Overcoming Catastrophic Forgetting for Continual Learning via Feature Propagation

Xuejun Han[1]
xuejunhan@cmail.carleton.ca
Yuhong Guo[1,2]
yuhong.guo@carleton.ca

[1] Carleton University
Ottawa, Canada
[2] Canada CIFAR AI Chair
Amii, Canada

Abstract

Classical machine learners are designed only to tackle one task and suffer *catastrophic forgetting* as new tasks or classes emerge. To address this shortcoming, continual machine learners are elaborated to commendably learn a stream of tasks with domain and class shifts among different tasks. In this paper, we propose a general feature-propagation based contrastive continual learning method for *image recognition* in an online fashion which is capable of handling multiple continual learning scenarios. Specifically, we align the current and previous representation spaces by means of feature propagation and contrastive representation learning to bridge the domain shifts among distinct tasks. To further mitigate the class-wise shifts of the feature representation, a supervised contrastive loss is exploited to make the image embeddings of the same class closer than those of different classes. The extensive experimental results demonstrate the outstanding performance of the proposed method in multiple image classification tasks (MNIST, CIFAR-10/100 and Tiny ImageNet) compared to other cutting-edge continual learning methods.

1 Introduction

In the real world, the environment is not set in stone. The machine learner is desired to favorably respond to the changing environment like humans, by acquiring the new knowledge rapidly without forgetting what has learned in the past. Towards this end, continual learning (CL) [35, 46] came into being and has attracted a surge of interest in computer vision tasks such as image recognition or segmentation, video recognition, etc. In this work, we focus on the continual learning applied to *image recognition*. Specifically, the model is presented with a stream of non-i.i.d. image data and can only learn one task at a time without accessing past task data. Therefore, the major challenge is the issue of *catastrophic forgetting* [17, 29] for previously learned images. To tackle such problem, a plethora of CL methods were proposed, as well as a variety of evaluation protocols and a systematic categorization of CL scenarios.

As categorized by [19, 50], continual learning has three distinct scenarios - task incremental learning, domain incremental learning and class incremental learning - with increasing difficulty. The *task incremental learning* [26, 54] is the easiest CL setting where task identifiers are provided at test time and a multi-head network is applied. Explicitly, the unique

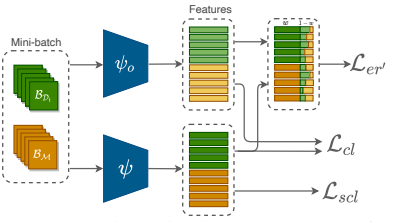


Figure 1: An overview of the proposed method CCL-FP+. ψ_o is the feature extractor whose state is after learning task $t-1$ but before task t, and remains fixed during the training. ψ is the feature extractor that is being optimized. The minibatch for each training iteration in generated by concatenating a minibatch $\mathcal{B}_{\mathcal{D}_t}$ from the current task data $\mathcal{D}_t$ with another minibatch $\mathcal{B}_{\mathcal{M}}$ from the memory set $\mathcal{M}$.

feature extractor is shared across different tasks but the classifiers are task-specific. By contrast, the *domain incremental learning* [10, 27] makes use of a single-head network thereby the task identifiers are not required at the test stage. The *class incremental learning* [33] as a more realistic but challenging CL scenario learns new tasks continually with a single-head network but the units of the classifier increases with the advent of new classes.

Distinct from the majority of prior continual learning work which just tackles certain continual learning scenarios, in this paper we propose a general feature propagation based contrastive continual learning method to manage all of them. Most existing continual learning methods focus the efforts on retaining either model parameters [22, 39] or functions [8, 31] whereas we lay our stress on the representation space. To protect the feature space from drastically changing by aligning the current and past embedding spaces during learning the new task, we desire the model to preserve considerable past knowledge without losing the ability to adapt to new tasks. Concretely, the method consists of three components. First, the current image embeddings are re-represented by being integrated with the previous corresponding ones via feature propagation, while experience replay [32, 36] is implemented on such propagated image embeddings. Additionally, a contrastive loss is deployed to explicitly enforce the current embeddings to approach the previous ones. To further eliminate the domain and classes shifts among distinct tasks, we also use a supervised contrastive loss to discriminatively make the image embeddings of the same class closer than those from different classes. Notably, we adopt the online continual learning setting where the model can only experience the data stream once [9, 27]. Extensive experiments are conducted on six image classification tasks: Split MNIST, Permuted MNIST, Rotated MNIST, Split CIFAR-10, Split CIFAR-100 and Split Tiny ImageNet. The results demonstrate the superiority of our proposed method over a number of the state-of-the-art continual learning competitors. An overview of the proposed method is illustrated in Figure 1.

2 Related Work

Continual learning Generally, the continual learning methods can be grouped into three categories. First, *regularization-based methods* alleviate the issue of forgetting by either regularizing the parameter changes in terms of the importance of parameters for old tasks [1, 8, 14, 22, 39, 52], or aligning the current and previous function space by means of knowledge distillation [26, 33] or KL divergence [3, 7, 31]. Second, *rehearsal-based methods* reserve a memory set of past examples which can be directly enrolled in training as training

data [2, 6, 11, 31, 34, 51] known as experience replay [32, 36] or indirectly used as gradient constraints [9, 27, 45] or functional regularizations [33]. To update the memory set, [2] selects the memory data to maximize the diversity of memory samples in terms of parameters gradients. [4] constructs the buffer via cardinality-constrained bilevel optimization. [41] contributes an Adversarial Shapley value scoring method to score memory samples. Instead of directly storing past examples, [42, 47] train a generative model (GANs or VAEs) on previous tasks and rehearsal pseudo-examples during learning new tasks, which is however hard to perform well on complex datasets. [30] utilizes the conditional generative adversarial networks (cGANs) to synthesize samples of previously seen tasks where the generator is endowed with a sparse mask for the its weights. Additionally, rather than storing the past examples, [15, 38, 53] reserves a set of gradient directions to alleviate the interference with past tasks. Lastly, *model-based methods* [20, 25, 37, 40, 48, 52] modify the network dynamically by fixing some units for previous tasks, adding extra units for new tasks or merging a couple of units for similar tasks.

