Not Just Selection, but Exploration: Online Class-Incremental Continual Learning via Dual View Consistency

Yanan Gu, Xu Yang, Kun Wei*, Cheng Deng*
School of Electronic Engineering, Xidian University, Xi'an 710071, China
{yanangu.xd,xuyang.xd,weikunsk,chdeng.xd}@gmail.com

Abstract

Online class-incremental continual learning aims to learn new classes continually from a never-ending and single-pass data stream, while not forgetting the learned knowledge of old classes. Existing replay-based methods have shown promising performance by storing a subset of old class data. Unfortunately, these methods only focus on selecting samples from the memory bank for replay and ignore the adequate exploration of semantic information in the single-pass data stream, leading to poor classification accuracy. In this paper, we propose a novel yet effective framework for online class-incremental continual learning, which considers not only the selection of stored samples, but also the full exploration of the data stream. Specifically, we propose a gradient-based sample selection strategy, which selects the stored samples whose gradients generated in the network are most interfered by the new incoming samples. We believe such samples are beneficial for updating the neural network based on back gradient propagation. More importantly, we seek to explore the semantic information between two different views of training images by maximizing their mutual information, which is conducive to the improvement of classification accuracy. Extensive experimental results demonstrate that our method achieves state-of-the-art performance on a variety of benchmark datasets. Our code is available on https://github.com/YananGu/DVC.

1. Introduction

Intelligent systems [24, 30, 33, 49, 53, 56] based on convolutional neural networks have achieved excellent performance on various tasks, some of which even exceed human-level performance. However, these intelligent systems, which need to restart the training process when new data is available, lack the ability to accumulate knowledge over time as humans do. Actually, such a restart practice is often not applicable in real scenarios because of training costs and privacy concerns. In order to achieve a higher level intelligent system, Continual Learning [17, 35, 38] studies the problem of learning from a never-ending data stream. A significant problem such a never-ending learning process faces is catastrophic forgetting—the inability to retain previously learned knowledge when learning new tasks.

Existing methods [1, 32, 42, 55] often relax the problem of continual learning to a more accessible task-incremental setting. In a task-incremental setting, the data stream is divided into several tasks with clear boundaries, and each task is learned offline. However, this setting lacks flexibility, because data in real world scenarios is often streamed online without task identity. In this paper, we focus on a more general online class-incremental continual setting, where a stream of samples is seen only once and task identity is unavailable during the training and testing phases.

There are already many types of methods proposed for task-incremental setting [1, 32, 34, 37], which can be primarily divided into three categories: regularization-based, parameter-isolation, and replay-based methods. In these methods, replay-based methods [2, 13, 45] have been proved to be simple yet efficient compared to other methods in the online class-incremental continual setting. Such methods usually keep previous performance by recording some samples of old classes for replay. Specifically, the recorded and incoming samples are put together to train the model, which makes the model preserve the previous knowledge to the greatest extent and learn new knowledge simultaneously. However, these methods only focus on finding the optimal recorded samples for replay and lack a full exploration of semantic information in the single-pass data stream, leading to poor classification accuracy.

In this paper, we propose a novel yet effective framework for online class-incremental continual learning to address the deficiencies observed above. Specifically, we propose a Maximally Gradient Interfered (MGI) retrieval strategy, which selects the stored samples whose gradients generated in the network are most interfered by the new incoming samples. We believe such samples are beneficial for updat-

* Corresponding author

ing the neural network based on back gradient propagation. More importantly, we propose a Dual View Consistency (DVC) strategy to fully leverage the data stream, including the incoming and retrieved images. Besides learning the traditional input-output mapping, the network is also forced by DVC to explore a consistent mapping between the representations of two transformed inputs with the same label (dual view image pairs). More specifically, we maximize the mutual information among dual view image pairs, which constrains the network to mine the common semantic information between image pairs. In this way, the model can learn the invariant image representations, which is beneficial for improving classification accuracy. Extensive experiments demonstrate both MGI and DVC improve the performance of the proposed method, which is found to achieve state-of-the-art performance on three commonly used datasets.

To sum up, our contributions are as follows:

- Unlike existing methods that only focus on selecting samples from the memory bank for replay, we propose a novel yet effective framework for online class-incremental continual learning, which simultaneously considers optimal samples selection and sufficient exploration of the single-pass data stream.
- We propose a Maximally Gradient Interfered retrieval strategy to better maintain the performance of old classes, and offer a Dual View Consistency strategy to further improve the classification accuracy.
- Extensive empirical results demonstrate our method performs significantly better than existing methods on several benchmark datasets.

2. Related Work

2.1. Continual Learning

Continual learning [21, 32, 35, 37, 51, 52] aims to build a model that can learn sequentially and accumulate acquired knowledge over time. During the learning process, only a small portion of input data from one or a few tasks is available at once. The main challenge of continual learning is to learn without catastrophic forgetting: with the increase of new tasks or fields, the performance of previously learned tasks or fields should not significantly degrade over time. There are three major categories of continual learning methods based on the techniques they use: regularization-based [1, 32, 42, 55], parameter-isolation [31, 37, 54] , and replay-based methods [2, 13, 45]. Regularization-based methods attempt to impose constraints on the update of network parameters to mitigate catastrophic forgetting. Some regularization-based approaches [1, 32, 42, 55] add well-designed regularization terms into the loss function to penalize the update of critical model parameters. Some methods [12, 25, 34] adjust model's gradient during optimization to preserve previous knowledge, and others utilize knowledge distillation techniques [27] to reduce the feature drift on old tasks. Parameter-isolation methods expand the network and mask parameters to prevent forgetting, and each task has different parameters. Specifically, Parameter-isolation methods can be divided into two types: Fixed Architecture (FA) [19, 37, 44] and Dynamic Architecture (DA) [3, 43, 54]. The main difference between them is whether new parameters are introduced into the model. FA methods only activate relevant parameters for each task without modifying the architecture while DA methods introduce new parameters for new tasks and maintain old parameters unchanged.

Replay-based methods [2, 41, 45] utilize a memory bank to store a subset of data from the previous task, which have achieved appealing results in the online class-incremental continual setting. For each batch of incoming images, replay-based methods retrieve another mini-batch from the memory bank. Then these methods use the incoming and the retrieved images to update the model. Specifically, Chaudhry *et al.* proposed an Experience Replay (ER) [13] method, retrieving samples randomly from the memory bank for replay. Further, Maximally Interfered Retrieval (MIR) [2] takes a controlled sampling strategy of stored samples for replay, which retrieves the samples whose losses are most interfered by the new incoming samples. Shim *et al.* proposed an Adversarial Shapley value Experience Replay (ASER) [45] method, which leverages Shapley value adversarially in memory retrieval.

