PCR: Proxy-based Contrastive Replay for Online Class-Incremental Continual Learning

Huiwei Lin, Baoquan Zhang, Shanshan Feng*, Xutao Li, Yunming Ye
Harbin Institute of Technology, Shenzhen
{linhuiwei, zhangbaoquan}@stu.hit.edu.cn, {victor_fengss, lixutao, yeyunming}@hit.edu.cn

Abstract

Online class-incremental continual learning is a specific task of continual learning. It aims to continuously learn new classes from data stream and the samples of data stream are seen only once, which suffers from the catastrophic forgetting issue, i.e., forgetting historical knowledge of old classes. Existing replay-based methods effectively alleviate this issue by saving and replaying part of old data in a proxy-based or contrastive-based replay manner. Although these two replay manners are effective, the former would incline to new classes due to class imbalance issues, and the latter is unstable and hard to converge because of the limited number of samples. In this paper, we conduct a comprehensive analysis of these two replay manners and find that they can be complementary. Inspired by this finding, we propose a novel replay-based method called proxy-based contrastive replay (PCR). The key operation is to replace the contrastive samples of anchors with corresponding proxies in the contrastive-based way. It alleviates the phenomenon of catastrophic forgetting by effectively addressing the imbalance issue, as well as keeps a faster convergence of the model. We conduct extensive experiments on three real-world benchmark datasets, and empirical results consistently demonstrate the superiority of PCR over various state-of-the-art methods [1].

1. Introduction

Online class-incremental continual learning (online CICL) is a special scenario of continual learning [12]. Its goal is to learn a deep model that can achieve knowledge accumulation of new classes and not forget information learned from old classes. In the meantime, the samples of a continuously non-stationary data stream are accessed only once during the learning process. At present, catastrophic forgetting (CF) is the main problem of online CICL. It is associated with the phenomenon that the model has a significant performance drop for old classes when learning new classes. The main reason is historical knowledge of old data would be overwritten by novel information of new data.

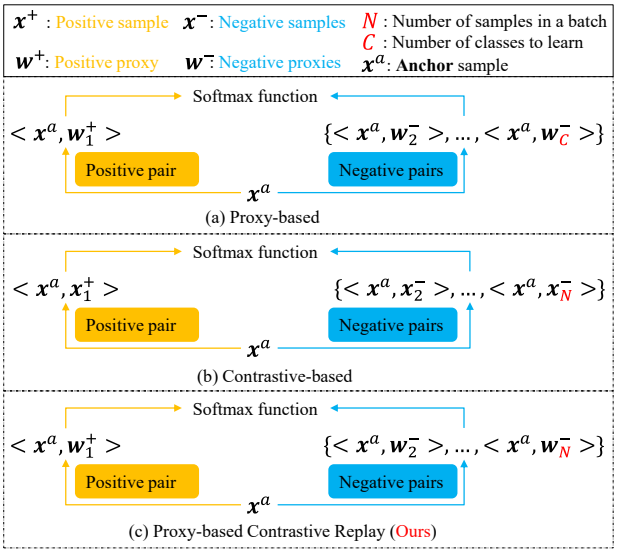


Figure 1. Illustration of our work. (a) The example of proxy-based replay manner. For each anchor sample, it calculates similarities of all anchor-to-proxy pairs. (b) The example of contrastive-based replay manner. For each anchor sample, it calculates similarities of all anchor-to-sample pairs in the same batch. (c) The example of our method. It calculates similarities of anchor-to-proxy pairs, which is similar to the proxy-based method. However, the anchor-to-proxy pairs are selected by the anchor-to-sample pairs in the same batch, which performs in the contrastive-based manner.

Among all types of methods proposed in continual learning, the replay-based methods have shown superior performance for online CICL [25]. In this family of methods, part of previous samples are saved in an episodic memory buffer and then used to learn together with current samples. In general, there are two ways to replay. The first is the proxy-based replay manner, which is to replay by using the proxy-based loss and softmax classifier. As shown in Figure 1(a), it calculates similarities between each anchor with

*Corresponding author
[1] https://github.com/FelixHuiweiLin/PCR

all proxies belonging to C classes. A proxy can be regarded as the representative of a sub-dataset [38], and the anchor is one of the samples in the training batch. The second is the contrastive-based replay manner that replays by using the contrastive-based loss and nearest class mean (NCM) classifier [27]. Shown as Figure 1(b), it computes similarities between each anchor with all N samples in the same training batch. Although these two manners are effective, they have their corresponding limitations. The former is subjected to the "bias" issue caused by class imbalance, tending to classify most samples of old classes into new categories. The latter is unstable and hard to converge in the training process due to the small number of samples.

In this work, we comprehensively analyze their characteristics and find that the coupling of them can achieve complementary advantages. On the one hand, the proxy-based manner enables fast and reliable convergence with the help of proxies. On the other hand, although the contrastive-based manner is not very robust, it has advantages in the selection of anchor-to-sample pairs. Only the classes associated with samples in anchor-to-sample pairs can be selected to learn. Previous studies [1, 6] have proved that suitably selecting of anchor-to-proxy pairs is effective to address the "bias" issue. Therefore, it is necessary to develop a coupling manner to jointly keep these advantages at the same time. In other words, it not only takes proxies to improve the robustness of the model as proxy-based manner, but also overcomes the "bias" problem by selecting anchor-to-proxy pairs as the pairs selection of contrastive-based manner.

With these inspirations, we propose a novel replay-based method called proxy-based contrastive replay (PCR) to alleviate the phenomenon of CF for online CICL. The core motivation is the coupling of proxy-based and contrastive-based loss, and the key operation is to replace anchor-to-sample pairs with anchor-to-proxy pairs in the contrastive-based loss. As shown in Figure 1(c), our method calculates similarities between each anchor and other proxies, which is similar to the proxy-based loss. However, it does not straightly make full use of proxies from all classes. It only takes the proxies whose associated classes of samples appear in the same batch, which is analogous to the contrastive-based loss. For one thing, it keeps fast convergence and stable performance with the help of proxies. For another thing, it addresses the "bias" issue by only choosing part of anchor-to-proxy pairs to calculate categorical probability. And the selected anchor-to-proxy pairs are generally better than the ones selected by existing solutions [1, 6].

Our main contributions can be summarized as follows:

1) We theoretically analyze the characteristics of proxy-based and contrastive-based replay manner, discovering the coupling manner of them is beneficial. To the best of our knowledge, this work is the first one to combine these two manners for the online CICL problem.

2) We develop a novel online CICL framework called PCR to mitigate the forgetting problem. By replacing the samples for anchor with proxies in contrastive-based loss, we achieve the complementary advantages of two existing approaches.

3) We conduct extensive experiments on three real-world datasets, and the empirical results consistently demonstrate the superiority of our PCR over various state-of-the-art methods. We also investigate and analyze the benefits of each component by ablation studies.

