Class Incremental Learning with Multi-Teacher Distillation

Haitao Wen, Lili Pan, Yu Dai, Heqian Qiu, Lanxiao Wang*, Qingbo Wu, Hongliang Li*
University of Electronic Science and Technology of China, Chengdu, China
{haitaowen, ydai, lanxiao.wang}@std.uestc.edu.cn, {lilipan, hqqiu, qbwu, hlli}@uestc.edu.cn

Abstract

Distillation strategies are currently the primary approaches for mitigating forgetting in class incremental learning (CIL). Existing methods generally inherit previous knowledge from a single teacher. However, teachers with different mechanisms are talented at different tasks, and inheriting diverse knowledge from them can enhance compatibility with new knowledge. In this paper, we propose the MTD method to find multiple diverse teachers for CIL. Specifically, we adopt weight permutation, feature perturbation, and diversity regularization techniques to ensure diverse mechanisms in teachers. To reduce time and memory consumption, each teacher is represented as a small branch in the model. We adapt existing CIL distillation strategies with MTD and extensive experiments on CIFAR-100, ImageNet-100, and ImageNet-1000 show significant performance improvement. Our code is available at https://github.com/HaitaoWen/CLearning.

1. Introduction

Continual learning is crucial for intelligent machines to adapt to various environments [6, 23]. Class incremental learning (CIL) as one of the most challenging scenarios, requires the model to incrementally learn a sequence of tasks without task identification [45]. However, when traditional learning strategies are applied in such a setting, it often incurs the catastrophic forgetting of old tasks [31, 38].

Distillation strategies are essentially functional regularization methods [16, 46], which encourage the input-output mapping of the model to be invariant and are the most direct approaches to incremental learning. LwF [25] is the pioneer in distilling the output logits of the teacher to the student, similar distillation strategies are also used in iCaRL [37] and BiC [52]. In addition, existing CIL methods also distill the intermediate features of the model, such as LUCIR [17], PODNet [8], GeoDL [42], and AFC [19]. Multiple teachers can provide diverse knowledge that is beneficial for training the student [13]. Some work has introduced multi-teacher distillation to CIL in recent times. DT-CIL [4] maximizes the mutual information between the student and two teachers, which are trained on old and new classes respectively.

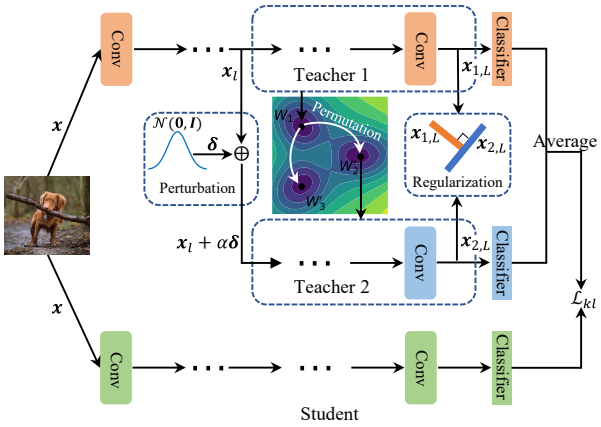

Figure 1. The diagram of MTD for finding diverse teachers from a basic model. Each teacher is represented as a small branch. Based on the properties of diverse teachers, weight permutation is used to teleport parameters from basic W_1 in one low-loss region to W_2' in another low-loss region, feature perturbation is adopted to perturb the input $\boldsymbol{x}_l$ of the permuted branch by adding $\boldsymbol{\delta}$ sampled from the normal distribution $\mathcal{N}(\mathbf{0}, \boldsymbol{I})$, and diversity regularization is applied to orthogonalize embeddings $\boldsymbol{x}_{1,L}$ and $\boldsymbol{x}_{2,L}$ of teachers.

To analyze the multi-teacher distillation in CIL and find what properties should teachers have, we first conduct preliminary experiments considering two teacher generation methods: "Oracle", where teachers are completely trained with different random seeds; "PFT", where teachers are obtained by finetuning on the memory of previous tasks with a periodic learning rate. Through effectiveness analysis, we find that *improving diversity between teachers while maintaining prediction quality can improve CIL performance.* Although multi-teacher distillation is beneficial for CIL, one problem must be faced to apply it, how to effectively find diverse teachers. Instead of directly retraining multiple teachers using different random seeds [18, 24], different initialization or architectures [53], and different data [4, 51],

*Corresponding authors.

we propose MTD for effectively finding diverse teachers from a basic model, Figure 1 shows the diagram.

Diverse teachers mean that they have different responses to the same inputs, i.e., they use dissimilar mechanisms for making their predictions. Furthermore, [29] demonstrates that the lack of linear connectivity [9, 11] between models implies they use dissimilar mechanisms. The lack of linear connectivity means that model parameters are in different low-loss regions and blocked by the high-loss ridge [47]. We analyze the properties of "Oracle" by visualizing loss landscapes in Figure 4, which confirms this inference. To find diverse teachers from a basic model instead of retraining, the problem before us is how to transform the basic model parameters from one low-loss region to another low-loss region. There are generally several transformations in parameter space, including translation [22], rotation [49], scaling [49, 54], and permutation [35]. We choose weight permutation to transform parameters as it can invariantly teleport parameters from one region to another region [3, 10, 43]. However, the invariance between teleported parameters and original parameters conflicts with our intended diversity. This drives us to explore the region around the teleported parameters along a different optimization trajectory compared with the original parameters for breaking invariance. To this end, we apply the feature perturbation for teleported parameters. In addition, Figure 6 shows that the cosine similarities between embeddings of teachers found by "Oracle" tend to be mutually orthogonal. Therefore, we propose diversity regularization to minimize the absolute cosine similarities between embeddings of teachers.

To reduce time and memory consumption, most of the feature extraction layers are shared among teachers, each teacher has a specific prediction branch containing layers that are the same as the structure of the basic model, but the parameters are different from each other. Finally, our main contributions can be summarized as follows:

- We find that improving diversity between teachers while maintaining prediction quality can improve performance;
- We find two properties of diverse teachers: parameters of teachers are in different low-loss regions, and embeddings of teachers tend to be mutually orthogonal;
- We propose MTD including weight permutation, feature perturbation, and diversity regularization based on the properties of "Oracle" to effectively find diverse teachers;
- We adapt existing CIL distillation strategies with MTD and extensive experiments on various benchmarks show significant performance improvement.

2. Related Work

2.1. Class Incremental Learning

The existing CIL methods can be generally divided into three categories. *Memory replay* stores a small amount of representative samples for each old class and replays them in the new task. iCaRL [37] selects samples that are close to the average embeddings. Mnemonics [26] parameterizes exemplars and uses the bilevel optimization to make exemplars approximate old data as much as possible. CIM [30] adaptively downsamples non-discriminative pixels to save more compressed exemplars in fixed-size memory. *Knowledge distillation* transfers knowledge from the previous model to the new model to mitigate forgetting. iCaRL [37] and BiC [52] distill the output logits on old classes. LUCIR [17] adopts the cosine distillation loss between old and new embeddings. PODNet [8] proposes pooled feature distillation to balance old and new classes. GeoDL [42] further constrains the cosine similarities between embeddings along the geodesic path in the manifold of feature subspaces. AFC [19] adaptively distills old features according to their importance. DT-CIL [4] uses two teachers trained on old and new classes respectively for distillation. *Dynamic structure* assigns a sub-network or a new independent network for each task. AANet [27] uses stable blocks and plastic blocks, and adaptively weights features of these two types of blocks to balance old and new tasks.