Contrastive Continual learning An increasing number of continual learning approaches integrating with contrastive learning have been emerging in recent years. [16] hypothesizes that the self-supervised pre-training could yield better representations for continual learning and tests this hypothesis using two contrastive algorithms MoCo-V2 [13] and SwAV [5] and one non-contrastive algorithm Barlow Twins [21]. [28] explicitly encourages samples of the same class to be close together and those from different classes to be far apart by use of the supervised contrastive loss. [6] claims the contrastively learned representations are more robust and proposes a rehearsal-based continual learning algorithm based on that. [44] utilizes the contrastive representations for continual domain adaptation. In this paper, we propose a novel CL method to exploit rehearsal memories while regularizing feature representations in a contrastive manner.

3 Continual Learning Setup

A continual learner experiences a stream of data triplets $(\mathbf{x}_i, y_i, t_i)$ over time where $t_i \in \{1, \cdots, T\}$ is the task identifier. For each task t, the i.i.d. examples $(\mathbf{x}, y, t)$ are drawn from a distribution $\mathcal{D}_t$ whereas the whole data stream is not independently and identically distributed (non-i.i.d.); i.e. there are domain and class shifts among different tasks. The continual learner is trained on one task at a time and not able to revisit the data of learned tasks aside from a few pieces of data. The goal of the continual learner is to generally perform well on all learned tasks, namely, rapidly adapting to new tasks and meanwhile preserving the previously learned knowledge to a great extent. The particular challenge faced by continual learning is the problem of *catastrophic forgetting* that means learning new tasks may hurt the performance on past tasks due to the non-i.i.d stream of data. In addition to this challenge, the continual learner is expected to fast acquire and adapt to the new tasks, hence a more compelling setting named *online continual learning* is considered in some prior continual learning works [9, 10, 27, 28] and will be adopted in this paper, where the continual learner can only experience the data stream once. Specifically, the model receives a small batch of data at a time and is only trained on the batch once. For rehearsal-based CL methods, a small memory set $\mathcal{M}$ storing a few past examples is reserved and can be revisited multiple times along the learning of new tasks. At the time of task t, by incorporating the memory set $\mathcal{M}$ into the current task data $\mathcal{D}_t$ as the training set which is also known as *experience replay*

[32, 36], the objective of the continual learner g is to minimize the following loss:

$$\mathcal{L}_{er} = \mathbb{E}_{(\mathbf{x},y)\sim\mathcal{D}_t\cup\mathcal{M}} l(g(\mathbf{x}),y) \tag{1}$$

where l is generally the cross-entropy loss function. It is worth noting that the experience replay is a very simple and strong CL baseline [10] and outperforms significantly in terms of either performance or efficiency the rehearsal-based methods that utilize the memory set indirectly during training such as [27, 33]. Therefore, our proposed method is developed on the basis of experience replay.

4 Methodology

Instead of enforcing the model outputs to be close to the previous ones in the output space by means of knowledge distillation loss in some prior continual learning works [26, 31, 33], The key idea of the proposed method is to preserve the representation space from drastic changes by means of feature propagation and contrastive loss. Specifically, the current representation space first absorbs some past knowledge by feature propagation from previous feature space. The fusion of the two terms is motivated by the exponential moving average which can move forward without forgetting the past. A contrastive regularization loss is then employed to reinforce such effect by encouraging the current example embeddings to approach the previous ones. As a complement, a supervised contrastive loss is leveraged to push the examples of the same class to cluster tightly in the representation space whereas the examples' embeddings from different classes are driven to be far apart. We will give an exposition in following sections.

4.1 Experience Replay with Feature Propagation

Suppose the model consists of two components: the feature extractor ψ and the classifier f. It is noted that for class incremental learning and domain incremental learning, there is an only classifier for all tasks whereas for task incremental learning, there is one classifier for each task. The feature extractor remains unique all the time. When the task t arrives, we have two labelled datasets on hand: the current task data $\mathcal{D}_t$ and a small memory set $\mathcal{M}$ of a few past examples. Before retraining the model on new data, we first copy the feature extractor ψ to ψ_o and keep ψ_o fixed during the ensuing training. Afterwards, a feature propagation procedure is carried out for the training data $\mathcal{D}_t \cup \mathcal{M}$ in the representation space. Concretely, the example embeddings derived from ψ are amended by fusing a weighted sum of all example embeddings come from ψ_o. We denote the modified example embedding for $\mathbf{x}_i \in \mathcal{D}_t \cup \mathcal{M}$ by $\tilde{\psi}(\mathbf{x}_i)$ defined as follows,

$$\tilde{\psi}(\mathbf{x}_i) = (1-w)\cdot\psi(\mathbf{x}_i) + w\cdot\sum_{\mathbf{x}_j\in\mathcal{D}_t\cup\mathcal{M}} A_{ij}\psi_o(\mathbf{x}_j) \tag{2}$$