However, these replay-based methods for online class-incremental continual learning only focus on finding the optimal samples from the memory bank for replay, lacking a full utilization of the single-pass data stream. The proposed DVC strategy seeks to find a consistent mapping between the representations of dual view image pairs, which makes most of the single-pass data stream.

2.2. Contrastive learning

Contrastive learning is widely used in self-supervised representation learning [7, 8, 14, 23, 26, 47]. The general goal of contrastive learning is to learn a hidden space in which the representation of "similar" samples should be mapped close together, while that of "dissimilar" samples should be further away. Chen *et al.* proposed a simple framework SimCLR [14] to perform contrastive learning, where positive pairs are composed of two random augmented views of the same image and negative ones are obtained with different images. Moco [23] maintains the negative samples queue and converts a branch of Siamese network [16] into a momentum encoder to improve consistency of the queue. DenseCL [50] performs dense contrastive learning at the pixel level. Recently, contrastive learning has been used in

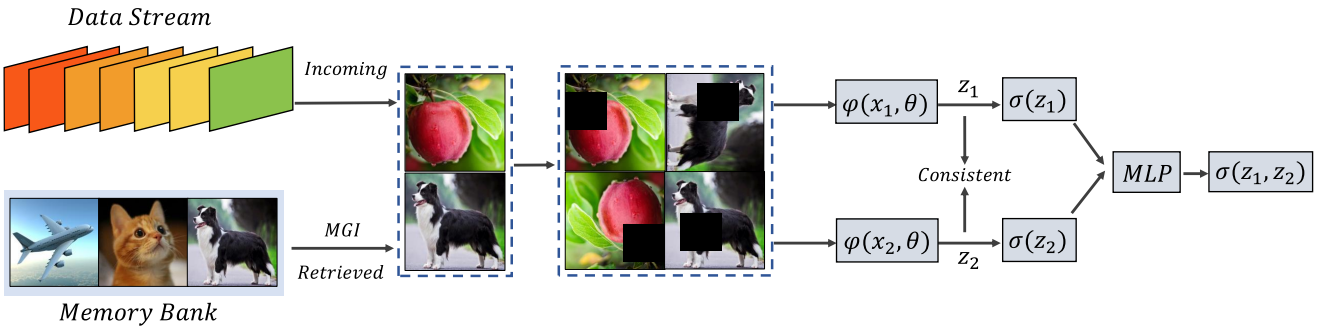

Figure 1. This is the flow of our method. At time t, the model receives an incoming batch of images from the data stream. Then the MGI retrieval strategy is used to select some samples from the memory bank for replay. The incoming and retrieved images are combined to form a training batch. Finally, the training batch of images is transformed into image pairs with different views, and sent to the same network to maximize the agreement of their representations. $\varphi(x_1, \theta)$ and $\varphi(x_2, \theta)$ share the same parameters θ.

various areas and shown promising results, including computer vision [10, 20], natural language processing [15, 18], graph [22, 40] and multi-modal data [6, 46]. Some key components contribute to the success of contrastive learning in learning useful representations, including proper data augmentations, the learnable nonlinear transformation between the representation, contrastive loss, and large batch size for negative samples.

Unlike the traditional contrastive learning methods which needs many negative samples, our method only requires dual view image pairs (positive samples). Therefore, our method can work well in cases where the batch size is small, while the performance of traditional contrastive learning methods may be affected.

3. Proposed Method

In this section, we first introduce the problem definition of online class-incremental continual learning, after which we detail the sample selection strategy based on the proposed MGI. Finally, we present the derivation and calculation process of the proposed DVC strategy.

3.1. Problem Definition

Following recent continual learning literature [2, 4, 5, 35, 45], we consider a supervised online class-incremental continual learning setting, where a model needs to learn new classes continually from an online data stream (the samples in the data stream can be seen only once). Consider a data stream $D = \{D_1, D_2, \ldots, D_N\}$ over $X \times Y$. X represents the samples, Y represents the labels of X, and N denotes the number of total tasks. Note that there is no overlap in the classes between tasks, which means $\{D_i\} \bigcap \{D_j\} = \emptyset$ for $i \neq j$. $\{D_t\}$ indicates the set of data for task t. In the training phase, the data stream can be seen only once, which means the data $\{D_t\}$ can be used to train the network one epoch in task t.

In addition, we adopt the single-head evaluation setup [11], where the task identity is unavailable during both the training and testing phases. Thus, the classifier must choose among all labels. The goal of task t is to train a model that can classify the classes belonging to D_t, while still preserving the ability to classify the classes belonging to $D_i, i < t$.

3.2. Maximally Gradient Interfered Retrieval

In online class-incremental continual learning, replay-based methods mitigate catastrophic forgetting by storing a subset of the samples from past tasks in a memory bank. For every incoming batch of images, replay-based methods retrieve another mini-batch of images from the memory bank. Then these methods update the model using both the incoming and retrieved images. Specially, the incoming batch of images can be used to train the model only once, meaning that the number of times old class images are retrieved is limited. Thus, every opportunity to retrieve samples from the memory bank is important to maintain the performance of old classes. In this paper, we select the stored samples whose gradients generated in the network are most interfered by the new incoming samples. We believe such samples are beneficial for updating the neural network based on back gradient propagation.

In the training process, the model receives a small batch B_t of size n at time t. We use the samples x_t of B_t to perform a virtual parameter update of current model $F(\theta)$, and the virtual updated model is denoted as $F_v(\theta_v)$. $\theta_v = \theta - \alpha \nabla \mathcal{L}(F(x_t), y_t)$, where α denotes the learning rate. Then we randomly select S candidate samples x_r from the memory bank, and compute the magnitude of the gradient vector caused by x_r in F and F_v, respectively:

$$G(x_r; \theta) = ||\nabla_\theta \mathcal{L}(y_t, F(x_r, \theta))||_1 \quad (1)$$
$$G(x_r; \theta_v) = ||\nabla_{\theta_v} \mathcal{L}(y_t, F_v(x_r, \theta_v))||_1,$$

then we compute the changes in gradient and sort the

changes in a descending order:

$$Score = Sort(G(x_r;\theta_v) - G(x_r;\theta)), \tag{2}$$

we then select the top k samples as the retrieved samples.

