Few-Shot Class-Incremental Learning via Class-Aware Bilateral Distillation

Linglan Zhao[1,*], Jing Lu[2,*], Yunlu Xu[2], Zhanzhan Cheng[2,†], Dashan Guo[1], Yi Niu[2], Xiangzhong Fang[1]
[1]Department of Electronic Engineering, Shanghai Jiao Tong University [2]Hikvision Research Institute
{llzhao,dmlab_gds,xzfang}@sjtu.edu.cn, {lujing6,xuyunlu,chengzhanzhan,niuyi}@hikvision.com

Abstract

Few-Shot Class-Incremental Learning (FSCIL) aims to continually learn novel classes based on only few training samples, which poses a more challenging task than the well-studied Class-Incremental Learning (CIL) due to data scarcity. While knowledge distillation, a prevailing technique in CIL, can alleviate the catastrophic forgetting of older classes by regularizing outputs between current and previous model, it fails to consider the overfitting risk of novel classes in FSCIL. To adapt the powerful distillation technique for FSCIL, we propose a novel distillation structure, by taking the unique challenge of overfitting into account. Concretely, we draw knowledge from two complementary teachers. One is the model trained on abundant data from base classes that carries rich general knowledge, which can be leveraged for easing the overfitting of current novel classes. The other is the updated model from last incremental session that contains the adapted knowledge of previous novel classes, which is used for alleviating their forgetting. To combine the guidances, an adaptive strategy conditioned on the class-wise semantic similarities is introduced. Besides, for better preserving base class knowledge when accommodating novel concepts, we adopt a two-branch network with an attention-based aggregation module to dynamically merge predictions from two complementary branches. Extensive experiments on 3 popular FSCIL datasets: mini-*ImageNet, CIFAR100 and CUB200 validate the effectiveness of our method by surpassing existing works by a significant margin. Code is available at https://github.com/LinglanZhao/BiDistFSCIL.*

1. Introduction

Real-world applications often face novel data in continuous stream format. In contrast, traditional models can only make predictions on a pre-defined label set, and are not flexible enough to tackle novel classes which may emerge after deployment. To address this issue, Class-Incremental Learning (CIL) has become an active area of recent research [2, 13, 20, 26]. The main focus of CIL is to effectively learn new concepts from abundant labeled samples and to simultaneously alleviate catastrophic forgetting over old classes. However, the requirement of sufficient training data from novel classes still makes CIL impractical in many scenarios, especially when annotated samples are hard to obtain due to privacy or the unaffordable collecting cost. For instance, to train an incremental model for face recognition, one or only few images are uploaded for recognizing the newly occurred person. To this end, Few-Shot Class-Incremental Learning (FSCIL) is proposed to learn novel concepts given only a few samples [30]. FSCIL defines a challenging task where abundant training samples are available only in the base session for initial model pre-training and the model should continually absorb novel concepts from few data points in each incremental session.

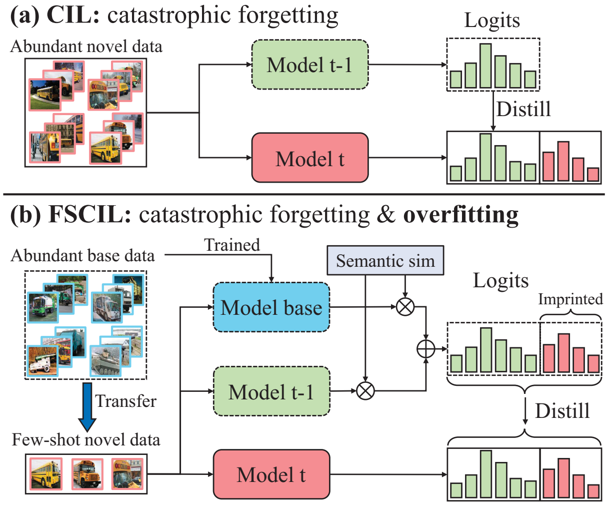


Figure 1. Comparisons of (a) vanilla knowledge distillation in CIL and (b) our adapted class-aware bilateral distillation for FSCIL.

A prevailing technique in CIL is to leverage knowledge distillation for alleviating the forgetting problem. The general routine is to calibrate the output logits between current and previous model, as illustrated in Fig. 1 (a). The output of current model t is restrained to be consistent with the output of model t-1 in last incremental session. Nevertheless, such paradigm is not suitable for FSCIL [30, 41, 42], since the scarcity of novel class samples will cause model

[*]Equal contribution. [†]Corresponding author.

t-1 severely overfitting to classes which occurred in that session (*i.e.*, session t-1), making the model lack generalization ability, which further leads to the biased incremental learning in current session t.

Therefore, to adapt the powerful distillation technique for the challenging FSCIL task, we are devoted to designing a new distillation structure that can simultaneously handle the forgetting and overfitting challenges. To this end, we propose the class-aware bilateral distillation module, by adaptively drawing knowledge from two complementary teachers. One of them is the base model trained on abundant data from base classes. By distilling from the base model, we transfer the rich general knowledge learned from base classes to the few-shot novel classes, hence easing their overfitting. The other teacher is the updated model in the last session t-1, which carries the adapted knowledge of previously seen novel classes (from session 1 to t-1), we can prevent the knowledge from forgetting by distilling from model t-1. Moreover, a class-aware coefficient is learned to dynamically merge the above two guidance by considering class-aware semantic similarities between novel and base classes as priors. Intuitively, the more similar between base classes and a novel category, the more knowledge from base classes can be leveraged for alleviating the overfitting. As presented in Fig. 1 (b), instead of solely utilizing last session's model t-1 for guiding the novel class adaptation, we selectively merge the output logits from both model t-1 and the base model as the guidance for distillation.

For further preserving base class knowledge when adapting to novel classes, an attention-based aggregation module is proposed to automatically combine predictions from the base model and the current model t. Considering that the lower layers of a convolutional neural network capture fundamental visual patterns [35], we set these layers shared and integrate the above models into a unified framework. For clarity, we also refer to the base and the current model as the base and novel branch, respectively. The two branches can be viewed as two individual experts for handling samples from different categories. For a test sample from base classes, the aggregation module will pay more attention to predictions from base branch since it specializes in base classes without forgetting. In contrast, the focus will be moved to novel branch when evaluated on novel class test samples, because novel branch is well adapted to those incremental classes. Our contributions are three-fold:

- To adapt the prevailing distillation technique for addressing the unique overfitting challenges posed by FSCIL, we propose a class-aware bilateral distillation method by adaptively drawing knowledge from two complementary teachers, which proves to be effective both in reducing the overfitting risk and preventing the aggravated catastrophic forgetting.
- We propose a two-branch network where the two branches are well associated by the class-aware bilateral distillation and attention-based aggregation module. The framework can simultaneously accommodate novel concepts and retain base knowledge, without sophisticated meta-training and can be conveniently applied to arbitrary pre-trained models, making it more practical in real-world applications.
- The superiority of our approach is validated on three public FSCIL datasets: *mini*-ImageNet, CIFAR100, and CUB200 by achieving remarkable state-of-the-art performance. For example, we surpass the second best result on *mini*-ImageNet over 3%.