2. Related work

2.1. Continual Learning

Recent advances on continual learning are driven by three main directions. 1) Architecture-based methods [41], also known as parameter-isolation methods, divide each task into a set of specific parameters of the model. They dynamically extend the model as the number of tasks increases [31] or gradually freeze part of parameters to overcome the forgetting problem [28]. 2) Regularization-based methods [41], also called prior-based methods, store previous knowledge learned from old data as prior information of network. It takes the historical knowledge to consolidate past knowledge by extending the loss function with additional regularization term [13, 20]. 3) Replay-based methods, which set a fixed-size memory buffer [9, 14, 22, 24, 33] or generative model [10, 11, 34, 37] to store, produce, and replay historical samples in the training process, also go by the name rehearsal-based methods. This kind of methods [4, 7, 8, 23, 36] that replay old samples in the buffer are still the most effective for anti-forgetting at present [5].

2.2. Online Class-Incremental Continual Learning.

Replay-based methods based on experience replay (ER) [30] are the main solutions of online CICL. Some approaches use the **memory retrieval strategy** to select valuable samples from memory, such as MIR [2] and ASER [32]. In the meantime, some approaches [3, 17, 19] focus on saving more effective samples to the memory, belonging to the **memory update strategy**. The others [6, 15, 16, 26, 39] utilize the **model update strategy** to improve the learning efficiency. Recently, a new method AOP based on orthogonal projection has been proposed without buffer. Most of them are proxy-based manners except SCR [26], which is a contrastive-based manner.

The proposed PCR in this work exploits a new model update strategy for online CICL, belonging to the family of replay-based methods. Different from existing approaches, it aims to combine the contrastive-based replay manner with the proxy-based replay manner. By complementing their advantages, the coupling manner can more effectively alleviate the phenomenon of catastrophic forgetting.

3. Problem Statement and Analysis

3.1. Problem Formulation

Online CICL divides a data stream into a sequence of learning tasks as $\mathcal{D} = \{\mathcal{D}_t\}_{t=1}^{T}$, where $\mathcal{D}_t = \{\mathcal{X}_t \times \mathcal{Y}_t, \mathcal{C}_t\}$ contains the samples $\mathcal{X}_t$, corresponding labels $\mathcal{Y}_t$, and task-specific classes $\mathcal{C}_t$. Different tasks have no overlap in the classes. The neural network is made up of a feature extractor $\boldsymbol{z} = h(\boldsymbol{x}; \boldsymbol{\Phi})$ and a proxy-based classifier $f(\boldsymbol{z}; \boldsymbol{W}) = \langle \boldsymbol{z}, \boldsymbol{W} \rangle \cdot \gamma$ [18], where $\boldsymbol{W}$ contains trainable proxies of all classes, $\langle \cdot, \cdot \rangle$ is the cosine similarity, and γ is a scale factor. All of learned classes are denoted as $\mathcal{C}_{1:t} = \bigcup_{k=1}^{t} \mathcal{C}_k$. The categorical probability that sample x belongs to class c is

$$p_c = \frac{e^{\langle h(\boldsymbol{x};\boldsymbol{\Phi}), \boldsymbol{w}_c \rangle \cdot \gamma}}{\sum_{j \in \mathcal{C}_{1:t}} e^{\langle h(\boldsymbol{x};\boldsymbol{\Phi}), \boldsymbol{w}_j \rangle \cdot \gamma}}. \tag{1}$$

In the training process, the model can only access D_t and each sample can be seen only once. Its objective function is

$$L = E_{(\boldsymbol{x},y)\sim\mathcal{D}_t}[-log(\frac{e^{\langle h(\boldsymbol{x};\boldsymbol{\Phi}), \boldsymbol{w}_y \rangle \cdot \gamma}}{\sum_{j \in \mathcal{C}_{1:t}} e^{\langle h(\boldsymbol{x};\boldsymbol{\Phi}), \boldsymbol{w}_j \rangle \cdot \gamma}})]. \tag{2}$$

3.2. Analysis of Catastrophic Forgetting.

A direct cause of CF is the unbalanced gradient propagation between old and new classes. The gradient for a single sample $\boldsymbol{x}$ can be expressed as

$$\frac{\partial L}{\partial \boldsymbol{W}} = \begin{cases} h(\boldsymbol{x};\boldsymbol{\Phi})(p_y - 1), & i = y \\ h(\boldsymbol{x};\boldsymbol{\Phi})(p_c), & c \neq y \end{cases}. \tag{3}$$

As Equation (3) shows, if a training sample $\boldsymbol{x}$ belongs to class y, it not only increases the logits value of the y-th dimension by $p_y - 1 < 0$, but also decreases the logits value of other dimension by $p_c > 0$. Combining with the chain rule, it provides the positive gradient for the proxy of class y as $\boldsymbol{w}_y = \boldsymbol{w}_y - \eta h(\boldsymbol{x};\boldsymbol{\Phi})(p_y - 1)$, and propagates the negative gradient to the other proxies as $\boldsymbol{w}_c = \boldsymbol{w}_c - \eta h(\boldsymbol{x};\boldsymbol{\Phi})(p_c)$. Since η is a positive learning rate and $h(\boldsymbol{x};\boldsymbol{\Phi})$ is usually non-negative by Relu [29]. Furthermore, the gradient is transferred to the feature extractor, making it focus on the features that can distinguish this class from other classes.

When directly optimizing Equation (2), which is known as Finetune, the learning of new classes dominates the gradient propagation, causing the phenomenon of CF. To better analyse it, we show a case that learns the samples of cat and dog at the first task (Figure 2(a)), and then learns the samples of ship and airplane at the next task (Figure 2(b)-(f)). As seen in the left part of Figure 2(b), the gradient is produced by learning new classes. As a result, the proxies of new classes receive more positive gradient (↑) and the others obtain more negative gradient (↓). Shown as the red arrows in the left part of Figure 2(b), it causes the proxies of new classes are close to the samples of new classes, while the proxies of old classes are far away from them. Meanwhile, the feature extractor pays more attention to the features of new classes. It causes the samples of new and old classes are close [6] in the unit embedding space. Hence, it is easy to classify samples to new classes.

3.3. Analysis of Proxy-based Manner

ER [30] allocates a memory buffer $\mathcal{M}$ to temporarily store part of previous samples of old classes, which are re-trained with current samples. And its objective function is

$$L_{ER} = E_{(\boldsymbol{x},y)\sim\mathcal{D}_t\cup\mathcal{M}}[-log(\frac{e^{\langle h(\boldsymbol{x};\boldsymbol{\Phi}), \boldsymbol{w}_y \rangle \cdot \gamma}}{\sum_{j \in \boxed{\mathcal{C}_{1:t}}} e^{\langle h(\boldsymbol{x};\boldsymbol{\Phi}), \boldsymbol{w}_j \rangle \cdot \gamma}})], \tag{4}$$

where the samples of all classes take the same way to calculate categorical probability. As described in Figure 2(c), previous samples of old classes acquire some advantages in the propagation of gradient. Not only the proxies of old classes obtain more positive gradient, but also the proxies of new classes receive more negative gradient. Although the phenomenon of CF can be alleviated to some extent, its effect is still limited. Since the number of samples for each class in the fixed buffer will decrease as the learning process goes on, the gradient of old classes are not enough.