Specifically, we focus on the changes in the last Fully Connected (FC) layer of the network. The weight matrix of the FC layer is denoted by $w = [w_1, \cdots, w_C]^\top \in R^{C\times f}$. Taking the calculation process of the magnitude of the gradient $G(x_r; w)$ as an example. Let $h = [h_1, \cdots, h_L]^\top$ be a hidden feature vector in the model, which is the input of the FC layer. $F(x_r, w) = [f_1, \cdots, f_C]^\top$ is the output of the FC layer after being processed by softmax function, and $y_r = [y_1, \ldots, y_C]$ is the ground truth.

The gradient of the cross-entropy loss w.r.t. w_{cl} is formulated as follows:

$$\frac{\partial}{\partial w_{cl}}\mathcal{L}(y_r, F(x_r, w)) = (f_c - y_c)h_l. \tag{3}$$

Thus, the calculation process of magnitude of the gradient $G(x_r; w)$ can be formulated as follows:

$$\begin{aligned} G(x_r; w) &= \sum_{c}^{C}\sum_{l}^{L}|(f_c - y_c)h_l| \\ &= \sum_{c}^{C}|(f_c - y_c)|\sum_{l}^{L}|h_l| \\ &= ||F(x_r, w) - y_r||_1 \cdot ||h||_1. \end{aligned} \tag{4}$$

$G(x_r; w_v)$ can be obtained in a manner similar to $G(x_r; w)$.

3.3. Dual View Consistency

In online class-incremental continual learning, the incoming data can be seen only once for every task. In other words, the incoming images can be used to update the model only one epoch in the training process, which means the passing images are far from being fully utilized. To make full use of the single-pass data stream, the proposed DVC strategy forces the network not only to learn to classify the input images, but also to make the representations of dual view image pairs consistent. Specifically, we use mutual information to estimate the consistency.

Mutual Information Estimation. Mutual Information (MI) is a fundamental quantity to measure the relationship between random variables. The MI between X_1 and X_2 can be understood as the decrease of the uncertainty in X_1 given X_2:

$$I(X_1; X_2) = H(X_1) - H(X_1|X_2), \tag{5}$$

and $I(X_1; X_2)$ can also be expressed equivalently as follows:

$$\begin{aligned} I(X_1; X_2) &= H(X_2) - H(X_2|X_1) \\ &= H(X_1, X_2) - H(X_1|X_2) - H(X_2|X_1). \end{aligned} \tag{6}$$

According to Eqs. (5) and (6), we reformulate $I(X_1; X_2)$ as follows:

$$\begin{aligned} 3I(X_1; X_2) = H(X_1) + H(X_2) + H(X_1, X_2) \\ - 2H(X_1|X_2) - 2H(X_2|X_1). \end{aligned} \tag{7}$$

Further, Eq. (7) can be rewritten as follows:

$$\begin{aligned} I(X_1; X_2) = \frac{1}{3}(H(X_1) + H(X_2) + H(X_1, X_2)) \\ + \frac{2}{3}(\sum_{x_1,x_2} p(x_1, x_2) log\frac{p(x_1, x_2)}{p(x_2)} \\ + \sum_{x_1,x_2} p(x_1, x_2) log\frac{p(x_1, x_2)}{p(x_1)}), \end{aligned} \tag{8}$$

where $p(x_1), p(x_2)$ represent the marginal distributions of x_1 and x_2, respectively. And $p(x_1, x_2)$ represents the joint distribution of x_1 and x_2. By forcing the joint distribution to be the same as the marginal distribution, $I(X_1; X_2)$ can be approximated as follows:

$$I(X_1; X_2) \approx \frac{1}{3}(H(X_1) + H(X_2) + H(X_1, X_2)). \tag{9}$$

Joint Distribution Estimation. In the training process, the model receives a small batch B_t of size n at time t. The joint probability matrix $\boldsymbol{P} \in \mathbb{R}^{C\times C}$ can be computed by Invariant Information Clustering (IIC) [28] as :

$$\boldsymbol{P} = \frac{1}{n}\sum_{i=1}^{n} F(\boldsymbol{x}_i^1) \cdot F(\boldsymbol{x}_i^2)^\top, \tag{10}$$

where x_i^1 and x_i^2 are two transformed versions of the same image x_i, $x_i \in B_t$. $F(x) = \sigma(\varphi(x)) = softmax(z) \in [0, 1]^C$. This can be interpreted as the distribution of a discrete random variable z over C classes, formally given by $P(z = c|x) = F_c(x)$. Each element of $\boldsymbol{P}$ at row c_1 and column c_2 denotes the joint probability $\boldsymbol{P}_{c1c2} = P(z_1 = c_1, z_2 = c_2)$. The marginal distributions $\boldsymbol{P}_{c_1} = P(z_1 = c_1)$ and $\boldsymbol{P}_{c_2} = P(z_2 = c_2)$ can be obtained by summing over the rows and columns of this matrix $\boldsymbol{P}$. Since we generally consider symmetric problems and $\boldsymbol{P}_{c1c2} = \boldsymbol{P}_{c2c1}$, $\boldsymbol{P}$ is symmetrized by $\boldsymbol{P} = \frac{\boldsymbol{P}+\boldsymbol{P}^\top}{2}$.

Such a joint distribution can also be estimated by a 2-layer Multi-Layer Perceptron (MLP) [7]. As shown in Fig. 1, after we add a 2-layer MLP at the end of the backbone network, the outputs of the two transformed images are concatenated and sent to the MLP layer to get the joint distribution. The features of the two inputs belong to the same class and so is the joint distribution.

Optimization Objective Function. In order to maximize $I(\boldsymbol{z}_1; \boldsymbol{z}_2)$, the MI loss can be formulated as follows:

$$\mathcal{L}_{MI} = -I(\boldsymbol{z}_1; \boldsymbol{z}_2). \tag{11}$$

We use the approximate calculation of MI mentioned above (Eq. (9)) to compute $\mathcal{L}_{MI}$, thus we must have a distance loss to constrain the distance between the joint distribution and the marginal distributions:

$$\mathcal{L}_{DL} = L_1(p(z_1, z_2), p(z_1)) + L_1(p(z_1, z_2), p(z_2)), \quad (12)$$

where L_1 indicates Mean Absolute Error (MAE) loss. The total loss function $\mathcal{L}_T$ can be formulated as follows:

$$\mathcal{L}_T = \lambda_1 \mathcal{L}_{CE} + \lambda_2 \mathcal{L}_{MI} + \lambda_3 \mathcal{L}_{DL}, \quad (13)$$

where λ_1, λ_2 and λ_3 are the balance coefficients of the three losses. $\mathcal{L}_{CE}$ denotes Cross-Entropy loss.

