First Session Adaptation: A Strong Replay-Free Baseline for Class-Incremental Learning

Aristeidis Panos
University of Cambridge
ap2313@cam.ac.uk

Yuriko Kobe
University of Cambridge
yk384@cam.ac.uk

Daniel Olmeda Reino
Toyota Motor Europe
daniel.olmeda.reino@toyota-europe.com

Rahaf Aljundi
Toyota Motor Europe
rahaf.al.jundi@toyota-europe.com

Richard E. Turner
University of Cambridge
ret26@cam.ac.uk

Abstract

In Class-Incremental Learning (CIL) an image classification system is exposed to new classes in each learning session and must be updated incrementally. Methods approaching this problem have updated both the classification head and the feature extractor body at each session of CIL. In this work, we develop a baseline method, First Session Adaptation (FSA), that sheds light on the efficacy of existing CIL approaches, and allows us to assess the relative performance contributions from head and body adaption. FSA adapts a pre-trained neural network body only on the first learning session and fixes it thereafter; a head based on linear discriminant analysis (LDA), is then placed on top of the adapted body, allowing exact updates through CIL. FSA is replay-free i.e. it does not memorize examples from previous sessions of continual learning. To empirically motivate FSA, we first consider a diverse selection of 22 image-classification datasets, evaluating different heads and body adaptation techniques in high/low-shot offline settings. We find that the LDA head performs well and supports CIL out-of-the-box. We also find that Featurewise Layer Modulation (FiLM) adapters are highly effective in the few-shot setting, and full-body adaption in the high-shot setting. Second, we empirically investigate various CIL settings including high-shot CIL and few-shot CIL, including settings that have previously been used in the literature. We show that FSA significantly improves over the state-of-the-art in 15 of the 16 settings considered. FSA with FiLM adapters is especially performant in the few-shot setting. These results indicate that current approaches to continuous body adaptation are not working as expected. Finally, we propose a measure that can be applied to a set of unlabelled inputs which is predictive of the benefits of body adaptation.

1. Introduction

Continual learning (CL) is needed to bring machine learning models to many real-life applications. After a model is trained on a given set of data, and once deployed in its test environment, it is likely that new classes will naturally emerge. For example, in an autonomous driving scenario, new types of road transportation and traffic signage can be encountered. The deployed model needs to efficiently acquire this new knowledge with minimal cost (e.g. annotation requirements) without deteriorating the performance on existing classes of objects. The process of efficiently acquiring new knowledge while preserving what is already captured by the model is what continual learning methods target.

Continual learning can be mainly divided into task incremental learning and class incremental learning. Task incremental learning (TIL) sequentially learns independent sets of heterogeneous tasks and is out of this paper's scope. Class incremental learning (CIL), in contrast, assumes that all classes (already learnt and future ones) form part of a single classification task. In class-incremental learning, data arrive in a sequence of sessions with new classes appearing as the sessions progress. Critically, the data in each session cannot be stored in its entirety and cannot be revisited. The goal is to train a single classification model under these constraints.

In both task incremental and class incremental learning, the data distribution will change as the sessions progress. However, these changes tend to be smaller in real-world class incremental learning settings than in the task incremental setting. For, example consider a human support robot learning about new objects in a home in an incremental manner (CIL) vs. the same robot learning to adapt to different homes (TIL).

Practically, CIL has two major uses. First, in situations where large amounts of data arrive in each session and so retraining the model is computationally prohibitive. Second, in situations where data are not allowed to be revisited due to privacy reasons such as under General Data Protection Regulation (GDPR). The latter situation is relevant for applications such as personalization where small numbers of data points are available i.e. few-shot continual learning. So-called replay-free methods are necessary for such settings, as samples of previous sessions are not allowed to be memorized, but CIL is known to be challenging in such settings.

Current SOTA methods for class incremental learning start from a pre-trained backbone [13, 19, 35, 34] and then adapt the features at each session of continual learning. The use of a pre-trained backbone has been shown to lead to strong performance especially in the few-shot CIL setting due to the lack of data [21]. However, it is unclear to what extent continuous adaption of the features in each session is helpful. In general, in CIL there is a trade-off between adaptation (which helps adapt to the new statistics of the target domain) and catastrophic forgetting (whereby earlier classes are forgotten as the representation changes). When memorization of previous samples is restricted, the benefits from adapting the feature extractor body to learn better features might well be out-weighted by the increase in forgetting of old classes. Moreover, body adaptation in earlier sessions is arguably more critical (e.g. adapting the backbone to the new domain in the first session), whilst it is less essential in later sessions where the changes in the ideal representation between sessions are smaller.

This work explores under what conditions continuous adaptation of the body is beneficial, both in principle and in current practice. In order to do this, we develop a new replay-free method, inspired by the first encoding method of [2], called First Session Adaptation (FSA). FSA adapts the body of a pre-trained neural network on only the first session of continual learning. We investigate adapting all body parameters and the use of Feature-wise Layer Modulation (FiLM) adapters [23] which learn only a small number of parameters. The head, in contrast, is adapted at each session using an approach that is similar to Linear Discriminate Analysis (LDA), which suffers from no forgetting, and improves over the Nearest Class Mean classifier (NCM) [18] approach. The efficacy of this general approach, including comparisons to a standard linear head, is first motivated through experiments in the offline setting (Sec. 4.2). We then carry out experiments under three CIL settings. First, we consider the high-shot setting (high-shot CIL). The other two settings consider few-shot continual learning. Specifically, one setting follows previous approaches that employ an initial session with a large number of data points and few-shot sessions thereafter (few-shot+ CIL) while the other exclusively includes sessions with only a small amount of data (few-shot CIL).

The contributions of the paper are as follows: (1) We develop a replay-free CIL baseline, namely FSA, that is extremely simple to implement and performs well in many different scenarios. (2) We empirically motivate FSA through a set of offline experiments that evaluate different forms of neural network head and body adaptation; (3) We then compare FSA to a set of strong continual learning baseline methods in a fair way using the same pre-trained backbone for all methods. (4) We show that the FSA-FiLM baseline performs well in the high-shot CIL setting, outperforming the SOTA whilst avoiding data memorization. (5) In the few-shot learning settings, FSA outperforms existing continual learning methods on eight benchmarks, often by a substantial margin, and is statistically tied with the best performing method on the one remaining dataset. (6) Finally, we propose a measure that can be applied to a set of unlabelled inputs which is predictive of the benefits of body adaptation.

2. Related Work

Class Incremental Learning is more challenging than task-incremental continual learning as a shared output layer is used for the classes of different learning sessions. We refer to [7] for a survey on both class and task incremental learning. While softmax cross entropy loss is considered a standard in classification models, it is shown to be a source of class interference in continual learning [38, 4]. Thus, recent works either focus on fixing the softmax classifier [4, 1, 44] or deploy nearest class mean classifiers (NCM) as an alternative [25, 5, 11]. In this work, we deploy an LDA classifier that uses the mean embedding of each class and an incrementally updated covariance matrix shared across all classes. We show that this approach is comparable to a softmax classifier in the offline setting and that it outperforms NCM.