where $w \in (0,1)$ is a trade-off parameter to balance the current and previous embedding spaces. The propagation weight A_{ij} for examples $\mathbf{x}_i$ and $\mathbf{x}_j$ is set as:

$$A_{ij} = \frac{\exp(-d(\psi(\mathbf{x}_i),\psi_o(\mathbf{x}_j))\cdot\eta)}{\sum_{\mathbf{x}_{j'}\in\mathcal{D}_t\cup\mathcal{M}}\exp(-d(\psi(\mathbf{x}_i),\psi_o(\mathbf{x}_{j'}))\cdot\eta)} \tag{3}$$

where $d(\cdot,\cdot)$ is the Euclidean distance and η is a temperature parameter. In consequence, the amended experience replay loss is

$$\mathcal{L}_{er'} = \mathbb{E}_{(\mathbf{x},y)\sim\mathcal{D}_t\cup\mathcal{M}} l(f(\tilde{\psi}(\mathbf{x})),y) \tag{4}$$

where l is the cross-entropy loss function. In the practical training implementation, the $\mathbf{x}_j$ in Eq. 2 is sampled in a mini-batch level. At test time, we only use ψ and f to make the inference simple and efficient. By being trained on the basis of the fused representation space, the model is expected to absorb certain propagated global information from the previous embedding space so as to avoid the significant forgetting issue.

4.2 Contrastive Representation Rehearsal

The contrastive learning shows the excellent capacity for representation learning by pushing the 'similar' examples to be close together and 'dissimilar' examples to be far apart [12, 18, 50]. Inspired by such idea but distinct from some existing work such as [50] where the 'similar' examples are formed from the representation space of the previous training iteration, our model makes use of the example embeddings derived from the previous feature extractor ψ_o. Explicitly, we enforce the example embeddings to stay near the previous corresponding ones by a contrastive loss, so that to realize the aspiration of protecting the representation space from dramatically changing after being retrained on new tasks. The proposed contrastive loss is defined as follows,

$$\mathcal{L}_{cl} = -\mathbb{E}_{\mathbf{x}\sim\mathcal{D}_t\cup\mathcal{M}} \log \frac{\exp(-d(\psi(\mathbf{x}),\psi_o(\mathbf{x}))\cdot\tau)}{\sum_{\mathbf{x}_j\in\mathcal{D}_t\cup\mathcal{M}}\exp(-d(\psi(\mathbf{x}),\psi_o(\mathbf{x}_j))\cdot\tau)} \tag{5}$$

where $d(\cdot,\cdot)$ is the Euclidean distance and τ is a temperature parameter. It is noted that instead of merely applying the contrastive learning to memory data $\mathcal{M}$ which the model has previously seen and learned, the whole training data $\mathcal{D}_t\cup\mathcal{M}$ is deployed in this loss to enhance the contrastive effect which we empirically found performs better.

Up to this point, we derive our proposed method which we named as *contrastive continual learning with feature propagation (CCL-FP)*. The overall objective during task t is

$$\mathcal{L}_{ccl-fp} = \mathcal{L}_{er'} + \alpha\mathcal{L}_{cl} \tag{6}$$

where $\alpha\in(0,1)$ is a trade-off parameter. By minimizing the above objective, the representation space is expected to stay steady throughout the training of all tasks, which is intuitively adequate to cope with the task incremental learning where the model is equipped with task specific classifiers or the class incremental learning in the case that the classes for different tasks are disjoint. Nevertheless, for the domain incremental learning where all tasks share an only classifier, it may be reliable to a limited degree.

4.3 Supervised Contrastive Replay

In the domain incremental learning, the classes for each task are identical whereas the domain shifts among different tasks are relatively substantial. In this case, merely retaining the representation space may not be a panacea, especially as all tasks share a unique classifier. Out of such concern, we resort to supervised contrastive learning as a complement to our overall loss function. Since the labels are on hand in our continual learning setup, the 'similar'

examples here can be formed by the data of the same class from $\mathcal{D}_t \cup \mathcal{M}$. Hence, a supervised contrastive loss is defined as follows,

$$\mathcal{L}_{scl} = -\mathbb{E}_{\mathbf{x}_i \sim \mathcal{D}_t \cup \mathcal{M}} \mathbb{E}_{\mathbf{x}_k \sim \mathcal{S}_i} \log \frac{\exp(-d(\psi(\mathbf{x}_i), \psi(\mathbf{x}_k)) \cdot \tau)}{\sum_{\mathbf{x}_j \in (\mathcal{D}_t \cup \mathcal{M}) \setminus \mathbf{x}_i} \exp(-d(\psi(\mathbf{x}_i), \psi(\mathbf{x}_j)) \cdot \tau)} \tag{7}$$

where $d(\cdot,\cdot)$ is the Euclidean distance and τ is a temperature parameter. $\mathcal{S}_i$ is a set consisting of the examples from $\mathcal{D}_t \cup \mathcal{M}$ of the same class as $\mathbf{x}_i$ but excluding $\mathbf{x}_i$ itself. The model thereby learns to make example embeddings of the same class which may come from both $\mathcal{D}_t$ and $\mathcal{M}$ close together and those of different classes distinct, so that the domain shifts of the same class among different tasks can be eliminated if there is any.