SS-IL [1] separately calculates categorical probability for old and new classes by separated softmax as

$$\begin{aligned} L_{SS} = & E_{(\boldsymbol{x},y)\sim\mathcal{D}_t}[-log(\frac{e^{\langle h(\boldsymbol{x};\boldsymbol{\Phi}), \boldsymbol{w}_y \rangle \cdot \gamma}}{\sum_{j \in \boxed{\mathcal{C}_t}} e^{\langle h(\boldsymbol{x};\boldsymbol{\Phi}), \boldsymbol{w}_j \rangle \cdot \gamma}})] \\ & + E_{(\boldsymbol{x},y)\sim\mathcal{M}}[-log(\frac{e^{\langle h(\boldsymbol{x};\boldsymbol{\Phi}), \boldsymbol{w}_y \rangle \cdot \gamma}}{\sum_{j \in \boxed{\mathcal{C}_{1:t-1}}} e^{\langle h(\boldsymbol{x};\boldsymbol{\Phi}), \boldsymbol{w}_j \rangle \cdot \gamma}})]. \end{aligned} \tag{5}$$

As demonstrated in Figure 2(d), it cuts off the propagation from the learning of old classes to the proxies of new classes, and prevents the propagation from the learning of new classes to the proxies of old classes. It is able to avoid that the gradient of new classes affect the proxies of old classes. However, the model can not well distinguish new classes from old classes, since the lacking of gradient makes it difficult for the model to classify classes across tasks.

ER-ACE [6] is also proposed to address the same issue by an asymmetric cross-entropy loss, which is expressed as

$$\begin{aligned} L_{ACE} = & E_{(\boldsymbol{x},y)\sim\mathcal{D}_t}[-log(\frac{e^{\langle h(\boldsymbol{x};\boldsymbol{\Phi}), \boldsymbol{w}_y \rangle \cdot \gamma}}{\sum_{j \in \boxed{\mathcal{C}_t}} e^{\langle h(\boldsymbol{x};\boldsymbol{\Phi}), \boldsymbol{w}_j \rangle \cdot \gamma}})] \\ & + E_{(\boldsymbol{x},y)\sim\mathcal{M}}[-log(\frac{e^{\langle h(\boldsymbol{x};\boldsymbol{\Phi}), \boldsymbol{w}_y \rangle \cdot \gamma}}{\sum_{j \in \boxed{\mathcal{C}_{1:t}}} e^{\langle h(\boldsymbol{x};\boldsymbol{\Phi}), \boldsymbol{w}_j \rangle \cdot \gamma}})]. \end{aligned} \tag{6}$$

Its categorical probability of new classes is similar with SS-IL, and the categorical probability of old classes is the same as ER. In detail, it only selects part of anchor-to-proxy pairs for the learning of new classes. As shown in Figure 2(e),

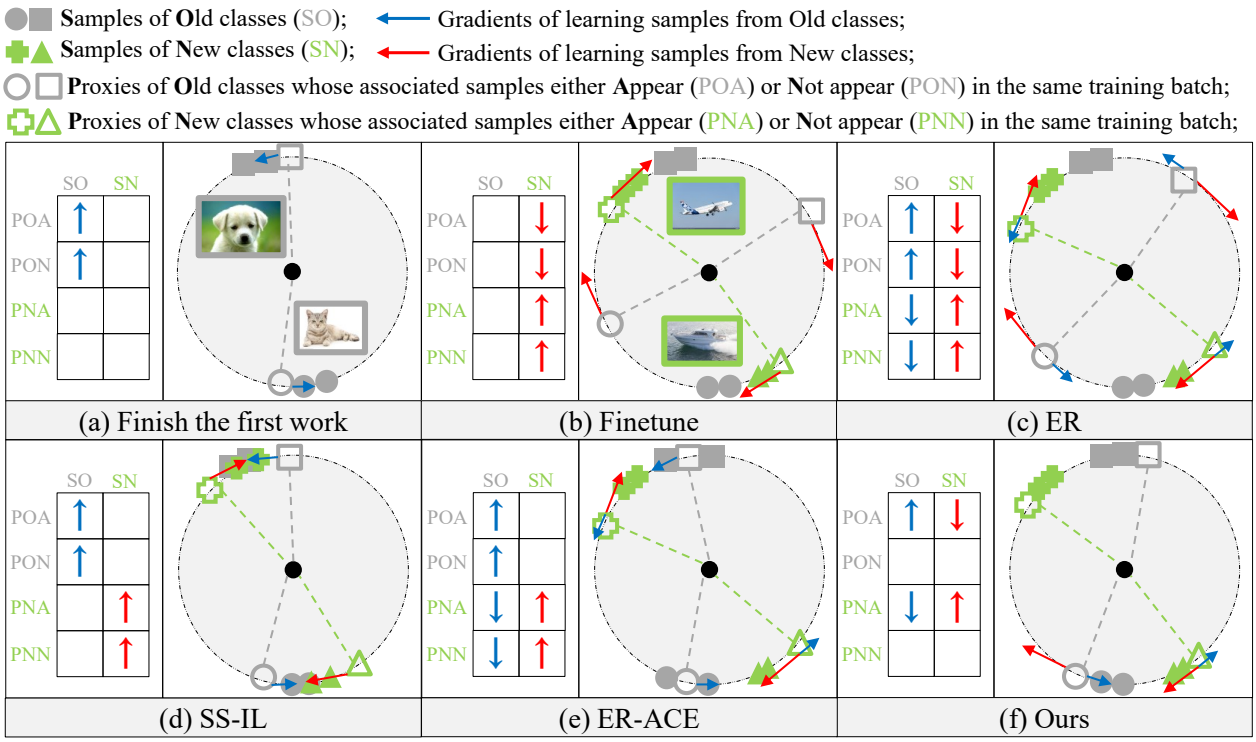


Figure 2. Analysis of existing proxy-based manners. In each sub-figure, the left part is the process of gradient propagation from samples to all proxies, and the right part is the unit embedding space of samples and proxies. (a) The learning of the first task. The gradient propagation only exists in current two classes, denoted as the blue arrows. (b) The learning of the second task by Finetune. The new classes dominate the gradient propagation, denoted as the red arrows. (c) The learning of the second task by ER. (d) The learning of the second task by SS-IL. (e) The learning of the second task by ER-ACE. (f) The learning of the second task by our method. Different from existing studies, our method controls the process of gradient propagation more effectively, improving the recognition of new and old classes.

it only breaks the gradient propagation from the learning of new classes to the proxies of old classes. Keeping the gradient from the learning of old classes to the proxies of new classes helps to avoid the inseparable situation of SS-IL. Although it is beneficial for old classes, the performance on new classes is harmed.

