Must Unsupervised Continual Learning Relies on Previous Information?

Haoyang Cheng, Haitao Wen, Heqian Qiu*, Lanxiao Wang, Minjian Zhang, Hongliang Li*
University of Electronic Science and Technology of China, Chengdu, China
{chenghaoyang, haitaowen, lanxiao.wang, mjzhang_ivip}@std.uestc.edu.cn,
{hqqiu, hlli}@uestc.edu.cn

Abstract

Open-world recognition has recently gained significant attention owing to its ability to bridge the gap between experimental scenarios and real-world applications. Since continual learning can learn from a sequence of dynamic data streams, it obtains extensive applications in open-world recognition. However, because of the production of data annotation is usually time-consuming and labor-intensive in real-world scenarios, it's necessary to develop unsupervised continual learning. Recent studies start to investigate unsupervised continual learning (i.e., UCL), but mainly focus on rehearsal and regularization strategies to enhance the anti-forgetting capability of UCL. In practice, rehearsal and regularization are information-dependent, which require information from previous data as supervised signals, e.g., replayed data and previous model. In this paper, we propose an information-free method, Alternate Task Discrimination (ATD), which is a self-supervised pretext task for continuity and improves anti-forgetting capability via encouraging the model to discriminate which data stream current sample is from. The whole process doesn't rely on any previous information. In order to perform ATD effectively in UCL framework, we design an alternating optimization algorithm where UCL and ATD are optimized respectively. We validate the effectiveness of the proposed method on multiple standard UCL benchmarks, where it obtains considerable improvements compared with baseline methods. In addition, our approach can be used as a plug-in unit, which makes further achievements when collaborated with existing popular UCL methods.

1. Introduction

Recently, open-world recognition has built a bridge between laboratory algorithms and real-world applications, where data is fundamentally dynamic and the model is required to process new data constantly. Continual learning (CL) [1, 6, 20, 26, 33, 40, 41, 47, 50] aims to learn from a sequence of data streams and demands the model to encode new information without forgetting previous knowledge (i.e., resist catastrophic forgetting), which has a wide range of applications in open-world recognition. However, since the production of data annotation is usually time-consuming and labor-intensive in real-world scenarios, the development of unsupervised continual learning is particularly important. Recently, there are some pioneering researches [15, 34] to investigate unsupervised continual learning by introducing unsupervised visual representation learning (i.e., self-supervised learning) [7, 11, 12, 18, 22, 49] to address the challenge of missing annotations. LUMP [34] proposes a novel rehearsal strategy based on dark experience sampling strategy [6], which constructs the

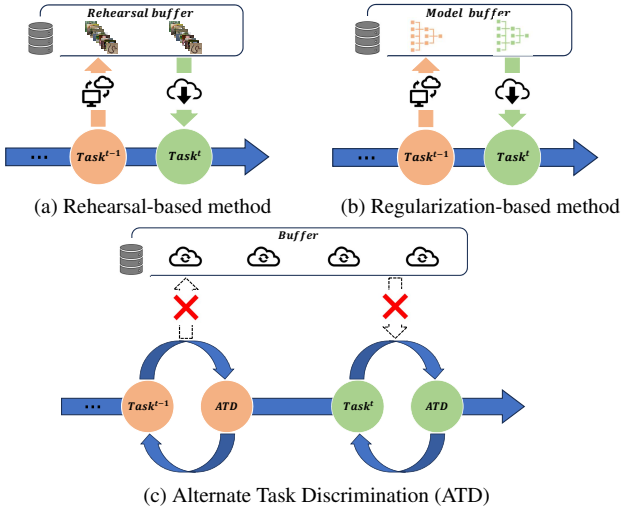

Figure 1. A brief illustration of Rehearsal-based method, Regularization-based method and the proposed ATD. (a) Rehearsal-based method is to replay some traversed samples in future learning. (b) Regularization-based method aims to store learned model to distill previous information. (c) The proposed ATD is a self-supervised pretext task to alleviate catastrophic forgetting without relying on previous information.

*Corresponding authors: Hongliang Li, Heqian Qiu

training set by utilizing mixup [51] to combine replayed samples and current samples. Regularization-based method CaSSLe [15] encourages the invariant feature orientation between current model and previous model via a prediction head. However, rehearsal and regularization strategies are information-dependent, which require information from previous data as supervised signals, e.g., replayed data and previous model, as shown in Figure 1a and 1b respectively. In unsupervised continual learning, rehearsal and regularization strategies pose a great challenge for storage and computing resource with the rapid increase of data amount brought by annotation liberation.

In order to address catastrophic forgetting without introducing previous information to alleviate resource consumption, we propose a novel method, Alternate Task Discrimination (ATD), which is a self-supervised pretext task for continuity without relying on any previous information, as shown in Figure 1c. Specifically, ATD encourages the model to discriminate which data stream current sample is from to alleviate catastrophic forgetting. The intension is that the model is able to capture some discrimination information of the data streams in the process of completing ATD (e.g., data sequence, data attribution, data distribution), which is helpful for the model to preserve the memory of seen data streams, as well as improve anti-forgetting capability. In order to perform ATD effectively in UCL framework, we design an alternating optimization algorithm where UCL and ATD are optimized respectively. Specifically, UCL optimization step is responsible for learning a beneficial representation for various downstream tasks, and ATD optimization step is responsible for preventing the representation from catastrophic forgetting via a self-supervised manner.

In addition, the proposed ATD can be used as a plug-in unit, which can be easily combined with existing UCL methods to obtain further improvements. For rehearsal-based methods, like DER [6] and LUMP [34], ATD can additionally discriminate which data stream the replayed data belongs to, instead of being limited to current data. For regularization-based methods, like CaSSLe [15], ATD can additionally discriminate which data stream the feature representation encoded by previous model belongs to, instead of being limited to current feature mapping.

The proposed method builds the bridge between anti-forgetting ability and self-supervised pretext task for the first time, which doesn't rely on previous information as supervised signals for continuity. We evaluate the performance of the proposed method on several standard UCL benchmarks, including the average accuracy and average forgetting on popular UCL datasets (e.g., Split CIFAR-10 [28], Split CIFAR-100 [28] and Split Tiny-ImageNet [13]) and the average accuracy on out of distribution (OOD) datasets. The proposed method obtains considerable improvements compared with the baseline method, and it achieves consistent improvements while collaborating with different existing UCL methods.

2. Related Work

2.1. Continual learning

Continual learning aims to empower intelligent agent with the ability that learn from a sequence of data without forgetting what it has learned on traversed data. Existing popular continual learning methods can be roughly separated into three categories, including rehearsal-based, regularization-based and architecture-based.