2.2. Multi-Teacher Distillation

Inheriting diverse knowledge from multiple teachers can improve the generalization of the student model [13, 50]. The study of multi-teacher distillation involves three aspects. *How to find multiple teachers.* [18, 24] train the basic model multiple times with different random seeds to obtain teachers. [51] trains models with different types of data to obtain teachers. [53] uses teachers that have different initialization or architectures from the basic model. However, retraining models from scratch is time-consuming. [41] injects noise into the logits of the basic model to simulate multiple teachers. [32] exploits stochastic blocks and skip connections to generate teachers. *How to represent multiple teachers.* Similarly, [24, 51, 53] represent multiple teachers in the form of independent models, which makes them time-and-memory consuming. [32, 41] obtain multiple teachers within a single basic model and are lightweight. Besides, [15, 44] adopt the structure of multiple branches to represent teachers. *How to learn from multiple teachers.* The most direct way is to distill the average output response of teachers [18, 44, 53]. [39, 51] further distill the sum of output responses weighted by normalized coefficients. [32, 41] randomly select one output response of teachers for each training iteration. In addition, due to significant differences between intermediate features of teachers, [33] use nonlinear layers to transform the features of the student to find more general solutions.

We review additional related work about weight permutation and discuss the differences and similarities between our work and existing work in Section 7 of supplementary.

3. Analyzing Multi-Teacher Distillation in CIL

Class incremental learning (CIL) requires a model $\mathcal{F}_t(\boldsymbol{x})$ to continually learn a sequence of T tasks associated with training data $\{\mathcal{D}_t\}_{t=1}^{T}$, where t is the task identity. The model $\mathcal{F}_t$ is composed of a classifier G_t and a embedding extractor containing L layers' F, i.e., $\mathcal{F}_t = G_t \circ F_{t,L} \circ \cdots \circ F_{t,1}$, correspondingly, their parameters form a set $W_t = \{\boldsymbol{\theta}_t, \boldsymbol{W}_{t,L}, \ldots, \boldsymbol{W}_{t,1}\}$. The data $\mathcal{D}_t = \{(\boldsymbol{x}_t, y_t)\}$ is a set of samples with the input $\boldsymbol{x}$ and the label y. Under the settings of CIL, classes between different tasks are not overlapped, i.e., $\{y_k\} \cap \{y_{i\neq k}\} = \varnothing$ and t is not known during both the training and testing phases [45]. Memory replay and knowledge distillation are effective strategies to mitigate forgetting [8, 19, 26, 37]. The accumulated episodic memory of t learned tasks is denoted as $\mathcal{M}_t$. When learning task t, the final model $\mathcal{F}_{t-1}$ of task $t-1$ is generally taken as the teacher of the current (student) model $\mathcal{F}_t$. The distillation loss between output logits of teacher and student models for an input $\boldsymbol{x} \sim \mathcal{D}_t \cup \mathcal{M}_{t-1}$ measured by Kullback-Leibler (KL) divergence can be formalized as follows,

$$\mathcal{L}_{kl}(\boldsymbol{x}) = \mathrm{KL}(\mathcal{F}_{t-1}^{1:c_{t-1}}(\boldsymbol{x}) \| \mathcal{F}_{t}^{1:c_{t-1}}(\boldsymbol{x})), \tag{1}$$

where $c_{t-1} = |\cup_{k=1}^{t-1}\{y_k\}|$ is the number of learned classes after task $t-1$ and the superscript $1:c_{t-1}$ means KL divergence is only computed from the 1-st logit to the c_{t-1}-th logit. We omit the operation of softmax [17, 37] for a brief description. Then, the loss of task t with $\mathcal{L}_{kl}$ is,

$$\mathcal{L}_t = \mathcal{L}_{other} + \lambda \mathcal{L}_{kl}, \tag{2}$$

where $\mathcal{L}_{other}$ denotes the sum of other losses, such as cross-entropy (CE) loss [37], NCA loss [12], pooled feature distillation loss [8], or cosine embedding loss [17], and λ is the coefficient for $\mathcal{L}_{kl}$.

3.1. Preliminary Experiments

To study the effectiveness of multi-teacher distillation in CIL and find important properties for performance improvement, we conduct preliminary experiments considering two basic multi-teacher finding methods: "Oracle" and "PFT". The "Oracle" method generates teachers by training incremental models with different random seeds, which is similar to [18, 24]. For example, we use three teachers for task t, the first teacher is the final model of task $t-1$, denoted as $\mathcal{F}_{t-1,1} \coloneqq \mathcal{F}_{t-1}$, the other two teachers are the final model of task $t-1$ in other two CIL processes respectively controlled with different random seeds, denoted as $\mathcal{F}^*_{t-1,2}$ and $\mathcal{F}^*_{t-1,3}$. These three teachers are independently and completely trained, their predictions are of high quality and their mechanisms have significant differences (please see Figure 5) compared with other teacher finding methods [32, 41]. The "PFT" method is the abbreviation for "Periodic Fine-Tuning", which finetune W_{t-1} with a periodic learning rate

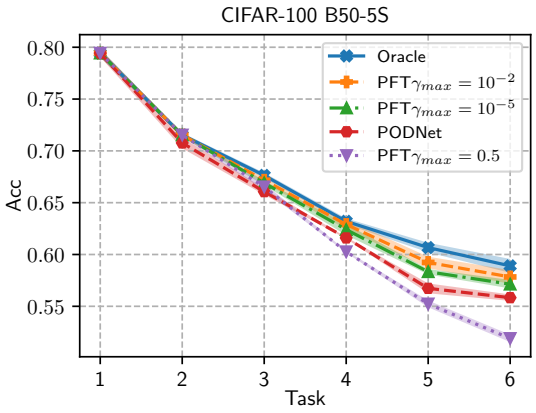

Figure 2. Testing accuracy curves of multi-teacher finding methods "Oracle" and "PFT" applied in CIL on the benchmark CIFAR100 B50-5S (Section 5). PODNet [8] is chosen as the baseline. Each experiment is run 3 times, and results are reported with mean and standard deviation.

on the memory $\mathcal{M}_{t-1}$, the learning rate γ is defined as,

$$\gamma = \frac{\gamma_{max}}{2}(\sin(\frac{2\pi i}{N} + \frac{3\pi}{2}) + 1), \tag{3}$$

where γ_{max} is the maximum learning rate in a period, i is the index of current iteration, and N is the total iterations in a period. Figure 3 shows the learning rate curve when $\gamma_{max} = 10^{-2}$ and $N = 100$. When γ increases, the

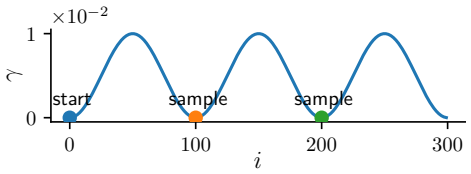

Figure 3. Illustration of the periodic learning rate. The oscillation of the learning rate helps parameters break free of the start point. Teachers are sampled at $\gamma = 0$.