3.4. Analysis of Contrastive-based Manner

SCR [26] is proposed as a good alternative for online CICL by contrastive-based loss, which is denoted as

$$L_{SCR} = E_{(\boldsymbol{x},y)\sim\mathcal{D}_t\cup\mathcal{M}} [\frac{-1}{|P(\boldsymbol{x})|}\sum_{p\in P(\boldsymbol{x})} log\frac{e^{\langle h(\boldsymbol{x};\boldsymbol{\Phi}),h(\boldsymbol{x}_p;\boldsymbol{\Phi})\rangle/\mathcal{T}}}{\sum_{j\in \boxed{J(\boldsymbol{x})}} e^{\langle h(\boldsymbol{x};\boldsymbol{\Phi}),h(\boldsymbol{x}_j;\boldsymbol{\Phi})\rangle/\mathcal{T}}}]. \quad (7)$$

It splices current samples and previous samples into the same batch and calculates the similarities of anchor-to-samples pairs. $J(\boldsymbol{x})$ is the indices set of samples except for anchor $\boldsymbol{x}$ in the same batch, while $P(\boldsymbol{x})$ denotes the set of samples with the same labels as anchor $\boldsymbol{x}$. Different from proxy-based loss, the selected pairs do not rely on the number of classes, but are related to the number of samples in a training batch. Hence, its effect is constrained by the size of memory buffer and batch size. And its performance would be not satisfactory when less samples to replay.

4. Methodology

4.1. Motivation

From above analysis, we can draw three conclusions. First and foremost, the unbalanced gradient propagation between new classes and old classes is the main cause of CF. The new classes dominate this process, making the samples of new classes highly distinguishable but the ones of old classes indivisible. Effectively controlling the gradient propagation between old and new classes can help the model alleviate the forgetting problem. Second, existing proxy-based approaches control the gradient propagation by selecting part of anchor-to-proxy pairs to calculate the objective function. Although they are effective, they are easy to hurt the generalization ability of model to learn new classes. Finally, the contrastive-based manner depends on the samples from the same batch but lacks the support of proxies. Its selection of anchor-to-sample pairs provides a heuristic way to select anchor-to-proxy pairs.

Based on these conclusions, we find that the coupling of these two manners would lead to a better solution. To avoid the limit caused by the size of samples, we do not take the coupling method in [38], which adds anchor-to-sample pairs to anchor-to-proxy pairs in cross-entropy loss. Specif-

ically, we replace the samples of anchor-to-sample pairs by proxies for in contrastive-based loss, and obtain our manner

$$L_{ours} = E_{(\boldsymbol{x},y)\sim\mathcal{D}_t\cup\mathcal{M}} [\frac{-1}{|P(\boldsymbol{x})|}\sum_{p\in P(\boldsymbol{x})} log\frac{e^{\langle h(\boldsymbol{x};\boldsymbol{\Phi}),\boldsymbol{w}_p\rangle/\mathcal{T}}}{\sum_{j\in\boxed{J(\boldsymbol{x})}} e^{\langle h(\boldsymbol{x};\boldsymbol{\Phi}),\boldsymbol{w}_j\rangle/\mathcal{T}}}]. \quad (8)$$

Different from existing studies, its way of computing categorical probability is changed for each mini-batch. On the one hand, such a loss has faster convergence speed and better robustness, and can cope with a small number of samples with the help of proxies. On the other hand, the replacing proxies are only from the classes that appear in the training batch. As a result, the gradient for propagation are only from the learning of these classes. As shown in Figure 2(f), the gradient among all proxies are not completely separated in the whole training process. The gradient propagation only occurs when the corresponding classes appear in the same batch. Meanwhile, in each learning step, only new and old classes in current batch participate in the gradient propagation. The proxies of old classes, which are affected by the negative gradient of new classes, can also generate the positive gradient for confrontation and further mitigate the forgetting problem. Hence, the samples of all classes can be recognized more correctly than existing methods.

4.2. Proxy-based Contrastive Replay

With these inspirations, we propose a novel proxy-based contrastive replay (PCR) framework, and the technical details will be stated in this section. The framework consists of a CNN-based backbone $h(\boldsymbol{x};\boldsymbol{\Phi})$ and a proxy-based classifier $f(\boldsymbol{z};\boldsymbol{W})$. The whole training and inference procedures of PCR are summarized in Algorithm 1.

4.2.1 The Training Procedure of PCR

In this part, the model is trained by learning samples of new classes and replaying samples of old classes. For each task, given current samples $(\boldsymbol{x}_c, y_c)$, it randomly retrieves previous samples $(\boldsymbol{x}_\mathcal{M}, y_\mathcal{M})$ from the memory buffer (line 1-4). Besides, these original samples and their augmented samples are spliced together for the batch of training (line 5-7). Then, the model is optimized by this training batch (line 8-9). The objective function is defined as

$$L_{PCR} = E_{(\boldsymbol{x},y)\sim\mathcal{D}_t\cup\mathcal{M}}[-log(\frac{e^{\langle h(\boldsymbol{x};\boldsymbol{\Phi}),\boldsymbol{w}_y\rangle\cdot\gamma}}{\sum_{j\in\boxed{\mathcal{C}_B}} e^{\langle h(\boldsymbol{x};\boldsymbol{\Phi}),\boldsymbol{w}_j\rangle\cdot\gamma}})], \quad (9)$$

where $\mathcal{C}_B$ is the classes indices in current batch of training, and the indices can be repeated. Finally, it updates the memory buffer by reservoir sampling strategy, which can ensure that the probability of each sample being extracted is equal. Conveniently, the memory buffer in our framework has a fix-sized, no matter how large the amount of samples is.

Algorithm 1 Proxy-based Contrastive Replay

Input:Dataset D, Learning Rate λ, Scale factor γ
Output:Network Parameters θ
Initialize:Memory Buffer $\mathcal{M} \leftarrow \{\}$, Network Parameters $\boldsymbol{\theta} = \{\boldsymbol{\Phi}, \boldsymbol{W}\}$

1: **for** $t \in \{1, 2, ..., T\}$ **do**
2: *//Training Procedure*
3: **for** mini-batch $(\boldsymbol{x}_c, y_c) \sim D_t$ **do**
4: $(\boldsymbol{x}_\mathcal{M}, y_\mathcal{M}) \leftarrow RandomRetrieval(\mathcal{M})$.
5: $(\boldsymbol{x}_{ori}, y_{ori}) \leftarrow Concat([(\boldsymbol{x}_c, y_c), (\boldsymbol{x}_\mathcal{M}, y_\mathcal{M})])$.
6: $(\boldsymbol{x}_{aug}, y_{aug}) \leftarrow DataAugmentation(\boldsymbol{x}_{ori}, y_{ori})$.
7: $(\boldsymbol{x}, y) \leftarrow Concat([(\boldsymbol{x}_{ori}, y_{ori}), (\boldsymbol{x}_{aug}, y_{aug})])$.
8: $L = -log(\frac{e^{\langle h(\boldsymbol{x};\boldsymbol{\Phi}),\boldsymbol{w}_y\rangle\cdot\gamma}}{\sum_{j\in\mathcal{C}_B} e^{\langle h(\boldsymbol{x};\boldsymbol{\Phi}),\boldsymbol{w}_j\rangle\cdot\gamma}})$
9: $\theta \leftarrow \theta + \lambda\nabla_\theta L$.
10: $\mathcal{M} \leftarrow ReservoirUpdate(\mathcal{M}, (\boldsymbol{x}_t, y_t))$.
11: **end for**
12: *//Inference Procedure*
13: **for** $k \in \{1, 2, ..., m\}$ **do**
14: $y_k^* \leftarrow= \arg\max_c \frac{e^{\langle h(\boldsymbol{x}_k;\boldsymbol{\Phi}),\boldsymbol{w}_c\rangle\cdot\gamma}}{\sum_{j\in\mathcal{C}_{1:t}} e^{\langle h(\boldsymbol{x}_k;\boldsymbol{\Phi}),\boldsymbol{w}_j\rangle\cdot\gamma}}, c \in C_{1:t}$
15: **end for**
16: **return** θ
17: **end for**