By integrating the supervised contrastive loss into the overall objective, we obtain our intensified model named by *CCL-FP+*:

$$\mathcal{L}_{ccl-fp+} = \mathcal{L}_{er'} + \alpha \mathcal{L}_{cl} + \beta \mathcal{L}_{scl} \tag{8}$$

where α and β are trade-off parameters in the range of $(0,1)$. We solve it by using a batch-wise gradient descent algorithm. The training algorithm for CCL-FP+ is provided in the supplementary file, while the memory set $\mathcal{M}$ is updated along the training by reservoir sampling [49].

5 Experiments

In this section, we compare our models CCL-FP and CCL-FP+ with several state-of-the-art continual learning methods in an online manner on a variety of image datasets. We start by reviewing the CL benchmarks and baselines and then report our experimental details as well as the analysis of experimental results and ablation study. Below we present our experimental setups, and report the results and analysis of our comparison experiments as well as ablation studies.

5.1 Experimental Setting

Datasets We conducted extensive experiments on six commonly used image datasets in the continual learning literature, of which four are applied in the class and task incremental learning and the other two are in the domain incremental learning. **Split MNIST** [54] is constructed by splitting the source MNIST dataset [24] into 5 disjoint binary-class subsets in sequence (e.g. 0/1, 2/3, 4/5, 6/7, 8/9), of which each is considered as a separate task. **Permuted MNIST** [22] is a variant of the MNIST dataset, where each task applies a certain random pixel-level permutation to all the original images. **Rotated MNIST** [27] is another variant of MNIST by rotating the original images with a certain random angel between 0 and 180 degrees in each task. For Permuted MNIST and Rotated MNIST, we consider 20 tasks and each task has 1000 images of 10 classes randomly sampled from the entire dataset. **Split CIFAR-10** [54] and **Split CIFAR-100** [53] are constructed by splitting the CIFAR-10 and CIFAR-100 datasets [23] into 5 disjoint binary-class subsets and 20 disjoint 5-class subsets, respectively. Similarly, **Split Tiny ImageNet** is a sequential split of the original Tiny ImageNet dataset [43] with 10 tasks, each of which introduces 20 classes.

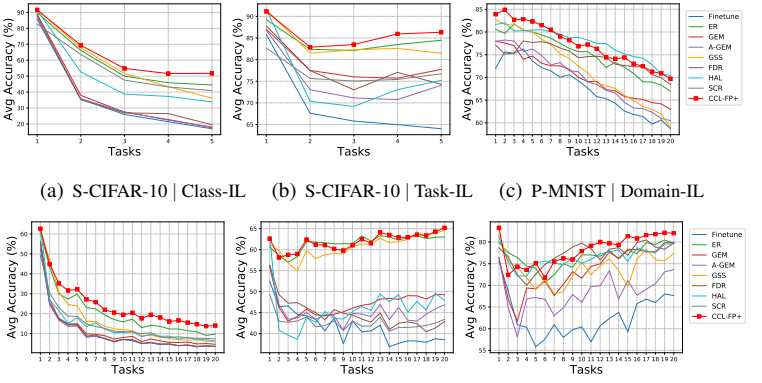
Model	S-MNIST		S-CIFAR-10		P-MNIST
	Class-IL	Task-IL	Class-IL	Task-IL	Domain-IL
Joint	**95.59 ± 0.31**	**99.33 ± 0.17**	**58.89 ± 3.26**	**87.58 ± 1.85**	**77.65 ± 1.09**
Finetune	19.62 ± 0.12	95.25 ± 1.66	17.00 ± 1.20	64.02 ± 3.53	58.68 ± 0.46
ER-Reservoir [10]	76.43 ± 3.08	98.77 ± 0.14	44.45 ± 3.69	84.42 ± 1.15	66.95 ± 1.40
GEM [27]	80.79 ± 1.47	97.68 ± 0.32	18.66 ± 0.91	77.74 ± 2.60	62.96 ± 1.14
A-GEM [9]	45.69 ± 3.77	98.66 ± 0.16	18.13 ± 0.27	74.07 ± 0.76	60.48 ± 2.04
GSS [2]	71.19 ± 1.25	98.45 ± 0.51	36.19 ± 4.38	81.47 ± 1.74	58.91 ± 0.96
FDR	81.03 ± 2.23	98.66 ± 0.52	19.51 ± 1.04	74.29 ± 3.49	68.41 ± 2.72
HAL [11]	79.15 ± 2.03	98.81 ± 0.18	33.86 ± 1.73	75.19 ± 2.57	**70.83 ± 1.86**
SCR [28]	–	–	40.91 ± 1.07	76.72 ± 2.28	–
CCL-FP (ours)	88.67 ± 0.97	**99.15 ± 0.37**	50.11 ± 3.69	85.44 ± 2.03	66.91 ± 0.95
CCL-FP+ (ours)	**89.16** ± 1.14	99.14 ± 0.05	**51.74 ± 2.41**	**86.33 ± 1.47**	69.22 ± 1.07