parameters W will be pushed away from the low-loss region around W_{t-1}. When γ decreases, the parameters W will fall back to the low-loss region again. The larger the γ_{max} is, the more oscillating of the optimization trajectory and possibly enhances the diversity between teachers. Finally, the "PFT" method samples the parameters at $\gamma = 0$ as teachers, denoted as $\mathcal{F}_{t-1,1}, \mathcal{F}_{t-1,2} \ldots, \mathcal{F}_{t-1,n}$. We take PODNet [8] as the baseline and use each method to generate two additional teachers (i.e., three teachers). The knowledge of teachers is integrated by averaging their output logits, i.e.,

$$\overline{\mathcal{F}}_{t-1}(\boldsymbol{x}) = \frac{1}{n}\sum_{i=1}^{n}\mathcal{F}_{t-1,i}(\boldsymbol{x}), \tag{4}$$

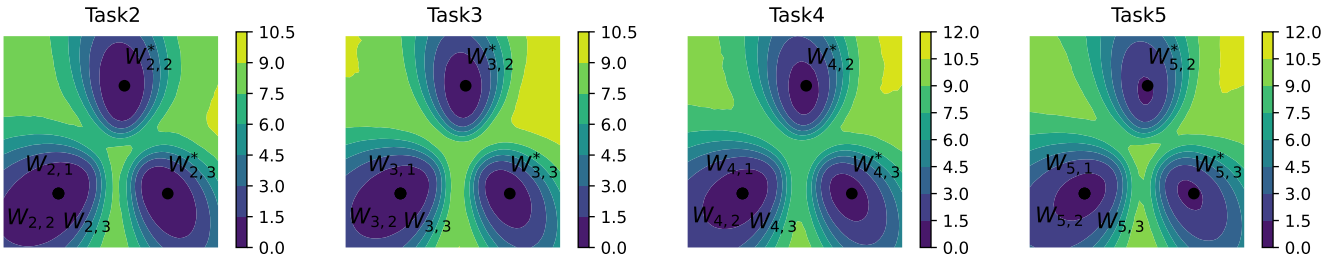

Figure 4. The loss landscape on the subspace spanned by parameters found by "Oracle", i.e., $W_{t,1}$, $W^*_{t,2}$, and $W^*_{t,3}$. The parameters $W_{t,2}$ and $W_{t,3}$ found by "PFT" are further projected to the subspace. The loss of each point in the subspace is evaluated on t learned tasks. We detail the loss landscape drawing in Section 8 of supplementary.

where n is the number of teachers. The knowledge is transferred to the student model by applying Equation (1) and replacing $\mathcal{F}_{t-1}$ with $\overline{\mathcal{F}}_{t-1}$.

3.2. Effectiveness Analysis

Figure 2 shows the results of these two methods adapting to PODNet on CIFAR-100 for 5 steps of incremental learning (i.e., B50-5S setting, Section 5). Since the knowledge of teachers is transferred through the output logits, we conduct effectiveness analysis from two aspects: the quality and diversity of teacher predictions. The prediction quality of teacher i is measured by the average accuracy on t learned tasks, i.e., $A_{i,t} = |S_{i,t}|/K_t$, where $S_{i,t}$ is the set of correctly predicted testing samples, and K_t is the number of testing samples in t learned tasks. To measure diversity, we define the prediction difference matrix $D_t = (d_{ij})_{n\times n}$,

$$d_{ij} = \frac{|S_{i,t}| - |S_{i,t} \cap S_{j,t}|}{t}, \tag{5}$$

where n is the number of teachers, t is the number of learned tasks. Different prediction mechanisms will have different responses to the same inputs, and the larger d_{ij} is, the more diversity that teacher i has compared with teacher j. To simplify the analysis, we report the average accuracy and average difference matrix of teachers during CIL,

$$\mathcal{TA} = \frac{1}{n}\sum_{i=1}^{n}\frac{1}{T-2}\sum_{t=2}^{T-1}A_{i,t}, \quad D = \frac{1}{T-2}\sum_{t=2}^{T-1}D_t, \tag{6}$$

where T is the total number of tasks, the reason why the sum range is from 2 to $T-1$ is that teachers are generated only after learning these tasks. Figure 5 shows $\mathcal{TA}$ and D of teachers generated by "PFT" and "Oracle". By comparing Figures 2 and 5, we can conclude that:

- When γ_{max} increases from 10^{-5} to 10^{-2}, the diversity between teachers also increases and the performance of baseline method PODNet also gets improved. This indicates that *improving diversity between teachers can improve CIL performance*.

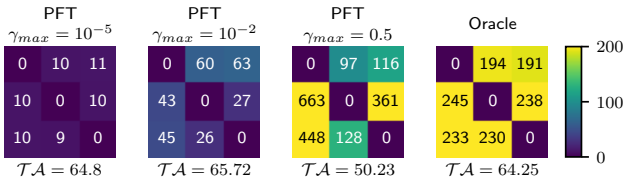

Figure 5. The average accuracy $\mathcal{TA}$ and average difference matrix D of teachers found by "Oracle" and "PFT" with different γ_{max}.

- When γ_{max} further increases from 10^{-2} to 0.5, the diversity between teachers further increases, but the CIL performance of baseline decreases. This is because the accuracy of teachers (i.e., $\mathcal{TA}$) has decreased significantly. Teachers obtained by "Oracle" have high diversity and accuracy, resulting in higher CIL performance. Therefore, *the prediction quality of teachers should be maintained while improving diversity*.

3.3. Properties of Oracle

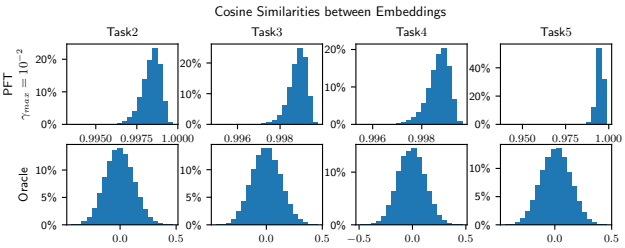

Figure 6. Cosine similarities between embeddings of teachers found by "PFT" and "Oracle". The horizontal axis is the value of similarity, and the vertical axis is the percentage of embeddings.

To find diverse teachers, we study the properties of teachers obtained by "Oracle" from two perspectives. *From the parameter space*, we visualize the loss landscape on the subspace spanned by parameters found by "Oracle" and project parameters found by "PFT" to the subspace to discover the spatial positional relationship between them, Figure 4 shows these loss landscapes from task 2 to task 5. *From the feature space*, we calculate the cosine similarities

between embeddings of two teachers found by "Oracle", and the similarities between embeddings of two teachers found by "PFT" to discover the special feature relationship between them, Figure 6 shows the results of statistics. We can conclude that:

- The parameters found by "PFT" almost overlap together, and *the parameters found by "Oracle" are in different low-loss regions, which are blocked by high-loss ridges.*
- The embeddings of teachers found by "PFT" have high consistency in direction in feature space, and *the embeddings of teachers found by "Oracle" tend to be mutually orthogonal.*

In the next section, we will adopt corresponding techniques based on the properties of "Oracle" to find diverse teachers. We also provide the cosine similarities between intermediate features of teachers in Section 9 of supplementary.