Rehearsal-based methods are to store appropriate samples from traversed data and replay them during future training. [8] proposes the representative replay strategy to store the representative samples and remove the leftover parts. [3] chooses the samples whose constraints best approximate the feasible region, which essentially increases the diversity of replayed data according to parameter gradients. DER [6] selects replayed samples by a dark experience, and preserves the memory by distilling the logits between old and new model. Rainbow Memory (RM) [5] proposes a replay strategy based on classification prediction uncertainty and data augmentation to increase the diversity replayed data. There are also some issues behind the effective anti-forgetting ability of rehearsal-based methods, e.g., the data imbalance of replayed samples and training samples. In order to mitigate this problem, LUCIR [24] proposes a cosine normalization classifier framework with less-forget constraint and inter-class separation. SS-IL [2] performs the separated softmax (SS) in the last layer to solve the bias problem caused by the data imbalance effect. DRI [44] starts with the quantity of samples and utilizes a generative model to supplement replayed data by generating previous data.

Regularization-based methods are to additionally add some regularization constraints to regulate the model optimization. EWC [26] slows down the learning rate on important weights for previous data to prevent catastrophic forgetting. SI [50] is committed to introduce biological networks, which proposes intelligent synapses to track the model parameters and fixes the important part to prevent catastrophic forgetting. LwF [33] introduces distillation [23] to encourage the consistent predictions from previous classification head. Piggyback [35] starts from the point of quantization and pruning, which proposes binary masks to perform on unmodified network weights to get a good performance. UCL [1] trades off catastrophic forgetting and plasticity of models by introducing two regularization terms to fix significant parameters for previous data and control the active parameters.

Architecture-based methods are to assign specific pa-

rameters to different tasks. PNN [40] instantiates a network for each task to prevent catastrophic forgetting, and introduces lateral connections to use previous knowledge to promote plasticity. RCL [46] introduces reinforcement learning to design special network for each task, which not only prevents catastrophic forgetting but also promotes plasticity. LtG [32] introduces architecture search to exploit the optimal network structure for each task. BSA [29] proposes a novel Bayesian framework to learn specific weights for each task to alleviate catastrophic forgetting.

2.2. Unsupervised visual representation learning

Unsupervised visual representation learning, or self-supervised learning, focuses on learning powerful representations from large-scale data with the absence of manual annotations. Early works are devoted to designing heuristic pretext tasks, such as Colorization [30], Inpainting [37], Jigsaw [36] and Rotate prediction [17], hoping to learn beneficial representations for various downstream tasks via solving these pretext tasks. Recently, unsupervised visual representation learning based on instance discrimination pretext task [45], or contrastive learning, has gradually become dominant. In general, contrastive learning advocates that input image should be as close as possible to its augmented view (i.e., positive sample) and far away from other images (i.e., negative sample) in the feature space. SimCLR [11] and MoCo [22] are the most representative methods. SimCLR [11] regards other samples in the batch as negative samples, and utilizes a large batch size to increase the number of negative samples. MoCo [22] decouples the number of negative samples and batch size, which constructs a queue to store negative samples to simulate the dataset dynamically and introduces the momentum encoder to ensure the consistency of negative samples. BYOL [18] empirically shows that negative sample is not necessary for contrastive learning, which additionally introduces a prediction head based on MoCo framework and only encourages similar feature orientation of positive pairs to obtain significant achievements. SimSiam [12] performs an in-depth research of siamese networks in contrastive learning, and provides proof-of-concept experiments to show that stop-gradient operation plays an important role in preventing collapsing. SimSiam [12] additionally gets rid of the momentum encoder in BYOL and obtains competitive achievements as well as decreasing the training cost. BarlowTwins [49] explores the solution of collapsing from another angle, which utilizes the identity matrix to supervise the cross-correlation matrix of two augmented views and achieves competitive results.

In addition, some contrastive learning methods [7, 14, 27, 31, 43] introduce clustering to improve contrastive learning framework by considering the feature similarity of samples. PCL [31] introduces clustering to replace instance discrimination with cluster discrimination, and models cluster discrimination as an expectation maximization process, where the semantic prototypes are optimized by K-means in step E and the augmentation invariance among prototypes is optimized in step M. SwAV [7] proposes an online clustering algorithm based on SeLa [4], which encourages the different augmented views of the same image to maintain the same cluster assignment. In order to alleviate some strong priors in clustering which is not necessary in contrastive learning, MSF [27] aggregates similar samples in the feature space and performs contrastive learning.

Recently, some researches start to focus on preventing the representation from suffering from catastrophic forgetting, whose insight is how to preserve the learned representation and utilize it for subsequent learning. iCaRL [39] is the precursor to perform knowledge distillation [23] on replayed samples and current samples, which utilizes it to learn an anti-forgetting representation. OML [25] and La-MAML [19], which are based on meta-learning, maintain a balance between catastrophic forgetting and plasticity for learned representations by specially designed meta-objectives. Co^{2}L [9] introduces a supervised contrastive learning constraint to improve the quality of learned representation, and finds it is of the ability to prevent catastrophic forgetting. LUMP [34] and CaSSLe [15] are the early work to focus on the continuity of unsupervised representation learning, i.e., Unsupervised Continual Learning (UCL). LUMP [34] introduces the rehearsal strategy of DER [6], and applies mixup [51] to merge the replayed samples and current samples and obtains significant achievements. CaSSLe [15] introduces knowledge distillation with prediction head to encourage the feature consistency between previous model and current model.

3. Method

3.1. Preliminaries

Unsupervised Visual Representation Learning. The proposition of unsupervised visual representation learning is to extract beneficial information for various downstream tasks from unlabeled data. Recent popular methods [12, 18, 49] mainly focus on the augmentation invariance (also called contrastive learning), whose framework details can be summarized as follows.

Given a batch of input image $x = \{x_1, \ldots, x_B\}$, we first generate two batch of augmented views $\mathcal{T}^1(x)$ and $\mathcal{T}^2(x)$ via the standard contrastive learning augmentation strategy $\mathcal{T}(\cdot)$. A view $\mathcal{T}^1(x)$ is fed to online encoder f_θ which is consist of a backbone f_{θ_b} (e.g., ResNet-50 [21]) and a projection head $f_{\theta_{pro}}$, and output the feature representations $z^1 = f_\theta\left(\mathcal{T}^1(x)\right) \in \mathbb{R}^{B\times N}$ where N is the channel size. Another view $\mathcal{T}^2(x)$ is fed to target encoder $f_{\theta'}$ which has the same structure as online encoder and outputs

the corresponding features $z^2 = f_{\theta'}\left(\mathcal{T}^2\left(x\right)\right)$.