2. Related work

2.1. Few-Shot Learning

Few-Shot Learning (FSL) aims to learn novel categories from scarce training examples. Previous FSL works can be divided into four categories. *Metric learning based* works [29, 31, 32, 37] attempt to learn appropriate distance metric between query (test) and support (training) samples. *Initialization based* FSL methods [9, 27] are proposed to learn good initialization of the model. Moreover, *weight generation based* methods [10, 25] directly generate classification weights for new classes to alleviate overfitting. In addition, *hallucination based* approaches [19, 34] train a generation network for data augmentation. However, FSL focuses solely on recognizing few novel classes, while ignoring the ability to handle the previously learned ones.

2.2. Class-Incremental Learning

It is a long-standing challenge in machine learning to learn novel concepts while preserving previous knowledge. To this end, Class-Incremental Learning (CIL) is proposed to learn new classes sequentially without forgetting the old ones. One line of CIL works focus on powerful regularization on network weights [16, 38] or predictions [12, 20, 36] to minimize the change between the current model and the previous one. Another line of CIL methods reveal that storing a small number of representative samples from old classes as an exemplar set for rehearsal [2, 13, 26] is helpful when learning novel concepts. Moreover, CIL works such as [1, 22] dynamically expand models to accommodate new classes. Nevertheless, CIL usually requires abundant novel class training samples which makes it unsuitable for many realistic applications such as incremental face recognition.

2.3. Few-Shot Class-Incremental Learning

Few-Shot Class-Incremental Learning (FSCIL) simultaneously takes into account the challenges from the above FSL and CIL. Concretely, FSCIL aims at incrementally learning from very few novel samples while preserving already learned knowledge. TOPIC [30] firstly defines the setting of FSCIL and adopts a neural gas for topology preservation in the embedding space. Following up works [3, 8, 23] modify existing approaches from CIL for undertaking FSCIL. Besides, [4, 5] utilize word vectors for alleviating the intrinsic difficulty of data scarcity in FSCIL. Another line of prevailing methods [6, 28, 39, 41–43] focus on meta-training on base class data by sampling a number of fake incremental episodes for test scenario simulation. Nevertheless, this relies on extra meta-training phases to prepare the model for future tasks, which is impractical in many real-world scenarios and limits its application to arbitrary pre-trained models. Moreover, most of these works [39, 41–43] freeze the parameters of the meta-trained model for explicit base knowledge preservation while sacrificing the plasticity of the model for novel concepts. Unlike the meta-training strategies, our adapted distillation method can be conveniently applied to any pre-trained models without sophisticated meta-training and preserve model's plasticity for adapting to novel knowledge.

3. Problem Setting

The aim of FSCIL [28, 30, 39] is to accommodate new knowledge from few novel class training samples and resist forgetting previously learned old classes. Formally, the model is trained on a series of incremental sessions $\left\{\mathcal{D}^0, \mathcal{D}^1, \cdots, \mathcal{D}^t\right\}$ where $\mathcal{D}^t = \{(\mathbf{x}_i, y_i)\}_i$ is the training set from session t and $\mathbf{x}_i$ is a sample from class $y_i \in \mathcal{C}^t$. The label space of dataset $\mathcal{D}^t$ is denoted by $\mathcal{C}^t$, which is mutually disjoint between different sessions, *i.e.* $\forall i, j$ and $i \neq j, \mathcal{C}^i \cap \mathcal{C}^j = \varnothing$. Following standard incremental learning paradigm, a model in each session t can only access $\mathcal{D}^t$ and an optional exemplar set $\mathcal{M}$ consisting of a small number of stored samples from the earlier sessions. Usually, the training set $\mathcal{D}^0$ in the base session contains a sufficient volume of data for base classes $\mathcal{C}^0$. In contrast, the training sets $\mathcal{D}^t$ $(t \geq 1)$ in the following sessions contains few training samples, which is also termed as a N-way K-shot support set, comprising of only K examples for each of the N categories from $\mathcal{C}^t$. Once the incremental learning in session t is finalized, the model is tested on query samples from all the seen classes so far: $\tilde{\mathcal{C}}^t = \mathcal{C}^0 \cup \mathcal{C}^1 \cdots \cup \mathcal{C}^t$.

4. Methodology

4.1. Overall Architecture

Considering the stability and plasticity dilemma in FSCIL with severe data imbalance between base and novel classes, our method adopts a two-branch architecture shown in Fig. 2 (a). For preventing base class knowledge from forgetting, the base branch is trained on the base training set $\mathcal{D}^0$ with abundant samples as in standard supervised learning in the base session. In each incremental session, feature extractor of base branch is frozen and its classification weights are expanded using mean feature embeddings [25] of the corresponding novel class training samples. To compensate for the lack of plasticity in base branch, novel branch can effectively adapt to new classes with learnable parameters. Moreover, for reducing computational complexity and mitigating overfitting, the feature extractors of base and novel branches have all the layers shared except the last residual layer [11]. Detailed discussions of the shared and learnable layers are also provided in Section 5.

Particularly, a feature extractor $f_\phi(\mathbf{x}) \in \mathbb{R}^d$ outputs d-dimensional features. Its parameters ϕ can be decomposed into $\phi = \{\psi, \theta\}$ where ψ and θ denote the parameters of shared layers and the last residual layer, respectively. Different from θ_b of the base branch which is frozen, θ_n^t of the novel branch, which is initialized with θ_b in the first incremental session, is learnable in each session. Hereinafter, we use superscripts $(0, 1, ..., t)$ for representing sessions and subscripts (b, n) to distinguish two branches. For example, we denote feature extractors of base and novel branch in session t as f_{ϕ_b} and $f_{\phi_n^t}$, respectively[1].

In session t, the novel branch is distilled under the combined guidance from base branch for transferring generalizable knowledge, and that from novel branch in last session $(t-1)$ for inheriting previously absorbed knowledge. Finally, an attention-based aggregation is used to merge complementary predictions from two branches. We first elaborate on the two key components and then discuss the flexibility of our method with none or a few exemplars available.

4.2. Class-Aware Bilateral Distillation

In each session t, for both the base and novel branches to further handle novel classes, corresponding classification weights $\{W_b^{t-1}, W_n^{t-1}\}$ in session $t-1$ are expanded to $\{\hat{W}_b^{t-1}, \hat{W}_n^{t-1}\}$ (shape $\mathbb{R}^{d\times|\tilde{\mathcal{C}}^{t-1}|} \rightarrow \mathbb{R}^{d\times|\tilde{\mathcal{C}}^t|}$) based on $\mathcal{D}^t$ shown in Fig. 2 (b). Concretely, the imprinted weights [25] for newly occurred classes in session t are calculated using centroids of feature embeddings from training samples with the same class labels:

$$\boldsymbol{w}_c = \frac{1}{N_c} \sum_{(\mathbf{x}_i, y_i)\in\mathcal{D}^t} \mathbf{I}\left[y_i = c\right] f_\phi\left(\mathbf{x}_i\right), \tag{1}$$

where $\boldsymbol{w}_c$ is the prototype [29] of class c, $\mathbf{I}$ denotes an indicator function and N_c is the sample number of class c in $\mathcal{D}^t$. f_ϕ refers to the fixed f_{ϕ_b} when expanding W_b^{t-1}, or $f_{\phi_n^{t-1}}$ when expanding W_n^{t-1}, respectively. Then base branch classification weights (W_b^t) are replaced by $\hat{W}_b^{t-1}$, and novel branch is initialized with $\{\theta_n^{t-1}, \hat{W}_n^{t-1}\}$ which can be further finetuned for novel class adapation.