4. Finding Diverse Teachers via MTD

To reduce time and memory consumption, teachers in our proposed MTD method are represented as small prediction branches and most of the feature extraction layers are shared. Take the ResNet-32 model [14] as an example, we first copy the last two stages of layers and the classifier in the original model as the branch of a teacher and use the feature outputted by the third to last stage as input to the branch. To find diverse and high-quality teachers, we apply the following three techniques and an optimization strategy to branches.

4.1. Weight Permutation

As illustrated in Figure 4, parameters found by "Oracle" are in different low-loss regions and blocked by high-loss ridges [47]. This can naturally be achieved by weight permutation. Weight permutation can teleport parameters from one low-loss region to another low-loss region, the teleported parameters and the original parameters are equivalent to each other in terms of output [10, 43]. Denote the initial branch model as $\mathcal{B}$ containing M layers and a specific classifier, which are copies of the last M layers and the classifier in the original model $\mathcal{F}$, i.e., $\mathcal{B} = G \circ F_L \circ \cdots \circ F_{L-M+1}$. The calculations for consecutive layer l and layer $l+1$ in $\mathcal{B}$ can be generally formalized as follows [40],

$$\boldsymbol{x}_{l+1} = F_{l+1} \circ F_l(\boldsymbol{x}_{l-1}) = \sigma[\boldsymbol{W}_{l+1}\sigma(\boldsymbol{W}_l\boldsymbol{x}_{l-1})], \quad (7)$$

where $\boldsymbol{x}_{l-1}$ is the output of layer $l-1$, σ is an element-wise nonlinear activation function, $\boldsymbol{W}_l$ and $\boldsymbol{W}_{l+1}$ are the parameters of F_l and F_{l+1}, respectively. The input of the branch model is $\boldsymbol{x}_{L-M}$, which is bridged from output of the $(L-M)$-th layer in original model, i.e.,

$$\boldsymbol{x}_{L-M} = F_{L-M} \circ \cdots \circ F_1(\boldsymbol{x}). \quad (8)$$

We omit the bias term in layers and the task identity subscript t of parameters for a brief description.

Weight permutation reorders the positions of parameters, which can implemented by the permutation matrix. The permutation matrix $\boldsymbol{P}_l$ for $\boldsymbol{W}_l \in \mathbb{R}^{m\times d}$ is obtained by permuting the columns of the identity matrix $\boldsymbol{I} \in \mathbb{R}^{m\times m}$, and the set of permutation matrices for layer l is formalized as,

$$\begin{aligned}\Pi_l = \{\boldsymbol{P} = (p_{ij})_{m\times m} | p_{ij} \in \{0,1\},\\ \boldsymbol{P}\mathbf{1}_m = \boldsymbol{P}^{\mathrm{T}}\mathbf{1}_m = \mathbf{1}_m\},\end{aligned} \quad (9)$$

where $\mathbf{1}_m \in \mathbb{R}^{m\times 1}$ is the all-ones vector and $|\Pi_l| = m!$. A permutation matrix is also an orthogonal matrix, i.e.,

$$\boldsymbol{P}^{\mathrm{T}}\boldsymbol{P} = \boldsymbol{P}\boldsymbol{P}^{\mathrm{T}} = \boldsymbol{I}. \quad (10)$$

To apply the permutation matrix $\boldsymbol{P}_l$ to parameters $\boldsymbol{W}_l$ and keep the output invariant, we can reformulate Equation (7) as follows,

$$\begin{aligned}\boldsymbol{x}_{l+1} &= \sigma[\boldsymbol{W}_{l+1}\sigma(\boldsymbol{P}_l^{\mathrm{T}}\boldsymbol{P}_l\boldsymbol{W}_l\boldsymbol{x}_{l-1})]\\ &= \sigma[\underline{\boldsymbol{W}_{l+1}\boldsymbol{P}_l^{\mathrm{T}}}\sigma(\underline{\boldsymbol{P}_l\boldsymbol{W}_l}\boldsymbol{x}_{l-1})].\end{aligned} \quad (11)$$

Considering there are also permutations in layer $l-1$ and layer $l+1$, the parameters of layer l and layer $l+1$ after applying permutation to layer l are changed to,

$$\boldsymbol{W}'_{l+1} = \boldsymbol{P}_{l+1}\underline{\boldsymbol{W}_{l+1}\boldsymbol{P}_l^{\mathrm{T}}}, \quad \boldsymbol{W}'_l = \underline{\boldsymbol{P}_l\boldsymbol{W}_l}\boldsymbol{P}_{l-1}^{\mathrm{T}}. \quad (12)$$

The permuted branch $\mathcal{B}' = G \circ F'_L \circ \cdots \circ F'_{L-M+1}$ is obtained by permuting M layers in $\mathcal{B}$ (except the classifier G), where the permutation matrix $\boldsymbol{P}_l$ of layer l is randomly chosen from Π_l and $l \in \{L-M+1, \ldots, L-1\}$. If we need to find n teachers, we can randomly permute the initial branch $\mathcal{B}$ for $n-1$ times to obtain different permuted branches (we already have the original model as a teacher). Finally, these $n-1$ branches are equivalent to the initial branch, i.e., $\mathcal{B}'_i(\boldsymbol{x}_{L-M}) = \mathcal{B}(\boldsymbol{x}_{L-M}), \forall \boldsymbol{x}_{L-M}, i \in \{2\ldots, n\}$, where $i=1$ is reserved for the original model.

4.2. Feature Perturbation

Although weight permutation teleports parameters to another low-loss region, the equivalence between the teleported parameters and the original parameters conflicts with our intended diversity. To break equivalence, the optimization trajectories of different branches should be different, and the most direct and simplest way is to make branches have different inputs. To avoid repetitively extracting features from different original inputs $\boldsymbol{x}$, we perturb the input (the bridged features from the original model) of each branch, i.e.,

$$\boldsymbol{x}_{i,L-M} = \boldsymbol{x}_{L-M} + \alpha\boldsymbol{\delta}_i, \quad \boldsymbol{\delta}_i \sim \mathcal{N}(\mathbf{0}, \boldsymbol{I}), \quad (13)$$

where $\mathcal{N}(\mathbf{0}, \boldsymbol{I})$ is the normal distribution, α is the scaling factor, and $i \in \{2, \ldots, n\}$. We also considered the mixup of features, however, it is susceptible to feature degradation, especially for a long sequence of incremental learning.

4.3. Diversity Regularization

As illustrated in Figure 6, the embeddings of diverse teachers found by "Oracle" tend to be mutually orthogonal. This phenomenon is consistent with the definition of diversity, diverse teachers should have different responses to the same inputs as much as possible. To improve the diversity between teachers, we try to minimize the absolute cosine similarities between embeddings of teachers. Denote the embedding of teacher i as $\boldsymbol{x}_{i,L}$, which is the output of the penultimate layer in permuted branch $\mathcal{B}'_i$ for the perturbed input $\boldsymbol{x}_{i,L-M}$, i.e.,

$$\boldsymbol{x}_{i,L} = F'_{i,L} \circ \cdots \circ F'_{i,L-M+1}(\boldsymbol{x}_{i,L-M}). \tag{14}$$

The diversity regularization loss $\mathcal{L}_{dr}$ is defined as,

$$\mathcal{L}_{dr} = \frac{1}{C_n^2} \sum_{i=1}^{n-1} \sum_{j=i+1}^{n} \frac{|\boldsymbol{x}_{i,L}^{\mathrm{T}} \boldsymbol{x}_{j,L}|}{\|\boldsymbol{x}_{i,L}\| \|\boldsymbol{x}_{j,L}\|}, \tag{15}$$

where C_n^2 is the number of 2-combinations for n teachers. Please note that $\boldsymbol{x}_{1,L} = F_L \circ \cdots \circ F_1(\boldsymbol{x})$ is the embedding of the original model without feature perturbation.