Due to the challenging nature of class incremental learning, some methods employ a buffer of stored samples from previous sessions [24, 39]. [24] suggests a simple baseline, GDumb, that performs offline training on both buffer data and new session data at each incremental step. GDumb shows strong performance compared to sophisticated continual learning solutions. Here we show that our simple yet effective solution often outperforms variants of GDumb that have a large memory without leveraging any buffer of stored samples.

In addition to the issues arising from the shared classification layer, learning a feature extractor that can provide representative features of all classes is a key element in continual learning. Some class incremental learning methods leverage large initial training session or start from a pre-trained network, e.g., [35, 9, 36] and show improved overall continual learning performance without questioning the role of such pre-training steps and whether the learned representations have in fact improved. Recently, [19] studies continual learning with strong pre-trained models and proposes deploying exemplar-based replay in the latent space using a small multilayer network on top of the fixed pre-trained model. The authors show that such latent replay improves performance, especially for tasks that are different from the distribution used to train the initial model. In this work, we show that adaptation of pre-trained models is essential for strong continual learning performance. However, different from all existing continual learning works, we show that adapting the representation only in the first session is sufficient to obtain representative features for the classes being learned and that this forms a strong baseline for future continual learning methods.

In addition to the incremental classification in the full data regime, we focus on the few-shot scenario where only a few labeled examples per class are provided at each training session. [9] utilize a meta-trained network for few-shot classification and explore multiple design choices including a no-adaptation baseline and a method to optimize orthogonal prototypes with the last classification layer only. The method heavily relies on a pre-training and meta-training step on a large subset of classes from the original dataset. In this work, we show that the meta-training step is not essential and that a simple LDA head is sufficient without the reliance on a big initial training stage. In [11] the authors propose a new distillation loss functioning at the feature map level where importance values are estimated per feature map along with replay and consider an NCM classifier at test time. [45] heavily relies on the training of an initial session using a specific loss function that takes into account classes that will be added in future. It further requires the total number of classes to be known *a priori* which is unrealistic for real applications. Our solution adapts FILM parameters to the first session data and fixes the representation for the remaining training sessions. We show that surprisingly few samples in the first session are sufficient for adaptation and that our solution is more powerful than current few-shot continual learning solutions.

3. Proposed Algorithm

In this section, first we discuss the main components of the proposed methods FSA/FSA-FiLM, namely body adaptation techniques and classifier heads. Then we formally introduce the two methods for tackling CIL.

3.1. Problem Formulation

In CIL, we are given a dataset $\mathcal{D}_s = \{\mathbf{x}_{i,s}, y_{i,s}\}_{i=1}^{N_s}$ for each session $s \in \{1, \dots, S\}$, where $X_s = \{\mathbf{x}_{i,s}\}_{i=1}^{N_s}$ is a set of images and $Y_s = \{y_{i,s}\}_{i=1}^{N_s}$ is the set of the corresponding labels with $y_{i,s} \in \mathcal{Y}_s$. Here $\mathcal{Y}_s$ is the label space of session s. It is common in CIL that label spaces are mutually exclusive across sessions, i.e. $\mathcal{Y}_s \cap \mathcal{Y}_{s'} = \varnothing$, $\forall s \neq s'$ and we only have access to $\mathcal{D}_s$ in the current session s to train our model. The proposed FSA method can naturally handle sessions with overlapping label spaces too, however, we follow the former paradigm in our experiments. Another typical assumption in CIL is that the data in all sessions come from the same dataset. We also adopt this assumption in this work, although our experiments on DomainNet are a step toward considering different datasets in each session.

In session s, we will use models defined as

$$f_s(\mathbf{x}) = W_s^\top g_{\boldsymbol{\theta}_s}(\mathbf{x}) + \mathbf{b}_s, \qquad (1)$$

where $g_{\boldsymbol{\theta}_s}(\mathbf{x}) \in \mathbb{R}^d$ is a feature extractor backbone[1] with session-dependent parameters $\boldsymbol{\theta}_s$. The linear classifier head comprises the class weights $W_s \in \mathbb{R}^{d \times |\mathcal{Y}_{1:s}|}$ and biases $\mathbf{b}_s \in \mathbb{R}^{|\mathcal{Y}_{1:s}|}$, and $\mathcal{Y}_{1:s} = \cup_{j=1}^{s} \mathcal{Y}_j$ is the label space of all distinct classes we have seen up to session s. Ideally, when a new dataset $\mathcal{D}_s$ is available at the s-th session, the model should be able to update the backbone parameters $\boldsymbol{\theta}$ and the linear classifier head W_s with the minimum computational overhead and without compromising performance over the previously seen classes $\mathcal{Y}_{1:s}$.

Backbone adaptation. Traditionally, the adaptation of the body at the current session s requires updating the full set of the network parameters $\boldsymbol{\theta}_s$ (full-body adaptation). Options include using specifically designed loss functions that mitigate catastrophic forgetting or using a memory buffer that stores a subset of the previously encountered data. See [16] for a review of continual learning techniques. Nevertheless, when training data availability is scarce relative to the size of the model then full-body adaptation ceases to be suitable, and few-shot learning techniques [33], such as meta-learning [10] and transfer learning [40] become favorable.

Recent works in few-shot learning [26] showed that a pre-trained backbone can be efficiently adapted by keeping its parameters $\boldsymbol{\theta}_s$ frozen and introducing Feature-wise Linear Modulation (FiLM) [23] layers with additional parameters $\boldsymbol{\xi}_s$ which scale and shift the activations produced by a convolutional layer. In [26], these parameters are generated by a meta-trained network when a downstream task is

[1] In this paper, we interchangeably use the terms feature extractor, backbone, and body.

given. Alternatively, $\boldsymbol{\xi}_s$ can be learned (fine-tuned) by the downstream data as in [27]. In this work, we consider both meta-learned (Supplement) and fine-tuned (Sec. 4) FiLM parameters.

Classifier heads. In both offline and CIL settings, (linear) classifier heads can be divided into two groups; parametrized and parameter-free. Parametrized heads require the weights/biases in Eq. (1) to be updated by an iterative gradient-based procedure whilst for parameter-free heads, a closed-form formula is available that directly computes the weights/biases after the update of the backbone. We consider three different heads (for notational simplicity we consider the offline setting, and thus suppress session s dependence), one parametrized, and two parameter-free.