4.2.2 The Inference Procedure of PCR

The inference procedure (line 13-15) is different from the training procedure. Each testing sample $\boldsymbol{x}_k$ obtains its class probability distribution by Equation (1). And we perform the inference prediction to $\boldsymbol{x}_k$ with highest probability as

$$y_k^* = \arg\max_c \frac{e^{\langle h(\boldsymbol{x}_k;\boldsymbol{\Phi}),\boldsymbol{w}_c\rangle\cdot\gamma}}{\sum_{j\in\boxed{\mathcal{C}_{1:t}}} e^{\langle h(\boldsymbol{x}_k;\boldsymbol{\Phi}),\boldsymbol{w}_j\rangle\cdot\gamma}}, c \in C_{1:t}. \quad (10)$$

5. Performance Evaluation

5.1. Experiment Setup

5.1.1 Datasets

We conduct experiments on three real-world image datasets for evaluation. Split CIFAR10 [21] is split into 5 taskes, and each task contains 2 classes. Split CIFAR100 [21] as well as Split MiniImageNet [35] are organized into 10 tasks, and each task is made up of samples from 10 classes.

5.1.2 Evaluated Baselines

To evaluate the effectiveness of PCR, we compare it with the following four methodological categories. **None-replay operations** contain IID and FINE-TUNE. **Memory update strategies** include ER [30], GSS [3], and GMED [19]. MIR [2] and ASER [32] are **memory retrieval strategies**. And **model update strategies** contains A-GEM [9], ER-WA [40], DER++ [4], SS-IL [1], SCR [26], ER-ACE [6], ER-DVC [15], and OCM [16].

Table 1. Final Accuracy Rate (higher is better). The best scores are in boldface, and the second best scores are underlined.

Datasets [sample size]	Split CIFAR10 [32×32]				Split CIFAR100 [32×32]				Split MiniImageNet [84×84]			
Buffer	100	200	500	1000	500	1000	2000	5000	500	1000	2000	5000
IID	55.9±0.4				17.1±1.0				17.3±1.7			
FINE-TUNE	17.9±0.4				5.9±0.2				4.3±0.2			
ER	33.8±3.2	41.7±2.8	46.0±3.5	46.1±4.3	14.5±0.8	17.6±0.9	19.7±1.6	20.9±1.2	11.2±0.6	13.4±0.9	16.5±0.9	16.2±1.7
GSS	23.1±3.9	28.3±4.6	36.3±4.1	44.8±3.6	14.6±1.3	16.9±1.4	19.0±1.8	20.1±1.1	10.3±1.5	13.9±1.0	14.6±1.1	15.5±0.9
GMED (NeurIPS2021)	32.8±4.7	43.6±5.1	52.5±3.9	51.3±3.6	15.0±0.9	18.8±0.7	21.1±1.2	23.0±1.5	11.9±1.2	15.3±1.3	18.0±0.8	19.6±1.0
MIR (NeurIPS2019)	34.8±3.3	40.3±3.3	42.6±1.7	47.4±4.1	14.8±0.7	18.1±0.7	20.3±1.6	21.6±1.7	11.9±0.6	14.8±1.1	17.2±0.8	17.2±1.2
ASER (AAAI2021)	33.7±3.7	31.6±3.4	42.1±3.0	42.3±2.9	13.0±0.9	16.1±1.1	17.7±0.7	18.9±1.0	10.5±1.1	13.8±0.9	16.1±0.9	18.1±1.1
A-GEM (ICLR2019)	17.5±1.7	17.4±2.1	17.9±0.7	18.2±1.5	5.4±0.6	5.6±0.5	5.4±0.7	4.6±1.0	5.0±1.0	4.7±1.1	5.0±2.3	4.8±0.8
ER-WA (CVPR2020)	36.9±2.9	42.5±3.4	48.6±2.7	45.9±5.3	18.3±0.7	21.7±1.2	23.6±0.9	24.0±1.8	15.1±0.7	17.1±0.9	18.9±1.4	18.5±1.5
DER++ (NeurIPS2020)	40.9±1.4	45.3±1.7	52.8±2.2	53.9±1.9	15.5±1.0	17.2±1.1	19.5±1.2	20.2±1.3	11.9±1.0	14.8±0.7	16.1±1.3	15.5±1.3
SS-IL (ICCV2021)	36.8±2.1	42.2±1.4	44.8±1.6	47.4±1.5	19.5±0.6	21.9±1.1	24.5±1.4	24.7±1.0	18.0±0.7	19.7±0.9	21.7±1.0	24.4±1.6
SCR (CVPR-W2021)	35.0±2.9	45.4±1.0	55.7±1.6	**59.8±1.6**	13.3±0.6	16.2±1.3	18.2±0.8	19.3±1.0	12.1±0.7	14.7±1.9	16.8±0.6	18.6±0.5
ER-DVC (CVPR2022)	36.3±2.6	45.4±1.4	50.6±2.9	52.1±2.5	16.8±0.8	19.7±0.7	22.1±0.9	24.1±0.8	13.9±0.6	15.4±0.7	17.2±0.8	19.1±0.9
OCM (ICML2022)	44.4±1.5	49.9±1.8	55.8±2.3	59.2±2.2	17.7±1.0	20.6±1.2	22.1±1.0	22.7±1.4	11.1±0.6	13.6±0.7	16.5±0.5	19.2±0.7
ER-ACE (ICLR2022)	44.3±1.5	49.7±2.4	54.9±1.4	57.5±1.9	19.7±0.8	23.1±0.8	24.8±0.9	27.0±1.2	18.1±0.5	20.3±1.3	24.8±1.1	26.2±1.0
↪ER-ACE-NCM	(45.0±1.3)	(51.0±1.2)	(56.8±1.1)	(60.1±1.0)	(21.0±0.6)	(24.2±0.7)	(26.6±0.7)	(29.1±1.0)	(18.1±0.8)	(22.3±0.5)	(25.3±0.5)	(27.1±0.9)
PCR (Ours)	**45.4±1.3**	**50.3±1.5**	**56.0±1.2**	58.8±1.6	**21.8±0.9**	**25.6±0.6**	**27.4±0.6**	**29.3±1.1**	**20.9±0.9**	**24.2±0.9**	**27.2±1.2**	**28.4±0.9**
↪PCR-NCM	(43.7±1.2)	(49.1±1.3)	(56.2±1.2)	(59.9±1.8)	(22.6±0.6)	(26.0±0.4)	(28.2±0.6)	(30.1±1.0)	(19.8±0.6)	(23.5±0.6)	(26.8±0.5)	(28.0±0.6)