4.4. Adaptation and Optimization

To seamlessly adapt to existing CIL methods, we embed the optimization of branches for finding teachers in the phase of class-balanced finetuning [8, 19], i.e., in addition to finetuning the classifier G_t of the original model $\mathcal{F}_t$ on the accumulated memory $\mathcal{M}_t$, we also optimize the parameters of the permuted branches $\mathcal{B}'_{t,i}, i \in \{2, \ldots, n\}$ on $\mathcal{M}_t$. As concluded in Section 3.2, the prediction quality of teachers should be maintained while improving diversity, we apply two distillation losses for branches. Denote the average output logits of branches and the original model as

$$\overline{\mathcal{B}}_t(\boldsymbol{x}) = \frac{1}{n}[\mathcal{F}_t(\boldsymbol{x}) + \sum_{i=2}^{n} \mathcal{B}'_{t,i}(\boldsymbol{x}_{i,L-M})]. \tag{16}$$

The first loss $\mathcal{L}_p$ aims to transfer knowledge from the previous teachers to the current teachers,

$$\mathcal{L}_p = \mathrm{KL}(\overline{\mathcal{B}}_{t-1}^{1:c_{t-1}}(\boldsymbol{x}) \| \overline{\mathcal{B}}_t^{1:c_{t-1}}(\boldsymbol{x})), \tag{17}$$

where c_{t-1} is the number of learned classes after task $t-1$. The second loss $\mathcal{L}_c$ aims to transfer knowledge from the current model to the current teachers,

$$\mathcal{L}_c = \mathrm{KL}(\hat{\mathcal{F}}_t^{c_{t-1}+1:c_t}(\boldsymbol{x}) \| \overline{\mathcal{B}}_t^{c_{t-1}+1:c_t}(\boldsymbol{x})), \tag{18}$$

where $\hat{\mathcal{F}}_t$ is the original model before finetuning. In addition, the classification loss $\mathcal{L}_{cs}$ of branches and the original model is denoted as,

$$\mathcal{L}_{cs} = \mathrm{CE}(\mathcal{F}_t(\boldsymbol{x}), y) + \sum_{i=2}^{n} \mathrm{CE}(\mathcal{B}'_{t,i}(\boldsymbol{x}_{i,L-M}), y), \tag{19}$$

where CE is the cross entropy loss [17, 37, 52], which can be replaced by the NCA loss [8, 19]. Finally, the total loss in the phase of class-balanced finetuning is,

$$\mathcal{L}_{total} = \mathcal{L}_{cs} + \beta \mathcal{L}_p + \eta \mathcal{L}_c + \rho \mathcal{L}_{dr}, \tag{20}$$

where β and η are coefficients for balancing between previous and current knowledge, ρ is the coefficient for diversity regularization. After finding multiple teachers, for adapting MTD to the learning of task t, we apply Equation (1) and replace $\mathcal{F}_{t-1}$ with $\overline{\mathcal{B}}_{t-1}$ without feature perturbation, i.e., the input of each branch in Equation (16) is $\boldsymbol{x}_{L-M}$.

5. Experiments

In this section, we adapt existing CIL distillation strategies with MTD for comparison experiments and conduct analytical experiments to study the effectiveness and properties of each technique in MTD. Next, we introduce the basic settings of experiments.

Benchmarks. There are three datasets used to construct benchmarks: **1) CIFAR-100** contains 100 classes and each class has 500 training samples and 100 testing samples with the image size 32×32 [21]. **2) ImageNet-1000** contains 1000 classes and each class has about 1500 training samples and 50 validation samples [7]. **3) ImageNet-100** is built by selecting 100 classes from ImageNet-1K according to a fixed random seed 1993. We apply two settings for constructing benchmarks. First, the dataset is divided into two parts, each part contains half of the classes, one part is taken as the first (basic) task, and the other part is further equally divided into S-step tasks, i.e., the total number of tasks is $T = N + 1$, where $N \in \{5, 10, 25\}$. Under this setting, we denote CIFAR100 for 10-step tasks as "B50-10S". Second, the dataset is directly and equally divided into T tasks, where $T \in \{5, 10, 20\}$. Under this setting, we denote CIFAR-100 for 10 tasks as "B0-10T".

Evaluation Metric. Same as existing work [8, 17, 19, 37], we adopt the average incremental accuracy to evaluate the performance of each CIL method. The definition is,

$$\mathcal{A} = \frac{1}{T} \sum_{t=1}^{T} A_t, \tag{21}$$

where A_t is the testing accuracy on all learned tasks after learning task t. A better method should have a higher $\mathcal{A}$.

We describe the implementation details, including model architecture, and hyperparameter settings in Section 10 of supplementary. It should be emphasized that we use MTD to obtain only one additional teacher, which is a small branch compared with the original model, in all comparison experiments, i.e., the total number of teachers is $n = 2$.

5.1. Comparison Experiments

We choose PODNet [8] and AFC [19] as the adaptation methods for MTD. We report two types of results, "MTD-

Method	CIFAR-100			ImageNet-100			ImageNet-1000		
$\mathcal{A}(\%)\uparrow$	S=5	10	25	5	10	25	5	10	25
BiC† [52]	59.36	54.20	50.00	70.07	64.96	57.73	62.65	58.72	53.47
Mnemonics† [26]	63.34	62.28	60.96	72.58	71.37	69.74	64.63	63.01	61.00
GeoDL† [42]	65.14	65.03	63.12	73.87	73.55	71.72	65.23	64.46	62.20
iCaRL [37]	57.83(±0.10)	52.63(±0.10)	49.02(±0.13)	64.75	58.80	52.46	51.60	47.42	41.03
LUCIR [17]	63.47(±0.34)	60.75(±0.29)	57.79(±0.34)	71.93	69.43	63.52	66.13	61.63	54.05
AANet [27]	65.97(±0.41)	64.08(±0.44)	60.44(±0.45)	77.98	74.70	68.65	68.87	65.65	60.07
PODNet [8]	65.07(±0.28)	62.93(±0.14)	59.45(±0.28)	76.32	73.54	63.05	68.33	65.35	58.62
w/ MTD-S	66.55(±0.16)	64.18(±0.30)	59.61(±0.14)	77.75	74.66	67.38	69.20	66.63	61.40
w/ MTD-T	**67.64**(±0.21)	65.58(±0.20)	60.94(±0.22)	**78.47**	75.43	68.54	69.65	67.21	62.19
AFC [19]	65.94(±0.08)	64.29(±0.31)	62.33(±0.35)	77.27	75.47	72.41	69.07	66.85	63.40
w/ MTD-S	66.92(±0.06)	65.38(±0.10)	62.74(±0.26)	77.82	**76.26**	**73.73**	69.62	67.42	64.20
w/ MTD-T	67.40(±0.02)	**65.69**(±0.14)	**62.81**(±0.18)	77.85	76.13	72.92	**70.40**	**68.15**	**65.01**

Table 1. Comparison results on CIFAR-100 B50, ImageNet-100 B50, and ImageNet-1000 B500. "MTD-S" denotes the student $\mathcal{F}_t$. "MTD-T" denotes teachers $\overline{\mathcal{B}}_t$. † denotes the results are referenced from [42]. The experiments on CIFAR-100 are run with 3 random seeds and results are reported with mean and standard deviation. Same as existing work, experiments on ImageNet are run with a fixed seed 1993.

