results demonstrate that our approach effectively enhances the flatness of loss landscapes, consequently enhancing the CIL performance. Contributions are three-fold:

- Our investigation discovers varying CIL performance among diverse PTMs across different datasets. Through visualization, we observe a direct correlation between the flatness of loss landscapes and performance of the PTMs.
- To achieve flatter loss landscapes, we propose an adapter-tuning method with random amalgamation of adapters (RAA) and refine the sharpness-aware minimization algorithm (as SAM+). Our methods notably amplify the flatness and stability of models.
- Experiments are conducted on extensive datasets to showcase the SOTA performance. Through visualization, we demonstrate that enhancing the flatness of loss landscapes plays a crucial role in alleviating catastrophic forgetting and improving PTMs' stability.

Related Works

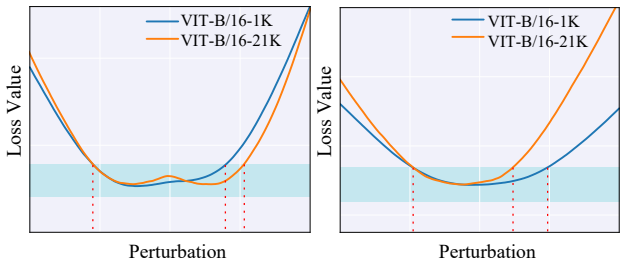
Class-incremental Learning. Class-incremental learning (CIL) aims to enable the model to learn new classes without forgetting the old knowledge during the continuous data stream. Current CIL methods can be mainly divided into three types: replay-based method (Zhao et al. 2022; Bang et al. 2021; Jodelet et al. 2023; Gao and Liu 2023), regularization-based method (Lee et al. 2020; Yang et al. 2023; Wang et al. 2021; Lopez-Paz and Ranzato 2017), and architecture-based method (Li et al. 2019; Wang et al. 2022b; Yang et al. 2022). Replay-based methods directly preserve or generate old instances during the new training phase to resist forgetting. These methods employ different strategies to retain or generate samples from the past, in order to efficiently use the past information. Regularization-based methods introduce regularization constraints during training or maintain the model's output on the old tasks, thus avoiding excessive modification of important parameters for old instances. Architecture-based methods dynamically adjust the network structure in response to ongoing learning tasks, typically by using different expansion strategies.

Continual Learning on Pre-Trained Models. With the deepening of research in pre-trained models (PTM), the strong representation capabilities of PTM have opened up new research directions for CIL. Numerous efficient parameter fine-tuning methods in NLP are leveraged for adapting to the stream of incoming data. Prompt-based methods (Wang et al. 2022d,c; Smith et al. 2023; Zhou et al. 2024a) strike a balance between model generalization and plasticity by introducing a small number of learnable prompt parameters. Adapter-based methods (Tan et al. 2024; Zhou et al. 2024a) integrate agile modules between layers of the transformer blocks and finetune the parameters within this specific segment. Alternative methods (McDonnell et al. 2024; Zhang et al. 2023) focus on feature level, aiming to achieve a representation space that sustains enhanced discrimination throughout the incremental process.

Flatness of Loss Landscapes. The concept of Flatness is initially introduced by (Hochreiter and Schmidhuber 1994), and its relationship with generalization has been studied in some works (Keskar et al. 2016; Dinh et al. 2017; Neyshabur et al. 2017; Mobahi 2016; Chaudhari et al. 2019; Sun et al. 2021). Sharpness-aware minimization (SAM) (Foret et al. 2020) introduces perturbations aligned with sharpness directions to parameters throughout optimization, enabling the model to attain low loss even in the worst case near the neighborhood of parameters. There are some works that attempt to optimize flatness for CIL with some success (Deng et al. 2021; Shi et al. 2021; Ma'sum et al. 2024; Shi et al. 2023). However, existing works primarily focus on loss design and optimization, overlooking the distinctive characteristics of PTM-based methods. We aim to address the limitation by leveraging a unique finetuning method for PTMs.

Preliminaries

Adapter-based Class-incremental Learning. CIL necessitates a learning system to integrate fresh class information while maintaining previously acquired knowledge consistently. Given a data stream comprising T non-overlapping sequential datasets, denoted as $\{\mathcal{D}^1, \mathcal{D}^2, ..., \mathcal{D}^T\}$, where $\mathcal{D}^t = \{(\mathbf{x}_i, y_i)\}_{i=1}^{N_t}$ represents the t-th training dataset, the instance $\mathbf{x}_i \in \mathbb{R}^D$ is from class $y_i \in Y_t$. Y_t is the label space



(a) CUB200 (b) ImageNetA

Figure 2: Loss landscapes of different PTMs on CUB200 (a) and ImageNetA (b).

PTM	CIFAR100		CUB200		ImageNetA		ImageNetR	
	$\mathcal{A}_{Last}$	$\mathcal{A}_{Avg}$	$\mathcal{A}_{Last}$	$\mathcal{A}_{Avg}$	$\mathcal{A}_{Last}$	$\mathcal{A}_{Avg}$	$\mathcal{A}_{Last}$	$\mathcal{A}_{Avg}$
VIT-B/16-1K	90.72	93.77	88.38	92.86	**64.88**	**72.68**	**80.15**	**83.87**
VIT-B/16-21K	**91.48**	**94.26**	**89.82**	**93.73**	62.43	70.83	79.38	83.63

Table 1: Experiment results with different pre-trained models on CIFAR100, CUB200, ImageNetA and ImageNetR.

of session t, and satisfies the condition $Y_t \cap Y_{t'} = \varnothing$ for $t \neq t'$. Consider a neural network $M_\theta = h_{\theta_c}(f_{\theta_r}(\cdot))$ with parameters $\theta = \{\theta_r, \theta_c\}$. f_{θ_r} denotes the feature extractor, which extracts representations of input images and h_{θ_c} represents the classification layer. We aim to learn a network M_θ, which performs well on test datasets for all seen classes $\mathcal{Y}_t = Y_1 \cup ...Y_t$ after t-th session. To leverage the generalization of the pre-trained model, we employ the parameters of the PTM for initialization of the feature extractor θ_r.