- The linear (learnable) head where $W, \mathbf{b}$ are learned.
- The Nearest Class Mean (NCM) classifier where the weight and bias of the k-th class is given by

  $$\mathbf{w}_k = \hat{\boldsymbol{\mu}}_k \text{ and } b_k = \ln \frac{|X^{(k)}|}{N} - \frac{1}{2}\hat{\boldsymbol{\mu}}_k^\top \hat{\boldsymbol{\mu}}_k, \tag{2}$$

  where $X^{(k)} = \{\mathbf{x}_i : y_i = k\}$ is the set of images belonging in class k and $\hat{\boldsymbol{\mu}}_k = \frac{1}{|X^{(k)}|}\sum_{\mathbf{x}\in X^{(k)}} g_{\boldsymbol{\theta}}(\mathbf{x})$ is the mean vector of the embedded images of class k.
- The Linear Discriminant Analysis (LDA) classifier with weights/biases defined as

  $$\mathbf{w}_k = \tilde{S}^{-1}\hat{\boldsymbol{\mu}}_k \text{ and } b_k = \ln \frac{|X^{(k)}|}{N} - \frac{1}{2}\hat{\boldsymbol{\mu}}_k^\top \tilde{S}^{-1}\hat{\boldsymbol{\mu}}_k, \tag{3}$$

  where $\tilde{S} = S + I_d$ is the regularized sample covariance matrix with sample covariance matrix $S = \frac{1}{|X|-1}\sum_{\mathbf{x}\in X}(g_{\boldsymbol{\theta}}(\mathbf{x}) - \hat{\boldsymbol{\mu}})(g_{\boldsymbol{\theta}}(\mathbf{x}) - \hat{\boldsymbol{\mu}})^\top$, $\hat{\boldsymbol{\mu}} = \frac{1}{|X|}\sum_{\mathbf{x}\in X} g_{\boldsymbol{\theta}}(\mathbf{x})$, and identity matrix $I_d \in \mathbb{R}^{d\times d}$.

Both NCM and LDA classifiers are suitable for CIL since their closed-form updates in Eq. (2) and Eq. (3) support exact, computationally inexpensive, continual updates (via running averages) that incur no forgetting. Updating the linear head, in contrast, requires more computational effort and machinery when novel classes appear in a new session. Notice that setting $\tilde{S} = I_d$ in LDA recovers NCM.

The continuous update of the LDA head is not as straightforward as it is for NCM, since LDA also requires updating the sample covariance S. We discuss how this can be attained in the next section where we introduce our proposed method FSA/FSA-FiLM.

3.2. First Session Adaptation

Motivated by the efficient adaptation techniques and CIL-friendly classifiers discussed in the previous section, we propose two simple and computationally efficient CIL methods called First Session Adaptation (FSA) and FSA-FiLM. FSA is based on a full-body adaptation while FSA-FiLM uses FiLM layers to adapt the body. Both methods utilize pre-trained backbones. FSA (-FiLM) (i) adapts the backbone only at the first session and then the backbone remains frozen for the rest of the sessions, and (ii) makes use of an LDA classifier which is continuously updated as new data become available. For adapting the body (either full or FiLM-based adaptation) at the first session, we use a linear head and a cross-entropy loss first, and then after the optimization is over, the linear head is removed and an LDA head is deployed instead, based on the optimized parameters $\boldsymbol{\theta}^*$ and Eq. (3). Updating the LDA head when the data of the next session becomes available, can be done by using Algorithm 1 recursively until the last session S. Specifically, by having access to the running terms $\{A, \mathbf{b}, \text{count}\}$, the sample covariance matrix is given by

$$S = \frac{1}{\text{count} - 1}\left(A - \frac{1}{\text{count}}\mathbf{b}\mathbf{b}^\top\right). \tag{4}$$

The time complexity of LDA scales as $\mathcal{O}(d^3 + |\mathcal{Y}_s|d^2)$ and its space complexity is $\mathcal{O}(d^2 + |\mathcal{Y}_{1:s}|d)$ at the s-th session. This computational burden is negligible compared to taking gradient steps for learning a linear head and as we will see in Sec. 4.2 using covariance information boosts performance significantly against the vanilla NCM head.

Algorithm 1 Update of sample covariance running terms

```
Require: $g(\cdot) \equiv$ a feature extractor backbone
Require: $X_s = \{\mathbf{x}_{i,s}\}_{i=1}^{N_s}$: Images of session $s$
Require: $A \in \mathbb{R}^{d\times d}, \mathbf{b} \in \mathbb{R}^d, \text{count} \in \mathbb{N}^*$ if $s > 1$
function IncUpdate($g(\cdot), X_s, A, \mathbf{b}, \text{count}$)
    if $s = 1$ then                                  ▷ Initialize $A, \mathbf{b}, \text{count}$
        $A \leftarrow \mathbf{0}_{d\times d}, \mathbf{b} \leftarrow \mathbf{0}_d, \text{count} \leftarrow 0$
    end if
    $A \leftarrow A + \sum_{\mathbf{x}\in X_s} g(\mathbf{x})g(\mathbf{x})^\top$
    $\mathbf{b} \leftarrow \mathbf{b} + \sum_{\mathbf{x}\in X_s} g(\mathbf{x})$
    count $\leftarrow$ count $+ N_s$
    return $A, \mathbf{b}, \text{count}$
end function
```

4. Experiments

In this section, we present a series of experiments to investigate the performance of FSA (-FiLM) under different CIL settings. First, we detail the datasets used for the experiments and discuss our implementation. Second, we perform a large empirical comparison between LDA, linear, and NCM classifiers, in combination with different body adaptation techniques, in the offline setting. For the CIL settings, the results are compared to the vanilla method, *No*

Adaptation (NA), where a pretrained backbone is combined with an LDA head and remains frozen across sessions. We also consider how these findings are affected by the number of shots available. Third, we compare FSA (-FiLM) with state-of-the-art continual learning methods and conduct ablation studies. Finally, we investigate the effect of adapting the backbone when the CIL dataset is "similar" to ImageNet-1k and discuss a similarity metric that indicates whether body adaptation is required.

4.1. Datasets and Implementation details

Datasets. We employ a diverse collection of 26 image-classification datasets across the offline and CIL experiments. For the offline experiments of Sec. 4.2, we use the 19 datasets of VTAB [42], a low-shot transfer learning benchmark, plus three additional datasets, FGVC-Aircraft [17], Stanford Cars [12], and Letters [6]. We refer to this group of datasets as VTAB+. For the CIL-based experiments, we choose 5 VTAB+ datasets from different domains, having an adequate number of classes to create realistic CIL scenarios. The datasets are CIFAR100, SVHN, dSprites-location, FGVC-Aircraft, Cars, and Letters while also including 4 extra datasets: CUB200 [32], CORE50 [15], iNaturalist [30], and DomainNet [22]. Exact details for each dataset are provided in the Supplement.