Model	S-CIFAR-100		S-Tiny-ImageNet		R-MNIST
	Class-IL	Task-IL	Class-IL	Task-IL	Domain-IL
Joint	**19.60 ± 2.14**	**69.80 ± 2.17**	**14.21 ± 0.66**	**43.89 ± 0.88**	**84.12 ± 0.61**
Finetune	3.58 ± 0.13	39.55 ± 4.42	4.77 ± 0.23	26.93 ± 1.59	67.64 ± 2.17
ER-Reservoir [10]	9.74 ± 0.98	63.05 ± 0.82	7.21 ± 0.29	36.75 ± 0.79	79.77 ± 0.86
GEM [27]	4.69 ± 0.41	49.29 ± 0.73	6.76 ± 0.45	29.05 ± 0.74	79.85 ± 2.17
A-GEM [9]	3.67 ± 0.10	46.88 ± 1.81	5.43 ± 0.11	29.67 ± 0.91	73.64 ± 3.68
GSS [2]	6.15 ± 0.49	64.58 ± 2.29	6.41 ± 0.42	39.71 ± 0.72	77.37 ± 3.55
FDR	3.65 ± 0.10	42.87 ± 2.62	4.83 ± 0.43	26.97 ± 2.69	79.75 ± 3.12
HAL [11]	6.31 ± 0.71	47.88 ± 2.76	3.85 ± 0.32	21.70 ± 1.12	78.65 ± 1.57
SCR [28]	7.38 ± 0.61	43.41 ± 1.31	2.61 ± 0.28	13.03 ± 0.85	–
CCL-FP (ours)	13.64 ± 1.04	**65.19 ± 0.65**	**10.52 ± 0.28**	39.44 ± 0.48	80.68 ± 1.74
CCL-FP+ (ours)	**14.05 ± 0.85**	65.19 ± 1.88	10.13 ± 0.44	**39.99 ± 0.59**	**82.06 ± 1.29**

Table 1: The average accuracy ± standard deviation (%) by the end of training for baselines and our models across 5 runs with different random seeds. The results for joint training, i.e. the upper bound, and the best accuracies for CL models on each benchmark are marked in bold. It is noted that '–' indicates experiments are unable to run because of compatibility issues (e.g. SCR on MNIST dataset).

Baselines We compared the proposed methods CCL-FP and CCL-FP+ with several rehearsal-based CL competitors in an online manner, including **ER-Reservoir** [10], **GEM** [27], **A-GEM** [9], **GSS** [2], **FDR** [3], **HAL** [11] and **SCR** [28], as well as the upper bound and lower bound. **Joint** trains the model with access to the data of all tasks at the same time, which serves as an upper bound in terms of performance. **Finetune** is a lower bound for CL baselines which simply trains the model using new task data without any effort to overcome forgetting of past tasks. Both Joint and Finetune train the model with a single pass over the data.

Implementation All baselines and our models adopt the same backbone architectures. We use a fully-connect network with two hidden layers of 100 RELU units for the variants of the MNIST dataset following [27, 34] and a ResNet18 for Split CIFAR-10, Split CIFAR-100 and Split Tiny ImageNet datasets following [31, 33]. All experiments are implemented under the single epoch training with minibatch size of 10. For temperature parameters η and τ, we set $\eta, \tau = 0.1$ for datasets in the class and task incremental learning, i.e. Split MNIST, Split CIFAR-10, Split CIFAR-100 and Split Tiny ImageNet, and $\eta = 0.1, \tau = 1$ for datasets in domain incremental learning, i.e. Permuted MNIST and Rotated MNIST. The hyperparameters w are selected from $\{0.1, 0.3, 0.5\}$ and α, β are from $\{0.01, 0.1, 0.5, 1\}$. The buffer size $|\mathcal{M}|$ is set to be 200 across all experiments for all CL methods. It is noted that the buffer of size 200 is fairly small and makes the rehearsal-based continual learning difficult especially on Split CIFAR-100 and Split Tiny ImageNet datasets with 200 classes in total.



(a) S-CIFAR-10 | Class-IL (b) S-CIFAR-10 | Task-IL (c) P-MNIST | Domain-IL

(d) S-CIFAR-100 | Class-IL (e) S-CIFAR-100 | Task-IL (f) R-MNIST | Domain-IL

Figure 2: The evolution of average accuracy on test data of all seen tasks as new tasks are learned. All results are obtained across 5 runs with different random seeds.

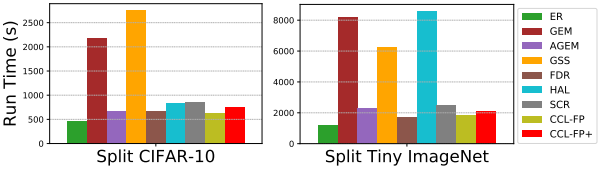


Figure 3: Run time (training + inference) for baselines and our model on Split CIFAR-10 and Split Tiny-ImageNet datasets.

For fair comparison, we use the same memory batch size for our proposed models and SCR for representation learning. The impact of the memory batch size will be investigated in the ablation study and provided in the supplementary material. Besides, we consider the average accuracy to evaluate the overall performance of CL models and the definition can be found in the supplementary material.

5.2 Experimental Results

The overall average accuracy of baselines and proposed models on all benchmarks is reported in Table 1 and the evolution curves of average accuracy with respect to the number of tasks on selected benchmarks are shown in Figure 2. It is worth noting that our models achieve the best average accuracy on all datasets under different settings, except Permuted MNIST in which HAL behaves better, whereas our models are more computational efficient as shown in Figure 3. In the class incremental learning setting, the model CCL-FP outperforms all compared methods on corresponding benchmarks by great margins and CCL-FP+ further obtains slight improvement except on S-Tiny-ImageNet. With a relatively small buffer of size 200, most of rehearsal-based methods are barely satisfactory especially on the complicated datasets such as S-CIFAR-100 and S-Tiny-ImageNet. The task incremental learning is generally considered as the easiest continual learning scenario, where all baselines perform fairly well on S-MNIST because of its simplicity. On three other datasets, some rehearsal-based methods underperform in the setup of the small buffer size in the online setting, whereas our models still gain outstanding performance for all corresponding benchmarks. It is noted that there is