5.1.3 Evaluation Metrics

We need to measure the performance of model for online CICL. Most important of all, we define $a_{i,j}(j <= i)$ as the accuracy evaluated on the held-out test samples of the jth task after the network has learned the training samples in the first i tasks. Similar with [32], we can acquire average accuracy rateA_i at the ith task based on $a_{i,j}(j <= i)$.

$$A_i = \frac{1}{i}\sum_{j=1}^{i} a_{i,j} \tag{11}$$

If the model finish learning all of T tasks, A_T is equivalent to the final accuracy rate. Furthermore, we decompose the accuracy rate to obtain the average accuracy rate of new data $A_i^n = a_{i,i}$ and the one of old data A_i^o at the ith task, where

$$A_i^o = \frac{1}{i-1}\sum_{j=1}^{i-1} a_{i,j}. \tag{12}$$

5.1.4 Implementation Details

The basic setting of backbone model is the same as the latest work [6]. In detail, we take the Reduced ResNet18 (the number of filters is 20) as the feature extractor for all datasets. During the training phase, the network is trained with the SGD optimizer and the learning rate is set as 0.1.

For all datasets, the classes are shuffled before division. And we set the memory buffer with {100, 200, 500, 1000} for Split CIFAR10, and with {500, 1000, 2000, 5000} for other two datasets. The model receives 10 current samples from data stream and 10 previous samples from the memory buffer at a time irrespective of the size of the memory.

Table 2. Final Accuracy Rate (higher is better) on Split CIFAR100.

Buffer	1000	2000	5000
SCR [16]	26.5±0.2	31.6±0.5	36.5±0.2
OCM [16]	28.1±0.3	35.0±0.4	42.4±0.5
PCR	29.3±0.6	36.3±0.9	46.5±0.8
PCR-NCM	31.0±0.9	37.7±0.8	47.9±0.6

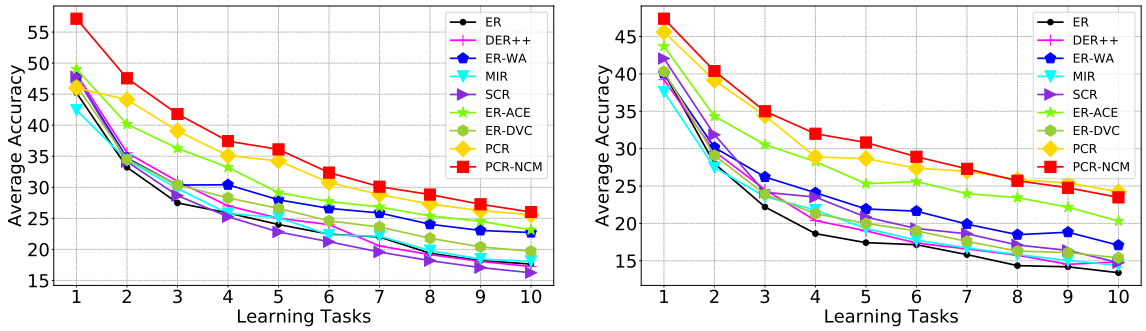
Moreover, we employ a combination of various augmentation techniques to get the augmented images. And the usage of data augmentation is fair for all methods. As for the testing phase, we set 256 as the batch size of validation.

5.2. Overall Performance

In this section, we conduct experiments to compare with various state-of-the-art baselines of continual learning. Table 1 shows the final average accuracy rate for Split CIFAR10, Split CIFAR100 and Split MiniImageNet. All reported scores are the average score of 10 runs with the 95% confidence interval. It is easy to find that our method obtains significantly improved performance on three datasets.

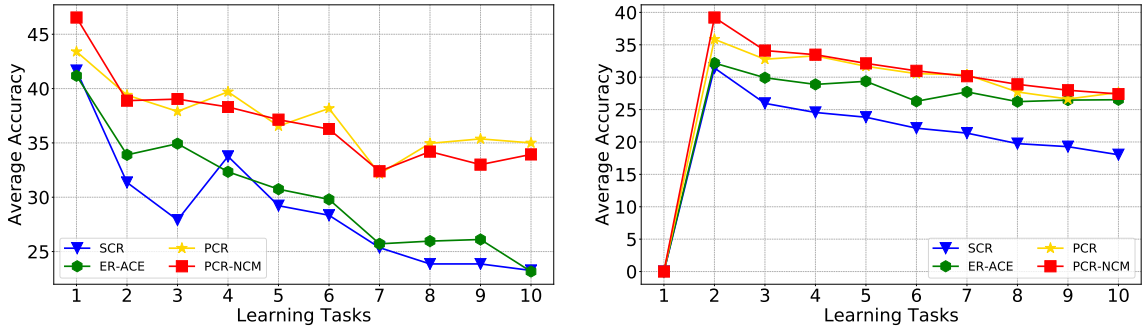
Comparison on final accuracy. Table 1 reports the accuracy performance of all baseline methods on three datasets. By comparing the final accuracy of all methods, we can draw two conclusions. First, the model update strategies are more effective and can greatly improve the performance among all replay-based methods. Second, ER-ACE achieves the highest accuracy rate as the latest method.

Our method PCR achieves the best performance, which confirms its effectiveness. In general, PCR achieves the best performance under most experimental settings in which each dataset contains four memory buffer of different sizes. On the one hand, it has the most outstanding performance



(a) Split CIFAR100 (b) Split MiniImageNet

Figure 3. Average accuracy rate on observed learning stages on Split CIFAR100 and Split MiniImageNet while the buffer size is 1000.



(a) Performance on novel knowledge (b) Performance on historical knowledge

Figure 4. Average accuracy rate on observed learning stages Split MiniImageNet while the buffer size is 5000.

on Split MiniImageNet among three benchmarks. On the other hand, the growth of buffer size further improve the performance of PCR. For example, PCR improves the proxy-based baseline ER and contrastive-based baseline SCR with a gap of more than 10%. Meanwhile, PCR outperforms the strongest baseline ER-ACE by 2.8%, 3.9%, 2.4%, 2.2% on Split MiniImagenet when the size of memory buffer is 500, 1000, 2000 and 5000, respectively. In addition, PCR defeats ER-ACE with an improvement of 2.1%, 2.5%, 2.6% and 2.3% on Split CIFAR100 with 500, 1000, 2000 and 5000 size of memory buffer, respectively.

Although there is a scenario on Split CIFAR10 that does not achieve the strongest result, the overall performance on this dataset is the best. By the buffer with 1000 samples, SCR that uses NCM classifier beats PCR on Split CIFAR10. We replace the proxy-based classifier by the NCM one and get the ER-ACE-NCM as well as PCR-NCM. By NCM classifier, there will be some significant improvement in their performance. And the results state that NCM classifier is more suitable for situation with smaller samples (e.g. Split-CIFAR100) and a larger buffer. Since small-sized samples are easy to obtain reliable features, and large-sized buffer trends to obtain accurate classification centers. In addition, OCM performs better than PCR when the buffer is large with the advantages of multiple data augmentations.