Adapter-based methods have demonstrated efficacy in PTM-based CIL. A typical adapter consists of a downsampled MLP layer $\mathbf{W}_{down} \in \mathbb{R}^{d\times h}$, a non-linear activation function, and an upsampled MLP layer $\mathbf{W}_{up} \in \mathbb{R}^{h\times d}$. We equip the original MLP structure in ViT with the adapter. For the input x_l, we formalize the output as

$$out = \text{MLP}(x_l) + \mathbf{W}_{up} \cdot \text{Relu}(\mathbf{W}_{down} \cdot x_l). \quad (1)$$

With the original parameters of PTM frozen, we only update the adapters of each layer and the classification head. We use the cosine classifier with the margin following (Peng et al. 2022). The training loss can be formulated as follows:

$$\mathcal{L}_t(\theta) = -\frac{1}{N_t}\sum_{i=1}^{N_t} log \frac{e^{s(cos\beta_{i,j}-m)}}{e^{s(cos\beta_{i,j}-m)} + \sum_{c=1}^{Y_t-\{j\}} e^{s(cos\beta_{i,c})}}, \quad (2)$$

where s and m are scale factor and margin factor, respectively, and $cos\beta_{i,j} = \frac{f_i}{\|f_i\|}\frac{p_j}{\|p_j\|}$, where f_i and p_j are representation of input $\mathbf{x_i}$ and prototype of class j.

Exploring Flatness in PTMs. Pre-trained models have demonstrated efficacy in CIL, owing to the robust generalization. However, our experiments, with two commonly used pre-trained models: *vit-base-patch16-224-in21k* pre-trained on ImageNet-21k (Russakovsky et al. 2015) and *vit-base-patch16-224* further fine-tuned on ImageNet-1K (Deng et al. 2009) after pre-training on ImageNet-21k in Tab. 1, showcase that models exhibiting strong CIL performance on certain datasets may demonstrate degraded performance on others. We investigate such inconsistency from the perspective of the flatness of loss landscapes.

Unlike geometric flatness, loss landscape flatness reflects a model's ability to maintain low loss when parameters vary near the global minimum. For visualization, we perturb model parameters along random Gaussian unit vectors, calculating loss across perturbation magnitudes to generate the curves (Fig. 2). The results show that a PTM with flatter loss landscapes, which demonstrates a broader range of parameter variations within a specific proximity to the loss minimum (depicted by the blue area), reaches better performance. It is evident that the performance of PTMs can be intuitively assessed through the flatness of loss landscapes.

Methodology

As the optimal minimum in parameter space varies across different sessions, ensuring parameter updates occur within flatter loss landscapes enables the retention of low-loss values for previous data in the subsequent sessions, thereby effectively improving stability in CIL. Therefore, we sought to improve the flatness to enhance the stability of PTMs.

We propose a multi-adapter architecture for downstream adaptation via diverse adapter interactions, reducing parameter sensitivity. To counter naive implementations' parameter growth, we introduce randomized adapter composition, achieving strong performance with minimal parameters. Additionally, an enhanced sharpness-aware minimization algorithm optimizes flatness explicitly. The overall framework is shown in Fig. 3

Individually Optimizing on Multiple Adapters

In adapter-based methods, only a single adapter per block is optimized during training to adapt to downstream data distributions. In contrast, our approach enables the PTM to maintain adaptability under multiple distinct adapters, ensuring robust performance in each session.

Specifically, we set N_A adapters for each block and individually update them with different initializations. For each adapter branch, we duplicate the input N_A times, distributing each copy to its corresponding branch, $\mathcal{A}^n = \{A_l^n\}_{l=1}^L$, which enables independent loss optimization across multiple adapters while maintaining isolated gradient computation paths. For the input x_l^n of the l-th MLP layer, the output of the n-th adapter A_l^n can be formulated as:

$$A_l^n(x_l^n) = \mathbf{W}_{up}^{n,l} \cdot \text{Relu}(\mathbf{W}_{down}^{n,l} \cdot x_l^n). \quad (3)$$

We optimize the loss computation across all adapters such that the model maintains consistently low loss values regardless of which adapter's parameters are activated. The loss can be denoted as:

$$\mathcal{L}_{single}(\theta; \mathcal{A}) = \frac{1}{N_A}\sum_{n=1}^{N_A} \mathcal{L}(\theta; \mathcal{A}^n). \quad (4)$$

This multiple mechanism simultaneously enhances parameter robustness and promotes flatter loss landscapes.

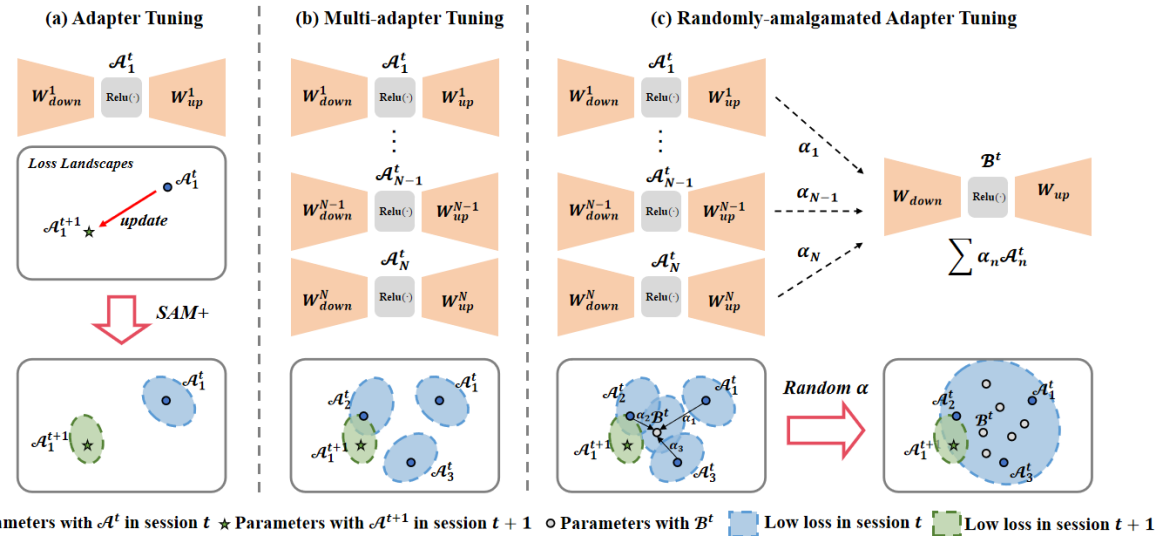


Figure 3: The comparison between adapter tuning, multi-adapter tuning and randomly-amalgamated adapter tuning.