For effectively learning from few-shot novel classes, our class-aware bilateral distillation combines the generalizable knowledge preserved in base branch and newly adapted knowledge from previous novel branch. The guidance signal for distillation is an adaptive convex combination of base and previous novel branch logits:

[1] For clarity, We omit the superscript of base branch feature extractor as it is fixed in each incremental session.

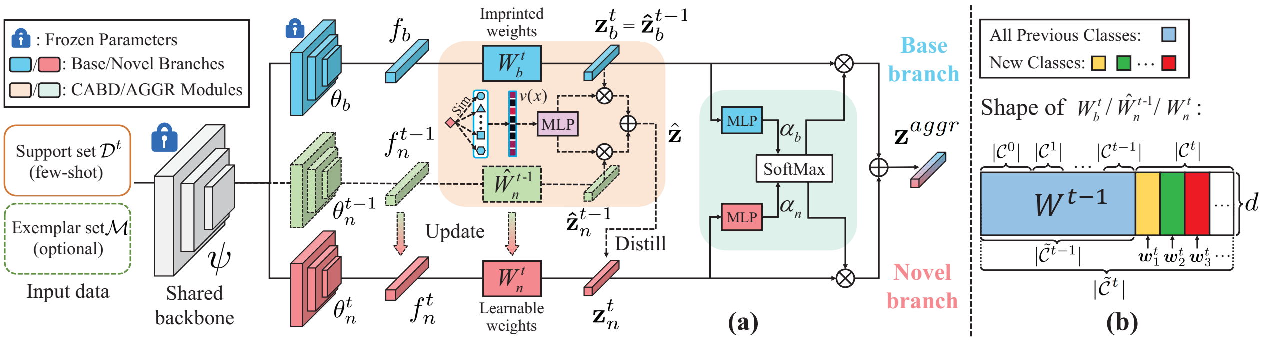


Figure 2. (a) An overview of our proposed method where modules with dashed lines are only used in the training process of session t. More details are discussed in Section 4. (b) Visualization of expanded classification weights in session t. Best viewed in color.

$$\hat{\mathbf{z}} = \rho(\mathbf{x}) \cdot \hat{\mathbf{z}}_b^{t-1} + (1 - \rho(\mathbf{x})) \cdot \hat{\mathbf{z}}_n^{t-1}, \tag{2}$$

where logits $\mathbf{z} = W^\top * f_\phi(\mathbf{x})$ is computed with corresponding weights and feature embeddings using cosine similarity[2], operators "$\cdot$" and "$*$" are used to distinguish between element-wise and matrix multiplication. Specifically, $\hat{\mathbf{z}}_b^{t-1} \in \mathbb{R}^{|\tilde{\mathcal{C}}^t|}$ is computed using imprinted $\hat{W}_b^{t-1}$ and features from the base branch, while $\hat{\mathbf{z}}_n^{t-1} \in \mathbb{R}^{|\tilde{\mathcal{C}}^t|}$ is computed with $\hat{W}_n^{t-1}$ and features from previous novel branch. The combination coefficient $\rho(\mathbf{x})$ is defined as:

$$\rho(\mathbf{x}) = \begin{cases} 1.0 & \text{if } y(\mathbf{x}) \in \mathcal{C}^0 \\ 1/(1+e^{-g_\vartheta(\boldsymbol{v}(\mathbf{x}))}) & \text{if } y(\mathbf{x}) \notin \mathcal{C}^0, \end{cases} \tag{3}$$

which adaptively controls the extent of transferred general knowledge from base to novel classes, conditioned on the class-wise semantic similarities. We set the coefficient of base classes to 1 for directly preserving base knowledge absorbed from abundant samples. As for a sample $\mathbf{x}$ from novel class c, a multi-layer perceptron (MLP) g_ϑ containing only one hidden layer takes the semantic vector $\boldsymbol{v}(\mathbf{x}) \in \mathbb{R}^{|\mathcal{C}^0|}$ as input and outputs a scalar:

$$g_\vartheta(\boldsymbol{v}(\mathbf{x})) = \text{MLP}\left(\left[cos(\boldsymbol{w}_c, \boldsymbol{w}_1), \ldots, cos(\boldsymbol{w}_c, \boldsymbol{w}_{|\mathcal{C}^0|})\right]; \vartheta\right). \tag{4}$$

$\boldsymbol{v}(\mathbf{x})$ encodes the prior semantic similarity between the category of $\mathbf{x}$ to all base classes where the k-th element is denoted as $cos(\boldsymbol{w}_c, \boldsymbol{w}_k)$, which is the cosine similarity between the classification weights $\boldsymbol{w}_c$ of novel class c and $\boldsymbol{w}_k$ of base class k, both directly taken from the expanded $\hat{W}_b^{t-1}$. As $\boldsymbol{w}_k$ can be reasonably regarded as the mean feature embedding for class k [25], $\boldsymbol{v}(\mathbf{x})_k$ naturally reveals the semantic similarity between class k and c. In addition, our distillation is a more general form of vanilla distillation by degenerating $\rho(\mathbf{x})$ to 0 in Eq. 2.

After obtaining the supervisory teacher logits $\hat{\mathbf{z}}$, the loss for class-aware bilateral distillation is defined as:

$$\begin{aligned} \mathcal{L}_{dst} &= \mathbb{E}_{(\mathbf{x},y)\sim\mathcal{D}^t\cup\mathcal{M}}\Big[\sum_{k=1}^{|\tilde{\mathcal{C}}^t|} -\tau_k(\hat{\mathbf{z}}) \log \tau_k(\mathbf{z}_n^t)\Big], \\ \tau_k(\mathbf{z}) &= \frac{e^{\gamma\cdot\mathbf{z}(k)/T}}{\sum_{j=1}^{|\tilde{\mathcal{C}}^t|} e^{\gamma\cdot\mathbf{z}(j)/T}}, \end{aligned} \tag{5}$$

where $\mathbf{z}(k)$ is the k-th element in $\mathbf{z}$, T is the distillation temperature and γ is a scalar to control the peakiness of output probability distribution [10, 24]. Note that our class-aware bilateral distillation is applied to $\mathcal{D}^t \cup \mathcal{M}$ on all the seen classes to simultaneously address both catastrophic forgetting and the unique overfitting challenge in FSCIL.