In the same time, we also compare PCR with SCR and OCM under the experiment setting in [16], shown as Table 2. In [16], the model is set as ResNet18 (the number of filters is 64), and it retrieves 64 samples from memory buffer for each training batch. Moreover, it is trained with Adam optimizer and the learning rate is set as 0.001. These experimental conditions are different from ours and can significantly improve the performance. The results suggest that our method is significantly better than SCR and OCM.

Comparison on learning process. For more detailed comparison, we reveal the accuracy performance in each task for part of effective approaches on all datasets, as shown in Figure 3. The results of line chart show that PCR not only achieves significant results in the accuracy of final task, but also performs better than other baselines in the whole learning process. In fact, with the help of NCM classifier, the overall performance of PCR-NCM is the best one. However, it has to calculate classification centers before its inference process, which greatly reduces its efficiency. Furthermore, the performance of PCR in the first few tasks does not outperform other baselines. However, the improvement of it become more and more visible as the number of tasks increases, proving its power to overcome CF. For instance, PCR has no obvious advantages in the first task, but it shows real effect in the remaining tasks on Split CIFAR100. Meanwhile, PCR and ER-ACE, especially PCR, are far superior in performance to other baselines on Split MiniImagenet. Therefore, our approach has a stronger ability to resist forgetting in the case of less data.

Comparison on knowledge balance. Actually, we should not only focus on the model's ability to retain histori-

Table 3. Final Accuracy Rate (higher is better) on three datasets for ablation studies. ER is a baseline method.

Datasets	Split CIFAR10				Split CIFAR100				Split MiniImageNet			
Buffer	100	200	500	1000	500	1000	2000	5000	500	1000	2000	5000
ER	33.8±3.2	41.7±2.8	46.0±3.5	46.1±4.3	14.5±0.8	17.6±0.9	19.7±1.6	20.9±1.2	11.2±0.6	13.4±0.9	16.5±0.9	16.2±1.7
ER+A	39.4±3.2	48.8±1.1	52.3±1.7	53.9±3.4	20.5±0.9	25.5±0.6	26.0±0.7	28.2±0.7	19.4±1.3	22.8±0.8	25.6±1.1	27.5±1.0
ER+A+B (PCR)	45.4±1.3	50.3±1.5	56.0±1.2	58.8±1.6	21.8±0.9	25.6±0.6	27.4±0.6	29.3±1.1	20.9±0.9	24.2±0.9	27.2±1.2	28.4±0.9
ER+A+B+C	41.9±2.0	48.3±2.4	55.8±1.2	57.0±2.6	19.8±1.0	24.1±0.5	25.9±0.5	27.3±0.7	19.3±1.2	21.8±0.8	24.5±0.9	26.3±1.0

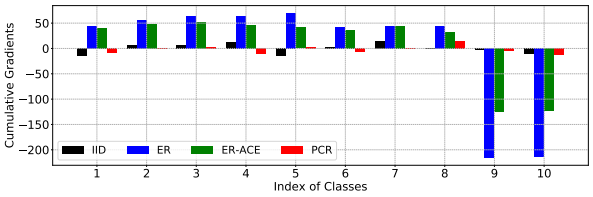


Figure 5. Cumulative gradients of different methods for all proxies when the model learns new classes (9/10) on Split CIFAR10.

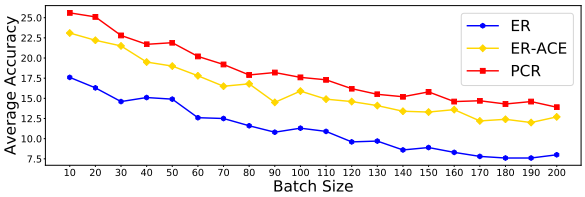


Figure 6. Final accuracy rate (higher is better) on Split CIFAR100 with different batch size when the buffer size is 1000.

cal knowledge, but also ensure the model's ability to quickly learn novel knowledge. Shown in Figure 4, we record the accuracy performance of novel and historical knowledge in each task for part of effective methods on Split MiniImageNet. Although SCR and ER-ACE can improve the anti-forgetting ability of the model, they have a serious impact on the generalization ability of the model. As the historical knowledge is consolidated, the learning performance of the model on novel knowledge becomes very poor. Different from existing studies, our model can not only effectively alleviate the phenomenon of CF, but also reduce the decline of the model at the generalization level as much as possible.

5.3. Ablation Study

In this section, we decompose PCR into several components, and further demonstrate their functions.

"A" means the **selection component**, which selects the anchor-to-proxy pairs as the contrastive-based replay manner. Actually, there are some samples from the same classes in a training batch. As a result, there are some duplicate anchor-to-proxy pairs in PCR. To effectively verify the role of the selected proxies, we remove the duplicate pairs. As shown in Table 3, the performance of ER has significant improvement with the help of this component.

"B" is the **duplication component** to keep the duplicate pairs in PCR. Although it contains the same knowledge as the anchor-to-proxy pairs of selection component, it still produces significant performance. Since it can provide anchor samples with more negative pairs, which are vital to contrastive-based loss. As stated in Table 3, PCR outperforms "ER+A" with the duplication component.

"C" denotes the **addition component** to exploit original anchor-to-sample pairs of SCR. Combining PCR with this part, we can get another coupling manner as [38]. It keeps anchor-to-sample pairs of contrastive-based loss, when replacing anchor-to-sample pairs by anchor-to-proxy pairs as PCR. Although it provides more knowledge about the relationships of samples, its performance is limited by little number of samples as indicated in Table 3.

In conclusion, the selection component and the duplication component are the keys of PCR. As displayed in Figure 5, PCR produces uniform gradients for all proxies to address the "bias" issue by the smart selection of anchor-to-proxy pairs. Furthermore, since the way of selection depends on the classes in the training batch, PCR is influenced by the batch size. Figure 6 reports the performance of several methods with different batch size. With the increase of batch size, PCR consistently maintains its advantages.

6. Conclusion

In this paper, we develop a novel online CICL method called PCR to alleviate the phenomenon of catastrophic forgetting by the coupling of proxy-based and contrastive-based replay manners. Based on the characteristics of these two manners, we propose to replace the samples of anchor-to-sample pairs by proxies. The coupling replay manner realizes complementary advantages. With the help of proxies, our method keeps the fast and stable convergence. In the meantime, the same selection of anchor-to-proxy pairs as contrastive samples is beneficial for addressing the "bias" issue of proxy-based manner. Extensive experiments on three datasets demonstrate the superiority of PCR over a large variety of state-of-the-art methods.

Acknowledgement

This work was supported in part by National Nature Science Foundation of China (No.62202124 and No.62272130), Shenzhen Science and Technology Program (No.KCXFZ20211020163403005) and Nature Science Program of Shenzhen (No.JCYJ20210324120208022 and No.JCYJ20200109113014456).