Nevertheless, such a method can only accommodate the model's adjustment to a finite set of adapters, constrained by the original number of adapters. Although increasing the number of adapters could yield better performance, it will require an impractical abundance of parameters.

Randomly-Amalgamated Adapter Tuning

We propose an efficient adapter fusion approach using limited adapters (only 2 or 3 in our method) to construct a continuous low-loss manifold. Our method creates smooth parameter interpolations by randomly blending the weights of these physical adapters during training, enforcing consistent performance across all possible convex combinations. Unlike fixed-weight combinations that only optimize discrete points in the adapter space, our stochastic weight sampling enables exploration of the entire interpolation spectrum.

The traditional adapter structure, containing nonlinear activation functions, does not align with our requirements due to its nonlinear nature. We propose a novel adapter fusion framework that enables adapter integration in each layer, $\mathcal{B} = \{B_l\}_{l=1}^{L}$, while maintaining parameter efficiency during incremental learning. For the input x_l, in the training stage, we formalize the output of amalgamated adapters as:

$$\begin{aligned} B_l(x_l) &= \sum_{n=1}^{N_A} \alpha_l^n \Big(\mathbf{W}_{up}^n \cdot \text{Relu}\Big(\sum_{n=1}^{N_A} \alpha_l^n (\mathbf{W}_{down}^n \cdot x_l) \Big) \Big) \\ &= \Big(\sum_{n=1}^{N_A} \alpha_l^n \mathbf{W}_{up}^n \Big) \cdot \text{Relu}\Big(\Big(\sum_{n=1}^{N_A} \alpha_l^n \mathbf{W}_{down}^n \Big) \cdot x_l \Big), \end{aligned} \tag{5}$$

where α_l^n is the random weight of adapter n and satisfies $\sum_{n=1}^{N_A} \alpha_l^n = 1, \alpha_l^n > 0$, N_A is the number of adapters. Within each batch of training data, α is stochastically altered to encompass a broad spectrum of parameter configurations, subsequently optimized to achieve a more expansive and flatter region of minimized loss values. The loss can be denoted as:

$$\mathcal{L}_{fused}(\theta; \mathcal{B}; \alpha) = \mathcal{L}(\theta; \mathcal{B}; \alpha). \tag{6}$$

where $\alpha = \{\{\alpha_1^n\}_{n=1}^{N_A}, ...\{\alpha_L^n\}_{n=1}^{N_A}\}$. The training loss incorporates the utilization of existing adapters along with a random amalgamation of adapters in the following manner:

$$\mathcal{L}_{all}(\theta; \mathcal{A}; \mathcal{B}; \alpha) = \mathcal{L}_{single}(\theta; \mathcal{A}) + \mathcal{L}_{fused}(\theta; \mathcal{B}; \alpha). \tag{7}$$

Before each session following the initial one, to mitigate the increment of additional parameters, we initialize $\mathcal{A}_1$ with the mean value derived from all adapters and freeze it to solely update the parameters of other adapters. As we utilize the structure of a multi-adapter, the fusion process can be represented as

$$\mathbf{W}_{down}^1 = \frac{1}{N_A} \sum_{n=1}^{N_A} \mathbf{W}_{down}^n, \quad \mathbf{W}_{up}^1 = \frac{1}{N_A} \sum_{n=1}^{N_A} \mathbf{W}_{up}^n. \tag{8}$$

We further provide a theoretical view on why random amalgamation of adapters can bring generalization and can improve the performance of CIL.

Theorem 1 (Generalization Bound). Let $\boldsymbol{\omega}_0$ and $\boldsymbol{\omega}_{A_n}$ denote the weights of the pretrained model and the n-th adapter, respectively. Then, the overall predictor weights can be decomposed as $\boldsymbol{\omega} = \boldsymbol{\omega}_0 + \sum_{n=1}^{N_A} \alpha_n \boldsymbol{\omega}_{A_n}$. For any $\rho > 0$ and any distribution $\mathscr{D}$, with probability $1-\delta$ over the training set $\mathcal{D} \sim \mathscr{D}$,

$$\mathbb{E}_{\boldsymbol{\omega}}[L_{\mathscr{D}}(\boldsymbol{\omega})] \leq \mathbb{E}_{\boldsymbol{\omega}}[\max_{\|\boldsymbol{\epsilon}\|_2 \leq \rho} L_{\mathcal{D}}(\boldsymbol{\omega} + \boldsymbol{\epsilon})]$$

$$+ 3\sqrt{\frac{2k \log\left(1 + \frac{\|\boldsymbol{\omega}_0\|_2^2}{\sum_{n=1}^{N_A} k\alpha_n^2 \sigma_A^2 + \lambda\rho^2}\right) + 4\log\frac{2n}{\delta} + \tilde{O}(1)}{n}},$$

where $n = |\mathcal{D}|$, $\lambda = 1/(1+\sqrt{\log(n)/k})^2$, k is the number of parameters, $\boldsymbol{\omega}_{A_n} \sim \mathcal{N}(\mathbf{0}, \sigma_A^2 \boldsymbol{I})$, and we assumed