Compared with traditional contrastive learning, our method only needs to maximize the MI between dual view image pairs and does not require the participation of negative samples. We compare our method with the contrastive learning method SCR [36] in online class-incremental continual learning in Sec. 4.5.

4. Experiments

In this section, we review the benchmark datasets, metrics, baselines we compared against and our experimental setting. We then report and analyze the results to validate the effectiveness of our approach.

4.1. Datasets

Split CIFAR-10 splits the CIFAR-10 [29] dataset into 5 disjoint sub-datasets for 5 tasks, and each task has 2 classes.

Split CIFAR-100 splits the CIFAR-100 [29] dataset into 10 sub-datasets for 10 tasks with non-overlapping classes, and each task has 10 classes.

Split Mini-ImageNet splits the Mini-ImageNet [48] dataset into 10 sub-datasets for 10 disjoint tasks, and each task contains 10 classes.

We conduct several experiments on these datasets and take the average results of these experiments as the final results. For a fair comparison, the classes in each task and the order of tasks are fixed in all experiments.

In the original Mini-ImageNet, 100 classes are divided into three parts, including 64, 16, and 20 classes respectively. In this paper, following [45], we combine the three parts into one dataset. We then split the combined dataset into 10 sub-datasets. The first task contains the first 10 classes; the second task contains the next 10 classes, and so on.

4.2. Metrics

Continual learning aims to build a model that can learn sequentially and accumulate acquired knowledge over time. Thus, we use two standard metrics in the continual learning literature to measure performance: Average Accuracy and Average Forgetting. Average Accuracy measures the overall performance of testing sets from seen tasks, and Average Forgetting measures how much the learned knowledge the algorithm has forgotten.

Let $a_{i,j}$ be the performance of the model on the held-out testing set of task j after the model is trained from task 1 to i. f_j represents how much the model forgets about task j after being trained on task i. For T tasks :

$$Average\ Accuracy(A_T) = \frac{1}{T}\sum_{j=1}^{T} a_{T,j}. \quad (14)$$

$$\begin{aligned} Average\ Forgetting(F_T) &= \frac{1}{T-1}\sum_{j=1}^{T-1} f_{T,j}, \\ where\ f_{i,j} &= \max_{l\in\{1,\ldots,i-1\}} a_{l,j} - a_{i,j}. \end{aligned} \quad (15)$$

4.3. Baselines

We compare our method with several state-of-the-art online class-incremental continual learning algorithms:

- **fine-tune:** As an important baseline in previous works [2, 9, 11, 13, 45], it only trains the model in the order the data is presented without any specific method for forgetting avoidance.
- **iid-offline:** This is not the continual learning method, but the upper limit of the performance of the continual learning method; iid-offline trains the model over multiple epochs on the dataset with i.i.d. sampled mini-batch. More specifically, we use 50 epochs for iid-offline training in all experiments.
- **EWC [11]:** Elastic Weight Consolidation, a prior-focused method that limits the update of parameters that were important to the past tasks, as measured by the fisher information matrix.
- **A-GEM [12]:** Averaged Gradient Episodic Memory, a method that uses the samples in the memory bank to constrain the parameter updates.
- **ER** [13]: Experience Replay, which retrieves samples randomly from the memory bank and updates the memory bank via reservoir sampling.
- **MIR** [2]: Maximally Interfered Retrieval, which retrieves memory samples whose losses are most interfered by the foreseen parameters update.
- **GSS** [4]: Gradient-based Sample Selection, different from MIR, GSS pays attention to the memory update strategy. It aims to diversify the gradients of the samples in the replay memory.

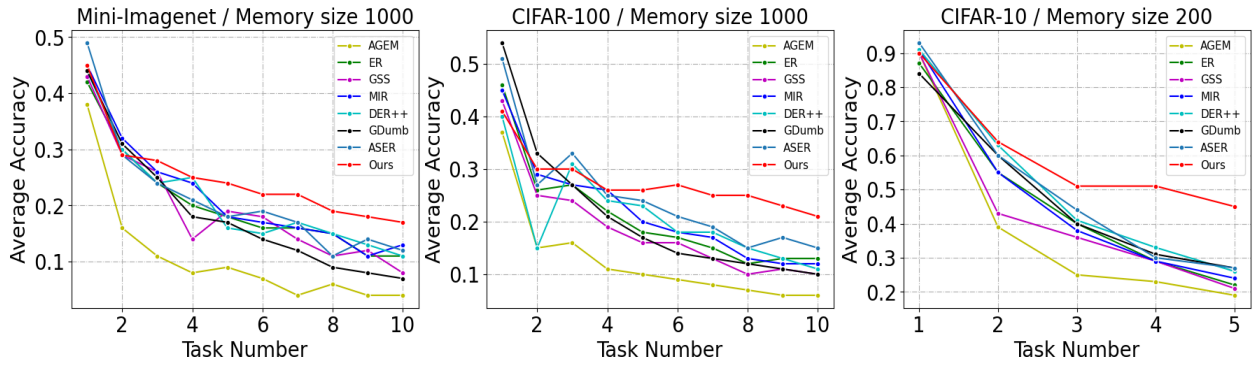

Figure 2. Average Accuracy on observed tasks. Our method outperforms other baselines especially when the model sees more classes.

- **DER++** [9]: Dark Experience Replay, which leverages Knowledge Distillation for retaining past experience.
- **GDumb** [39]: Greedy sampler and Dumb learner, which greedily updates the memory buffer with the constraint to keep a balanced class distribution. At inference, it trains a model from scratch using the balanced memory buffer only.
- **ASER** [45]: Adversarial Shapley value Experience Replay, which scores the samples in the memory bank according to their ability to preserve latent decision boundaries for previously observed classes while interfering with latent decision boundaries of current classes being learned.

4.4. Implementation Detail

Following the existing online class-incremental continual learning methods [2,4,45], we use a reduced Resnet18 as our backbone network for all the datasets. We use Stochastic Gradient Descent (SGD) to optimize the network and set the learning rate to 0.1. The model receives a batch of size 10 at a time from the data stream, and the size k of the retrieved batch is also set to 10 irrespective of the size of the memory. The number S of candidate samples is set to 50. For CIFAR-100 and Mini-ImageNet, $\lambda_1 = \lambda_2 = 1$, $\lambda_3 = 4$. For CIFAR-10, $\lambda_1 = \lambda_2 = 1$, $\lambda_3 = 2$.