To further prevent the novel branch from overfitting and stabilize training, we also introduce a regularization loss $\mathcal{L}_{reg}$ to explicitly control coefficients $\rho(\mathbf{x})$ by encouraging more knowledge transferred from the base branch:

$$\mathcal{L}_{reg} = \mathbb{E}_{(\mathbf{x},y)\sim\mathcal{D}^t\cup\mathcal{M}} \left|\rho(\mathbf{x}) - 1\right|. \tag{6}$$

After that, combined with the classification loss $\mathcal{L}_{cls}$:

$$\mathcal{L}_{cls} = \mathbb{E}_{(\mathbf{x},y)\sim\mathcal{D}^t\cup\mathcal{M}}\left[\text{CE}(\gamma \cdot \mathbf{z}_n^t, y)\right], \tag{7}$$

where CE is the cross-entropy loss function and $\mathbf{z}_n^t$ is the output logits from current novel branch, we formulate the final loss function for novel branch as:

$$\mathcal{L}_{novel} = \mathcal{L}_{cls} + w_{dst} \cdot \mathcal{L}_{dst} + w_{reg} \cdot \mathcal{L}_{reg}, \tag{8}$$

where w_{dst} and w_{reg} are the balancing hyper-parameters.

4.3. Attention-based Prediction Aggregation

Despite the remarkable classification performance of novel branch trained using our class-aware bilateral distillation, it is inevitable to incur base class knowledge forgetting due to finetuning on novel class training set. To further improve the performance on base classes, an attention-based aggregation module that selectively merges predictions from both base and novel branches is proposed.

As shown in the right part of Fig. 2 (a), logits $\mathbf{z}_b^t$ and $\mathbf{z}_n^t$ from base and novel branch are fed into two MLPs, *i.e.*, h_{φ_b} and h_{φ_n}, to get the confidence score α_b and α_n for each

[2] Following [13], we use L2 normalization by mapping features and weights into a high-dimensional sphere for alleviating the imbalance of base and novel classes. Normalization is omitted to simplify notation.

Algorithm 1 Model Adaptation in Incremental Session t

```
Input: Feature extractor of base and novel branch $f_{\phi_b}$ and $f_{\phi_n^{t-1}}$ from session $t-1$, classification weights of base and novel branch $W_b^{t-1}$ and $W_n^{t-1}$.
Output: Evolved feature extractor $f_{\phi_n^t}$, classification weights $W_n^t$ of novel branch, updated exemplar set $\mathcal{M}$.
Expand $W_b^{t-1}$ and $W_n^{t-1}$ from $\mathbb{R}^{d\times|\tilde{\mathcal{C}}^{t-1}|}$ to $\mathbb{R}^{d\times|\tilde{\mathcal{C}}^{t}|}$ using training samples from $\mathcal{D}^t$ with Eq. 1.
Initialize current novel branch with $\{f_{\phi_n^{t-1}}, \hat{W}_n^{t-1}\}$ and randomly initialize MLP parameters $g_\vartheta$, $h_{\varphi_b}$ and $h_{\varphi_n}$.
while not done do
    $\{(\mathbf{x}, y)\} \leftarrow$ sample a batch of data from $\mathcal{D}^t \cup \mathcal{M}$.
    Calculate the novel branch adaption loss $\mathcal{L}_{novel}$ in Eq. 8 with proposed distillation strategy.
    Calculate the aggregated logits $\mathbf{z}^{aggr}$ in Eq. 9 and compute the overall loss function $\mathcal{L}$ in Eq. 11.
    Update $\{\phi_n^t, W_n^t, \vartheta, \varphi_b, \varphi_n\}$ with gradients $\nabla\mathcal{L}$.
end while
Select Top-$k$ (default $k$=1) samples for each novel class in $\mathcal{D}^t$ whose embedding $f_{\phi_n^t}(\mathbf{x})$ is the nearest to corresponding weights in $W_n^t$, and then add into $\mathcal{M}$.
```

branch: $\alpha_l = h_{\varphi_l}(\text{softmax}(\gamma \cdot \mathbf{z}_l^t))$, $l \in \{b, n\}$. We softmax the logits to probabilities for the ease of MLP training and γ is the scalar in Eq. 5. Then aggregated logits $\mathbf{z}^{aggr}$ can be computed with the instance-wise confidence:

$$\mathbf{z}^{aggr} = [\mathbf{z}_b^t, \mathbf{z}_n^t] * \text{softmax}([\alpha_b, \alpha_n])^\top, \tag{9}$$

Intuitively, the two complementary branches specialize in the classification of samples from base and novel categories, respectively. Namely, base branch is no doubt more competent in handling samples of base classes $\mathcal{C}^0$, where the forgetting of base knowledge does not exist. By contrast, novel branch is more equipped to classify incremental classes due to the flexible adaption to novel data. To produce appropriate attention weights for different classes, a binary classification constraint is adopted:

$$\mathcal{L}_{bin} = \mathbb{E}_{(\mathbf{x},y)\sim\mathcal{D}^t\cup\mathcal{M}}\left[\text{CE}([\alpha_b, \alpha_n], \mathbf{I}[y \in \mathcal{C}^0])\right], \tag{10}$$

the loss encourages a larger value of α_b if the sample comes from base classes, and a larger α_n for the novel class. Finally, combined with the novel branch learning loss in Eq. 8, the overall loss function can be computed as:

$$\mathcal{L} = \mathcal{L}_{novel} + \mathcal{L}_{attn} + w_{bin} \cdot \mathcal{L}_{bin}, \tag{11}$$

where $\mathcal{L}_{attn}$ is the classification loss in the same form as Eq. 7 on the aggregated logits $\mathbf{z}^{aggr}$. During inference, the final predictions are made using $\mathbf{z}^{aggr}$.

Combined with the proposed bilateral distillation and attention-based aggregation, our model is end-to-end trainable in each incremental session t using Eq. 11 and the pseudo-code is given in Algorithm 1.

4.4. Discussions of Different Exemplar Settings

Following previous works [3,8,18,28], so far we assume an exemplar set $\mathcal{M}$ (1 exemplar per class by default) can be accessed. Nevertheless, our method is flexible enough to provide different modes to trade off between memory cost and accuracy. In the scenario where more exemplars can be stored [8,28], we show in experiments that better results can be further obtained. More importantly, our method can also tackle the extreme case where exemplars are unavailable. To this end, we only need to modify the exemplar set from $\mathcal{M}$ to $\bar{\mathcal{M}}$, in which instead of saving original exemplar images, we simply use the mean feature embedding of each class as the substitution to acquire the pseudo exemplar set $\bar{\mathcal{M}}$. Besides, we can directly treat the corresponding classification weight in W_b^{t-1} and W_n^{t-1} as a good approximation for the mean feature embedding of previous categories [25], thus no extra memory is required for saving $\bar{\mathcal{M}}$. Namely, the corresponding weight vector taken from W_b^{t-1} and W_n^{t-1} are used to replace f_b and f_n in Fig. 2.

5. Experiments

In this section, experimental setups of FSCIL are first presented. Then we compare with state-of-the-arts on 3 popular benchmarks. After that, detailed ablative experiments are conducted to validate each proposed component.