Method	ImageNetA		ImageNetR		CUB200		Stanford-Cars		CIFAR100		**Average**	
	$\mathcal{A}_{Last}\uparrow$	$\mathcal{A}_{Avg}\uparrow$	$\mathcal{A}_{Last}\uparrow$	$\mathcal{A}_{Avg}\uparrow$	$\mathcal{A}_{Last}\uparrow$	$\mathcal{A}_{Avg}\uparrow$	$\mathcal{A}_{Last}\uparrow$	$\mathcal{A}_{Avg}\uparrow$	$\mathcal{A}_{Last}\uparrow$	$\mathcal{A}_{Avg}\uparrow$	$\mathcal{A}_{Last}\uparrow$	$\mathcal{A}_{Avg}\uparrow$
Adam-adapter	48.81	58.84	65.79	72.42	85.84	91.33	45.55	56.97	87.30	91.19	66.66	74.15
Adam-ssf	48.94	58.79	66.61	74.36	85.67	90.99	37.42	49.67	85.28	89.92	64.78	72.75
Adam-prompt	29.60	38.57	63.68	71.63	85.28	90.89	40.28	52.19	84.96	89.68	60.76	68.59
L2P	44.04	51.24	72.34	77.36	67.02	79.62	39.83	50.88	84.06	88.26	61.46	69.47
DualPrompt	47.29	56.40	69.10	74.28	68.48	80.59	31.29	45.46	81.77	86.44	59.59	68.63
CODAPrompt	52.08	63.92	73.31	78.47	75.09	84.61	46.05	58.08	83.21	87.71	65.95	74.56
LAE	42.02	50.47	70.38	76.00	65.92	78.46	52.98	56.73	80.48	85.86	62.36	69.50
Ease	57.25	66.50	75.88	81.21	82.15	89.45	58.42	64.35	88.23	92.06	72.39	78.71
RanPAC	63.18	70.97	77.70	82.79	90.16	**93.52**	68.89	77.60	91.29	94.24	78.24	83.82
SLCA	60.72	68.23	79.35	83.29	87.52	92.39	65.17	73.15	91.26	94.29	76.80	82.27
SSIAT	62.43	70.83	79.38	83.63	88.38	92.86	60.63	67.76	91.35	94.35	76.43	81.89
Ours	**64.06**	**71.34**	**80.31**	**83.96**	**90.18**	93.47	**72.39**	**77.83**	**91.72**	**94.49**	**79.73**	**84.22**

Table 2: Average and last accuracy comparison on ImageNetA, ImageNetR, CUB200, Stanford-Cars and CIFAR100. All the methods are reproduced using the same three seeds, and we report the **Average** CIL performance.

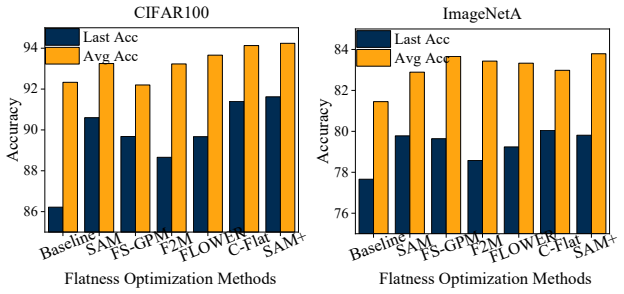


Figure 4: CIL performance of the PTM with various flatness optimization methods.

Method	$\mathcal{A}_{Last}\uparrow$		$\mathcal{A}_{Avg}\uparrow$	
	w/o SAM+	w/ SAM+	w/o SAM+	w/ SAM+
Adam-adapter	71.94	**72.35**	78.45	**78.72**
Adam-ssf	71.63	**72.31**	78.52	**78.93**
Adam-prompt	65.88	**67.37**	72.69	**74.23**
L2P	66.87	**68.25**	74.12	**75.25**
DualPrompt	66.66	**66.84**	74.43	**74.69**
CODAPrompt	70.92	**72.55**	78.68	**79.43**
LAE	64.70	**65.26**	72.70	**73.12**
Ease	75.88	**76.35**	82.31	**83.33**
RanPAC	80.58	**80.78**	85.38	**86.09**
SLCA	79.71	**80.15**	84.55	**85.02**
SSIAT	80.39	**81.05**	85.42	**85.59**

Table 3: The average results of $\mathcal{A}_{Last}$ and $\mathcal{A}_{Avg}$ on all evaluated datasets. Experiments are conducted on all the compared methods with or without SAM+.

$L_{\mathscr{D}}(\boldsymbol{\omega}) \leq \mathbb{E}_{\boldsymbol{\epsilon}}[L_{\mathscr{D}}(\boldsymbol{\omega}+\boldsymbol{\epsilon})]$, $\boldsymbol{\epsilon}$ is chosen as $\epsilon_i \sim \mathcal{N}(0, \sigma_{\boldsymbol{\epsilon}}^2)$ in every direction. Note that the size of the adapter weights is negligible compared to the pre-trained model; therefore, it is reasonable to assume that $\boldsymbol{\omega}_{A_n}$ has zero mean. See the Appendix for detailed proof.

Enhanced Sharpness-Aware Minimization

Various conventional flatness optimization methods are assessed in Fig. 4 and show enhanced CIL performance compared to the baseline. Existing methods are based on the design of loss function, which are plug-and-in for PTMs. SAM (Foret et al. 2020), for instance, utilizes the direct way involving perturbing parameters to ensure that the model maintains lower loss in the vicinity of optimal parameters θ. The problem for the t-th session can be formulated as:

$$\min_{\theta} \mathcal{L}_t^{SAM}(\theta) \quad \text{where} \quad \mathcal{L}_t^{SAM}(\theta) \triangleq \max_{\|\epsilon\|_2 \leq \rho} \mathcal{L}_t(\theta+\epsilon), \tag{9}$$

where ϵ is perturbation on parameters and ρ is a hyperparameter to control the optimized neighborhood range of θ. The sharpness-aware loss hinders the model from converging towards a narrow and steep minimum. By the classical solution to a dual norm problem, the approximated $\hat{\epsilon}$ can be solved as: $\hat{\epsilon}(\theta) = \rho \frac{\nabla_\theta \mathcal{L}_t(\theta)}{\|\nabla_\theta \mathcal{L}_t(\theta)\|_2}$ (see details in Appendix).