S" is the results that the student achieves, i.e., the original model $\mathcal{F}_t$, and "MTD-T" is the results that teachers achieve, i.e., the average logits of teachers $\overline{\mathcal{B}}_t$ (Equation 16). Tables 1 and 2 show the comparison results on benchmarks with different settings.

Results on CIFAR-100. We apply "B50" and "B0" settings to CIFAR-100 benchmarks. Table 1 shows the results under the "B50" setting. It can be seen that MTD can consistently improve PODNet and AFC for different steps of incremental learning. For example, MTD-T can improve PODNet by 2.57%, 2.65%, and 1.49% for 5, 10, and 25 steps respectively, and improve AFC by 1.46%, 1.4%, and 0.48% for 5, 10, and 25 steps respectively. Table 2 shows the results under the "B0" setting. Compared with more stability required in the setting of "B50", "B0" requires more plasticity [28]. MTD can still improve PODNet and AFC in this setting. For example, MTD-T improves AFC by 1.15%, 0.74%, and 0.68% for 5, 10, and 25 steps respectively.

Results on ImageNet. Table 1 shows the results on ImageNet-100 under the "B50" setting and ImageNet-1000 under the "B500" setting. It can be seen that MTD-T can improve PODNet by 2.15%, 1.89%, and 5.49% for 5, 10, and 25 steps respectively, and MTD-S improves AFC by 0.55%, 0.79%, and 1.32% for 5, 10, and 25 steps respectively. When "AFC w/ MTD" learning tasks for 10 and 25 steps, the student exceeds the teachers, which indicates that the ensemble of teachers does not always achieve better performance. The results on ImageNet-1000 show that MTD-T can improve PODNet by 1.32%, 1.86%, and 3.57% for 5, 10, and 25 steps respectively, and improve AFC by 1.33%, 1.3%, and 1.61% for 5, 10, and 25 steps respectively.

We compare the existing dual-teacher distillation method DT-CIL [4] in section 11.1 of supplementary.

Method	CIFAR-100		
$\mathcal{A}(\%)\uparrow$	T=5	10	20
iCaRL [37]	56.10(±0.07)	51.15(±0.19)	47.81(±0.07)
LUCIR [17]	62.95(±0.16)	56.54(±0.12)	50.17(±0.64)
AANet [27]	63.95(±0.22)	55.37(±0.21)	48.00(±0.41)
PODNet [8]	63.43(±0.21)	55.59(±0.06)	48.20(±0.22)
w/ MTD-S	64.60(±0.08)	57.48(±0.25)	48.92(±0.28)
w/ MTD-T	65.11(±0.07)	57.94(±0.48)	49.92(±0.40)
AFC [19]	64.19(±0.25)	57.50(±0.52)	50.41(±0.31)
w/ MTD-S	64.79(±0.13)	58.01(±0.24)	50.71(±0.37)
w/ MTD-T	**65.34**(±0.13)	**58.24**(±0.14)	**51.09**(±0.32)

Table 2. Comparison results on CIFAR-100 B0. Experiments are run 3 times and reported with mean and standard deviation.

5.2. Analytical Experiments

Ablative Studies. There are three techniques in MTD, weight permutation, feature perturbation, and diversity regularization. We take PODNet [8] as the baseline and gradually add each technique to the baseline to study the effectiveness. Table 3 shows the ablative results on CIFAR-100 B50. It can be seen that feature perturbation and diversity regularization can consistently improve performance under different settings. Weight permutation brings certain improvements, but relatively small. We think that this is because our permutation matrices are random, although parameters found by "Oracle" are blocked by the high-loss ridge, and random permutation can also separate parameters by the high-loss ridge (please see Figure 7), but random permutation may not necessarily be consistent with the separation pattern of parameters found by "Oracle".

Effects of the Number of Teachers. More teachers may bring more diverse knowledge. We increase the number of

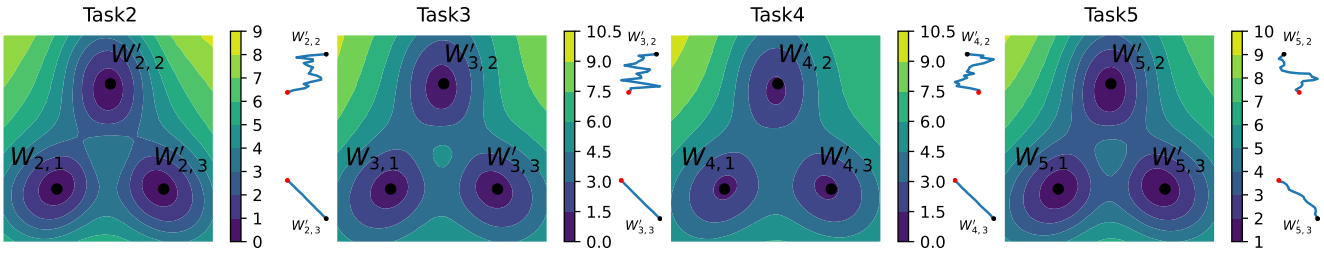

Figure 7. The loss landscape on the subspace spanned by parameters generated by weight permutation, i.e, $W_{t,1}$, $W'_{t,2}$, and $W'_{t,3}$, where $W_{t,1}$ is the original parameters, $W'_{t,2}$ and $W'_{t,3}$ are permuted parameters. These three parameters are blocked by the high-loss ridges and equivalent to each other in terms of output. To break equivalence, we apply the feature perturbation and diversity regularization to branches ($W'_{t,2}$ and $W'_{t,3}$) to make their trajectories different each other for improving diversity between them. On the right of each subfigure, we show the optimization trajectories of permuted parameters in the subspace, the upper and lower trajectories tend to be different. The black point is the start point of the trajectory, and the red point is the endpoint (i.e., the final teacher) of the trajectory.

			CIFAR-100		
FP	WP	DR	S =5	10	25
			65.07(±0.28)	62.93(±0.14)	59.45(±0.28)
✓			66.76(±0.14)	64.33(±0.11)	59.61(±1.48)
✓	✓		66.99(±0.23)	64.39(±0.29)	59.74(±0.10)
✓		✓	67.53(±0.08)	65.50(±0.34)	60.74(±1.08)
✓	✓	✓	67.64(±0.21)	65.58(±0.20)	60.94(±0.22)

Table 3. Ablative results of weight permutation (WP), feature perturbation (FP), and diversity regularization (DR) on CIFAR-100 B50 taking PODNet as baseline. We run experiments 3 times and report the results of MTD-T with mean and standard deviation.