		CIFAR-10		CIFAR-100		R-MNIST
Buffer	Model	Class-IL	Task-IL	Class-IL	Task-IL	Domain-IL
0.2k	ER-Res.	44.45	84.42	9.74	63.05	79.77
	HAL	33.86	75.19	6.31	47.88	78.65
	SCR	40.91	76.72	7.38	40.41	–
	CCL-FP+	51.74	86.33	14.05	65.19	82.06
0.5k	ER-Res.	56.64	87.02	14.55	68.87	82.23
	HAL	37.48	75.87	8.09	50.04	82.30
	SCR	45.19	79.70	9.97	50.58	–
	CCL-FP+	57.03	87.37	19.44	69.03	83.10
1k	ER-Res.	57.19	88.35	21.26	73.30	84.48
	HAL	45.13	80.87	11.02	56.59	84.55
	SCR	46.05	81.66	12.06	54.95	–
	CCL-FP+	60.29	88.87	23.99	74.27	84.95
5k	ER-Res.	61.71	90.94	24.27	79.71	85.87
	HAL	48.83	83.35	15.31	65.40	85.46
	SCR	46.12	82.13	14.71	60.39	–
	CCL-FP+	63.84	91.03	27.38	80.46	86.96

Table 2: The average accuracy across 5 runs with different random seeds over a range of buffer size of ER, HAL, SCR and CCL-FP+ on selected datasets.

			S-MNIST		S-CIFAR-10		R-MNIST
$w\neq 0$	$\alpha\neq 0$	$\beta\neq 0$	Class-IL	Task-IL	Class-IL	Task-IL	Domain-IL
			76.43	98.77	44.45	84.42	79.77
✓			83.48	98.72	50.45	85.23	79.38
	✓		78.04	99.13	44.82	84.63	79.89
		✓	77.37	98.67	46.41	85.55	81.01
✓	✓		88.67	99.15	50.11	85.44	80.68
✓		✓	82.95	98.73	50.36	84.73	81.78
	✓	✓	78.31	98.89	47.52	85.91	81.79
✓	✓	✓	89.16	99.14	51.74	86.33	82.06

Table 3: The ablation study on selected datasets. All results are the average accuracy across 5 runs with different random seeds. The results in the first row is for ER-Reservoir.

no substantially difference between CCL-FP and CCL-FP+, suggesting that the supervised contrastive loss is removable in this setting. For domain incremental learning we have two benchmarks where CCL-FP+ is consistently better than CCL-FP, confirming the importance of supervised contrastive loss for this setting.

Figure 3 gives the run time (training + inference) of our methods and baselines on S-CIFAR-10 and S-Tiny-ImageNet datasets and we can observe that our model CCL-FP+ is consistently computational efficient over different benchmarks. Additionally, we further study the impact of buffer size in terms of average accuracy by evaluating ER, HAL, SCR and CCL-FP+ on S-CIFAR-10, S-CIFAR-100 and R-MNIST datasets with a range of different buffer size. The results are reported in Table 2, in which we can see that the average accuracy improves with the increase of the buffer size and our model consistently outperforms ER, HAL and SCR on three selected datasets for a wide range of buffer sizes. Besides, more experimental results are provided in the supplementary materials.

5.3 Ablation Study

We conducted an ablation study to explore the impact of each component in our proposed model on S-MNIST and S-CIFAR-10 and R-MNIST datasets with default buffer size of 200. The results are reported in Table 3. Our models have been empirically demonstrated to be superior to the baseline ER with reservoir buffer on all benchmarks in Table 1 which confirms the effectiveness of proposed components in our models. Here we will investigate and analyze the specific effect of each component in different continual learning scenarios:

$w \neq 0$ indicates the inclusion of the feature propagation component, $\alpha \neq 0$ indicates the inclusion of the contrastive representation rehearsal component $\mathcal{L}_{cl}$, and $\beta \neq 0$ indicates the inclusion of the supervised contrastive replay component $\mathcal{L}_{scl}$.

In the class incremental learning, we observe that the feature propagation ($w \neq 0$) yields remarkable performance gains compared to the other two components, which is approximately 7% gains on S-MNIST and 6% gains on S-CIFAR-10. The other two components contribute in total about 5.5% gains on S-MNIST and 1.3% gains on S-CIFAR-10. In the task incremental learning, there is no prominent gains from a certain component. By considering the results in Table 1 we conclude that the supervised contrastive loss ($\beta \neq 0$) achieves marginal performance gains so it is removable in this setting. In the domain incremental learning, the supervised contrastive loss appears to be particularly important compared to two other components, which produce about 1.3% performance gains on R-MNIST. In addition, as shown in Table 1, embracing the supervised contrastive loss into the model can obtain about 2.3% performance gains on the P-MNIST benchmark.

6 Conclusion

In this paper, we proposes an effective and also computational efficient online continual learning method applied to image recognition task. Instead of pure experience replay training, we first re-represent the image embeddings by incorporating the information of previous representation space via feature propagation and the model is then trained on the modified image embeddings. Moreover, to largely preserve the representation space from dramatical changes when experiencing new tasks, we encourage current image embeddings to approach previous corresponding ones by a contrastive loss whereby the model is expected to keep a competent memory of what has learned in the past and overcome the problem of catastrophic forgetting after being exposed to new tasks. Furthermore, a supervised contrastive loss is leveraged in the model training to explicitly encourage the images of the same class to cluster closely in representation space and meanwhile push image embeddings from different classes to be far apart. The extensive experiments demonstrated the superiority of our models in a variety of image classification tasks.