We propose SAM+ to enhance SAM, which utilizes the estimated $\hat{\epsilon}$ to impose regularization on the model's update $\Delta\theta$, preventing the model from updating towards directions associated with increasing loss values. Specifically, we compute the similarity between the model's update direction and the estimated sharpness direction: $cos\mu = \frac{\hat{\epsilon}(\theta)\Delta\theta}{\|\hat{\epsilon}(\theta)\|\|\Delta\theta\|}$. Subsequently, we introduce a penalty factor γ multiplied on gradient to regulate the model's update direction towards a trajectory distinct from sharpness: $\gamma = \min(1, \tau(1-e^{cos\mu-1}))$, where τ denotes the temperature. The regularized model update with the common gradient descent method can be denoted as: $\theta = \theta - \eta\gamma\Delta\theta$. In Fig. 4, SAM+ achieves superior performance compared to other methods.

Experiments

Datasets: To evaluate various continual learning techniques, we carried out extensive experiments using six datasets. Specifically, **CIFAR100** (Krizhevsky, Hinton et al. 2009) consists of 60,000 images of 100 classes. **CUB200** (Wah et al. 2011) contains bird images for 200 classes with around 60 images per class. **ImageNetR** (Hendrycks et al. 2021a) consists of 30,000 images with 200 categories. Each class encompasses a diverse range of image styles and challenging

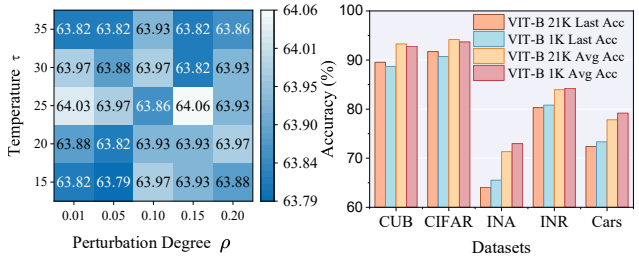


Figure 5: Further analysis on parameter robustness (**left**) and different pre-trained models (**right**).

samples extracted from the original ImageNet. This benchmark presents a formidable challenge due to the substantial divergence between these styles and the data used for pre-training. **ImageNetA** (Hendrycks et al. 2021b) is a practical dataset comprising 200 distinct classes. The dataset exhibits significant class imbalance, with certain classes containing only a small number of training instances. **Stanford-Cars** (Krause et al. 2013) contains 196 fine classes of car images.

Evaluation protocols: Following previous works (Zhang et al. 2023; Zhou et al. 2024a), we utilize common evaluation metrics in CIL. Specifically, we present the average accuracy of all classes after learning the last task, denoted as $\mathcal{A}_{Last}$, and average accuracy of the whole incremental sessions $\mathcal{A}_{Avg} = \frac{1}{T}\sum_{i=1}^{T}\mathcal{A}_i$.

Implementation details: We adopt ViT-B/16 (Alexey 2020) as the pre-trained model, which is pre-trained on ImageNet-21K (Russakovsky et al. 2015). The initial learning rate is set as 0.01 and we train the first session for 20 epochs and 10 epochs for later sessions. To mitigate experimental variability, following previous works (Zhang et al. 2023; Zhou et al. 2024a), we conduct experiments on three specific seeds: 1993, 1996, and 1997, and report the average results.

Comparison with the State-of-the-art

We compare with SOTA CIL methods on ViT-B/16 models pre-trained on ImageNet-21K. The methods we compare include L2P (Wang et al. 2022d), DualPrompt (Wang et al. 2022c), CODAPrompt (Smith et al. 2023), LAE (Gao et al. 2023), Ease (Zhou et al. 2024b), RanPAC (McDonnell et al. 2024), SLCA (Zhang et al. 2023), SSIAT (Tan et al. 2024) and other parameter-efficient tuning methods (Zhou et al. 2024a). All methods use the original setup and hyperparameters to reproduce their reported performance.

We report CIL performance in Tab. 2. ImageNetA and ImageNetR present challenges for pre-trained models because of factors like domain shifts and disparities in category distributions compared to pre-training data. It is evident that the performance of each approach on these two datasets is comparatively inferior to that achieved on others. RanPAC outperforms other previous methods on ImageNetA, while SSIAT achieves better performance on ImageNetR. Besides, adapter-based methods perform better than other similar methods (Adam-adapter achieves superior performance compared to other Adam methods), indicating that adapter-based methods are more appropriate for PTM-based CIL. Our method achieves SOTA performance on the two datasets. In ImageNetA, our method achieves the last accuracy of 64.06%, surpassing SSIAT and RanPAC by 0.88% and 1.63%, respectively. Additionally, our method achieves the highest final accuracy on fine-grained classification datasets, CUB-200 and Stanford Cars. Notably, it outperforms the second-best approach by 3.5% on the Stanford Cars. Besides, our method outperforms existing approaches, attaining SOTA performance with the average CIL performance gain across all benchmark datasets (in **Average**).

SAM	SAM+	RAA	ImageNetA		ImageNetR	
			$\mathcal{A}_{Last}\uparrow$	$\mathcal{A}_{Avg}\uparrow$	$\mathcal{A}_{Last}\uparrow$	$\mathcal{A}_{Avg}\uparrow$
×	×	×	62.43	70.83	79.38	83.63
✓	×	×	63.53	71.18	79.78	83.83
×	✓	×	63.86	71.23	79.81	83.79
×	×	✓	64.01	71.28	80.22	83.81
✓	×	✓	64.06	71.30	80.25	83.90
×	✓	✓	**64.06**	**71.34**	**80.31**	**83.96**

Table 4: Ablation results on ImageNetA and ImageNetR. We analyze the role of each component of our methodology.