Training details. All models are implemented with PyTorch [20]. We use a pre-trained EfficientNet-B0 [28] on Imagenet-1k as the main backbone for all methods. For the few-shot+ CIL experiment in Sec. 4.3, we also consider two ResNet architectures, ResNet-18 and ResNet-20 [8] to enable direct comparison to the original settings used in [45]. All the deployed backbones (except ResNet-20, due to the unavailability of pre-trained weights on ImageNet-1k), are pre-trained on ImageNet-1k for all methods. We keep the optimization settings the same across all baselines for fairness. Optimization details are given in the Supplement.

Evaluation protocol. We report the Top-1 accuracy after the last continual learning session evaluated on a test set including instances from all the previously seen classes. In the Supplement, we provide the accuracy after each session. To quantify the forgetting behavior, we use a scaled modification of the performance dropping rate [29], namely percent performance dropping rate (PPDR), defined as PPDR $= 100 \times \frac{\mathcal{A}_1 - \mathcal{A}_S}{\mathcal{A}_1}\%$, where $\mathcal{A}_1$ denotes test accuracy after the first session, and $\mathcal{A}_S$ the test accuracy after the last session.

4.2. Head Comparisons

To motivate the choice of the LDA head for FSA (-FiLM), we compare its average accuracy across all VTAB+ datasets with that of the linear and NCM classifiers under the *offline* setting where all the classes are available and no incremental learning is required. For the body, we use no adaptation (NA) as a baseline (i.e. using the unadapted original backbone), FiLM-based adaptation, and full-body adaptation while ranging the number of shots from 5 to 10, then 50, and finally using all available training data. As Fig. 1 illustrates, LDA consistently outperforms both NCM and linear heads when the number of shots is equal to 5 and 10, regardless of adaptation technique. Notice also that as the number of learnable parameters increases, the choice of the head plays a less significant role in the predictive power of the method. Overall, LDA performs similarly to the linear head in the high-shot settings and performs the best of all heads in the low-shot settings. Finally, the full covariance information of LDA provides a consistent advantage over its isotropic covariance counterpart, i.e. the NCM classifier. This is in agreement with the results presented in [27] for the LDA and NCM classifier. In addition to the results presented here, we have also tested meta-learning based body adaptation methods [3, 26, 2] which support continual learning out of the box. We find these perform poorly compared to the fine-tuning based methods. See next section for more results.

4.3. Class-Incremental Learning Comparisons

We consider three different CIL scenarios: (i) high-shot CIL, (ii) few-shot+ CIL, and (iii) few-shot CIL. We compare our FSA/FSA-FiLM methods with recent state-of-the-art FSCIL methods, including Decoupled-Cosine [31], CEC [43], FACT [45], and ALICE. Additionally, for the high-shot CIL setting, we consider a strong replay-based baseline for CIL, GDumb [24] and a competitive replay-free method, E-EWC+SDC [41]. Finally, we introduce an additional baseline adapter inspired by [37], called FSA-LL (Last Layer). In FSA-LL only the parameters of the backbone's last block are fine-tuned which can be compared to the FSA-FiLM adaption method.

High-shot CIL. In this setting, we consider all the available training data for each class while keeping the number of novel classes in each session low. We use CIFAR100, CORE50, SVHN, dSprites-loc, FGVC-Aircraft, Cars, and Letters for our experiments. For CIFAR100, CORE50, and FGVC-Aircraft, the first session consists of 10 classes, 4 for dSprites-loc, 16 for Cars, and 12 for letters. The rest of the sessions include 10 classes for CIFAR100, FGVC-Aircraft, 20 for Cars, 5 for CORE50 and Letters, 2 for SVHN and dSprites-loc. Therefore, the total number of incremental sessions for CIFAR100, FGVC-Aircraft, and Cars is 10 while for CORE50 we have 9 sessions. A pre-trained on Imagenet-1k EfficientNet-B0 is deployed as a backbone for all methods. FSA-FiLM outperforms all the competitors by a significant margin on all datasets except dSprites-loc. We attribute the performance gap on dSprites-loc due to the

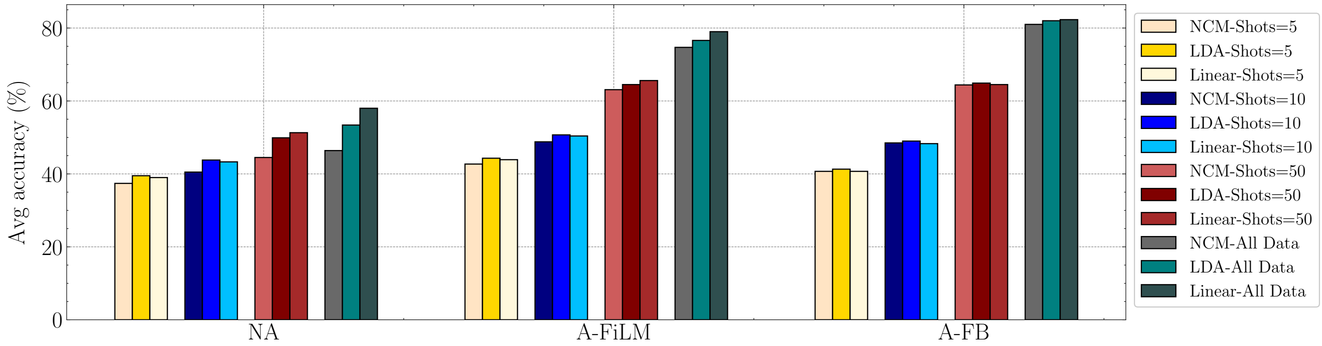

Figure 1. Average accuracy across all VTAB+ datasets using no-adaptation (NA), FiLM adaptation (A-FiLM), and full body adaptation (A-FB) for different classifier heads (NCM, LDA, Linear) and number of shots (5, 10, 50, All Data). The results correspond to the offline setting where all classes are available without any incremental learning.