References

[1] Rahaf Aljundi, Francesca Babiloni, Mohamed Elhoseiny, Marcus Rohrbach, and Tinne Tuytelaars. Memory aware synapses: Learning what (not) to forget. In *European Conference on Computer Vision (ECCV)*, 2018.

[2] Rahaf Aljundi, Min Lin, Baptiste Goujaud, and Yoshua Bengio. Gradient based sample selection for online continual learning. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2019.

[3] Ari S. Benjamin, D. Rolnick, and K. Körding. Measuring and regularizing networks in function space. In *International Conference on Learning Representations (ICLR)*, 2019.

[4] Zalán Borsos, Mojmír Mutný, and Andreas Krause. Coresets via bilevel optimization for continual learning and streaming. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2020.

[5] Mathilde Caron, Ishan Misra, Julien Mairal, Priya Goyal, Piotr Bojanowski, and Armand Joulin. Unsupervised learning of visual features by contrasting cluster assignments. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2020.

[6] Hyuntak Cha, Jaeho Lee, and Jinwoo Shin. Co2L: Contrastive continual learning. In *International Conference on Computer Vision (ICCV)*, 2021.

[7] Sungmin Cha, Hsiang Hsu, Taebaek Hwang, Flavio Calmon, and Taesup Moon. {CPR}: Classifier-projection regularization for continual learning. In *International Conference on Learning Representations*, 2021.

[8] Arslan Chaudhry, Puneet K Dokania, Thalaiyasingam Ajanthan, and Philip HS Torr. Riemannian walk for incremental learning: Understanding forgetting and intransigence. In *European Conference on Computer Vision (ECCV)*, 2018.

[9] Arslan Chaudhry, Marc'Aurelio Ranzato, Marcus Rohrbach, and Mohamed Elhoseiny. Efficient lifelong learning with a-gem. In *International Conference on Learning Representations (ICLR)*, 2019.

[10] Arslan Chaudhry, Marcus Rohrbach, Mohamed Elhoseiny, Thalaiyasingam Ajanthan, Puneet K Dokania, Philip HS Torr, and Marc'Aurelio Ranzato. Continual learning with tiny episodic memories. In *International Conference on Machine Learning (ICML)*, 2019.

[11] Arslan Chaudhry, Albert Gordo, P. Dokania, Philip H. S. Torr, and David Lopez-Paz. Using hindsight to anchor past knowledge in continual learning. In *The Thirty-Fifth AAAI Conference on Artificial Intelligence (AAAI)*, 2021.

[12] Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. A simple framework for contrastive learning of visual representations. In *Proceedings of the 37th International Conference on Machine Learning (ICML)*, 2020.

[13] Ting Chen, Simon Kornblith, Kevin Swersky, Mohammad Norouzi, and Geoffrey Hinton. Big self-supervised models are strong semi-supervised learners. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2020.

[14] Prithviraj Dhar, Rajat Vikram Singh, Kuan-Chuan Peng, Ziyan Wu, and Rama Chellappa. Learning without memorizing. *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019.

[15] Mehrdad Farajtabar, Navid Azizan, Alex Mott, and Ang Li. Orthogonal gradient descent for continual learning. In *Proceedings of the Twenty Third International Conference on Artificial Intelligence and Statistics (AISTATS)*, 2020.

[16] Jhair Gallardo, Tyler L. Hayes, and Christopher Kanan. Self-supervised training enhances online continual learning. In *The 32nd British Machine Vision Conference (BMVC)*, 2021.

[17] Ian J. Goodfellow, Mehdi Mirza, Da Xiao, Aaron Courville, and Yoshua Bengio. An empirical investigation of catastrophic forgeting in gradientbased neural networks. In *International Conference on Learning Representations (ICLR)*, 2014.

[18] Kaiming He, Haoqi Fan, Yuxin Wu, Saining Xie, and Ross Girshick. Momentum contrast for unsupervised visual representation learning. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020.

[19] Yen-Chang Hsu, Yen-Cheng Liu, Anita Ramasamy, and Zsolt Kira. Re-evaluating continual learning scenarios: A categorization and case for strong baselines. In *NeurIPS Continual learning Workshop*, 2018.

[20] Ghassen Jerfel, Erin Grant, Thomas L. Griffiths, and Katherine Heller. Reconciling meta-learning and continual learning with online mixtures of tasks. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2019.

[21] Li Jing Jure Zbontar, Ishan Misra, Yann LeCun, and Stephane Deny. Barlow twins: Self-supervised learning via redundancy reduction. In *Proceedings of the 38th International Conference on Machine Learning (ICML)*, 2021.

[22] James Kirkpatrick, Razvan Pascanu, Neil Rabinowitz, Joel Veness, Guillaume Desjardins, Andrei A. Rusu, Kieran Milan, John Quan, Tiago Ramalho, Agnieszka Grabska-Barwinska, Demis Hassabis, Claudia Clopath, Dharshan Kumaran, and Raia Hadsell. Overcoming catastrophic forgetting in neural networks. *Proceedings of the National Academy of Sciences (PNAS)*, 114(13):3521–3526, 2017.

[23] Alex Krizhevsky. Learning multiple layers of features from tiny images. 2009.