4.5. Comparative Performance Evaluation

Tab. 1 shows Average Accuracy by the end of the data stream for Mini-ImageNet, CIFAR-100 and CIFAR-10. As the table shows, our method shows significantly improved performance on three standard datasets.

From a dataset perspective, the improvement of our method on CIFAR-10 is greater than that on CIFAR-100 and Mini-ImageNet. More specifically, our method has an average improvement of 3% on Mini-ImageNet, 3.7% on CIFAR-100, and 11.7% on CIFAR-10 compared to the strongest baseline ASER. Actually, the overall performance of all mentioned online class-incremental continual learning methods on Mini-ImageNet and CIFAR-100 is lower than on CIFAR-10. The reason for this performance difference is that Mini-ImageNet and CIFAR-100 contain more classes and are divided into more tasks than CIFAR-10. More classes and longer learning sequences increase the difficulty of online continual learning.

From a memory bank size perspective, our method performs better is cases where the size of the memory bank is small. Taking the experiments on CIFAR-10 as an example, the improvements of our method with M=1k, 0.5k and 0.2k are 7%, 12.4% and 15.8% compared to ASER, respectively. The smaller the memory size, the more significant the performance improvement of our method. This phenomenon also occurs on the experiments of CIFAR-100 and Mini-ImageNet. In other words, our method can perform better in the case of limited storage space, which is also in line with practical needs.

From the perspective of the online continual learning methods, our method exceeds existing online continual learning methods in most cases. Only a few cases (M=5k on Mini-ImageNet) where the performance of our method is slightly lower than GDumb. Strictly speaking, GDumb is not specifically designed for continual learning problems, but it has competitive performance on continual learning tasks. As mentioned in [35], GDumb achieves the best performance with a large memory buffer, but it achieves poor performance when the memory buffer is small. Unlike GDumb, our method performs well on all sizes of memory buffer settings, especially on small ones.

We also compare our method with SCR [36], which applies supervised contrastive learning into online class-incremental continual learning. Similar to traditional contrastive learning methods, SCR requires a large batch size of images for training, so it retrieves 100 images from the memory bank for each update of the model. However, the performance of SCR is poor when the retrieved size is small. Tab. 3 shows the comparison results when the number of

Method	M=1k	M=2k	M=5k	M=1k	M=2k	M=5k	M=0.2k	M=0.5k	M=1k
finetune	4.3 ± 0.2	4.3 ± 0.2	4.3 ± 0.2	5.8 ± 0.3	5.8 ± 0.3	5.8 ± 0.3	18.1 ± 0.3	18.1 ± 0.3	18.1 ± 0.3
iid offline	51.4 ± 0.2	51.4 ± 0.2	51.4 ± 0.2	49.6 ± 0.2	49.6 ± 0.2	49.6 ± 0.2	81.7 ± 0.1	81.7 ± 0.1	81.7 ± 0.1
EWC	3.1 ± 0.3	3.1 ± 0.3	3.1 ± 0.3	4.8 ± 0.2	4.8 ± 0.2	4.8 ± 0.2	17.9 ± 0.3	17.9 ± 0.3	17.9 ± 0.3
A-GEM	4.4 ± 0.2	4.3 ± 0.2	4.3 ± 0.2	6.0 ± 0.1	6.0 ± 0.1	5.9 ± 0.2	18.1 ± 0.3	18.3 ± 0.1	18.3 ± 0.1
ER	10.2 ± 0.5	12.9 ± 0.8	16.4 ± 0.9	11.6 ± 0.5	15.0 ± 0.5	20.5 ± 0.8	23.2 ± 1.0	31.2 ± 1.4	39.7 ± 1.3
GSS	9.3 ± 0.8	14.1 ± 1.1	15.0 ± 1.1	9.7 ± 0.2	12.4 ± 0.6	16.8 ± 0.8	23.0 ± 0.9	28.3 ± 1.7	37.1 ± 1.6
MIR	10.1 ± 0.6	14.2 ± 0.9	18.5 ± 1.0	11.3 ± 0.3	15.1 ± 0.3	22.2 ± 0.7	24.6 ± 0.6	32.5 ± 1.5	42.8 ± 1.4
GDumb	7.3 ± 0.3	11.4 ± 0.2	$\mathbf{19.5 \pm 0.5}$	10.0 ± 0.2	13.3 ± 0.4	19.2 ± 0.4	26.6 ± 1.0	31.9 ± 0.9	37.5 ± 1.1
DER++	10.9 ± 0.6	15.0 ± 0.7	17.4 ± 1.5	11.8 ± 0.4	15.7 ± 0.5	20.8 ± 0.8	28.1 ± 1.2	35.4 ± 1.3	42.8 ± 1.9
ASER	11.5 ± 0.6	13.5 ± 0.8	17.8 ± 1.0	14.3 ± 0.5	17.8 ± 0.5	22.8 ± 1.0	29.6 ± 1.0	38.2 ± 1.0	45.1 ± 2.0
Ours	$\mathbf{15.4 \pm 0.7}$	$\mathbf{17.2 \pm 0.8}$	19.1 ± 0.9	$\mathbf{19.7 \pm 0.7}$	$\mathbf{22.1 \pm 0.9}$	$\mathbf{24.1 \pm 0.8}$	$\mathbf{45.4 \pm 1.4}$	$\mathbf{50.6 \pm 2.9}$	$\mathbf{52.1 \pm 2.5}$
	(a) Mini-ImageNet			(b) CIFAR-100			(c) CIFAR-10		

Table 1. Average Accuracy (higher is better), M denotes the memory buffer size. All numbers are the average of 15 runs. The data in the table represents Average Accuracy $\pm$ variance.

Method	M=1k	M=2k	M=5k
Baseline	10.2 ± 0.5	12.9 ± 0.8	$\mathbf{16.3 \pm 0.9}$
+ SeparateAug	11.7 ± 0.5	11.8 ± 0.9	12.7 ± 1.4
+ CombineAug	$\mathbf{12.3 \pm 0.8}$	$\mathbf{13.3 \pm 1.0}$	14.8 ± 0.9

Table 2. Impact of different ways in which augmented images are sent to network on performance. The experiments are performed on Mini-ImageNet. All numbers are the average of 15 runs.