Method	**CIFAR100**	**CORE50**	**SVHN**	**dSprites-loc**	**FGVC-Aircraft**	**Cars**	**Letters**	**Avg Diff**
NA	68.2 (26.8)	82.6 (14.2)	39.9 (51.1)	20.6 (54.0)	41.3 (1.7)	43.3 (40.3)	68.4 (24.1)	0.0
E-EWC+SDC [41]	32.4 (66.7)	21.7 (78.1)	39.5 (60.1)	18.6 (81.4)	25.6 (55.8)	30.0 (62.6)	33.6 (66.3)	-23.3
FACT [45]	10.2 (89.4)	22.0 (77.7)	33.8 (65.9)	6.4 (93.6)	4.7 (90.3)	0.6 (99.3)	20.9 (79.1)	-38.0
ALICE [21]	52.4 (45.8)	72.8 (25.8)	46.1 (53.6)	68.3 (31.7)	39.8 (35.0)	36.4 (56.0)	75.7 (24.2)	+3.9
FSA	62.8 (34.5)	82.8 (15.5)	71.3 (26.6)	**91.5** (**8.5**)	50.8 (6.3)	50.3 (36.6)	78.4 (21.4)	+17.7
FSA-LL	60.5 (37.2)	79.0 (19.2)	64.6 (33.1)	91.3 (8.5)	45.4 (21.7)	45.7 (43.8)	77.2 (22.7)	+14.2
FSA-FiLM	**73.8** (**23.4**)	**85.4** (**13.3**)	**75.9** (**23.4**)	76.9 (22.8)	**55.9** (**-5.7**)	**55.9** (**30.2**)	**79.7** (**20.0**)	**+19.9**
GDumb [24]	54.5 (42.2)	82.4 (15.0)	78.2 (19.6)	79.5 (12.9)	25.3 (56.9)	14.2 (82.6)	70.1 (27.0)	+5.7
Offline-FiLM	78.2	88.4	93.1	98.5	67.5	67.3	85.2	+30.6

Table 1. Last session's test accuracy (%) (↑) and the PPDR (%) (↓) in parentheses, for the high-shot CIL setting (Sec. 4.3). The last column reports the average accuracy difference (↑) across all datasets between a baseline and NA. A pre-trained EfficientNet-B0 on Imagenet-1k is used as a backbone for all methods. For reference, we include the replay-based baseline GDumb where 1k images are used for the memory buffer.

large number of data points (25k) and the portion of the classes (25%) in the first session. It is also apparent the efficiency of fine-tuning FiLM parameters over fine-tuning the parameters of the model's last layer; note the number of FiLM parameters is only 20.5k while the last layer of an EfficientNet-B0 comprises 2.9M parameters. Furthermore, fine-tuning the FiLM parameters offer almost 20% accuracy increase on average compared to using a pretrained model without doing any further adaptation. Interestingly, the replay-free FSA-FILM is able to outperform significantly the replay-based method GDumb with a 1k memory buffer on most of the datasets; accuracy and time comparisons between FSA-FiLM and GDumb with varying buffer sizes can be found in Figure 1 of the Supplement. Regarding FACT's perfromance, although it starts from a pre-trained backbone, it was developed under the assumption that the first session contains lots of data and a high fraction of the classes that will be encountered ($>$ 50%) which is not true for the setting treated in Table 1. FACT overfits in this setting and this results in poor performance. It turns out that FACT's assumptions about the first session are a strong requirement which is not necessary for obtaining good performance as our FSA baseline shows.

Few-shot+ CIL. This setting is the one that has most commonly been used for few-shot CIL (FSCIL) and it involves an initial session that contains a large number of classes (around 50-60% of the total number of classes in the dataset) and all the available training images of these classes. The remaining sessions comprise a small number of classes and shots (typically 5/10-way 5-shot). Here we follow the exact FSCIL settings as described in [45, 21] for CIFAR100 and CUB200. We use ResNet-18/20 and EfficientNet-B0 as backbones. Table 2 summarizes the performance comparison between baselines. FSA performs on par with FACT on CIFAR100 when we use the original backbone used in [45], and it outperforms FACT by almost 10% and ALICE by 3.4% when EfficientNet-B0 is utilized while FSA-FiLM exhibits the lowest PPDR score. Notice also that FSA is only marginally worse than its offline counterpart, meaning there is little room for continuous body adaptation to improve things further. For CUB200, FSA with an EfficientNet-B0 performs on par with ALICE. Interestingly, we observe that NA performs well on this dataset. This indicates that CUB200 is not far from ImageNet-1k. The current results of FSA set new SOTA performance on CIFAR100 for the FSCIL setting.

		Datasets	
Method	**Backbone**	**CIFAR100**	**CUB200**
CEC [43]	RN-20	49.1 (32.7)*	-
FACT [45]		52.1 (30.2)*	-
FSA		52.0 (30.8)	-
NA	RN-18	50.4 (26.8)	50.0 (29.3)
CEC [43]		-	52.3 (31.1)*
FACT [45]		49.5 (34.8)*	56.9 (25.0)*
ALICE [21]		54.1 (31.5)†	60.1 (22.4)†
FSA-FiLM		55.2 (24.4)	52.7 (27.6)
FSA		61.4 (25.1)	57.6 (24.3)
NA	EN-B0	55.2 (25.8)	63.2 (**19.6**)
FACT [45]		56.5 (34.6)	62.9 (23.3)
ALICE [21]		62.7 (28.4)	**63.5** (22.2)
FSA-LL		61.4 (25.7)	55.9 (22.3)
FSA-FiLM		61.8 (**22.4**)	62.9 (20.4)
FSA		**66.1** (24.6)	63.4 (20.9)
Offline	EN-B0	67.0	65.1

Table 2. Baseline comparison under the few-shot+ CIL setting (Sec. 4.3). We report the accuracy (%) (↑) of the last session and the PPDR (%) (↓) in parentheses. Asterisk (*) indicates that the reported results are from [45] and † indicates results reported in [21]. We use three different backbones, EfficientNet-B0 (EN-B0) and ResNet-18/20 (RN-18/20); EN-B0 and RN-18 are pre-trained on Imagenet-1k.

Few-shot CIL. The final setting, which is firstly introduced in this work, considers an alternative FSCIL setting in which only a small number of data points are available in all sessions, including the first. We use 50 shots per session while the first session includes 20% of the total number of classes of the dataset. Each of the remaining sessions includes around 10% of the total number of classes; more details are available in the Supplement. We repeat experiments 5 times and we report the mean accuracy and PPDR in Table 3. FSA-FiLM outperforms all the other baselines by a large margin in terms of both accuracy and PPDR, indicating that transfer learning is considerably advantageous for CIL when the data availability is low. Notice that both ALICE and FACT struggle to achieve good performance under this setting due to the limited amount of data in the first session. We find that FGVC-Aircraft exhibits a positive backward transfer behavior that we attribute to a difficult initial session followed by sessions that contain classes that are comparatively easier to distinguish.

Finally, we demonstrate the suitability of the LDA head in FSA-FiLM compared to a nearest class mean head, denoted FSA-FiLM-NCM. In the continual learning setting, LDA gives far larger improvements over NCM; e.g. 10% on average as presented in Table 3. This highlights the importance of a strong classification layer. Note that the incremental update of LDA does not require any replay buffer as opposed to a linear head.