BYOL [18] and SimSiam [12] add an extra prediction head $f_{\theta_{pre}}$ for online features z^1 to match target features z^2, i.e. $z^1 = f_{\theta_{pre}}\left(z^1\right)$, and minimize the cosine similarity between the pairwise features:

$$\mathcal{L} = \frac{1}{B}\sum_{i=1}^{B} -\frac{z_i^1}{\|z_i^1\|_2} \cdot \frac{z_i^2}{\|z_i^2\|_2} \tag{1}$$

In practice, BYOL and SimSiam adopt the swap symmetrization strategy to improve the performance. Additionally, target encoder $f_{\theta'}$ is a momentum-based moving average of online encoder f_θ in BYOL, i.e., $\theta' = m * \theta' + (1 - m) * \theta$ where m is a momentum coefficient, which does not participates in the gradient back propagation update. SimSiam shows that the stop-gradient operation plays a important role in avoiding collapsing in the two branch contrastive learning framework, which copies the parameters of online encoder to target encoder and obtains a considerable result, i.e., $\theta' = \theta$.

Instead of applying the stop-gradient operation to avoid collapsing, BarlowTwins [49] constrains the cross-correlation matrix between the pairwise augmentation features to be close to the identity matrix, whose online encoder and target encoder have no difference:

$$\begin{aligned} \mathcal{L} &= \sum_{i=1}^{N}\left(1 - c_{ii}\right)^2 + \lambda \cdot \sum_{i=1}^{N}\sum_{j=1, j\neq i}^{N} {c_{ij}}^2, \\ c_{ij} &= \frac{\sum_{k=1}^{B} z_{k,i}^1 z_{k,j}^2}{\sqrt{\sum_{k=1}^{B}\left(z_{k,i}^1\right)^2}\sqrt{\sum_{k=1}^{B}\left(z_{k,j}^2\right)^2}} \end{aligned} \tag{2}$$

where $C = (c_{ij}) \in \mathbb{R}^{N\times N}$ is the cross-correlation matrix, λ ia a hyper-parameter to trade off the importance of diagonal and non-diagonal elements.

In this work, we mainly focus on SimSiam and BarlowTwins to construct fundamental UCL framework due to their remarkable performance and representativeness, like [34].

Continual Learning (CL). Continual learning aims to learn from non-stationary data distributions, giving the neural networks the ability to continuously learn. Specifically, given a series of task streams $\{1, \ldots, t, \ldots, T\}$ which have different data distributions, continual learning dedicates to training a model across the task streams, keeping acquiring the fresh information without forgetting what it has learned.

Considering task stream t and corresponding data stream $\mathcal{D}_t = \left\{\left(x_{i,t},\ y_{i,t}\right)_{i=1}^{n_t}\right\}$ which is consist of n_t pairs of samples and matching annotations, the trivial continual learning loss $\mathcal{L}_{CL}^t$ (finetune the encoder f_θ and classifier h_ϕ on task stream t) is usually constructed by cross entropy loss (CE):

$$\mathcal{L}_{CL}^t = \mathbb{E}_{\left(x_{i,t},\ y_{i,t}\right)\sim\mathcal{D}_t}\left[CE\left(h_\phi\left(f_\theta\left(x_{i,t}\right)\right),\ y_{i,t}\right)\right] \tag{3}$$

In practice, the model tends to forget what it has learned from previous task streams, which is the notorious catastrophic forgetting. Many methods [3, 6, 8, 26, 33, 40, 48, 50] focus on designing regularization-based, rehearsal-based and architecture-based strategies to alleviate catastrophic forgetting.

Unsupervised Continual Learning (UCL). Recently, [15, 34] introduce unsupervised visual representation learning to learn beneficial representations from non-stationary data distributions without relying on data annotations. The trivial UCL is to finetune the models across the task streams. Formally, considering task stream t and corresponding data stream $\mathcal{D}_t = \left\{\left(x_{i,t}\right)_{i=1}^{n_t}\right\}$ which is consist of n_t samples without any manual annotations, the trivial UCL loss $\mathcal{L}_{UCL}^t$ is constructed based on the unsupervised visual representation learning loss $\mathcal{L}$:

$$\mathcal{L}_{UCL}^t = \mathbb{E}_{\left(x_{i,t}\right)\sim\mathcal{D}_t}\left[\mathcal{L}\left(x_{i,t}\right)\right] \tag{4}$$

UCL also suffers from catastrophic forgetting in practice, where the learned representations prone to lose the information encoded in the previous task streams. In order to alleviate catastrophic forgetting in UCL, LUMP [34] and CaSSLe [15] make an attempt in rehearsal and regularization respectively. However, rehearsal and regularization strategies require information from previous data as supervised signal in practice, e.g., replayed data and previous model, which is information-dependent. In this paper, we propose a novel method, Alternate Task Discrimination (ATD), which is a self-supervised pretext task for continuity without relying on any previous information and alleviates catastrophic forgetting by discriminating which task current sample is from. We will discuss the details of the proposed ATD in Section 3.2.

3.2. Alternate Task Discrimination (ATD)

Generally, since information-dependent UCL methods [15, 34] face storage and computation challenges with the rapid increase of data amount brought by annotation liberation, this paper focuses on designing information-dependent-free strategy to alleviate catastrophic forgetting which doesn't rely on information from previous data streams. In this section, we propose Alternate Task Discrimination (ATD), which alleviate catastrophic forgetting by distinguishing which data stream current sample is from, as shown in Figure 2. Essentially, ATD is a self-supervised pretext task for continuity, which doesn't need previous information to assist future learning but develops the anti-forgetting capability through completing this pretext task. The intension is that the model captures discrimination information through completing ATD, such as data sequence, data attribution and data distribution. With the help of these information, the model is able to indirectly preserve the memory of seen data to prevent from catastrophic forgetting. Specifically,