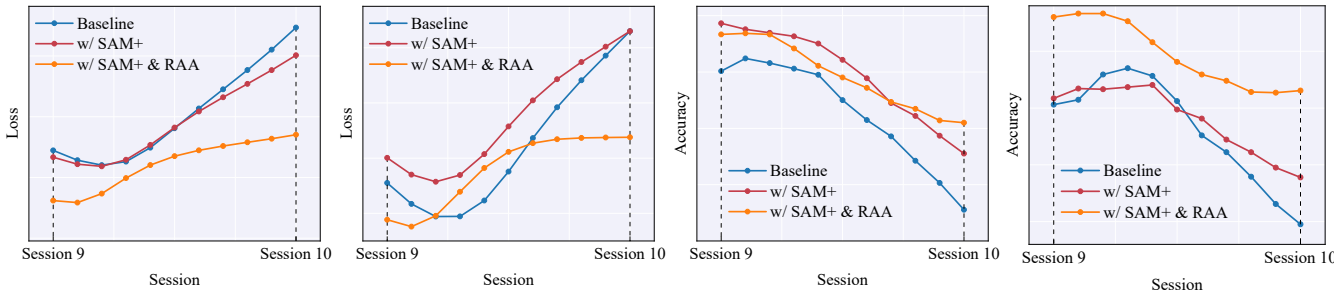
N_A	ImageNetA		ImageNetR		CIFAR100		CUB200	
	$\mathcal{A}_{Last}$	$\mathcal{A}_{Avg}$	$\mathcal{A}_{Last}$	$\mathcal{A}_{Avg}$	$\mathcal{A}_{Last}$	$\mathcal{A}_{Avg}$	$\mathcal{A}_{Last}$	$\mathcal{A}_{Avg}$
2	**64.06**	**71.34**	**80.31**	**83.96**	**91.72**	**94.19**	89.94	93.44
3	64.05	70.88	79.04	82.65	91.31	93.85	**90.18**	**93.47**
4	63.27	70.32	78.05	82.02	90.96	93.64	89.81	93.44

Table 5: Ablation study for the number of adapters on ImageNetA, ImageNetR, CIFAR100 and CUB200.

Ablation Study

The effectiveness of flattening methods: We perform an ablation analysis to examine the efficacy of each element within our methodology. Specifically, we report the CIL performance in various situations with original SAM (Foret et al. 2020), our proposed enhanced SAM and random amalgamation of adapters (RAA) in Table 4. It is evident that SAM achieves an enhancement in performance compared to the baseline, signifying that the CIL performance can be improved by the flattening of the loss landscapes. After the enhancement of the optimization process, SAM+ brings further improvements. In addition, employing only amalgamated adapters in conjunction with the training approach involving randomized weights can also result in enhancements in performance. The best performance is achieved after combining the two elements. In Fig. 7, we visualize the loss landscapes of the first session on ImageNetA and ImageNetR. Through the integration of distinct modules within our method, the loss landscapes gradually transition to a flatter form, leading to a notable decrease in parameter sensitivity in the vicinity of the optimal point.

The plug-and-play ability of SAM+: We integrate the SAM+ with existing approaches to ascertain its universal efficacy in enhancing CIL across diverse methods seamlessly,



(a) ImageNetA: Loss (b) ImageNetR: Loss (c) ImageNetA: Accuracy (d) ImageNetR: Accuracy

Figure 6: Visualization of loss and accuracy in the process of model updating from session 9 to session 10.

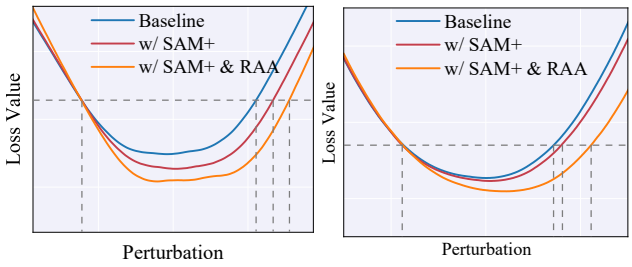


Figure 7: Visualization of loss landscapes in ablation study on ImageNetA (**left**) and ImageNetR (**right**).

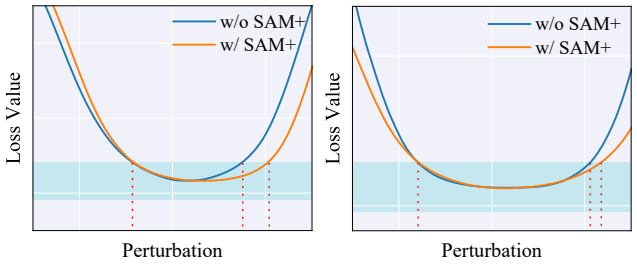


Figure 8: Visualization of loss landscapes of CODAPrompt (**left**) and SLCA (**right**) with or without SAM+.

thereby validating its adaptability and utility. The results of four datasets are averaged and shown in Tab. 3. To demonstrate the plug-and-play capabilities of SAM+, the experimental results depicted in gray rows reflect the integration of SAM+ within some compared methods. For all compared methods, the integration of SAM+ results in performance enhancements, demonstrating the efficacy of SAM+. Although all methods show enhancements following the introduction of the proposed SAM+ method, they still fall short in comparison to our method. Moreover, SAM+ effectively flattens the loss landscapes of previous methods, as illustrated in Fig. 8, showcasing the plug-and-play capability.

Parameter robustness and different PTMs: We conduct experiments on five datasets with different numbers of adapters. As shown in Tab. 5, increasing the number of adapters does not invariably lead to a commensurate improvement in CIL performance, notwithstanding the introduction of additional parameters. The introduction of more adapters introduces additional randomness, leading to training instability and, consequently, performance degradation. Besides, we conduct a comprehensive analysis of our method's sensitivity to hyper-parameters and its performance across different pretrained models in Fig. 5. The results demonstrate that our approach consistently achieves performance gains under varying hyperparameter configurations and with different PTMs.

Visualization on the effectiveness of flat loss landscapes: To further demonstrate the impact of flatter loss landscapes on enhancing model stability during the CIL process, we save the checkpoints of several models and evaluate their performance on previous data when transitioning from session 9 to session 10 on ImageNetA and ImageNetR (refer to Appendix for more results). In Fig. 6a and Fig. 6b, it is evident that the model's loss on previous data exhibits a slower increase following the implementation of flattening methods. The corresponding decline in the model's accuracy on previous data also occurs at a reduced rate (as illustrated in Fig. 6c and Fig. 6d), indicating a mitigation of catastrophic forgetting. The visualization highlights that our approach successfully promotes a flatter loss landscape, enabling the model to sustain performance on previous data during update between sessions, enhancing model stability and effectively addressing the catastrophic forgetting.