4.4. Inhomogeneous Class-Incremental Learning

FSA adapts the body only on the first session of continual learning and it is therefore likely to be sensitive to the particular classes which are present in this session. To investigate the degree of performance sensitivity for FSA, we devise a setting similar to the few-shot CIL setting of Sec. 4.3, where each session includes classes that share some common attribute. We select CIFAR100 which provides superclass information, and DomainNet which consists of different domains and also has superclass information available. We create three distinct CIL configurations for each dataset, each of which has different types of data in the first session. For CIFAR100, we split the data into 19 sessions. The first session includes 10 classes from super-classes (i) aquatic mammals and fish (*Aq*), (ii) electric devices and furniture (*DevF*), or (iii) Vehicles 1 and Vehicles 2 (*Veh*). Each of the other 18 sessions include images from the remaining 18 super-classes. Similarly, for DomainNet, we split the data into 6 sessions using 50 shots with 10 classes in each session. The first session includes 10 classes of (i) the "electricity" superclass of the real domain (*El-R*), (ii) the "furniture" superclass of the clipart domain (*F-Clp*), or (iii) the "transportation" superclass of the sketch domain (*Tr-Sk*). In this way, we can vary the content of the first session and analyze the effect this has on performance. Table 4 reveals that FSA-FiLM's performance is similar even if the data in the first session used for the body adaptation appears disparate from that contained in the remaining sessions. Even for DomainNet where the distribution shift of the data across sessions is considerable, performance is only marginally affected in each of the three settings. This provides evidence that adapting the body only on the first session achieves competitive performance regardless of the class order, given the assumption that the data come from a single dataset (albeit a varied one in the case of DomainNet).

4.5. When to adapt the body?

Despite the strong performance of FSA (-FiLM) in the few-shot settings of Sec. 4.3, there are cases where the NA method achieves very close accuracy to FSA (e.g. see CUB200 results in Table 2). This implies that there may be datasets where adaptation is not required and all we need is the pre-trained backbone. In order to decide whether we require body adaptation or not, we compute the minimum cosine distance in the embedding space of the pre-

Method/Dataset	CIFAR100	SVHN	dSprites-loc	FGVC-Aircraft	Letters	DomainNet	iNaturalist	Avg Diff
NA	57.4 (28.4)	28.3 (61.5)	11.9 (66.6)	41.0 (-15.5)	57.6 (29.9)	69.0 (17.2)	49.7 (4.3)	0.0
FACT [45]	16.8 (79.7)	24.1 (66.2)	11.7 (64.1)	8.3 (79.9)	49.8 (40.9)	20.6 (75.6)	14.3 (74.0)	-24.2
ALICE [21]	58.0 (33.8)	23.0 (65.9)	23.0 (46.0)	42.0 (27.5)	66.5 (31.4)	66.5 (25.1)	47.9 (13.7)	+1.7
FSA	60.3 (27.0)	32.9 (53.5)	33.7 (**41.3**)	50.1 (-16.8)	62.2 (28.5)	70.3 (17.5)	51.5 (**1.1**)	+6.6
FSA-LL	62.0 (25.7)	43.5 (47.0)	18.8 (61.7)	45.8 (-2.1)	69.4 (26.1)	67.6 (16.9)	49.2 (14.1)	+5.9
FSA-FiLM-NCM	66.4 (24.8)	38.8 (56.5)	25.1 (59.4)	42.5 (-7.5)	57.1 (34.9)	68.8 (18.5)	51.5 (12.1)	+5.0
FSA-FiLM	**70.9** (**20.5**)	**51.3** (**43.5**)	**35.7** (43.1)	**55.8** (**-19.8**)	**73.4** (**22.1**)	**74.0** (**15.6**)	**58.8** (4.8)	+15.0
Offline-FiLM	73.8	77.2	83.7	65.1	79.7	75.1	62.1	+28.0

Table 3. Accuracy (%) (↑) of the last session and PPDR (%) (↓) in parentheses for the few-shot CIL setting of Sec. 4.3. The last column reports the average accuracy difference (↑) across all datasets between a baseline and NA. A pre-trained EfficientNet-B0 on ImageNet-1k is used as a backbone for all methods. FSA-FiLM-NCM utilizes an NCM classifier while NA, FSA (-LL, FiLM) and Offline uses an LDA head.

Dataset	NA	Aq	DevF	Veh
CIFAR100	57.4±1.0	66.1±1.6	66.2±1.9	67.9±1.2
	NA	**El-R**	**F-Clp**	**Tr-Sk**
DomainNet	69.0±0.4	70.6±0.6	71.7±0.6	72.8±0.5

Table 4. Last session accuracy (%) (↑) of FSA-FiLM for three different session splits on CIFAR100 and DomainNet as described in Sec. 4.4. For reference, we also include the accuracy of NA. Results are averaged over 5 runs (mean±std).

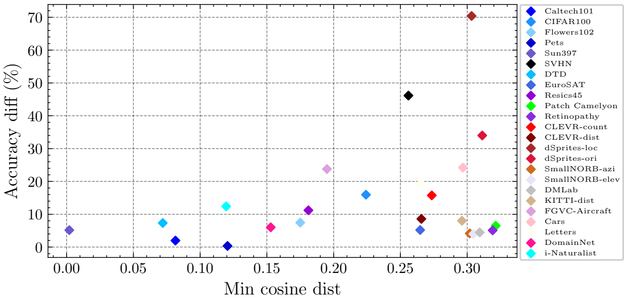

Figure 2. Scatter plot of the accuracy differences between FSA-FiLM and NA against the minimum cosine distance between a dataset and *mini*Imagenet dataset evaluated using the NA method. We consider the offline setting with 50 shots. A pre-trained EfficientNet-B0 on ImageNet-1k is used as a backbone.

trained backbone between the downstream dataset and the *mini*Imagenet [31] dataset. We use *mini*Imagenet (60k images) as a proxy of Imagenet-1k (1.3M images) to reduce the computational overhead of evaluating pairwise distances. This allows us to approximately measure the dissimilarity between the downstream dataset and Imagenet-1k.

Fig. 2 shows the accuracy difference between FSA-FiLM and NA as a function of the cosine distance for the offline setting with 50 shots whilst Fig. 3 illustrates the same accuracy difference where the datasets are grouped based on the VTAB+ categorization. For this experiment, we use 24 datasets (VTAB+, DomainNet, and iNaturalist). We observe that adaptation is more beneficial for datasets in the structured domain, which contain images that are dissimilar to those of ImageNet-1k.

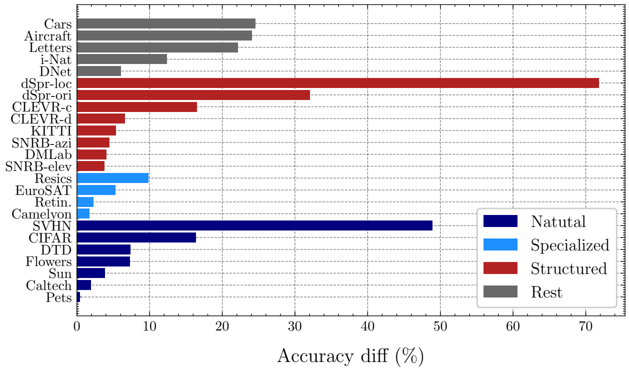

Figure 3. Bar plot of the accuracy differences between FSA-FiLM and NA for the offline case with 50 shots.