5.1. Experimental Setups

Datasets and Evaluation. Following mainstream setting [30], our experiments are conducted on three benchmark datasets: *mini*-ImageNet [32], CIFAR100 [17] and CUB200 [33]. *mini*-ImageNet is a subset of ImageNet [7] including 60,000 images with resolution 84×84 from 100 chosen classes. CIFAR100 is comprised of 60,000 tiny images of size 32×32 from 100 categories. CUB200 is a fine-grained classification dataset for 200 bird species with similar appearance where the image size is 224×224. We follow the splits in [30], for *mini*-ImageNet and CIFAR100, 60 categories are selected as base classes while the remaining are split into 8 incremental sessions with only 5 training examples per novel class (*i.e.*, 5-way 5-shot). As for CUB200 dataset, 100 categories are selected as the base training set, while the rest forms 10-way 5-shot tasks for 10 sessions in total. We leave more details of dataset splits and visualizations in the supplementary material.

Implementation Details. Our method is conducted with PyTorch library, and results are averaged over 5 runs. ResNet18 [11] is adopted as the feature extractor f_ϕ which is trained by SGD optimizer with Nesterov momentum 0.9.

In the base session, we simply pre-train on the base class training set as in standard supervised learning without extra self-supervised techniques [15, 23] or sophisticated meta-training strategies [6,41]. For *mini*-ImageNet and CIFAR100, we train 200 epochs from scratch and the learning

Method	Accuracy in each session (%)									Avg.	Final Impro.
	0	1	2	3	4	5	6	7	8		
iCaRL$^{*\diamond}$ [26]	61.31	46.32	42.94	37.63	30.49	24.00	20.89	18.80	17.21	33.29	+35.01
TOPIC [30]	61.31	50.09	45.17	41.16	37.48	35.52	32.19	29.46	24.42	39.64	+27.80
ERL++** [8]	61.70	57.58	54.66	51.72	48.66	46.27	44.67	42.81	40.79	49.87	+11.43
IDLVQ* [3]	64.77	59.87	55.93	52.62	49.88	47.55	44.83	43.14	41.84	51.16	+10.38
CEC [39]	72.00	66.83	62.97	59.43	56.70	53.73	51.19	49.24	47.63	57.75	+4.59
F2M** [28]	72.05	67.47	63.16	59.70	56.71	53.77	51.11	49.21	47.84	57.89	+4.38
CLOM [44]	73.08	68.09	64.16	60.41	57.41	54.29	51.54	49.37	48.00	58.48	+4.22
Replay* [21]	71.84	67.12	63.21	59.77	57.01	53.95	51.55	49.52	48.21	58.02	+4.01
MetaFSCIL [6]	72.04	67.94	63.77	60.29	57.58	55.16	52.90	50.79	49.19	58.85	+3.03
FACT$^{\natural}$ [41]	**75.32**	70.34	65.84	62.05	58.68	55.35	52.42	50.42	48.51	59.88	+3.71
Ours (0 exemplar)	74.65	69.89	65.44	61.76	59.49	56.11	53.28	51.74	50.49	60.32	
Ours (1 exemplar)[default]*	74.65	**70.43**	**66.29**	**62.77**	**60.75**	**57.24**	**54.79**	**53.65**	**52.22**	**61.42**	
Ours (5 exemplars)**	74.65	70.70	66.81	63.63	61.36	58.14	55.59	54.23	53.39	62.06	

*: method with 1 exemplar per class. **: method with 5 exemplars per class. $^{\diamond}$: results from [30]. $^{\natural}$: results using the publicly available code from [41].

Table 1. Comparisons to state-of-the-art FSCIL methods on *mini*-ImageNet. "Final Impro." highlights the improvement in the final session.

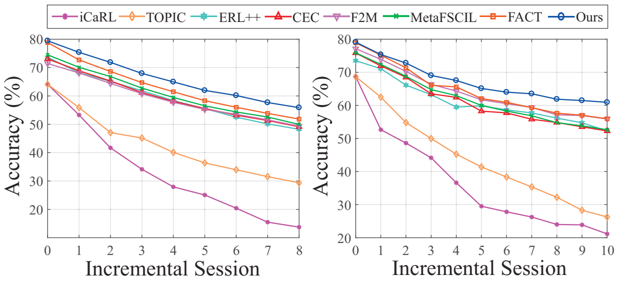
rate is set to 0.1 which is dropped by 0.1 at the 120-th and 160-th epoch. Since the model for CUB200 is initialized by ImageNet [7] pre-trained parameters [30, 39, 41], we further train 120 epochs with learning rate 0.01 dropped by 0.1 at the 50-th, 70-th and 90-th epoch. Following [10, 24], scalar γ is initialized to 10 which is learnable in the base session and keeps fixed afterward. Data augmentations including left-right flip, random crop and color jitter are applied.

In each incremental session, we further finetune the novel branch with distillation temperature $T = 16$ and learning rate 0.001 for 100 iterations. The learning rate for MLPs (g_ϑ, h_{φ_b} and h_{φ_n}) is set to 0.01. The selection of other hyper-parameters is provided in Section 5.3.

5.2. Comparisons with State-of-The-Arts

We conduct comparisons with recent state-of-the-arts on *mini*-ImageNet, CIFAR100 and CUB200 datasets. By default, our method utilizes an exemplar set $\mathcal{M}$ where only 1 training example per class is stored as in [3, 26]. As reported in Table 1, we surpass the second-best approach by 3.03% for the improvement in the final session and boost the average performance by 1.54% on *mini*-ImageNet. With more exemplars stored (5 per class as [8, 28]), we further improve the final result by 1.17% thanks to our distillation module for effectively exploring useful knowledge from the data. When exemplar is unavailable, our flexible framework still surpasses all previous methods due to better novel class adaptation in extreme scenarios discussed in Section 4.4.

The performance curves on CIFAR100 and CUB200 are presented in Fig. 3, and more detailed results on the two datasets are given in our supplementary material. As shown in Fig. 3, our method consistently outperforms previous state-of-the-arts in all sessions. The above observations verify the superiority of our approach for effective adaptation to novel classes with few training samples.



(a) CIFAR100 (b) CUB200

Figure 3. Performance curves of our method comparing to state-of-the-art FSCIL methods on (a) CIFAR100 and (b) CUB200.

5.3. Ablation Studies

We first validate our two key components: Class-Aware Bilateral Distillation (CABD) in Section 4.2 and Attention-based Prediction Aggregation (AGGR) in Section 4.3, then provide hyper-parameter sensitivity test experiments.