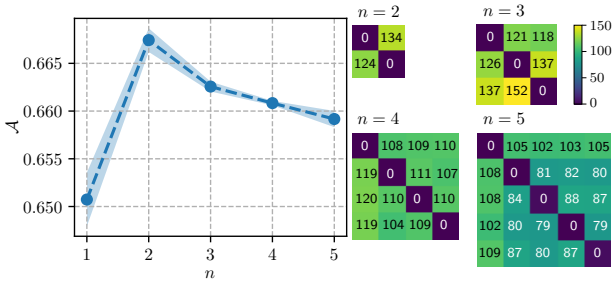

Figure 8. **Left:** Average incremental accuracies of MTD-S under different numbers of teachers. **Right:** Prediction difference matrices of teachers under different numbers of teachers.

teachers to study if CIL performance can be improved. The left of Figure 8 shows the results of MTD-S with different numbers of teachers. It can be seen that we only need to add one teacher to achieve the best performance, i.e., $n = 2$. When the number of teachers is $n > 2$, the performance gradually decreases. We think this is due to two reasons. First, the ensemble of knowledge from teachers by averaging logits is crude, and this is still a challenge for existing work to effectively ensemble knowledge from teachers [13, 50]. Therefore, as the number of teachers increases, it becomes difficult to ensemble knowledge. Second, the right of Figure 8 shows the average prediction difference matrices under different numbers of teachers. It can be seen that the diversity between teachers gradually decreases as the number of teachers increases. However, Section 3.2 shows that diversity is crucial for improving performance.

Visualization of Loss Landscape and Optimization Trajectory. To check whether the parameters are in different low-loss regions before and after permutation, and whether the optimization trajectories between teachers are different, we visualize the loss landscapes and optimization trajectories of teachers in Figure 7. By comparing with Figure 4, we can find that MTD mimics the properties of "Oracle" in parameter space. In addition, by relating Figure 7 to the difference matrix when $n = 3$ in the right of Figure 8, we can find that optimizing along different trajectories in parameter space results in different prediction mechanisms.

6. Conclusion

In this paper, we investigated the multi-teacher distillation in CIL. We found two key properties of diverse teachers: parameters are in different low-loss regions and embeddings tend to be mutually orthogonal. Then, we proposed MTD including weight permutation, feature perturbation, and diversity regularization to find diverse teachers. Finally, extensive experiments show MTD can significantly improve performance. One limitation of our method is permutation matrices are random, which may not be consistent with the real separation pattern in "Oracle". A promising approach is to find permutation matrices or the operator of teleportation in a data-driven way, we leave this in our future work.

Acknowledgement. This work was support in part by National Science and Technology Major Project (2021ZD0112001), National Natural Science Foundation of China (No. U23A20286, No. 62171111), and Project funded by China Postdoctoral Science Foundation 2023TQ0046.

References

[1] Ryan Prescott Adams and Richard S Zemel. Ranking via sinkhorn propagation. *arXiv preprint arXiv:1106.1925*, 2011. 1

[2] Hongjoon Ahn, Jihwan Kwak, Subin Lim, Hyeonsu Bang, Hyojun Kim, and Taesup Moon. Ss-il: Separated softmax for incremental learning. In *Proceedings of the IEEE/CVF International conference on computer vision*, pages 844–853, 2021. 3

[3] Samuel K Ainsworth, Jonathan Hayase, and Siddhartha Srinivasa. Git re-basin: Merging models modulo permutation symmetries. *arXiv preprint arXiv:2209.04836*, 2022. 2, 1

[4] Yoojin Choi, Mostafa El-Khamy, and Jungwon Lee. Dual-teacher class-incremental learning with data-free generative replay. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3543–3552, 2021. 1, 2, 7

[5] Mircea Cimpoi, Subhransu Maji, Iasonas Kokkinos, Sammy Mohamed, and Andrea Vedaldi. Describing textures in the wild. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3606–3613, 2014. 3

[6] Matthias Delange, Rahaf Aljundi, Marc Masana, Sarah Parisot, Xu Jia, Ales Leonardis, Greg Slabaugh, and Tinne Tuytelaars. A continual learning survey: Defying forgetting in classification tasks. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2021. 1

[7] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *2009 IEEE conference on computer vision and pattern recognition*, pages 248–255. Ieee, 2009. 6

[8] Arthur Douillard, Matthieu Cord, Charles Ollion, Thomas Robert, and Eduardo Valle. Podnet: Pooled outputs distillation for small-tasks incremental learning. In *European Conference on Computer Vision*, pages 86–102. Springer, 2020. 1, 2, 3, 6, 7

[9] Felix Draxler, Kambis Veschgini, Manfred Salmhofer, and Fred Hamprecht. Essentially no barriers in neural network energy landscape. In *International conference on machine learning*, pages 1309–1318. PMLR, 2018. 2

[10] Rahim Entezari, Hanie Sedghi, Olga Saukh, and Behnam Neyshabur. The role of permutation invariance in linear mode connectivity of neural networks. *arXiv preprint arXiv:2110.06296*, 2021. 2, 5, 1

[11] Timur Garipov, Pavel Izmailov, Dmitrii Podoprikhin, Dmitry Vetrov, and Andrew Gordon Wilson. Loss surfaces, mode connectivity, and fast ensembling of dnns. In *Proceedings of the 32nd International Conference on Neural Information Processing Systems*, pages 8803–8812, 2018. 2

[12] Jacob Goldberger, Geoffrey E Hinton, Sam Roweis, and Russ R Salakhutdinov. Neighbourhood components analysis. *Advances in neural information processing systems*, 17, 2004. 3

[13] Jianping Gou, Baosheng Yu, Stephen J Maybank, and Dacheng Tao. Knowledge distillation: A survey. *International Journal of Computer Vision*, 129:1789–1819, 2021. 1, 2, 8

[14] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016. 5, 1

[15] Xiaoxi He, Zimu Zhou, and Lothar Thiele. Multi-task zipping via layer-wise neuron sharing. *Advances in Neural Information Processing Systems*, 31, 2018. 2, 1

[16] Geoffrey Hinton, Oriol Vinyals, and Jeff Dean. Distilling the knowledge in a neural network. *arXiv preprint arXiv:1503.02531*, 2015. 1

[17] Saihui Hou, Xinyu Pan, Chen Change Loy, Zilei Wang, and Dahua Lin. Learning a unified classifier incrementally via rebalancing. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 831–839, 2019. 1, 2, 3, 6, 7

[18] Artur Ilichev, Nikita Sorokin, Irina Piontkovskaya, and Valentin Malykh. Multiple teacher distillation for robust and greener models. In *Proceedings of the International Conference on Recent Advances in Natural Language Processing (RANLP 2021)*, pages 601–610, 2021. 1, 2, 3

[19] Minsoo Kang, Jaeyoo Park, and Bohyung Han. Class-incremental learning by knowledge distillation with adaptive feature consolidation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 16071–16080, 2022. 1, 2, 3, 6, 7

[20] Sanghwan Kim, Lorenzo Noci, Antonio Orvieto, and Thomas Hofmann. Achieving a better stability-plasticity trade-off via auxiliary networks in continual learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 11930–11939, 2023. 3

[21] Alex Krizhevsky, Geoffrey Hinton, et al. Learning multiple layers of features from tiny images. *https://www.cs.toronto.edu/kriz/cifar.html*, 2009. 6