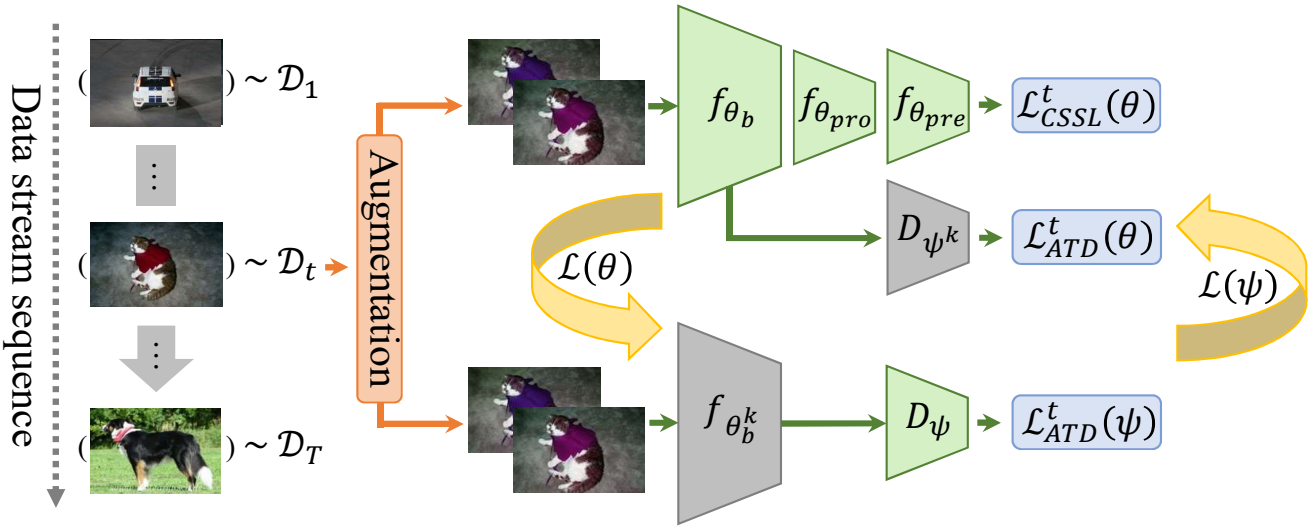

Figure 2. **The framework of proposed Alternate Task Discrimination (ATD)**. For each data stream, an alternating optimization algorithm is introduced to solve ATD effectively, which is consist of the UCL optimization step and ATD optimization step. UCL optimization step aims to learn an effective representation with the assistance of discriminator $D_{\psi^k}(\cdot)$ (discriminator parameters don't participate in gradient updating in this step, as shown in gray) from last ATD optimization step. ATD optimization step aims to acquire discriminative information with the assistance of backbone encoder $f_{\theta_b^k}$ (encoder parameters don't participate in gradient updating in this step, as shown in gray) from last UCL optimization step.

ATD introduces a discriminator D_ψ to discriminate the data stream attribution of current samples based on the backbone f_{θ_b} feature mapping:

$$\mathcal{L}^t_{ATD} = \mathbb{E}_{(x_{i,t})\sim\mathcal{D}_t}\left[CE\left(D_\psi\left(f_{\theta_b}\left(x_{i,t}\right)\right),t\right)\right] \tag{5}$$

In practice, we introduce augmentation invariance to enhance the reliability and robustness of discriminator:

$$\begin{aligned}\mathcal{L}^t_{ATD} = &\mathbb{E}_{(x_{i,t})\sim\mathcal{D}_t}\left[CE\left(D_\psi\left(f_{\theta_b}\left(\mathcal{T}^1\left(x_{i,t}\right)\right)\right),t\right)\right]+\\ &\mathbb{E}_{(x_{i,t})\sim\mathcal{D}_t}\left[CE\left(D_\psi\left(f_{\theta_b}\left(\mathcal{T}^2\left(x_{i,t}\right)\right)\right),t\right)\right]\end{aligned} \tag{6}$$

In order to perform ATD effectively in UCL framework, we design an alternating optimization algorithm. It is consist of the UCL optimization step and ATD optimization step, where the UCL optimization step guarantees the effectiveness of the learned representations and ATD optimization step guarantees the continuity of the learned representations:

$$\theta^k \leftarrow \arg\min_\theta\left(\lambda\mathcal{L}^t_{ATD}\left(\theta,\psi^k\right)+\mathcal{L}^t_{UCL}\left(\theta\right)\right) \tag{7}$$

$$\psi^{k+1} \leftarrow \arg\min_\psi\ \lambda\mathcal{L}^t_{ATD}\left(\theta^k,\psi\right) \tag{8}$$

where λ is a hyper-parameter to trade off the strengths of base UCL and ATD.

UCL optimization step. UCL optimization step aims to learn an effective representation with the assistance of discriminator $D_{\psi^k}(\cdot)$ from last ATD optimization step, where discriminator parameters don't participate in gradient updating. The loss function for this step is as follows:

$$\begin{aligned}\mathcal{L}\left(\theta\right) = &\lambda\mathbb{E}_{(x_{i,t})\sim\mathcal{D}_t}\left[CE\left(D_{\psi^k}\left(f_{\theta_b}\left(x_{i,t}\right)\right),t\right)\right]+\\ &\mathbb{E}_{(x_{i,t})\sim\mathcal{D}_t}\left[\mathcal{L}\left(x_{i,t}\right)\right]\end{aligned} \tag{9}$$

Formally, UCL optimization step captures beneficial information to ensure the effectiveness of the representation based on the last step discriminator which preserves the discriminative information across data streams.

ATD optimization step. ATD optimization step aims to acquire discriminative information with the assistance of backbone encoder $f_{\theta_b^k}$ from last UCL optimization step, where encoder parameters don't participate in gradient updating. The loss function for this step is as follows:

$$\mathcal{L}\left(\psi\right) = \lambda\mathbb{E}_{(x_{i,t})\sim\mathcal{D}_t}\left[CE\left(D_\psi\left(f_{\theta_b^k}\left(x_{i,t}\right)\right),t\right)\right] \tag{10}$$

Formally, ATD optimization step acquires discriminative information across data streams based on the last step backbone encoder which provides an effective representation.

In addition, the proposed ATD can be used as a plug-in unit to easily collaborate with existing UCL methods. Take rehearsal-based methods as an example, like DER [6] and LUMP [34], ATD can additionally distinguish which data stream the replayed data belongs to:

$$\begin{aligned}\mathcal{L}^{\mathcal{B}}_{ATD} = &\mathbb{E}_{(b_i,y_i)\sim\mathcal{B}}\left[CE\left(D_\psi\left(f_{\theta_b}\left(\mathcal{T}^1\left(b_i\right)\right)\right),y_i\right)\right]+\\ &\mathbb{E}_{(b_i,y_i)\sim\mathcal{B}}\left[CE\left(D_\psi\left(f_{\theta_b}\left(\mathcal{T}^2\left(b_i\right)\right)\right),y_i\right)\right]\end{aligned} \tag{11}$$

The rehearsal buffer which is consist of K replayed samples is denoted as $\mathcal{B} = \left\{(b_i, y_i)_{i=1}^{K}\right\}$. b_i and y_i are i-th sample and corresponding data stream attribution label respectively. For regularization-based methods, like CaSSLe [15], ATD can additionally discriminate which data stream the feature representation encoded by previous model belongs to:

$$\begin{aligned}\mathcal{L}_{ATD}^{p} = \mathbb{E}_{(x_{i,t})\sim\mathcal{D}_t}\left[CE\left(D_\psi\left(f_p\left(\mathcal{T}^1\left(x_{i,t}\right)\right)\right), t\right)\right] + \\ \mathbb{E}_{(x_{i,t})\sim\mathcal{D}_t}\left[CE\left(D_\psi\left(f_p\left(\mathcal{T}^2\left(x_{i,t}\right)\right)\right), t\right)\right]\end{aligned} \tag{12}$$

where previous backbone encoder is denoted as f_p. Essentially, ATD optimization step can capture more discriminative information with the assistance of existing UCL methods.