Method	M=1k	M=2k	M=5k
SCR	13.1 ± 0.3	14.9 ± 0.2	16.3 ± 0.4
Ours	$\mathbf{19.7 \pm 0.7}$	$\mathbf{22.1 \pm 0.9}$	$\mathbf{24.1 \pm 0.8}$

Table 3. Comparison of our method with SCR [36]. To be fair, for both methods, we retrieve 10 images from the memory bank in each iteration. The experiments are performed on CIFAR-100. All numbers are the average of 15 runs.

the retrieved images is 10. As the table shows, our method is significantly better than SCR. This is because SCR requires a large number of negative samples. Unlike SCR, our method only requires dual view image pairs, so it can work well in cases where the batch size is small.

In addition, we find that the improvement of previous methods which focus on selecting samples is unstable. For example, the performance of MIR is better than that of ER on Mini-ImageNet with M=2k, but the situation is reversed with M=1k. ASER has better performance than MIR on Mini-ImageNet with M=1k, but it is slightly lower than MIR with M=2k and M=5k. As a comparison, our method performs better than the strongest baseline ASER on all the datasets with all memory buffer sizes.

Fig. 2 shows that our method is consistently better than other baselines on three datasets. Our method achieves better accuracies as the tasks increase, proving that it is a powerful method to overcome catastrophic forgetting.

Tab. 4 shows the Average Forgetting by the end of the data stream for Mini-ImageNet, CIFAR-100 and CIFAR-10. Our method shows the lowest Average Forgetting on three datasets in most cases. In CIFAR-10, compared to the strongest baseline ASER, our method achieves 29.2% , 26.2% and 19.9 % reduction of Average Forgetting with M=0.2k, 0.5k and 1k, respectively. Similarly, the reduction of Average Forgetting is also significant on Mini-ImageNet and CIFAR-100.

In summary, we have demonstrated the effectiveness of our method in overcoming catastrophic forgetting. Our method achieves competitive performance on three commonly used benchmark datasets.

4.6. Ablation Study

In this section, we verify the effectiveness of each component of our method. As shown in Tab. 5, the DVC dramatically improves the performance of the baseline, both on Average Accuracy and Average Forgetting. In particular, the DVC can significantly improve the performance of the baseline when the memory bank size is small. For example, at a memory bank size of 1k, the model with DVC improves 7.1% Average Accuracy and reduces 7.8% Average Forgetting compared to baseline. In addition, MGI also improves the performance of the model in various experiments, although the improvements are less than DVC. In

Method	M=1k	M=2k	M=5k	M=1k	M=2k	M=5k	M=0.2k	M=0.5k	M=1k
fine-tune	35.6 ± 0.9	35.6 ± 0.9	35.6 ± 0.9	50.4 ± 1.0	50.4 ± 1.0	50.4 ± 1.0	81.7 ± 0.7	81.7 ± 0.7	81.7 ± 0.7
EWC	28.1 ± 0.8	28.1 ± 0.8	28.1 ± 0.8	39.1 ± 1.2	39.1 ± 1.2	39.1 ± 1.2	81.5 ± 1.4	81.5 ± 1.4	81.5 ± 1.4
A-GEM	35.5 ± 0.8	35.7 ± 1.1	35.1 ± 0.8	43.3 ± 0.7	43.3 ± 0.7	43.1 ± 0.7	66.1 ± 1.0	66.4 ± 0.8	66.4 ± 0.9
ER	32.7 ± 0.9	29.1 ± 0.7	26.0 ± 1.0	39.1 ± 0.9	34.6 ± 0.9	30.6 ± 0.9	60.9 ± 1.0	50.2 ± 2.5	39.5 ± 1.6
GSS	33.5 ± 0.8	28.0 ± 0.7	27.5 ± 1.2	38.2 ± 0.7	34.3 ± 0.6	30.2 ± 0.8	62.2 ± 1.3	55.3 ± 1.3	44.9 ± 1.4
MIR	31.5 ± 1.2	25.6 ± 1.1	$\mathbf{20.4 \pm 1.0}$	39.5 ± 0.6	33.3 ± 0.8	28.3 ± 0.7	61.8 ± 1.0	51.5 ± 1.4	38.0 ± 1.5
DER++	33.8 ± 0.8	28.6 ± 0.8	27.1 ± 1.3	41.9 ± 0.6	36.7 ± 0.5	33.5 ± 0.8	55.9 ± 1.8	45.0 ± 1.0	34.6 ± 2.8
ASER	33.8 ± 1.3	30.5 ± 1.3	25.1 ± 0.8	43.0 ± 0.5	37.9 ± 0.6	29.6 ± 0.9	56.4 ± 1.6	47.5 ± 1.3	39.6 ± 2.0
Ours	$\mathbf{25.1 \pm 0.7}$	$\mathbf{23.1 \pm 0.7}$	21.9 ± 0.8	$\mathbf{30.6 \pm 0.7}$	$\mathbf{27.8 \pm 1.0}$	$\mathbf{26.1 \pm 0.5}$	$\mathbf{27.2 \pm 2.5}$	$\mathbf{21.3 \pm 3.1}$	$\mathbf{19.7 \pm 2.9}$
	(a) Mini-ImageNet			(b) CIFAR-100			(c) CIFAR-10		

Table 4. Average Forgetting (lower is better). M denotes the memory buffer size. The data in the table represents Average Forgetting $\pm$ variance.

Method	M=1k (AA ↑ / AF ↓)	M=2k (AA ↑ / AF ↓)	M=5k (AA ↑ / AF ↓)
Baseline	11.6 ± 0.5 / 39.1 ± 0.9	15.0 ± 0.5 / 34.6 ± 0.9	20.5 ± 0.8 / 30.6 ± 0.9
Baseline + DVC	18.7 ± 0.8 / 31.3 ± 1.0	21.7 ± 0.8 / 28.2 ± 1.0	22.4 ± 1.4 / 29.2 ± 1.0
Baseline + DVC + MGI	$\mathbf{19.7 \pm 0.7}$ / $\mathbf{30.6 \pm 0.7}$	$\mathbf{22.1 \pm 0.9}$ / $\mathbf{27.8 \pm 1.0}$	$\mathbf{24.1 \pm 0.8}$ / $\mathbf{26.1 \pm 0.5}$

Table 5. Ablation studies on CIFAR-100. "Baseline" represents the model with ER method. "DVC" represents the proposed Dual View Consistency strategy. "MGI " represents the proposed Maximally Gradient Interfered strategy. "AA" represents the Average Accuracy and "AF" represents the Average Forgetting. All numbers are the average of 15 runs.

contrast to DVC, MGI works better when the memory bank size is large. The reason is apparent—as the number of samples available increases, the strategy of selecting samples becomes more critical.