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*, pages 844–853, 2021. 2, 3, 5

[2] Rahaf Aljundi, Eugene Belilovsky, Tinne Tuytelaars, Laurent Charlin, Massimo Caccia, Min Lin, and Lucas Page-Caccia. Online continual learning with maximal interfered retrieval. *Advances in Neural Information Processing Systems*, 32:11849–11860, 2019. 2, 5

[3] Rahaf Aljundi, Min Lin, Baptiste Goujaud, and Yoshua Bengio. Gradient based sample selection for online continual learning. *Advances in neural information processing systems*, 32, 2019. 2, 5

[4] Pietro Buzzega, Matteo Boschini, Angelo Porrello, Davide Abati, and Simone Calderara. Dark experience for general continual learning: a strong, simple baseline. *Advances in neural information processing systems*, 33:15920–15930, 2020. 2, 5

[5] Pietro Buzzega, Matteo Boschini, Angelo Porrello, and Simone Calderara. Rethinking experience replay: a bag of tricks for continual learning. In *2020 25th International Conference on Pattern Recognition (ICPR)*, pages 2180–2187. IEEE, 2021. 2

[6] Lucas Caccia, Rahaf Aljundi, Nader Asadi, Tinne Tuytelaars, Joelle Pineau, and Eugene Belilovsky. New insights on reducing abrupt representation change in online continual learning. In *International Conference on Learning Representations*, 2021. 2, 3, 5, 6

[7] 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, 2021. 2

[8] Arslan Chaudhry, Albert Gordo, Puneet Dokania, Philip Torr, and David Lopez-Paz. Using hindsight to anchor past knowledge in continual learning. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 35, pages 6993–7001, 2021. 2

[9] Arslan Chaudhry, Marc'Aurelio Ranzato, Marcus Rohrbach, and Mohamed Elhoseiny. Efficient lifelong learning with a-gem. In *International Conference on Learning Representations*, 2018. 2, 5

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

[11] Bo Cui, Guyue Hu, and Shan Yu. Deepcollaboration: Collaborative generative and discriminative models for class incremental learning. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 35, pages 1175–1183, 2021. 2

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

[13] Prithviraj Dhar, Rajat Vikram Singh, Kuan-Chuan Peng, Ziyan Wu, and Rama Chellappa. Learning without memorizing. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5138–5146, 2019. 2

[14] Mehrdad Farajtabar, Navid Azizan, Alex Mott, and Ang Li. Orthogonal gradient descent for continual learning. In *International Conference on Artificial Intelligence and Statistics*, pages 3762–3773. PMLR, 2020. 2

[15] Yanan Gu, Xu Yang, Kun Wei, and Cheng Deng. Not just selection, but exploration: Online class-incremental continual learning via dual view consistency. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7442–7451, 2022. 2, 5

[16] Yiduo Guo, Bing Liu, and Dongyan Zhao. Online continual learning through mutual information maximization. In *International Conference on Machine Learning*, pages 8109–8126. PMLR, 2022. 2, 5, 6, 7

[17] Jiangpeng He and Fengqing Zhu. Online continual learning for visual food classification. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 2337–2346, 2021. 2

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

[19] Xisen Jin, Arka Sadhu, Junyi Du, and Xiang Ren. Gradient-based editing of memory examples for online task-free continual learning. *Advances in Neural Information Processing Systems*, 34, 2021. 2, 5

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

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

[22] Huiwei Lin, Shanshan Feng, Xutao Li, Wentao Li, and Yunming Ye. Anchor assisted experience replay for online class-incremental learning. *IEEE Transactions on Circuits and Systems for Video Technology*, 2022. 2

[23] Yaoyao Liu, Bernt Schiele, and Qianru Sun. Rmm: Reinforced memory management for class-incremental learning. *Advances in Neural Information Processing Systems*, 34, 2021. 2

[24] David Lopez-Paz and Marc'Aurelio Ranzato. Gradient episodic memory for continual learning. *Advances in neural information processing systems*, 30, 2017. 2

[25] 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. 1

[26] Zheda Mai, Ruiwen Li, Hyunwoo Kim, and Scott Sanner. Supervised contrastive replay: Revisiting the nearest class

mean classifier in online class-incremental continual learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3589–3599, 2021. 2, 4, 5

[27] 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

[28] Zichen Miao, Ze Wang, Wei Chen, and Qiang Qiu. Continual learning with filter atom swapping. In *International Conference on Learning Representations*, 2021. 2

[29] Vinod Nair and Geoffrey E Hinton. Rectified linear units improve restricted boltzmann machines. In *Icml*, 2010. 3

[30] David Rolnick, Arun Ahuja, Jonathan Schwarz, Timothy Lillicrap, and Gregory Wayne. Experience replay for continual learning. *Advances in Neural Information Processing Systems*, 32, 2019. 2, 3, 5

[31] Andrei A Rusu, Neil C Rabinowitz, Guillaume Desjardins, Hubert Soyer, James Kirkpatrick, Koray Kavukcuoglu, Razvan Pascanu, and Raia Hadsell. Progressive neural networks. *arXiv preprint arXiv:1606.04671*, 2016. 2

[32] Dongsub Shim, Zheda Mai, Jihwan Jeong, Scott Sanner, Hyunwoo Kim, and Jongseong Jang. Online class-incremental continual learning with adversarial shapley value. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 35, pages 9630–9638, 2021. 2, 5, 6

[33] Shixiang Tang, Dapeng Chen, Jinguo Zhu, Shijie Yu, and Wanli Ouyang. Layerwise optimization by gradient decomposition for continual learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9634–9643, 2021. 2

[34] Gido M van de Ven, Hava T Siegelmann, and Andreas S Tolias. Brain-inspired replay for continual learning with artificial neural networks. *Nature communications*, 11(1):1–14, 2020. 2

[35] Oriol Vinyals, Charles Blundell, Timothy Lillicrap, Daan Wierstra, et al. Matching networks for one shot learning. *Advances in neural information processing systems*, 29:3630–3638, 2016. 5

[36] Liyuan Wang, Xingxing Zhang, Kuo Yang, Longhui Yu, Chongxuan Li, HONG Lanqing, Shifeng Zhang, Zhenguo Li, Yi Zhong, and Jun Zhu. Memory replay with data compression for continual learning. In *International Conference on Learning Representations*, 2021. 2

[37] Ye Xiang, Ying Fu, Pan Ji, and Hua Huang. Incremental learning using conditional adversarial networks. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 6619–6628, 2019. 2

[38] Xufeng Yao, Yang Bai, Xinyun Zhang, Yuechen Zhang, Qi Sun, Ran Chen, Ruiyu Li, and Bei Yu. Pcl: Proxy-based contrastive learning for domain generalization. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7097–7107, 2022. 2, 4, 8

[39] Haiyan Yin, Ping Li, et al. Mitigating forgetting in online continual learning with neuron calibration. *Advances in Neural Information Processing Systems*, 34, 2021. 2

[40] 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. 5

[41] Fei Zhu, Zhen Cheng, Xu-yao Zhang, and Cheng-lin Liu. Class-incremental learning via dual augmentation. *Advances in Neural Information Processing Systems*, 34, 2021. 2