4. Experiments

We provide a variety of experimental results on popular UCL benchmarks in this section. In subsection 4.1, we first describe the experimental details, including implementation details of proposed method, popular UCL datasets, evaluation metrics. Secondly, we provide the main results (i.e., average accuracy and average forgetting) on popular UCL datasets in subsection 4.2. Thirdly, we complete the experiments on out of distribution datasets (OOD datasets) and provide the experimental results in subsection 4.3. Finally, we give more ablation studies to further verify the efficiency of proposed method in subsection 4.4.

4.1. Experimental details

Implementation details. ResNet-18 [21] is used as the backbone encoder in our experiments to make a fair comparison with existing UCL methods. The projection head and extra prediction head are the same as that in [12]. The discriminator is essentially a classifier based on cosine similarity to solve the data imbalance effect during training, inspired by [16, 38, 42]. We train the model with SGD optimizer for 200 epoches, including base unsupervised visual representation learning framework and discriminator. We set learning rate to 0.015, weight decay to 5e-4 and momentum to 0.9. The batchsize is 128 in our implementation.

Datasets. We follow [34] to divide CIFAR-10 [28] (10-class, 32×32 resolution), CIFAR-100 [28] (100-class, 32×32 resolution) and Tiny-ImageNet [13] (100-class, 64×64 resolution) in category order, i.e., Split CIFAR-10 (2 categories per task stream, 5 task streams), Split CIFAR-100 (5 categories per task stream, 20 task streams) and Split Tiny-ImageNet (5 categories per task stream, 20 task streams).

Evaluation metrics. In terms of evaluating the quality of learned representations, we adopt KNN classification performance [45] as the evaluation metric, which is able to evaluate the discriminative ability of the representations from feature level. Specifically, a KNN classifier is performed on the frozen pre-trained representations to report the classification performance. As for evaluating the continuity of learned representations, we utilize standard "Average Accuracy" and "Average Forgetting" as the evaluation metric.

"Average Accuracy" refers to the average performance on each task stream after training across all T task streams:

$$A = \frac{1}{T}\sum_{j=1}^{T} a_{Tj} \tag{13}$$

where a_{Tj} refers to the KNN classification performance by training across T task streams and testing on task stream j.

"Average Forgetting" refers to the average gap between optimal performance and final performance on first $T-1$ task stream after training across all T task streams:

$$F = \frac{1}{T-1}\sum_{j=1}^{T-1}\left(\max_{1\leqslant i\leqslant T} a_{ij} - a_{Tj}\right) \tag{14}$$

where $\max\limits_{1\leqslant i\leqslant T}$ refers to the optimal performance for training across 1 to T task streams and testing on task stream j.

4.2. Main results

In this subsection, we perform TAD on Split CIFAR-10, Split CIFAR-100 and Split Tiny-ImageNet, and report the main results (i.e., the average accuracy and average forgetting on all task streams) in Table 1. In addition to reporting the performance of standard TAD, we provide the performance of TAD†, which is the improved version of proposed TAD by incorporating with LUMP. Compared with the standard baseline method FINETUNE, the proposed TAD obtains considerable improvements on all benchmarks of both SimSiam-based and BarlowTwins-based UCL frameworks. Compared with other unsupervised continual learning methods, e.g., DER [6] which is introduced to unsupervised continual learning from supervised continual learning, the proposed TAD gets competitive results on multiple benchmarks. It's worth noting that TAD is a resource-free strategy which doesn't need the assistance of previous information, where rehearsal-based methods DER [6] and LUMP [34] require rehearsal buffer to store previous data.

In addition, the proposed improved version TAD† obtains significant achievements on all benchmarks, which achieves top-2 performance on all metrics. Compared with its baseline LUMP, TAD† gets consistent and observable improvements, showing the effectiveness of TAD in improving anti-forgetting ability of rehearsal-based method. For example, TAD† based on SimSiam [12] obtains 0.79%, 0.22%, 0.25% average accuracy improvements on Split CIFAR-10, Split CIFAR-100 and Split Tiny-ImageNet respectively, and achieves 1.08%, 1.53%, 0.54% average forgetting drops respectively.

Method	Split CIFAR-10		Split CIFAR-100		Split Tiny-ImageNet	
	Accuracy	Forgetting	Accuracy	Forgetting	Accuracy	Forgetting
Supervised Continual Learning						
FINETUNE	82.87(±0.47)	14.26(±0.52)	61.08(±0.04)	31.23(±0.41)	53.10(±1.37)	33.15(±1.22)
PNN [40]	82.74(±2.12)	-	66.05(±0.86)	-	64.38(±0.92)	-
SI [50]	85.18(±0.65)	11.39(±0.77)	63.58(±0.37)	27.98(±0.34)	44.96(±2.41)	26.29(±1.40)
A-GEM [10]	82.41(±1.24)	13.82(±1.27)	59.81(±1.07)	30.08(±0.91)	60.45(±0.24)	24.94(±1.24)
GSS [3]	89.49(±1.75)	7.50(±1.52)	70.78(±1.67)	21.28(±1.52)	70.96(±0.72)	14.76(±1.22)
DER [6]	91.35(±0.46)	5.65(±0.35)	79.52(±1.88)	12.80(±1.47)	68.03(±0.85)	17.74(±0.65)
MULTITASK	97.77(±0.15)	-	93.89(±0.78)	-	91.79(±0.46)	-
Unsupervised Continual Learning based on SimSiam [12]						
FINETUNE	90.11(±0.12)	5.42(±0.08)	75.42(±0.78)	10.19(±0.37)	71.07(±0.20)	9.48(±0.56)
PNN [40]	90.93(±0.22)	-	66.58(±1.00)	-	62.15(±1.35)	-
SI [50]	**92.75**(±0.06)	**1.81**(±0.21)	80.08(±1.30)	5.54(±1.30)	72.34(±0.42)	8.26(±0.64)
DER [6]	91.22(±0.30)	4.63(±0.26)	77.27(±0.30)	9.31(±0.09)	71.90(±1.44)	8.36(±2.06)
LUMP [34]	91.00(±0.40)	2.92(±0.53)	82.30(±1.35)	4.71(±1.52)	76.66(±2.39)	3.54(±1.04)
TAD (Ours)	91.27(±0.21)	5.02(±0.39)	78.59(±0.77)	8.74(±0.80)	73.99(±0.46)	7.02(±0.47)
TAD† (Ours)	91.79(±0.28)	1.84(±0.35)	**82.52**(±0.54)	**3.18**(±0.73)	**76.91**(±1.38)	**3.00**(±0.37)
MULTITASK	95.76(±0.08)	-	86.31(±0.38)	-	82.89(±0.49)	-
Unsupervised Continual Learning based on BarlowTwins [49]						
FINETUNE	87.72(±0.32)	4.08(±0.56)	71.97(±0.54))	9.45(±1.01)	66.28(±1.23)	8.89(±0.66)
PNN [40]	87.52(±0.33)	-	57.93(±2.98)	-	48.70(±2.59)	-
SI [50]	90.21(±0.08)	2.03(±0.22)	75.04(±0.63)	7.43(±0.67)	56.96(±1.48)	17.04(±0.89)
DER [6]	88.67(±0.24)	2.41(±0.26)	73.48(±0.53)	7.98(±0.29)	68.56(±1.47)	7.87(±0.44)
LUMP [34]	90.31(±0.30)	**1.13**(±0.18)	80.24(±1.04)	3.53(±0.83)	72.17(±0.89)	2.43(±1.00)
TAD (Ours)	89.38(±0.39)	3.21(±0.66)	75.65(±0.28)	7.80(±1.33)	70.71(±0.88)	6.95(±0.56)
TAD† (Ours)	**91.28**(±0.77)	1.93(±0.15)	**80.79**(±0.53)	**3.21**(±0.48)	**74.75**(±0.83)	**2.38**(±0.70)
MULTITASK	95.48(±0.14)	-	87.16(±0.52)	-	82.42(±0.74)	-