Distillation Module. The upper part of Table 2 shows that, with fixed feature extractor and imprinted classification weights, the base branch achieves joint accuracy of 48.97% in the final session. However, the accuracy is biased to base classes while the novel class accuracy Acc$_{novel}$ only reaches 13.68% due to the lack of plasticity. Instead, our novel branch (middle part of Table 2) with learnable parameters is much better adapted to novel classes as Acc$_{novel}$ implies. However, vanilla distillation only using imprinted novel branch logits $\hat{\mathbf{z}}_n^{t-1}$ (part of Eq. 2) leads to degraded performance, since base class accuracy Acc$_{base}$ decreases dramatically caused by overfitting to novel classes. Although directly using logits $\hat{\mathbf{z}}_b^{t-1}$ from base branch maintains base class results, but fails to facilitate better novel performance. Only by combining the general knowledge from $\hat{\mathbf{z}}_b^{t-1}$ and the well adapted concepts from $\hat{\mathbf{z}}_n^{t-1}$ in Eq. 2, the best balance between Acc$_{base}$ and Acc$_{novel}$ is obtained,

Branch	CABD		AGGR	Joint Accuracy of base and novel classes in each session (%)									Final session (8)	
	$\hat{\mathbf{z}}_n^{t-1}$	$\hat{\mathbf{z}}_b^{t-1}$	$\mathbf{z}^{aggr}$	0	1	2	3	4	5	6	7	8	Acc$_{base}$	Acc$_{novel}$
Base				74.65	69.85	65.30	61.67	58.65	55.48	52.74	50.79	48.97	72.50	13.68
Novel	✓			74.65	69.46	64.17	59.75	56.88	52.65	49.44	47.16	45.33	54.33	31.50
		✓		74.65	69.49	65.09	62.09	59.95	56.56	53.73	52.23	50.83	68.08	24.95
	✓	✓		74.65	69.68	65.91	62.32	60.15	56.78	54.52	53.26	51.47	66.15	29.45
Aggregate	✓	✓	✓	**74.65**	**70.43**	**66.29**	**62.77**	**60.75**	**57.24**	**54.79**	**53.65**	**52.22**	67.83	28.80

Table 2. Ablation studies of our proposed method on *mini*-ImageNet dataset. "Acc$_{base}$" and "Acc$_{novel}$" denote the performance of recognizing test samples from base and novel classes in the final session (8), respectively.

AGGR Type	AGGR Strategy	Acc.
Feature	(a) $W^\top * [f_b^t \odot f_n^t]$	51.58
Prediction	(b) $\mathbf{z}_b^t$ if $(\arg\max \mathbf{z}_b^t \in \mathcal{C}^0)$ else $\mathbf{z}_n^t$	49.39
	(c) $\mathbf{z}_b^t$ if $(\arg\max \mathbf{z}_n^t \in \mathcal{C}^0)$ else $\mathbf{z}_n^t$	51.81
	(d) $\frac{1}{2} \cdot \mathbf{z}_b^t + \frac{1}{2} \cdot \mathbf{z}_n^t$	51.28
	(e) $[\mathbf{z}_b^t, \mathbf{z}_n^t] * \text{softmax}([\alpha_b, \alpha_n])^\top$	**52.22**

Table 3. Prediction aggregation choices on *mini*-ImageNet.

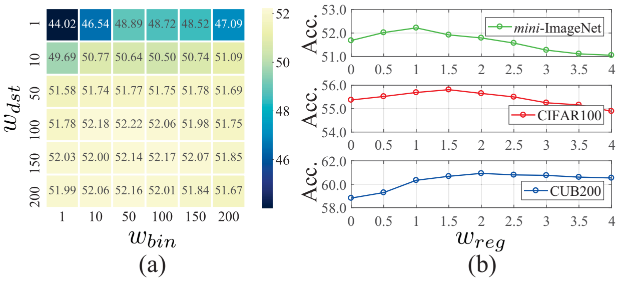


Figure 4. Ablations of hyper-parameter sensitivity: (a) w_{dst} and w_{bin} on *mini*-ImageNet and (b) w_{reg} on three datasets.

achieving 2.5% improvement than the naive base branch.

Prediction Aggregation Module. Although only using the novel branch already makes promising results, we hope to further suppress the drop in base classes as they contribute a large proportion of encountered classes. To this end, an attention-based prediction aggregation module is adopted to adaptively combine predictions from base and novel branches. As shown in the lower part of Table 2, the proposed aggregation module achieves consistent improvements compared to a single novel branch in all sessions with better trade-off between Acc$_{base}$ and Acc$_{novel}$. To further prove the advantages of our aggregation design, we compare it to other widely used aggregation choices: (a) classifying with concatenated features from base and novel branches [14,15,40]; (b) adopting predictions from the base branch if $\arg\max(\mathbf{z}_b^t)$ belongs to base classes, otherwise adopting predictions from the novel branch; (c) similar to (b) but using $\arg\max(\mathbf{z}_n^t)$ instead of $\arg\max(\mathbf{z}_b^t)$ for routing; (d) simply averaging predictions from two branches without attention mechanism. It is observed from Table 3 that our design (e) outperforms other choices, attributing to the flexible attention mechanism.

Hyper-Parameter Sensitivity. In Eq. 8 and Eq. 11, three key hyper-parameters are included during training: w_{dst}, w_{reg} and w_{bin}. See from Fig. 4 (a), our model can achieve satisfactory results on *mini*-ImageNet dataset with relative larger values of w_{dst} and w_{bin}, and becomes insensitive to the selection of the two hyper-parameters in the wide range from 50 to 200. It is because the model can prevent more knowledge from forgetting with the help of a larger distillation weight w_{dst}, and the aggregation module can better distinguish a test sample whether from a base or novel class thanks to the regularization effect of binary classification weight w_{bin}. Moreover, consistent experimental results are also observed from the other two datasets. As a result, we set $w_{dst} = 100$ and $w_{bin} = 50$ respectively throughout our experiments.

In addition, we analyze the performance fluctuation under different loss weights w_{reg} which controls the regularization strength on $\rho(\mathbf{x})$ in Eq. 2. The larger value of w_{reg} brings stronger constraints to pull $\rho(\mathbf{x})$ close to 1, meaning more general knowledge from base branch should be transferred to novel branch. Shown in Fig. 4 (b), accuracy gradually improves as w_{reg} grows larger in the beginning, because transferring more base knowledge can alleviate overfitting. Then accuracy starts to decline when w_{reg} becomes too large, as the plasticity of novel branch is hurt. The optimal w_{reg} for three datasets is 1.0, 1.5 and 2.0, respectively. For classes from the fine-grained dataset CUB200 that share similar appearance, larger w_{reg} is required to pull the coefficient $\rho(\mathbf{x})$ close to 1, thus more knowledge is transferred from base to novel classes. Contrarily, for classes from *mini*-ImageNet and CIFAR100 that are less semantically related, a relatively smaller w_{reg} is preferred.

We also present ablations on the sensitivity of model parameters θ which can be finetuned for novel classes. Table. 4 shows that, with fewer parameters trainable, the model finds it difficult to accommodate new concepts due to sacrificed plasticity. We also tend to get dissatisfactory performance when finetuning excessive parameters due to potential overfitting in few-shot scenarios. Finally, our model achieves the best trade-off between stability and plasticity when adapting the last residual layer (*i.e.*, conv5_x in [11]).

Learnable Parameters θ	Final Acc.
NULL	49.96
last (conv + bn) of conv5_x [11]	51.53
last resblock of conv5_x	51.78
conv5_x	**52.22**
conv4∼5_x	51.81
conv3∼5_x	51.69
conv2∼5_x	51.42
all layers of backbone f_ϕ	51.34

Table 4. Sensitivity of learnable parameters θ on *mini*-ImageNet.