We also explore the impact of different ways to send augmented images to the network. As shown in Tab. 2, the "SeparateaAug" means that we send the two augmented images to the network separately. This operation can improve the performance of the model in case where the memory bank size is small (M=1k). However, as the size of the memory bank increases, "SeparateaAug" may hurt the performance (M=1k and M=5k). We think this is because the augmented images increase the burden of model learning under the single-pass setting. "CombineAug" means that we combine the two augmented images into one batch for training. It can reduce the learning burden brought by the augmented images and improve the performance of the model. We believe this is because a large batch size of images is helpful in updating the batch normalization layer, especially in online continual learning.

5. Conclusion

In this paper, we propose a novel yet effective framework for online class-incremental continual learning, which considers not only the selection of optimal samples, but also the full exploration of semantic information in the single-pass data stream. Specifically, we select the stored samples whose gradients generated in the network are most interfered by the new incoming samples. We believe such samples are beneficial for updating the neural network based on back gradient propagation. More importantly, we maximize the mutual information among dual view image pairs, which constrains the network to mine the common semantic information between image pairs. In this way, the model can learn the invariant image representations, which is beneficial for improving classification accuracy. Extensive experiments on three commonly used benchmark datasets demonstrate the effectiveness of our method.

6. Acknowledgement

Our work was supported in part by the National Key R&D Program of China under Grant 2017YFE0104100, in part by the National Natural Science Foundation of China under Grant 62132016, Grant 62171343, and Grant 62071361, in part by Key Research and Development Program of Shaanxi under Grant 2021ZDLGY01-03, and in part by the Fundamental Research Funds for the Central Universities ZDRC2102.

References

[1] Rahaf Aljundi, Francesca Babiloni, Mohamed Elhoseiny, Marcus Rohrbach, and Tinne Tuytelaars. Memory aware synapses: Learning what (not) to forget. In *ECCV*, pages 139–154, 2018. 1, 2

[2] Rahaf Aljundi, Eugene Belilovsky, Tinne Tuytelaars, Laurent Charlin, Massimo Caccia, Min Lin, and Lucas Page-Caccia. Online continual learning with maximal interfered retrieval. In *NeurIPS*, pages 11849–11860, 2019. 1, 2, 3, 5, 6

[3] Rahaf Aljundi, Punarjay Chakravarty, and Tinne Tuytelaars. Expert gate: Lifelong learning with a network of experts. In *CVPR*, pages 3366–3375, 2017. 2

[4] Rahaf Aljundi, Min Lin, Baptiste Goujaud, and Yoshua Bengio. Gradient based sample selection for online continual learning. In *NeurIPS*, pages 11816–11825, 2019. 3, 5, 6

[5] Rahaf Aljundi, Min Lin, Baptiste Goujaud, and Yoshua Bengio. Online continual learning with no task boundaries. *arXiv preprint arXiv:1903.08671*, 2019. 3

[6] Relja Arandjelovic and Andrew Zisserman. Objects that sound. In *ECCV*, pages 435–451, 2018. 3

[7] Rowel Atienza. Improving model generalization by agreement of learned representations from data augmentation. In *IEEE/CVF Winter Conference on Applications of Computer Vision*, 2022. 2, 4

[8] Philip Bachman, R Devon Hjelm, and William Buchwalter. Learning representations by maximizing mutual information across views. *arXiv preprint arXiv:1906.00910*, 2019. 2

[9] Pietro Buzzega, Matteo Boschini, Angelo Porrello, Davide Abati, and Simone Calderara. Dark experience for general continual learning: a strong, simple baseline. *arXiv preprint arXiv:2004.07211*, 2020. 5, 6

[10] Mathilde Caron, Ishan Misra, Julien Mairal, Priya Goyal, Piotr Bojanowski, and Armand Joulin. Unsupervised learning of visual features by contrasting cluster assignments. *arXiv preprint arXiv:2006.09882*, 2020. 3

[11] Arslan Chaudhry, Puneet K Dokania, Thalaiyasingam Ajanthan, and Philip HS Torr. Riemannian walk for incremental learning: Understanding forgetting and intransigence. In *ECCV*, pages 532–547, 2018. 3, 5

[12] Arslan Chaudhry, Marc'Aurelio Ranzato, Marcus Rohrbach, and Mohamed Elhoseiny. Efficient lifelong learning with a-gem. In *ICLR*, 2019. 2, 5

[13] Arslan Chaudhry, Marcus Rohrbach, Mohamed Elhoseiny, Thalaiyasingam Ajanthan, Puneet K Dokania, Philip HS Torr, and Marc'Aurelio Ranzato. On tiny episodic memories in continual learning. *arXiv preprint arXiv:1902.10486*, 2019. 1, 2, 5

[14] Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. A simple framework for contrastive learning of visual representations. pages 1597–1607. PMLR, 2020. 2

[15] Zewen Chi, Li Dong, Furu Wei, Nan Yang, Saksham Singhal, Wenhui Wang, Xia Song, Xian-Ling Mao, Heyan Huang, and Ming Zhou. Infoxlm: An information-theoretic framework for cross-lingual language model pre-training. *arXiv preprint arXiv:2007.07834*, 2020. 3

[16] Sumit Chopra, Raia Hadsell, and Yann LeCun. Learning a similarity metric discriminatively, with application to face verification. In *CVPR*, volume 1, pages 539–546. IEEE, 2005. 2

[17] 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 TPAMI*, 2021. 1

[18] Hongchao Fang, Sicheng Wang, Meng Zhou, Jiayuan Ding, and Pengtao Xie. Cert: Contrastive self-supervised learning for language understanding. *arXiv preprint arXiv:2005.12766*, 2020. 3

[19] Chrisantha Fernando, Dylan Banarse, Charles Blundell, Yori Zwols, David Ha, Andrei A Rusu, Alexander Pritzel, and Daan Wierstra. Pathnet: Evolution channels gradient descent in super neural networks. *arXiv preprint arXiv:1701.08734*, 2017. 2

[20] Jean-Bastien Grill, Florian Strub, Florent Altché, Corentin Tallec, Pierre H Richemond, Elena Buchatskaya, Carl Doersch, Bernardo Avila Pires, Zhaohan Daniel Guo, Mohammad Gheshlaghi Azar, et al. Bootstrap your own latent: A new approach to self-supervised learning. *arXiv preprint arXiv:2006.07733*, 2020. 3