Conclusion

In this work, we propose a novel class-incremental learning (CIL) method for pre-trained models (PTMs) that enhances performance by explicitly optimizing the flatness of loss landscapes. Our key innovation, Randomized Adapter Amalgamation (RAA), efficiently constructs a continuous low-loss space through stochastic adapter blending, while enhanced sharpness-aware optimization further smooths the landscapes. Experiments show our approach achieves SOTA results, demonstrating that flatter landscapes significantly improve continual learning. This work provides both a practical solution and theoretical insights into CIL performance.

Acknowledgements

This work was supported by the HUST Interdisciplinary Research Support Program (2025JCYJ077), the project of Peng Cheng Lab (PCL2025AS214), the 2026 Optics-Valley Excellence Project funded by Nat'l Graduate College for Elite Engineers of HUST, and School of Computer Science and Technology, School of Artificial Intelligence and Automation, and Hopcroft Center for Computing Science. Zhipeng Chen is appreciated for participating in initial experiments.

References

Alexey, D. 2020. An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv preprint arXiv: 2010.11929*.

Aljundi, R.; Babiloni, F.; Elhoseiny, M.; Rohrbach, M.; and Tuytelaars, T. 2018. Memory aware synapses: Learning what (not) to forget. In *Proceedings of the European conference on computer vision (ECCV)*, 139–154.

Bang, J.; Kim, H.; Yoo, Y.; Ha, J.-W.; and Choi, J. 2021. Rainbow Memory: Continual Learning With a Memory of Diverse Samples. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 8218–8227.

Chaudhari, P.; Choromanska, A.; Soatto, S.; LeCun, Y.; Baldassi, C.; Borgs, C.; Chayes, J.; Sagun, L.; and Zecchina, R. 2019. Entropy-sgd: Biasing gradient descent into wide valleys. *Journal of Statistical Mechanics: Theory and Experiment*, 2019(12): 124018.

Deng, D.; Chen, G.; Hao, J.; Wang, Q.; and Heng, P.-A. 2021. Flattening sharpness for dynamic gradient projection memory benefits continual learning. *Advances in Neural Information Processing Systems*, 34: 18710–18721.

Deng, J.; Dong, W.; Socher, R.; Li, L.-J.; Li, K.; and Fei-Fei, L. 2009. Imagenet: A large-scale hierarchical image database. In *2009 IEEE conference on computer vision and pattern recognition*, 248–255. Ieee.

Dinh, L.; Pascanu, R.; Bengio, S.; and Bengio, Y. 2017. Sharp minima can generalize for deep nets. In *International Conference on Machine Learning*, 1019–1028. PMLR.

Foret, P.; Kleiner, A.; Mobahi, H.; and Neyshabur, B. 2020. Sharpness-aware minimization for efficiently improving generalization. *arXiv preprint arXiv:2010.01412*.

Gao, Q.; Zhao, C.; Sun, Y.; Xi, T.; Zhang, G.; Ghanem, B.; and Zhang, J. 2023. A unified continual learning framework with general parameter-efficient tuning. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 11483–11493.

Gao, R.; and Liu, W. 2023. DDGR: Continual Learning with Deep Diffusion-based Generative Replay. In Krause, A.; Brunskill, E.; Cho, K.; Engelhardt, B.; Sabato, S.; and Scarlett, J., eds., *Proceedings of the 40th International Conference on Machine Learning*, volume 202 of *Proceedings of Machine Learning Research*, 10744–10763. PMLR.

Hendrycks, D.; Basart, S.; Mu, N.; Kadavath, S.; Wang, F.; Dorundo, E.; Desai, R.; Zhu, T.; Parajuli, S.; Guo, M.; et al. 2021a. The many faces of robustness: A critical analysis of out-of-distribution generalization. In *Proceedings of the IEEE/CVF international conference on computer vision*, 8340–8349.

Hendrycks, D.; Zhao, K.; Basart, S.; Steinhardt, J.; and Song, D. 2021b. Natural adversarial examples. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 15262–15271.

Hochreiter, S.; and Schmidhuber, J. 1994. Simplifying neural nets by discovering flat minima. *Advances in neural information processing systems*, 7.

Jodelet, Q.; Liu, X.; Phua, Y. J.; and Murata, T. 2023. Class-Incremental Learning Using Diffusion Model for Distillation and Replay. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV) Workshops*, 3425–3433.

Keskar, N. S.; Mudigere, D.; Nocedal, J.; Smelyanskiy, M.; and Tang, P. T. P. 2016. On large-batch training for deep learning: Generalization gap and sharp minima. *arXiv preprint arXiv:1609.04836*.

Kirkpatrick, J.; Pascanu, R.; Rabinowitz, N.; Veness, J.; Desjardins, G.; Rusu, A. A.; Milan, K.; Quan, J.; Ramalho, T.; Grabska-Barwinska, A.; et al. 2017. Overcoming catastrophic forgetting in neural networks. *Proceedings of the national academy of sciences*, 114(13): 3521–3526.

Krause, J.; Stark, M.; Deng, J.; and Fei-Fei, L. 2013. 3d object representations for fine-grained categorization. In *Proceedings of the IEEE international conference on computer vision workshops*, 554–561.

Krizhevsky, A.; Hinton, G.; et al. 2009. Learning multiple layers of features from tiny images.

Lee, J.; Hong, H. G.; Joo, D.; and Kim, J. 2020. Continual Learning With Extended Kronecker-Factored Approximate Curvature. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*.

Li, X.; Zhou, Y.; Wu, T.; Socher, R.; and Xiong, C. 2019. Learn to Grow: A Continual Structure Learning Framework for Overcoming Catastrophic Forgetting. In Chaudhuri, K.; and Salakhutdinov, R., eds., *Proceedings of the 36th International Conference on Machine Learning*, volume 97 of *Proceedings of Machine Learning Research*, 3925–3934. PMLR.

Lopez-Paz, D.; and Ranzato, M. A. 2017. Gradient Episodic Memory for Continual Learning. In Guyon, I.; Luxburg, U. V.; Bengio, S.; Wallach, H.; Fergus, R.; Vishwanathan, S.; and Garnett, R., eds., *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc.