	CFR100	SVHN	dSp-loc	FGVC	Letters	DNet	iNat
EffNet-B0 (FSA-FiLM)	70.9	**51.3**	**35.7**	**55.8**	**73.4**	74.0	58.8
ConvNext (NA)	**87.1**	43.6	13.6	50.2	63.1	**82.9**	**72.6**
ConvNext-CosD	0.2	0.6	0.6	0.3	0.4	0.3	0.1

Table 5. Accuracy comparison between a pre-trained ConvNext on ImageNet-22k without body adaptation (NA) and a pre-trained EfficientNet-B0 on ImageNet-1k which is adapted on the first session (FSA-FiLM). We use the few-shot CIL setting (Sec. 4.3) and we report the accuracy (%) (↑) after the last session. Results are averaged over 5 runs (mean±std). We also include the minimum cosine distance using the pre-trained ConvNext as in Figure 2.

To stress the importance of adaptation on datasets far from ImageNet, we compare FSA-FiLM with an EfficientNet-B0 backbone to the no-adaptation method with a ConvNext [14] pre-trained on Imagenet-22k. The total number of parameters for FSA-FiLM (backbone and FiLM parameters) is ∼4M while for ConvNext is 348M. Table 5 shows that a small adapted backbone can significantly surpass the accuracy of a much larger pre-trained backbone for datasets far from ImageNet. We also compute the cosine distance measure for ConvNext and find that it is predictive of the performance of the unadapted ConvNext model and the benefits of adaptation. We show this in Table 5; note that for CIFAR100 the ConvNext's cosine distance values are small relative to the other datasets (indicating that CIFAR100 is close to the pretraining distribution) whereas for EfficientNet the values in Figure 2 indicate that CIFAR100 is more out-of-distribution.

The main takeaway from these results is that when there is a large cosine distance (e.g. SVHN, dSprites, FGVC, Letters) FiLM adaptation of a light-weight backbone performs well – better even than applying LDA head adaptation to a much larger backbone trained on a much larger dataset. As the community employs ever-larger models and datasets, these results indicate that adapters are likely to continue to bring improvements over simply learning a classifier head (the NA baseline).

5. Discussion

We have presented FSA (-FiLM), a simple yet effective replay-free baseline for CIL which adapts a pre-trained backbone via full body adaptation or FiLM layers only at the first session of continual learning and then utilizes a flexible incrementally updated LDA classifier on top of the body. Extensive experiments in several CIL scenarios have shown that FSA outperforms previous SOTA baselines in most cases, thus, questioning the efficacy of current approaches to continuous body adaption. Furthermore, the experiments have revealed that FiLM layers are helpful when the amount of data available in the target domain is relatively low and the number of parameters in the neural network and the size of the source domain data set is large. For very large foundation models trained on very large source domain data sets, it is likely we are often in this situation and FiLM-like layers will be more effective than full-body adaptation, even when the target domain has a fairly large amount of data.

Limitations and future work. The main limitation of this work is inextricably linked with the distributional assumptions of CIL in general. If the data distribution at each session shifts considerably (e.g. the first session includes natural images while the next session includes images of numbers) then first session body adaptation is not suitable. In future work, we plan to combine FSA (-FiLM) with a memory method like GDumb to get the best of both worlds and deal with more challenging CIL scenarios.

References

[1] 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 (ICCV)*, pages 844–853, October 2021. 2

[2] Peyman Bateni, Raghav Goyal, Vaden Masrani, Frank Wood, and Leonid Sigal. Improved few-shot visual classification. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 14493–14502, 2020. 2, 5

[3] John Bronskill, Daniela Massiceti, Massimiliano Patacchiola, Katja Hofmann, Sebastian Nowozin, and Richard Turner. Memory efficient meta-learning with large images. *Advances in Neural Information Processing Systems*, 34:24327–24339, 2021. 5

[4] Lucas Caccia, Rahaf Aljundi, Nader Asadi, Tinne Tuytelaars, Joelle Pineau, and Eugene Belilovsky. New insights on reducing abrupt representation change in online continual learning. *arXiv preprint arXiv:2203.03798*, 2022. 2

[5] MohammadReza Davari, Nader Asadi, Sudhir Mudur, Rahaf Aljundi, and Eugene Belilovsky. Probing representation forgetting in supervised and unsupervised continual learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 16712–16721, 2022. 2

[6] T. E. de Campos, B. R. Babu, and M. Varma. Character recognition in natural images. In *Proceedings of the International Conference on Computer Vision Theory and Applications, Lisbon, Portugal*, February 2009. 5

[7] Matthias De Lange, Rahaf Aljundi, Marc Masana, Sarah Parisot, Xu Jia, Aleš Leonardis, Gregory Slabaugh, and Tinne Tuytelaars. A continual learning survey: Defying forgetting in classification tasks. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 44(7):3366–3385, 2021. 2

[8] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 770–778, 2016. 5

[9] Michael Hersche, Geethan Karunaratne, Giovanni Cherubini, Luca Benini, Abu Sebastian, and Abbas Rahimi. Constrained few-shot class-incremental learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9057–9067, 2022. 3

[10] Timothy Hospedales, Antreas Antoniou, Paul Micaelli, and Amos Storkey. Meta-learning in neural networks: A survey. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 44(9):5149–5169, 2021. 3

[11] 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. 2, 3

[12] Jonathan Krause, Michael Stark, Jia Deng, and Li Fei-Fei. 3d object representations for fine-grained categorization. In

Proceedings of the IEEE International Conference on Computer Vision Workshops, pages 554–561, 2013. 5

[13] Timothée Lesort, Oleksiy Ostapenko, Diganta Misra, Md Rifat Arefin, Pau Rodríguez, Laurent Charlin, and Irina Rish. Scaling the number of tasks in continual learning. *arXiv preprint arXiv:2207.04543*, 2022. 2

[14] Zhuang Liu, Hanzi Mao, Chao-Yuan Wu, Christoph Feichtenhofer, Trevor Darrell, and Saining Xie. A convnet for the 2020s. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 11976–11986, 2022. 8

[15] Vincenzo Lomonaco and Davide Maltoni. Core50: A new dataset and benchmark for continuous object recognition. In *Conference on Robot Learning*, pages 17–26. PMLR, 2017. 5

[16] Zheda Mai, Ruiwen Li, Jihwan Jeong, David Quispe, Hyunwoo Kim, and Scott Sanner. Online continual learning in image classification: An empirical survey. *Neurocomputing*, 469:28–51, 2022. 3