[24] Yann LeCun, Corinna Cortes, and Christopher J.C. Burges. The mnist database of handwritten digits. 1998. URL http://yann.lecun.com/exdb/mnist/.

[25] Soochan Lee, Junsoo Ha, Dongsu Zhang, and Gunhee Kim. A neural dirichlet process mixture model for task-free continual learning. In *International Conference on Learning Representations (ICLR)*, 2020.

[26] Zhizhong Li and Derek Hoiem. Learning without forgetting. In *European Conference on Computer Vision (ECCV)*, 2016.

[27] David Lopez-Paz and Marc'Aurelio Ranzato. Gradient episodic memory for continual learning. In *Advances in Neural Information Processing Systems (NIPS)*, 2017.

[28] 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 *CVPR workshop*, 2021.

[29] Michael McCloskey and Neal J. Cohen. Catastrophic interference in connectionist networks: The sequential learning problem. volume 24 of *Psychology of Learning and Motivation*, pages 109 – 165. Academic Press, 1989.

[30] Oleksiy Ostapenko, Mihai Marian Puscas, Tassilo Klein, Patrick Jähnichen, and Moin Nabi. Learning to remember: A synaptic plasticity driven framework for continual learning. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019.

[31] Buzzega Pietro, Boschini Matteo, Porrello Angelo, Abati Davide, and Calderara Simone. Dark experience for general continual learning: a strong, simple baseline. *Advances in Neural Information Processing Systems (NeurIPS)*, 2020.

[32] R. Ratcliff. Connectionist models of recognition memory: constraints imposed by learning and forgetting functions. *Psychological review*, 97 2:285–308, 1990.

[33] S. Rebuffi, A. Kolesnikov, G. Sperl, and C. H. Lampert. icarl: Incremental classifier and representation learning. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017.

[34] 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 *International Conference on Learning Representations (ICLR)*, 2019.

[35] Mark Bishop Ring. *Continual Learning in Reinforcement Environments*. PhD thesis, 1994.

[36] ANTHONY Robins. Catastrophic forgetting, rehearsal and pseudorehearsal. *Connection Science*, 7(2):123–146, 1995.

[37] Andrei A. Rusu, Neil C. Rabinowitz, Guillaume Desjardins, Hubert Soyer, J. Kirkpatrick, K. Kavukcuoglu, Razvan Pascanu, and R. Hadsell. Progressive neural networks. *ArXiv*, abs/1606.04671, 2016.

[38] Gobinda Saha, Isha Garg, and Kaushik Roy. Gradient projection memory for continual learning. In *International Conference on Learning Representations (ICLR)*, 2021.

[39] Jonathan Schwarz, Wojciech Czarnecki, Jelena Luketina, Agnieszka Grabska-Barwinska, Yee Whye Teh, Razvan Pascanu, and Raia Hadsell. Progress & compress: A scalable framework for continual learning. In *International Conference on Machine Learning (ICML)*, 2018.

[40] Joan Serra, Didac Suris, Marius Miron, and Alexandros Karatzoglou. Overcoming catastrophic forgetting with hard attention to the task. In *Proceedings of the 35th International Conference on Machine Learning (ICML)*, 2018.

[41] 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 (AAAI)*, 2021.

[42] Hanul Shin, Jung Kwon Lee, Jaehong Kim, and Jiwon Kim. Continual learning with deep generative replay. In *Advances in Neural Information Processing Systems (NIPS)*, 2017.

[43] Stanford. Tiny imagenet challenge (cs231n). 2015. URL http://tiny-imagenet.herokuapp.com/.

[44] Peng Su, Shixiang Tang, Peng Gao, Di Qiu, Ni Zhao, and Xiaogang Wang. Gradient regularized contrastive learning for continual domain adaptation. In *The Thirty-Fifth AAAI Conference on Artificial Intelligence (AAAI)*, 2021.

[45] Shixiang Tang, Dapeng Chen, Jinguo Zhu, Shijie Yu, and Wanli Ouyang. Layerwise optimization by gradient decomposition for continual learning. *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2021.

[46] S. Thrun. A lifelong learning perspective for mobile robot control. In *Proceedings of IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 1994.

[47] Gido M van de Ven and Andreas S Tolias. Generative replay with feedback connections as a general strategy for continual learning. *arXiv preprint arXiv:1809.10635*, 2018.

[48] Tom Veniat, Ludovic Denoyer, and MarcAurelio Ranzato. Efficient continual learning with modular networks and task-driven priors. In *International Conference on Learning Representations (ICLR)*, 2021.

[49] Jeffrey S. Vitter. Random sampling with a reservoir. *ACM Transactions on Mathematical Software*, 11(1):37–57, 1985.

[50] Zhirong Wu, Yuanjun Xiong, Stella X. Yu, and Dahua Lin. Unsupervised feature learning via non-parametric instance discrimination. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2018.

[51] Haiyan Yin, peng yang, and Ping Li. Mitigating forgetting in online continual learning with neuron calibration. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2021.

[52] Jaehong Yoon, Eunho Yang, Jeongtae Lee, and Sung Ju Hwang. Lifelong learning with dynamically expandable networks. In *International Conference on Learning Representations (ICLR)*, 2018.

[53] Guanxiong Zeng, Yang Chen, Bo Cui, and Shan Yu. Continuous learning of context-dependent processing in neural networks. *Nature Machine Intelligence*, 1:364–372, 2019.

[54] Friedemann Zenke, Ben Poole, and Surya Ganguli. Continual learning through synaptic intelligence. In *International Conference on Machine Learning (ICML)*, 2017.