Mallya, A.; Davis, D.; and Lazebnik, S. 2018. Piggyback: Adapting a single network to multiple tasks by learning to mask weights. In *Proceedings of the European conference on computer vision (ECCV)*, 67–82.

Mallya, A.; and Lazebnik, S. 2018. Packnet: Adding multiple tasks to a single network by iterative pruning. In *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*, 7765–7773.

Ma'sum, M. A.; Pratama, M.; Lughofer, E.; Liu, L.; Kowalczyk, R.; et al. 2024. Few-shot continual learning via flat-to-wide approaches. *IEEE Transactions on Neural Networks and Learning Systems*.

McDonnell, M. D.; Gong, D.; Parvaneh, A.; Abbasnejad, E.; and van den Hengel, A. 2024. Ranpac: Random projections and pre-trained models for continual learning. *Advances in Neural Information Processing Systems*, 36.

Mobahi, H. 2016. Training recurrent neural networks by diffusion. *arXiv preprint arXiv:1601.04114*.

Neyshabur, B.; Bhojanapalli, S.; McAllester, D.; and Srebro, N. 2017. Exploring generalization in deep learning. *Advances in neural information processing systems*, 30.

Peng, C.; Zhao, K.; Wang, T.; Li, M.; and Lovell, B. C. 2022. Few-shot class-incremental learning from an open-set perspective. In *European Conference on Computer Vision*, 382–397. Springer.

Rebuffi, S.-A.; Kolesnikov, A.; Sperl, G.; and Lampert, C. H. 2017. icarl: Incremental classifier and representation learning. In *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*, 2001–2010.

Russakovsky, O.; Deng, J.; Su, H.; Krause, J.; Satheesh, S.; Ma, S.; Huang, Z.; Karpathy, A.; Khosla, A.; Bernstein, M.; et al. 2015. Imagenet large scale visual recognition challenge. *International journal of computer vision*, 115: 211–252.

Shi, G.; Chen, J.; Zhang, W.; Zhan, L.-M.; and Wu, X.-M. 2021. Overcoming catastrophic forgetting in incremental few-shot learning by finding flat minima. *Advances in neural information processing systems*, 34: 6747–6761.

Shi, W.; Chen, Y.; Zhao, Z.; Lu, W.; Yan, K.; and Du, X. 2023. Create and find flatness: Building flat training spaces in advance for continual learning. In *ECAI 2023*, 2138–2145. IOS Press.

Smith, J. S.; Karlinsky, L.; Gutta, V.; Cascante-Bonilla, P.; Kim, D.; Arbelle, A.; Panda, R.; Feris, R.; and Kira, Z. 2023. Coda-prompt: Continual decomposed attention-based prompting for rehearsal-free continual learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 11909–11919.

Sun, X.; Zhang, Z.; Ren, X.; Luo, R.; and Li, L. 2021. Exploring the vulnerability of deep neural networks: A study of parameter corruption. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 35, 11648–11656.

Tan, Y.; Zhou, Q.; Xiang, X.; Wang, K.; Wu, Y.; and Li, Y. 2024. Semantically-Shifted Incremental Adapter-Tuning is A Continual ViTransformer. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 23252–23262.

Wah, C.; Branson, S.; Welinder, P.; Perona, P.; and Belongie, S. 2011. The caltech-ucsd birds-200-2011 dataset.

Wang, F.-Y.; Zhou, D.-W.; Ye, H.-J.; and Zhan, D.-C. 2022a. Foster: Feature boosting and compression for class-incremental learning. In *European conference on computer vision*, 398–414. Springer.

Wang, L.; Zhang, X.; Li, Q.; Zhu, J.; and Zhong, Y. 2022b. CoSCL: Cooperation of Small Continual Learners is Stronger than a Big One. arXiv:2207.06543.

Wang, S.; Li, X.; Sun, J.; and Xu, Z. 2021. Training Networks in Null Space of Feature Covariance for Continual Learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 184–193.

Wang, Z.; Zhang, Z.; Ebrahimi, S.; Sun, R.; Zhang, H.; Lee, C.-Y.; Ren, X.; Su, G.; Perot, V.; Dy, J.; et al. 2022c. Dualprompt: Complementary prompting for rehearsal-free continual learning. In *European Conference on Computer Vision*, 631–648. Springer.

Wang, Z.; Zhang, Z.; Lee, C.-Y.; Zhang, H.; Sun, R.; Ren, X.; Su, G.; Perot, V.; Dy, J.; and Pfister, T. 2022d. Learning to prompt for continual learning. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 139–149.

Yang, B.; Deng, X.; Shi, H.; Li, C.; Zhang, G.; Xu, H.; Zhao, S.; Lin, L.; and Liang, X. 2022. Continual Object Detection via Prototypical Task Correlation Guided Gating Mechanism. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 9255–9264.

Yang, Y.; Zhou, D.-W.; Zhan, D.-C.; Xiong, H.; Jiang, Y.; and Yang, J. 2023. Cost-Effective Incremental Deep Model: Matching Model Capacity With the Least Sampling. *IEEE Transactions on Knowledge and Data Engineering*, 35(4): 3575–3588.

Zhang, G.; Wang, L.; Kang, G.; Chen, L.; and Wei, Y. 2023. Slca: Slow learner with classifier alignment for continual learning on a pre-trained model. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 19148–19158.

Zhao, H.; Wang, H.; Fu, Y.; Wu, F.; and Li, X. 2022. Memory-Efficient Class-Incremental Learning for Image Classification. *IEEE Transactions on Neural Networks and Learning Systems*, 33(10): 5966–5977.

Zhou, D.-W.; Cai, Z.-W.; Ye, H.-J.; Zhan, D.-C.; and Liu, Z. 2024a. Revisiting class-incremental learning with pre-trained models: Generalizability and adaptivity are all you need. *International Journal of Computer Vision*, 1–21.

Zhou, D.-W.; Sun, H.-L.; Ye, H.-J.; and Zhan, D.-C. 2024b. Expandable subspace ensemble for pre-trained model-based class-incremental learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 23554–23564.