[17] Subhransu Maji, Esa Rahtu, Juho Kannala, Matthew Blaschko, and Andrea Vedaldi. Fine-grained visual classification of aircraft. *arXiv preprint arXiv:1306.5151*, 2013. 5

[18] Thomas Mensink, Jakob Verbeek, Florent Perronnin, and Gabriela Csurka. Distance-based image classification: Generalizing to new classes at near-zero cost. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 35(11):2624–2637, 2013. 2

[19] Oleksiy Ostapenko, Timothee Lesort, Pau Rodríguez, Md Rifat Arefin, Arthur Douillard, Irina Rish, and Laurent Charlin. Continual learning with foundation models: An empirical study of latent replay. In *Conference on Lifelong Learning Agents*, pages 60–91. PMLR, 2022. 2, 3

[20] 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, 2019. 5

[21] Can Peng, Kun Zhao, Tianren Wang, Meng Li, and Brian C Lovell. Few-shot class-incremental learning from an open-set perspective. In *Computer Vision–ECCV 2022: 17th European Conference, Tel Aviv, Israel, October 23–27, 2022, Proceedings, Part XXV*, pages 382–397. Springer, 2022. 2, 6, 7, 8

[22] Xingchao Peng, Qinxun Bai, Xide Xia, Zijun Huang, Kate Saenko, and Bo Wang. Moment matching for multi-source domain adaptation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1406–1415, 2019. 5

[23] Ethan Perez, Florian Strub, Harm De Vries, Vincent Dumoulin, and Aaron Courville. FiLM: Visual reasoning with a general conditioning layer. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 32, 2018. 2, 3

[24] Ameya Prabhu, Philip HS Torr, and Puneet K Dokania. GDumb: A simple approach that questions our progress in continual learning. In *European Conference on Computer Vision*, pages 524–540. Springer, 2020. 2, 5, 6

[25] Sylvestre-Alvise Rebuffi, Alexander Kolesnikov, Georg Sperl, and Christoph H Lampert. icarl: Incremental classifier and representation learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2001–2010, 2017. 2

[26] James Requeima, Jonathan Gordon, John Bronskill, Sebastian Nowozin, and Richard E Turner. Fast and flexible multi-task classification using conditional neural adaptive processes. *Advances in Neural Information Processing Systems*, 32, 2019. 3, 5

[27] Aliaksandra Shysheya, John Bronskill, Massimiliano Patacchiola, Sebastian Nowozin, and Richard E Turner. Fit: Parameter efficient few-shot transfer learning for personalized and federated image classification. *arXiv preprint arXiv:2206.08671*, 2022. 4, 5

[28] Mingxing Tan and Quoc Le. Efficientnet: Rethinking model scaling for convolutional neural networks. In *International Conference on Machine Learning*, pages 6105–6114. PMLR, 2019. 5

[29] Xiaoyu Tao, Xiaopeng Hong, Xinyuan Chang, Songlin Dong, Xing Wei, and Yihong Gong. Few-shot class-incremental learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 12183–12192, 2020. 5

[30] Grant Van Horn, Oisin Mac Aodha, Yang Song, Yin Cui, Chen Sun, Alex Shepard, Hartwig Adam, Pietro Perona, and Serge Belongie. The iNaturalist Species Classification and Detection Dataset. In *Proceedings of the IEEE Conference/CVF on Computer Vision and Pattern Recognition*, pages 8769–8778, 2018. 5

[31] Oriol Vinyals, Charles Blundell, Timothy Lillicrap, Daan Wierstra, et al. Matching networks for one shot learning. *Advances in Neural Information Processing Systems*, 29, 2016. 5, 8

[32] C. Wah, S. Branson, P. Welinder, P. Perona, and S. Belongie. Technical Report CNS-TR-2011-001, California Institute of Technology, 2011. 5

[33] Yaqing Wang, Quanming Yao, James T Kwok, and Lionel M Ni. Generalizing from a few examples: A survey on few-shot learning. *ACM Computing Surveys (CSUR)*, 53(3):1–34, 2020. 3

[34] Zifeng Wang, Zizhao Zhang, Sayna Ebrahimi, Ruoxi Sun, Han Zhang, Chen-Yu Lee, Xiaoqi Ren, Guolong Su, Vincent Perot, Jennifer Dy, et al. Dualprompt: Complementary prompting for rehearsal-free continual learning. In *Computer Vision–ECCV 2022: 17th European Conference, Tel Aviv, Israel, October 23–27, 2022, Proceedings, Part XXVI*, pages 631–648. Springer, 2022. 2

[35] Zifeng Wang, Zizhao Zhang, Chen-Yu Lee, Han Zhang, Ruoxi Sun, Xiaoqi Ren, Guolong Su, Vincent Perot, Jennifer Dy, and Tomas Pfister. Learning to prompt for continual learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 139–149, 2022. 2, 3

[36] Guile Wu, Shaogang Gong, and Pan Li. Striking a balance between stability and plasticity for class-incremental learning. In *Proceedings of the IEEE/CVF International Confer-*

ence on Computer Vision (ICCV), pages 1124–1133, 2021. 3

[37] Tz-Ying Wu, Gurumurthy Swaminathan, Zhizhong Li, Avinash Ravichandran, Nuno Vasconcelos, Rahul Bhotika, and Stefano Soatto. Class-incremental learning with strong pre-trained models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9601–9610, 2022. 5

[38] 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 International Conference on Computer Vision (ICCV)*, pages 374–382, 2019. 2

[39] Shipeng Yan, Jiangwei Xie, and Xuming He. Der: Dynamically expandable representation for class incremental learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3014–3023, 2021. 2

[40] Jason Yosinski, Jeff Clune, Yoshua Bengio, and Hod Lipson. How transferable are features in deep neural networks? *Advances in Neural Information Processing Systems*, 27, 2014. 3

[41] 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/CVF conference on computer vision and pattern recognition*, pages 6982–6991, 2020. 5, 6

[42] Xiaohua Zhai, Joan Puigcerver, Alexander Kolesnikov, Pierre Ruyssen, Carlos Riquelme, Mario Lucic, Josip Djolonga, Andre Susano Pinto, Maxim Neumann, Alexey Dosovitskiy, et al. A large-scale study of representation learning with the visual task adaptation benchmark. *arXiv preprint arXiv:1910.04867*, 2019. 5

[43] 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/CVF Conference on Computer Vision and Pattern Recognition*, pages 12455–12464, 2021. 5, 7

[44] Bowen Zhao, Xi Xiao, Guojun Gan, Bin Zhang, and Shu-Tao Xia. Maintaining discrimination and fairness in class incremental learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 13208–13217, 2020. 2

[45] 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/CVF Conference on Computer Vision and Pattern Recognition*, pages 9046–9056, 2022. 3, 5, 6, 7, 8