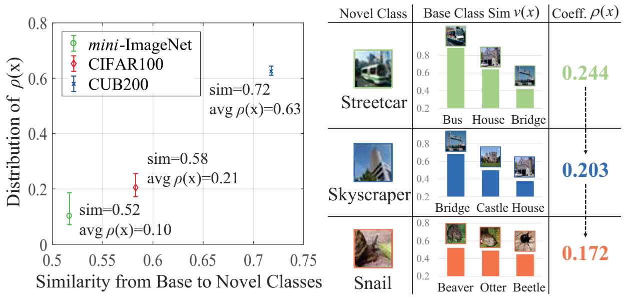
5.4. Further Analyses

Coefficients for Distillation. We provide more in-depth analyses of the distillation coefficients $\rho(\mathbf{x})$. From a dataset-wise view, the prior semantic similarity on different datasets varies, thus the semantic-aware coefficients $\rho(\mathbf{x})$ should be varied across benchmarks. We first compute the cosine distances between each novel class weight and its nearest base class weight, which are averaged to obtain the overall similarity level of each dataset, then visualize its relation with the average value of learned $\rho(\mathbf{x})$. As in Fig. 5 (a), $\rho(\mathbf{x})$ quantitatively shows a positive correlation with the average level of semantic similarity, validating the more general knowledge from base branch should be transferred when base and novel classes are more semantically related.

From a class-wise view, we present 3 novel classes from CIFAR100 generating the distillation coefficient $\rho(\mathbf{x})$ from large to small, as well as its corresponding semantic similarity vector $\boldsymbol{v}(\mathbf{x})$ to base classes which is the input to produce $\rho(\mathbf{x})$. For clarity, top-3 values from vector $\boldsymbol{v}(\mathbf{x})$ are shown. Fig. 5 (b) shows that $\rho(\mathbf{x})$ tends to be a larger value when the novel class is more similar to the base ones since more generalizable knowledge can be leveraged, which again accords with our intentions to design the module.

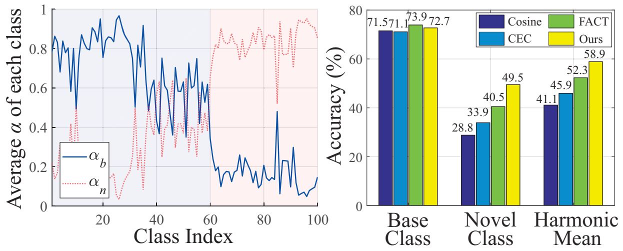
Attention Weights for Aggregation. To better understand the attention-based aggregation, we visualize the average score of α_b and α_n in Eq. 9 for each base and novel class on the test set of *mini*-ImageNet. From Fig. 6 (a), we observe that test samples from base classes (indices 1-60) induce larger base branch score α_b to leverage more predictions from base branch, while test samples from novel classes (indices 61-100) produce a larger α_n to pay more attention to novel branch predictions, as is expected.

In more detail, α_b of base classes with indices 1-35 tend to obtain larger values than those with indices 36-60. Because large proportions of classes within 36-60 and novel classes from 61-100 are inorganic objects, while classes from 1-35 belong to animal categories that are relatively less similar to novel classes, hence model puts more confidence on the predictions of base branch. This reveals the class-wise α_b and α_n are also correlated with different base-to-novel semantic similarity. Due to space limitations, we leave more detailed visualizations in the supplement.



(a) (b)

Figure 5. Further analyses of our distillation coefficients from (a) a dataset-wise view and (b) a class-wise view on CIFAR100.



(a) (b)

Figure 6. (a) Distribution of the averaged confidence score α_b and α_n of each class on *mini*-ImageNet; (b) Performance trade-off between base and novel classes on CUB200.

Trade-off between Base and Novel Classes. For better understanding FSCIL challenges, we analyze the ability to adapt to novel classes and to preserve base knowledge by delving into the individual accuracy of base and novel classes, as well as the harmonic mean. Since most existing works only focus on joint accuracy, we can only compare with few recent works [39, 41] that report harmonic mean results on CUB200 dataset. Fig. 6 (b) shows that our approach outperforms the second best result on novel classes by 9% verifying the power of adaption with our distillation module. Meanwhile, we still maintain competitive base class accuracy thanks to the prediction aggregation module for resisting base class forgetting. Finally, the best harmonic mean proves that we achieve a better trade-off between base and novel classes.

6. Conclusion

In the paper, we adapt knowledge distillation technique to handle the unique challenge of overfitting posed by FSCIL and introduce the Class-Aware Bilateral Distillation, which dynamically combines the general knowledge from base classes and the adapted concepts from previous novel classes. Besides, an attention-based aggregation module is used to bring a better balance between base and novel class performance. Extensive experiments and in-depth analysis prove that our approach can set a remarkable new state-of-the-art. In the future, we will investigate the more long-term FSCIL scenarios to verify our algorithm's robustness.

References

[1] Davide Abati, Jakub Tomczak, Tijmen Blankevoort, Simone Calderara, Rita Cucchiara, and Babak Ehteshami Bejnordi. Conditional channel gated networks for task-aware continual learning. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3931–3940, 2020. 2

[2] Francisco M Castro, Manuel J Marín-Jiménez, Nicolás Guil, Cordelia Schmid, and Karteek Alahari. End-to-end incremental learning. In *European Conference on Computer Vision*, pages 233–248, 2018. 1, 2

[3] Kuilin Chen and Chi-Guhn Lee. Incremental few-shot learning via vector quantization in deep embedded space. In *International Conference on Learning Representations*, 2021. 3, 5, 6

[4] Ali Cheraghian, Shafin Rahman, Pengfei Fang, Soumava Kumar Roy, Lars Petersson, and Mehrtash Harandi. Semantic-aware knowledge distillation for few-shot class-incremental learning. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2534–2543, 2021. 3

[5] Ali Cheraghian, Shafin Rahman, Sameera Ramasinghe, Pengfei Fang, Christian Simon, Lars Petersson, and Mehrtash Harandi. Synthesized feature based few-shot class-incremental learning on a mixture of subspaces. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 8661–8670, 2021. 3

[6] Zhixiang Chi, Li Gu, Huan Liu, Yang Wang, Yuanhao Yu, and Jin Tang. Metafscil: A meta-learning approach for few-shot class incremental learning. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 14166–14175, 2022. 3, 5, 6

[7] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 248–255, 2009. 5, 6

[8] Songlin Dong, Xiaopeng Hong, Xiaoyu Tao, Xinyuan Chang, Xing Wei, and Yihong Gong. Few-shot class-incremental learning via relation knowledge distillation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, pages 1255–1263, 2021. 3, 5, 6

[9] Chelsea Finn, Pieter Abbeel, and Sergey Levine. Model-agnostic meta-learning for fast adaptation of deep networks. In *International conference on machine learning*, pages 1126–1135, 2017. 2

[10] Spyros Gidaris and Nikos Komodakis. Dynamic few-shot visual learning without forgetting. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4367–4375, 2018. 2, 4, 6