Table 1. The main results (Average Accuracy and Average Forgetting) on Split CIFAR-10, Split CIFAR-100 and Split Tiny-ImageNet. TAD† is the improved version of proposed method by incorporating with LUMP. All the results are composed of the mean and standard deviation of the three trials. The best performance is indicated by bold, and secondary performance is underlined.

4.3. Evaluation on out of distribution (OOD) datasets

In this subsection, we transfer the representations pre-trained on Split CIFAR-10 and Split CIFAR-100 to out of distribution (OOD) datasets to evaluate the generalization of learned representations. We report the transfer performance (i.e., average accuracy) of the proposed method on OOD datasets in Table 2. The proposed method achieves top-2 performance on multiple metrics of both SimSiam-based and BarlowTwins-based unsupervised continual learning frameworks, showing strong competitiveness. Especially for the proposed method based on BarlowTwins [49], the best performance is achieved on almost all metrics. It's worth noting that compared LUMP, the proposed method obtains considerable improvements. For example, TAD† based on SimSiam [12] obtains 0.55%, 1.81%, 3.58%, 6.36% average accuracy improvements respectively when transferring from Split CIFAR-10 to MNIST, Fashion-MNIST (FMNIST), SVHN, CIFAR-100. It suggests that the proposed TAD further improves rehearsal-based method LUMP on transferring ability.

4.4. Ablation study

In this subsection, we supplement the experiments of improved TAD versions, where TAD is collaborated with existing popular UCL methods, including rehearsal-based method DER [6] and LUMP [34], regularization-based method CaSSLe [15]. We give the main results (i.e., the

In-class	Split CIFAR-10				Split CIFAR-100			
Out of class	MNIST	FMNIST	SVHN	CIFAR-100	MNIST	FMNIST	SVHN	CIFAR-10
Supervised Continual Learning								
FINETUNE	86.42(±1.11)	74.47(±0.84)	41.00(±0.85)	17.42(±0.96)	75.02(±3.97)	62.37(±3.20)	38.05(±0.73)	39.18(±0.83)
SI [50]	87.08(±0.79)	76.41(±0.81)	42.62(±1.31)	19.14(±0.91)	79.96(±2.63)	63.71(±1.36)	40.92(±1.64)	40.41(±1.71)
A-GEM [10]	86.07(±1.94)	74.74(±3.21)	37.77(±3.49)	16.11(±0.38)	77.56(±3.21)	64.16(±2.29)	37.48(±1.73)	37.91(±1.33)
GSS [3]	70.36(±3.54)	69.20(±2.51)	33.11(±2.26)	18.21(±0.39)	76.54(±0.46)	65.31(±1.72)	35.72(±2.37)	49.41(±1.81)
DER [6]	80.32(±1.91)	70.49(±1.54)	41.48(±2.76)	17.72(±0.25)	87.71(±2.23)	75.97(±1.29)	50.26(±0.95)	59.07(±1.06)
MULTITASK	88.79(±1.13)	79.50(±0.52)	41.26(±1.95)	27.68(±0.66)	92.29(±3.37)	86.12(±1.87)	54.94(±1.77)	54.04(±3.68)
Unsupervised Continual Learning based on SimSiam [12]								
FINETUNE	89.23(±0.99)	80.05(±0.34)	49.66(±0.81)	34.52(±0.12)	85.99(±0.86)	76.90(±0.11)	50.09(±1.41)	57.15(±0.96)
SI [50]	**93.72**(±0.58)	82.50(±0.51)	**57.88**(±0.16)	36.21(±0.69)	91.50(±1.26)	80.57(±0.93)	**54.07**(±2.73)	60.55(±2.54)
DER [6]	88.35(±0.82)	79.33(±0.62)	48.83(±0.55))	30.68(±0.36)	87.96(±2.04)	76.21(±0.63)	47.70(±0.94)	56.26(±0.16)
LUMP [34]	91.03(±0.22)	80.78(±0.88)	45.18(±1.57)	31.17(±1.83)	**91.76**(±1.17)	81.61(±0.45)	50.13(±0.71)	63.00(±0.53)
TAD† (Ours)	91.58(±0.67)	**82.59**(±0.75)	48.76(±0.94)	**37.53**(±1.31)	90.64(±1.68)	**82.10**(±0.83)	53.64(±1.26)	**64.33**(±1.09)
MULTITASK	90.69(±0.13)	80.65(±0.42)	47.67(±0.45)	39.55(±0.18)	90.35(±0.24)	81.11(±1.86)	52.20(±0.61)	70.19(±0.15)
Unsupervised Continual Learning based on BarlowTwins [49]								
FINETUNE	86.86(±1.62)	78.37(±0.74)	44.64(±2.39)	28.03(±0.52)	76.08(±2.86)	76.82(±0.83)	42.95(±0.90)	53.12(±0.13)
SI [50]	90.31(±0.69)	80.58(±0.68)	49.18(±0.51)	31.80(±0.40)	85.24(±0.99)	78.82(±0.67)	45.18(±1.37)	53.99(±0.56)
DER [6]	85.15(±2.19)	77.96(±0.59)	45.68(±0.93)	27.83(±0.86)	78.08(±1.95)	76.67(±0.68)	44.58(±1.01)	53.24(±0.82)
LUMP [34]	88.73(±0.54)	81.69(±0.45)	**51.53**(±0.41)	31.53(±0.36)	90.22(±1.39)	81.28(±0.91)	50.24(±0.95)	60.76(±0.87)
TAD† (Ours)	**91.19**(±0.79)	**82.26**(±0.77)	50.48(±0.52)	**32.96**(±0.56)	**91.47**(±1.55)	**82.23**(±0.75)	**51.37**(±1.42)	**61.71**(±0.73)
MULTITASK	88.63(±1.38)	79.49(±0.29)	49.24(±2.44)	36.33(±0.29)	86.98(±1.70)	79.40(±1.10)	50.19(±0.81)	49.50(±0.38)

Table 2. The transfer performance (average accuracy) on out of distribution (OOD) datasets. TAD† is the improved version of proposed method by incorporating with LUMP. All the results are composed of the mean and standard deviation of the three trials. The best performance is indicated by bold, and secondary performance is underlined.