[21] Yanan Gu, Cheng Deng, and Kun Wei. Class-incremental instance segmentation via multi-teacher networks. In *AAAI*, volume 35, pages 1478–1486, 2021. 2

[22] Kaveh Hassani and Amir Hosein Khasahmadi. Contrastive multi-view representation learning on graphs. In *ICCV*, pages 4116–4126. PMLR, 2020. 3

[23] Kaiming He, Haoqi Fan, Yuxin Wu, Saining Xie, and Ross Girshick. Momentum contrast for unsupervised visual representation learning. In *CVPR*, pages 9729–9738, 2020. 2

[24] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *CVPR*, pages 770–778, 2016. 1

[25] Xu He and Herbert Jaeger. Overcoming catastrophic interference using conceptor-aided backpropagation. In *ICLR*, 2018. 2

[26] Olivier Henaff. Data-efficient image recognition with contrastive predictive coding. pages 4182–4192. PMLR, 2020. 2

[27] Geoffrey Hinton, Oriol Vinyals, and Jeff Dean. Distilling the knowledge in a neural network. *Computer Science*, 14(7):38–39, 2015. 2

[28] Xu Ji, Joao F Henriques, and Andrea Vedaldi. Invariant information clustering for unsupervised image classification and segmentation. In *ICCV*, pages 9865–9874, 2019. 4

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

[30] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. Imagenet classification with deep convolutional neural networks. *NeurIPS*, 25:1097–1105, 2012. 1

[31] Soochan Lee, Junsoo Ha, Dongsu Zhang, and Gunhee Kim. A neural dirichlet process mixture model for task-free continual learning. In *ICLR*, 2020. 2

[32] Sang-Woo Lee, Jin-Hwa Kim, Jaehym Jun, Jung-Woo Ha, and Byoung-Tak Zhang. Overcoming catastrophic forgetting

by incremental moment matching. In *ECCV*, pages 4652–4662, 2017. 1, 2

[33] Xiangyu Li, Zhe Xu, Kun Wei, and Cheng Deng. Generalized zero-shot learning via disentangled representation. In *AAAI*, volume 35, pages 1966–1974, 2021. 1

[34] David Lopez-Paz and Marc'Aurelio Ranzato. Gradient episodic memory for continual learning. In *NeurIPS*, pages 6467–6476, 2017. 1, 2

[35] 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, 2, 3, 6

[36] 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 *CVPRW*, pages 3589–3599, 2021. 5, 6, 7

[37] Arun Mallya and Svetlana Lazebnik. Packnet: Adding multiple tasks to a single network by iterative pruning. In *CVPR*, pages 7765–7773, 2018. 1, 2

[38] Marc Masana, Xialei Liu, Bartlomiej Twardowski, Mikel Menta, Andrew D Bagdanov, and Joost van de Weijer. Class-incremental learning: survey and performance evaluation. *arXiv preprint arXiv:2010.15277*, 2020. 1

[39] Ameya Prabhu, Philip HS Torr, and Puneet K Dokania. Gdumb: A simple approach that questions our progress in continual learning. In *ECCV*, pages 524–540. Springer, 2020. 6

[40] Jiezhong Qiu, Qibin Chen, Yuxiao Dong, Jing Zhang, Hongxia Yang, Ming Ding, Kuansan Wang, and Jie Tang. Gcc: Graph contrastive coding for graph neural network pre-training. In *Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, pages 1150–1160, 2020. 3

[41] Matthew Riemer, Ignacio Cases, Robert Ajemian, Miao Liu, Irina Rish, Yuhai Tu, and Gerald Tesauro. Learning to learn without forgetting by maximizing transfer and minimizing interference. In *ICLR*, 2018. 2

[42] Hippolyt Ritter, Aleksandar Botev, and David Barber. Online structured laplace approximations for overcoming catastrophic forgetting. In *NeurIPS*, pages 3738–3748, 2018. 1, 2

[43] 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

[44] Joan Serra, Didac Suris, Marius Miron, and Alexandros Karatzoglou. Overcoming catastrophic forgetting with hard attention to the task. pages 4548–4557. PMLR, 2018. 2

[45] Dongsub Shim, Zheda Mai, Jihwan Jeong, Scott Sanner, Hyunwoo Kim, and Jongseong Jang. Online class-incremental continual learning with adversarial shapley value. In *AAAI*, 2021. 1, 2, 3, 5, 6

[46] Chen Sun, Fabien Baradel, Kevin Murphy, and Cordelia Schmid. Learning video representations using contrastive bidirectional transformer. *arXiv preprint arXiv:1906.05743*, 2019. 3

[47] Yonglong Tian, Dilip Krishnan, and Phillip Isola. Contrastive multiview coding. In *ECCV*, pages 776–794. Springer, 2020. 2

[48] Oriol Vinyals, Charles Blundell, Timothy Lillicrap, Koray Kavukcuoglu, and Daan Wierstra. Matching networks for one shot learning. In *NeurIPS*, 2016. 5

[49] Xiumei Wang, Yanan Gu, Xinbo Gao, and Zheng Hui. Dual residual attention module network for single image super resolution. *Neurocomputing*, 364:269–279, 2019. 1

[50] Xinlong Wang, Rufeng Zhang, Chunhua Shen, Tao Kong, and Lei Li. Dense contrastive learning for self-supervised visual pre-training. In *CVPR*, pages 3024–3033, 2021. 2

[51] Kun Wei, Cheng Deng, and Xu Yang. Lifelong zero-shot learning. In *IJCAI*, pages 551–557, 2020. 2

[52] Kun Wei, Cheng Deng, Xu Yang, and Dacheng Tao. Incremental zero-shot learning. *IEEE Transactions on Cybernetics*, 2021. 2

[53] Xu Yang, Cheng Deng, Tongliang Liu, and Dacheng Tao. Heterogeneous graph attention network for unsupervised multiple-target domain adaptation. *IEEE TPAMI*, 2020. 1

[54] Jaehong Yoon, Eunho Yang, Jeongtae Lee, and Sung Ju Hwang. Lifelong learning with dynamically expandable networks. 2018. 2

[55] Friedemann Zenke, Ben Poole, and Surya Ganguli. Continual learning through synaptic intelligence. In *Proceedings of machine learning research*, pages 3987–3995, 2017. 1, 2

[56] Han Zhao, Xu Yang, Zhenru Wang, Erkun Yang, and Cheng Deng. Graph debiased contrastive learning with joint representation clustering. In *IJCAI*, pages 3434–3440, 2021. 1