[11] 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. 3, 5, 7, 8

[12] Geoffrey Hinton, Oriol Vinyals, and Jeff Dean. Distilling the knowledge in a neural network. *arXiv preprint arXiv:1503.02531*, 2015. 2

[13] Saihui Hou, Xinyu Pan, Chen Change Loy, Zilei Wang, and Dahua Lin. Learning a unified classifier incrementally via rebalancing. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 831–839, 2019. 1, 2, 4

[14] Zhong Ji, Zhishen Hou, Xiyao Liu, Yanwei Pang, and Xuelong Li. Memorizing complementation network for few-shot class-incremental learning. *arXiv preprint arXiv:2208.05610*, 2022. 7

[15] Jayateja Kalla and Soma Biswas. S3c: Self-supervised stochastic classifiers for few-shot class-incremental learning. In *European Conference on Computer Vision*, pages 432–448, 2022. 5, 7

[16] James Kirkpatrick, Razvan Pascanu, Neil Rabinowitz, Joel Veness, Guillaume Desjardins, Andrei A Rusu, Kieran Milan, John Quan, Tiago Ramalho, Agnieszka Grabska-Barwinska, et al. Overcoming catastrophic forgetting in neural networks. *Proceedings of the national academy of sciences*, 114(13):3521–3526, 2017. 2

[17] Alex Krizhevsky, Geoffrey Hinton, et al. Learning multiple layers of features from tiny images. 2009. 5

[18] Anna Kukleva, Hilde Kuehne, and Bernt Schiele. Generalized and incremental few-shot learning by explicit learning and calibration without forgetting. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 9020–9029, 2021. 5

[19] Kai Li, Yulun Zhang, Kunpeng Li, and Yun Fu. Adversarial feature hallucination networks for few-shot learning. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 13470–13479, 2020. 2

[20] Zhizhong Li and Derek Hoiem. Learning without forgetting. *IEEE Transactions on Pattern analysis and machine intelligence*, 40(12):2935–2947, 2017. 1, 2

[21] Huan Liu, Li Gu, Zhixiang Chi, Yang Wang, Yuanhao Yu, Jun Chen, and Jin Tang. Few-shot class-incremental learning via entropy-regularized data-free replay. In *European Conference on Computer Vision*, 2022. 6

[22] Yaoyao Liu, Bernt Schiele, and Qianru Sun. Adaptive aggregation networks for class-incremental learning. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2544–2553, 2021. 2

[23] Pratik Mazumder, Pravendra Singh, and Piyush Rai. Few-shot lifelong learning. In *Proceedings of the AAAI Conference on Artificial Intelligence*, pages 2337–2345, 2021. 3, 5

[24] Boris Oreshkin, Pau Rodríguez López, and Alexandre Lacoste. Tadam: Task dependent adaptive metric for improved few-shot learning. In *Advances in neural information processing systems*, 2018. 4, 6

[25] Hang Qi, Matthew Brown, and David G Lowe. Low-shot learning with imprinted weights. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 5822–5830, 2018. 2, 3, 4, 5

[26] 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, 6

[27] Andrei A. Rusu, Dushyant Rao, Jakub Sygnowski, Oriol Vinyals, Razvan Pascanu, Simon Osindero, and Raia Hadsell. Meta-learning with latent embedding optimization. In *International Conference on Learning Representations*, 2019. 2

[28] Guangyuan Shi, Jiaxin Chen, Wenlong Zhang, Li-Ming Zhan, and Xiao-Ming Wu. Overcoming catastrophic forgetting in incremental few-shot learning by finding flat minima. *Advances in neural information processing systems*, pages 6747–6761, 2021. 3, 5, 6

[29] Jake Snell, Kevin Swersky, and Richard Zemel. Prototypical networks for few-shot learning. In *Advances in neural information processing systems*, 2017. 2, 3

[30] Xiaoyu Tao, Xiaopeng Hong, Xinyuan Chang, Songlin Dong, Xing Wei, and Yihong Gong. Few-shot class-incremental learning. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 12183–12192, 2020. 1, 2, 3, 5, 6

[31] Yonglong Tian, Yue Wang, Dilip Krishnan, Joshua B Tenenbaum, and Phillip Isola. Rethinking few-shot image classification: a good embedding is all you need? In *European Conference on Computer Vision*, pages 266–282, 2020. 2

[32] Oriol Vinyals, Charles Blundell, Timothy Lillicrap, Daan Wierstra, et al. Matching networks for one shot learning. In *Advances in neural information processing systems*, 2016. 2, 5

[33] Catherine Wah, Steve Branson, Peter Welinder, Pietro Perona, and Serge Belongie. The caltech-ucsd birds-200-2011 dataset. *Technical report*, 2011. 5

[34] Yu-Xiong Wang, Ross Girshick, Martial Hebert, and Bharath Hariharan. Low-shot learning from imaginary data. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 7278–7286, 2018. 2

[35] Jason Yosinski, Jeff Clune, Yoshua Bengio, and Hod Lipson. How transferable are features in deep neural networks? In *Advances in neural information processing systems*, 2014. 2

[36] Lu Yu, Bartlomiej Twardowski, Xialei Liu, Luis Herranz, Kai Wang, Yongmei Cheng, Shangling Jui, and Joost van de Weijer. Semantic drift compensation for class-incremental learning. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 6982–6991, 2020. 2

[37] Tianyuan Yu, Sen He, Yi-Zhe Song, and Tao Xiang. Hybrid graph neural networks for few-shot learning. In *Proceedings of the AAAI Conference on Artificial Intelligence*, pages 3179–3187, 2022. 2

[38] Friedemann Zenke, Ben Poole, and Surya Ganguli. Continual learning through synaptic intelligence. In *International conference on machine learning*, pages 3987–3995, 2017. 2

[39] Chi Zhang, Nan Song, Guosheng Lin, Yun Zheng, Pan Pan, and Yinghui Xu. Few-shot incremental learning with continually evolved classifiers. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 12455–12464, 2021. 3, 6, 8

[40] Hanbin Zhao, Yongjian Fu, Mintong Kang, Qi Tian, Fei Wu, and Xi Li. Mgsvf: Multi-grained slow vs. fast framework for few-shot class-incremental learning. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2021. 7

[41] Da-Wei Zhou, Fu-Yun Wang, Han-Jia Ye, Liang Ma, Shiliang Pu, and De-Chuan Zhan. Forward compatible few-shot class-incremental learning. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 9046–9056, 2022. 1, 3, 5, 6, 8

[42] Da-Wei Zhou, Han-Jia Ye, Liang Ma, Di Xie, Shiliang Pu, and De-Chuan Zhan. Few-shot class-incremental learning by sampling multi-phase tasks. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2022. 1, 3

[43] Kai Zhu, Yang Cao, Wei Zhai, Jie Cheng, and Zheng-Jun Zha. Self-promoted prototype refinement for few-shot class-incremental learning. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 6801–6810, 2021. 3

[44] Yixiong Zou, Shanghang Zhang, Yuhua Li, and Ruixuan Li. Margin-based few-shot class-incremental learning with class-level overfitting mitigation. In *Advances in neural information processing systems*, 2022. 6