	Accuracy	Forgetting
FINETUNE	90.11(±0.12)	5.42(±0.08)
TAD (Ours)	**91.27**(±0.21)	**5.02**(±0.39)
DER [6]	91.22(±0.30)	4.63(±0.26)
DER + TAD (Ours)	**91.52**(±0.25)	**3.23**(±0.46)
LUMP [34]	91.00(±0.40)	2.92(±0.53)
TAD† (Ours)	**91.79**(±0.28)	**1.84**(±0.35)
CaSSLe [15]	91.23(±0.34)	2.74(±0.39)
CaSSLe + TAD (Ours)	**91.35**(±0.19)	**1.72**(±0.43)

Table 3. **Collaboration with existing UCL methods**. We report the average accuracy, average forgetting of the combinations between TAD and other UCL methods based on SimSiam [12] on Split CIFAR-10. All the results are composed of the mean and standard deviation of the three trials. The best performance is indicated by bold.

average accuracy, average forgetting) of the combinations between TAD and other UCL methods in Table 3. The proposed TAD obtains consistent and considerable improvements compared with different "baselines", where the improved versions obtain 0.30%, 0.79%, 0.12% average accuracy improvements and 1.40%, 1.08%, 1.02% average forgetting drops compared with DER [6], LUMP [34], CaSSLe [15] respectively. The stable improvements suggest that TAD is an effective plug-in unit for different existing UCL methods to enhance the continual learning ability.

5. Conclusion

In this paper, we investigate how to alleviate catastrophic forgetting without relying on previous information in UCL, i.e., information-free unsupervised continual learning method. To this end, we propose Alternate Task Discrimination (ATD), which can be recognised as a self-supervised pretext task for continuity. Specifically, ATD aims to discriminate which data stream current sample is from, where we additionally design an alternating optimization algorithm to make ATD work effectively in UCL framework. In addition, ATD can be used as a plug-in unit, which is easily combined with existing UCL methods and makes further achievements. The extensive experiments on multiple standard UCL benchmarks demonstrate the effectiveness and competitiveness of ATD.

6. Acknowledgements

This work was supported in part by National Science and Technology Major Project (2021ZD0112001), National Natural Science Foundation of China (No. U23A20286), China Postdoctoral Science Foundation 2023TQ0046, and Sichuan Province Innovative Talent Funding Project for Postdoctoral Fellows BX202212.

References

[1] Hongjoon Ahn, Sungmin Cha, Donggyu Lee, and Taesup Moon. Uncertainty-based continual learning with adaptive regularization. In *Advances in Neural Information Processing Systems*, pages 4394–4404, Vancouver, BC, Canada, 2019. 1, 2

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

[3] Rahaf Aljundi, Min Lin, Baptiste Goujaud, and Yoshua Bengio. Gradient based sample selection for online continual learning. In *Advances in Neural Information Processing Systems*, pages 11816–11825, Vancouver, BC, Canada, 2019. 2, 4, 7, 8

[4] Yuki Markus Asano, Christian Rupprecht, and Andrea Vedaldi. Self-labelling via simultaneous clustering and representation learning. In *Proceedings of the International Conference on Learning Representations*, Addis Ababa, Ethiopia, 2020. 3

[5] Jihwan Bang, Heesu Kim, YoungJoon Yoo, Jung-Woo Ha, and Jonghyun Choi. Rainbow memory: Continual learning with a memory of diverse samples. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8218–8227, Virtual, 2021. 2

[6] Pietro Buzzega, Matteo Boschini, Angelo Porrello, Davide Abati, and Simone Calderara. Dark experience for general continual learning: a strong, simple baseline. In *Advances in Neural Information Processing Systems*, pages 15920–15930, virtual, 2020. 1, 2, 3, 4, 5, 6, 7, 8

[7] Mathilde Caron, Ishan Misra, Julien Mairal, Priya Goyal, Piotr Bojanowski, and Armand Joulin. Unsupervised learning of visual features by contrasting cluster assignments. In *Advances in Neural Information Processing Systems*, pages 9912–9924, Virtual, 2020. 1, 3

[8] Francisco M Castro, Manuel J Marín-Jiménez, Nicolás Guil, Cordelia Schmid, and Karteek Alahari. End-to-end incremental learning. In *Proceedings of the European Conference on Computer Vision*, pages 233–248, Munich, Germany, 2018. 2, 4

[9] Hyuntak Cha, Jaeho Lee, and Jinwoo Shin. Co2l: Contrastive continual learning. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 9516–9525, Virtual, 2021. 3

[10] Arslan Chaudhry, Marc'Aurelio Ranzato, Marcus Rohrbach, and Mohamed Elhoseiny. Efficient lifelong learning with a-gem. In *Proceedings the International Conference on Learning Representations*, New Orleans, LA, USA, 2019. 7, 8

[11] Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. A simple framework for contrastive learning of visual representations. In *Proceedings of the International Conference on Machine Learning*, pages 1597–1607, Virtual, 2020. 1, 3

[12] Xinlei Chen and Kaiming He. Exploring simple siamese representation learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 15750–15758, Virtual, 2021. 1, 3, 4, 6, 7, 8

[13] 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, Miami, Florida, USA, 2009. 2, 6

[14] Debidatta Dwibedi, Yusuf Aytar, Jonathan Tompson, Pierre Sermanet, and Andrew Zisserman. With a little help from my friends: Nearest-neighbor contrastive learning of visual representations. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 9588–9597, Virtual, 2021. 3

[15] Enrico Fini, Victor G Turrisi da Costa, Xavier Alameda-Pineda, Elisa Ricci, Karteek Alahari, and Julien Mairal. Self-supervised models are continual learners. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9621–9630, Virtual, 2022. 1, 2, 3, 4, 6, 7, 8

[16] 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, Salt Lake City, UT, USA, 2018. 6

[17] Spyros Gidaris, Praveer Singh, and Nikos Komodakis. Unsupervised representation learning by predicting image rotations. In *Proceedings of the International Conference on Learning Representations*, Vancouver, BC, Canada, 2018. 3

[18] Jean Bastien Grill, Florian Strub, Florent Altché, Corentin Tallec, Pierre Richemond, Elena Buchatskaya, Carl Doersch, Bernardo Avila Pires, Zhaohan Guo, Mohammad Gheshlaghi Azar, et al. Bootstrap your own latent-a new approach to self-supervised learning. In *Advances in Neural Information Processing Systems*, pages 21271–21284, Virtual, 2020. 1, 3, 4

[19] Gunshi Gupta, Karmesh Yadav, and Liam Paull. Look-ahead meta learning for continual learning. In *Advances in Neural Information Processing Systems*, pages 11588–11598, Virtual, 2020. 3