[22] Daniel Kunin, Javier Sagastuy-Brena, Surya Ganguli, Daniel LK Yamins, and Hidenori Tanaka. Neural mechanics: Symmetry and broken conservation laws in deep learning dynamics. *arXiv preprint arXiv:2012.04728*, 2020. 2

[23] Timothée Lesort, Vincenzo Lomonaco, Andrei Stoian, Davide Maltoni, David Filliat, and Natalia Díaz-Rodríguez. Continual learning for robotics: Definition, framework, learning strategies, opportunities and challenges. *Information fusion*, 58:52–68, 2020. 1

[24] Wanli Li, Tieyun Qian, Xuhui Li, and Lixin Zou. Adversarial multi-teacher distillation for semi-supervised relation extraction. *IEEE Transactions on Neural Networks and Learning Systems*, 2023. 1, 2, 3

[25] Zhizhong Li and Derek Hoiem. Learning without forgetting. *IEEE transactions on pattern analysis and machine intelligence*, 40(12):2935–2947, 2017. 1, 3

[26] Yaoyao Liu, Yuting Su, An-An Liu, Bernt Schiele, and Qianru Sun. Mnemonics training: Multi-class incremental learning without forgetting. In *Proceedings of the IEEE/CVF conference on Computer Vision and Pattern Recognition*, pages 12245–12254, 2020. 2, 3, 7

[27] Yaoyao Liu, Bernt Schiele, and Qianru Sun. Adaptive aggregation networks for class-incremental learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2544–2553, 2021. 2, 7, 1

[28] Yaoyao Liu, Yingying Li, Bernt Schiele, and Qianru Sun. Online hyperparameter optimization for class-incremental learning. *arXiv preprint arXiv:2301.05032*, 2023. 7

[29] Ekdeep Singh Lubana, Eric J Bigelow, Robert P Dick, David Krueger, and Hidenori Tanaka. Mechanistic mode connectivity. In *International Conference on Machine Learning*, pages 22965–23004. PMLR, 2023. 2

[30] Zilin Luo, Yaoyao Liu, Bernt Schiele, and Qianru Sun. Class-incremental exemplar compression for class-incremental learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 11371–11380, 2023. 2

[31] Michael McCloskey and Neal J Cohen. Catastrophic interference in connectionist networks: The sequential learning problem. In *Psychology of learning and motivation*, pages 109–165. Elsevier, 1989. 1

[32] Luong Trung Nguyen, Kwangjin Lee, and Byonghyo Shim. Stochasticity and skip connection improve knowledge transfer. In *2020 28th European Signal Processing Conference (EUSIPCO)*, pages 1537–1541. IEEE, 2021. 2, 3, 1

[33] SeongUk Park and Nojun Kwak. Feed: Feature-level ensemble for knowledge distillation. *arXiv preprint arXiv:1909.10754*, 2019. 2

[34] Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, et al. Pytorch: An imperative style, high-performance deep learning library. *Advances in neural information processing systems*, 32:8026–8037, 2019. 1

[35] Fidel A Guerrero Peña, Heitor Rapela Medeiros, Thomas Dubail, Masih Aminbeidokhti, Eric Granger, and Marco Pedersoli. Re-basin via implicit sinkhorn differentiation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 20237–20246, 2023. 2, 1

[36] Fabrizio Pittorino, Antonio Ferraro, Gabriele Perugini, Christoph Feinauer, Carlo Baldassi, and Riccardo Zecchina. Deep networks on toroids: removing symmetries reveals the structure of flat regions in the landscape geometry. In *International Conference on Machine Learning*, pages 17759–17781. PMLR, 2022. 1

[37] Sylvestre-Alvise Rebuffi, Alexander Kolesnikov, Georg Sperl, and Christoph H Lampert. icarl: Incremental classifier and representation learning. In *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*, pages 2001–2010, 2017. 1, 2, 3, 6, 7

[38] Anthony Robins. Catastrophic forgetting, rehearsal and pseudorehearsal. *Connection Science*, 7(2):123–146, 1995. 1

[39] Sebastian Ruder, Parsa Ghaffari, and John G Breslin. Knowledge adaptation: Teaching to adapt. *arXiv preprint arXiv:1702.02052*, 2017. 2

[40] Gobinda Saha, Isha Garg, and Kaushik Roy. Gradient projection memory for continual learning. *arXiv preprint arXiv:2103.09762*, 2021. 5

[41] Bharat Bhusan Sau and Vineeth N Balasubramanian. Deep model compression: Distilling knowledge from noisy teachers. *arXiv preprint arXiv:1610.09650*, 2016. 2, 3, 1

[42] Christian Simon, Piotr Koniusz, and Mehrtash Harandi. On learning the geodesic path for incremental learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1591–1600, 2021. 1, 2, 7

[43] Berfin Simsek, François Ged, Arthur Jacot, Francesco Spadaro, Clément Hongler, Wulfram Gerstner, and Johanni Brea. Geometry of the loss landscape in overparameterized neural networks: Symmetries and invariances. In *International Conference on Machine Learning*, pages 9722–9732. PMLR, 2021. 2, 5, 1

[44] Guocong Song and Wei Chai. Collaborative learning for deep neural networks. *Advances in neural information processing systems*, 31, 2018. 2, 1

[45] Gido M Van de Ven and Andreas S Tolias. Three scenarios for continual learning. *arXiv preprint arXiv:1904.07734*, 2019. 1, 3

[46] Gido M van de Ven, Tinne Tuytelaars, and Andreas S Tolias. Three types of incremental learning. *Nature Machine Intelligence*, 4(12):1185–1197, 2022. 1

[47] Eli Verwimp, Matthias De Lange, and Tinne Tuytelaars. Rehearsal revealed: The limits and merits of revisiting samples in continual learning. *arXiv preprint arXiv:2104.07446*, 2021. 2, 5

[48] Catherine Wah, Steve Branson, Peter Welinder, Pietro Perona, and Serge Belongie. The caltech-ucsd birds-200-2011 dataset. 2011. 3

[49] Ruosi Wan, Zhanxing Zhu, Xiangyu Zhang, and Jian Sun. Spherical motion dynamics: Learning dynamics of normalized neural network using sgd and weight decay. *Advances in Neural Information Processing Systems*, 34:6380–6391, 2021. 2

[50] Lin Wang and Kuk-Jin Yoon. Knowledge distillation and student-teacher learning for visual intelligence: A review and new outlooks. *IEEE transactions on pattern analysis and machine intelligence*, 44(6):3048–3068, 2021. 2, 8

[51] Meng-Chieh Wu, Ching-Te Chiu, and Kun-Hsuan Wu. Multi-teacher knowledge distillation for compressed video action recognition on deep neural networks. In *ICASSP 2019-2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 2202–2206. IEEE, 2019. 1, 2

[52] Yue Wu, Yinpeng Chen, Lijuan Wang, Yuancheng Ye, Zicheng Liu, Yandong Guo, and Yun Fu. Large scale incremental learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 374–382, 2019. 1, 2, 6, 7

[53] Shan You, Chang Xu, Chao Xu, and Dacheng Tao. Learning from multiple teacher networks. In *Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 1285–1294, 2017. 1, 2

[54] Bo Zhao, Iordan Ganev, Robin Walters, Rose Yu, and Nima Dehmamy. Symmetries, flat minima, and the conserved quantities of gradient flow. *arXiv preprint arXiv:2210.17216*, 2022. 2