[20] Mahmudul Hasan and Amit K Roy-Chowdhury. A continuous learning framework for activity recognition using deep hybrid feature models. *IEEE Transactions on Multimedia*, 17(11):1909–1922, 2015. 1

[21] 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, Las Vegas, NV, USA, 2016. 3, 6

[22] Kaiming He, Haoqi Fan, Yuxin Wu, Saining Xie, and Ross Girshick. Momentum contrast for unsupervised visual representation learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9729–9738, Virtual, 2020. 1, 3

[23] Geoffrey Hinton, Oriol Vinyals, Jeff Dean, et al. Distilling the knowledge in a neural network. *arXiv:1503.02531*, 2015. 2, 3

[24] Saihui Hou, Xinyu Pan, Chen Change Loy, Zilei Wang, and Dahua Lin. Learning a unified classifier incrementally via

rebalancing. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 831–839, Long Beach, CA, USA, 2019. 2

[25] Khurram Javed and Martha White. Meta-learning representations for continual learning. In *Advances in Neural Information Processing Systems*, pages 1818–1828, Vancouver, BC, Canada, 2019. 3

[26] 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. 1, 2, 4

[27] Soroush Abbasi Koohpayegani, Ajinkya Tejankar, and Hamed Pirsiavash. Mean shift for self-supervised learning. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 10326–10335, Virtual, 2021. 3

[28] Alex Krizhevsky, Geoffrey Hinton, et al. Learning multiple layers of features from tiny images, 2009. 2, 6

[29] Abhishek Kumar, Sunabha Chatterjee, and Piyush Rai. Bayesian structural adaptation for continual learning. In *Proceedings of the International Conference on Machine Learning*, pages 5850–5860, Virtual, 2021. 3

[30] Gustav Larsson, Michael Maire, and Gregory Shakhnarovich. Learning representations for automatic colorization. In *Proceedings of the European Conference on Computer Vision*, pages 577–593, Amsterdam, Netherlands, 2016. 3

[31] Junnan Li, Pan Zhou, Caiming Xiong, and Steven CH Hoi. Prototypical contrastive learning of unsupervised representations. In *Proceedings of the International Conference on Learning Representations*, Virtual, 2021. 3

[32] Xilai Li, Yingbo Zhou, Tianfu Wu, Richard Socher, and Caiming Xiong. Learn to grow: A continual structure learning framework for overcoming catastrophic forgetting. In *Proceedings of the International Conference on Machine Learning*, pages 3925–3934, Long Beach, California, USA, 2019. 3

[33] Zhizhong Li and Derek Hoiem. Learning without forgetting. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 40(12):2935–2947, 2017. 1, 2, 4

[34] Divyam Madaan, Jaehong Yoon, Yuanchun Li, Yunxin Liu, and Sung Ju Hwang. Representational continuity for unsupervised continual learning. In *Proceedings the International Conference on Learning Representations*, Virtual, 2022. 1, 2, 3, 4, 5, 6, 7, 8

[35] Arun Mallya, Dillon Davis, and Svetlana Lazebnik. Piggyback: Adapting a single network to multiple tasks by learning to mask weights. In *Proceedings of the European Conference on Computer Vision*, pages 72–88, Munich, Germany, 2018. 2

[36] Mehdi Noroozi and Paolo Favaro. Unsupervised learning of visual representations by solving jigsaw puzzles. In *Proceedings of the European Conference on Computer Vision*, pages 69–84, Amsterdam, Netherlands, 2016. 3

[37] Deepak Pathak, Philipp Krahenbuhl, Jeff Donahue, Trevor Darrell, and Alexei A Efros. Context encoders: Feature learning by inpainting. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2536–2544, Las Vegas, NV, USA, 2016. 3

[38] 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, Salt Lake City, UT, USA, 2018. 6

[39] 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, Honolulu, HI, USA, 2017. 3

[40] Andrei A Rusu, Neil C Rabinowitz, Guillaume Desjardins, Hubert Soyer, James Kirkpatrick, Koray Kavukcuoglu, Razvan Pascanu, and Raia Hadsell. Progressive neural networks. *arXiv:1606.04671*, 2016. 1, 3, 4, 7

[41] Selvarajah Thuseethan, Sutharshan Rajasegarar, and John Yearwood. Deep continual learning for emerging emotion recognition. *IEEE Transactions on Multimedia*, 2021. 1

[42] Xin Wang, Thomas E Huang, Trevor Darrell, Joseph E Gonzalez, and Fisher Yu. Frustratingly simple few-shot object detection. In *Proceedings of the International Conference on Machine Learning*, pages 9919–9928, Virtual, 2020. 6

[43] Xudong Wang, Ziwei Liu, and Stella X Yu. Unsupervised feature learning by cross-level instance-group discrimination. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 12586–12595, Virtual, 2021. 3

[44] Zhen Wang, Liu Liu, Yiqun Duan, and Dacheng Tao. Continual learning through retrieval and imagination. In *Proceedings of the AAAI Conference on Artificial Intelligence*, pages 8594–8602, Virtual, 2022. 2

[45] Zhirong Wu, Yuanjun Xiong, Stella X Yu, and Dahua Lin. Unsupervised feature learning via non-parametric instance discrimination. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3733–3742, Salt Lake City, UT, USA, 2018. 3, 6

[46] Ju Xu and Zhanxing Zhu. Reinforced continual learning. In *Advances in Neural Information Processing Systems*, pages 907–916, Montréal, Canada, 2018. 3

[47] Guanglei Yang, Enrico Fini, Dan Xu, Paolo Rota, Mingli Ding, Tang Hao, Xavier Alameda-Pineda, and Elisa Ricci. Continual attentive fusion for incremental learning in semantic segmentation. *IEEE Transactions on Multimedia*, 2022. 1

[48] Jaehong Yoon, Eunho Yang, Jeongtae Lee, and Sung Ju Hwang. Lifelong learning with dynamically expandable networks. In *Proceedings the International Conference on Learning Representations*, Vancouver, BC, Canada, 2018. 4

[49] Jure Zbontar, Li Jing, Ishan Misra, Yann LeCun, and Stéphane Deny. Barlow twins: Self-supervised learning via redundancy reduction. In *Proceedings of the International Conference on Machine Learning*, pages 12310–12320, Virtual, 2021. 1, 3, 4, 7, 8

[50] Friedemann Zenke, Ben Poole, and Surya Ganguli. Continual learning through synaptic intelligence. In *Proceedings of the International Conference on Machine Learning*, pages 3987–3995, Sydney, Australia, 2017. 1, 2, 4, 7, 8

[51] Hongyi Zhang, Moustapha Cisse, Yann N Dauphin, and David Lopez-Paz. mixup: Beyond empirical risk minimization. In *Proceedings of the International Conference on Learning Representations*, Vancouver, BC, Canada, 2018. 2, 3