

Attenuating Catastrophic Forgetting by Joint Contrastive and Incremental Learning

Quentin Ferdinand, Benoit Clement, Quentin Oliveau, Gilles Le Chenadec, Panagiotis Papadakis

▶ **To cite this version:**

Quentin Ferdinand, Benoit Clement, Quentin Oliveau, Gilles Le Chenadec, Panagiotis Papadakis. Attenuating Catastrophic Forgetting by Joint Contrastive and Incremental Learning. IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Jun 2022, New-Orleans, United States. 10.1109/CVPRW56347.2022.00423 . hal-03784379

HAL Id: hal-03784379

https://ensta.hal.science/hal-03784379v1

Submitted on 23 Sep 2022

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.



HAL Authorization

This CVPR workshop paper is the Open Access version, provided by the Computer Vision Foundation.
Except for this watermark, it is identical to the accepted version;
the final published version of the proceedings is available on IEEE Xplore.

Attenuating Catastrophic Forgetting by Joint Contrastive and Incremental Learning

Quentin Ferdinand[1,2], Benoit Clement[2], Quentin Oliveau[1], Gilles Le Chenadec[2], Panagiotis Papadakis[3]
[1] Naval Group Research*, France
[2] ENSTA Bretagne, Lab-STICC UMR 6285, Brest, France
[3] IMT Atlantique, Lab-STICC UMR 6285, Brest, France

Abstract

In class incremental learning, discriminative models are trained to classify images while adapting to new instances and classes incrementally. Training a model to adapt to new classes without total access to previous class data, however, leads to the known problem of catastrophic forgetting of the previously learnt classes. To alleviate this problem, we show how we can build upon recent progress on contrastive learning methods. In particular, we develop an incremental learning approach for deep neural networks operating both at classification and representation level which alleviates forgetting and learns more general features for data classification. Experiments performed on several datasets demonstrate the superiority of the proposed method with respect to well known state-of-the-art methods.

1. Introduction

Despite the popularity and strong performances of deep convolutional neural networks on computer vision tasks [9, 19], a number of problems are not yet fully addressed. Notably, the performance of a discriminative model is heavily dependent on the amount of data available during learning, whose availability and annotation in real-world problems is either questionable or time-consuming. However, these models are static, i.e. designed to be trained offline to solve a task and then used to solve the same task, therefore if the task changes overtime then these models will not automatically adapt. On the other hand, re-training a model with newly acquired data about either new knowledge or task data distribution changes will degrade performance on the tasks previously learnt, a problem well known as catastrophic forgetting [17] in the incremental learning research field.

To address this issue, prior works have focused on three main concepts, namely data rehearsal, task recency bias correction and knowledge distillation. Rehearsal-based methods generate or store a small portion of data from previous tasks and add them to the current task training data [14, 18, 21] in order to keep information about previous tasks in the dataset. As none or only a few samples from past tasks are usually stored and rehearsed, the dataset used for training is heavily imbalanced which leads to a score magnitude bias in the output of the last fully connected layer of the neural network towards most recent tasks; bias that some works attempt to minimize [16, 21, 24]. Finally, knowledge distillation approaches are regularisation-based methods borrowed from the field of transfer learning that add a term to the loss function in order to transfer knowledge of the previous tasks towards the model being trained for the current task [5, 6, 10, 18, 25].

Most of these methods operate at a classifier level, however we believe that maintaining a discriminative representation is equally important. In this context, in order to improve the representation of the model during the incremental learning process, we believe that we can draw an analogy with an emerging trend of representation learning called contrastive learning [3, 8] that has been shown to improve the discriminativeness of model representations [11]. We thus propose a new approach for joint training of the representation and the classification components of a model, via contrastive and incremental learning. On the one hand, we employ contrastive learning to learn a more discriminative representation for new classes while keeping the discriminative information of the previous representation for old classes. And in the other hand we make use of incremental learning methods to train an unbiased classifier that also adapts to new classes without forgetting previous ones.

2. Related works

Recently many advances have been proposed in representation, transfer, and incremental learning. In this section we will briefly describe the most important methods that can be used to alleviate catastrophic forgetting happening

*Thanks to Naval Group for supporting this work

in deep neural networks during incremental training.

2.1. Incremental learning

A significant amount of work has been proposed to alleviate catastrophic forgetting in the field of incremental learning. We refer the interested reader to the following surveys [4, 16] for a detailed presentation of the state-of-the-art and a comparison of the different algorithms of incremental learning, allowing us to focus on the main components in the following paragraphs.

Knowledge distillation was first introduced to the field of incremental learning by Li *et al*. [13] and amounts to the addition of a regularisation term to the training loss function with the aim of transferring the knowledge from a teacher model to a student model. In this work, the model from the previous incremental step was used as a teacher and the current model as a student to compare both models softened output probability distributions. This method was shown to transfer knowledge about the previous tasks into the new model, therefore alleviating catastrophic forgetting.

This idea has been employed in numerous works including iCarL [18] that first introduced the concept of data rehearsal to the incremental learning field. The optimal strategy in term of performance would be to rehearse all of the data previously seen, which is equivalent to ordinary classification. Conversely, in iCarL and most other works using rehearsal that emerged since then, the memory size was considered very limited and fixed during the whole incremental process in order to avoid oversimplifying the problem.

Another major bottleneck in incremental learning lies in the fact that the dataset used for training is imbalanced. In order to remove the bias towards recent classes, in iCarL, Rebuffi *et al*. proposed to use a nearest-exemplar-mean (NEM) classifier at test time instead of the classification layer. Nevertheless, better performance was then achieved in other works [1, 21, 24] by keeping the classification layer while adding a post training phase to remove the bias from the layer.

2.2. Contrastive learning

Initially introduced for unsupervised learning in simCLR [3], contrastive methods have been shown to learn discriminative representations also in supervised scenarios [11]. Conceptually, the idea is to use heavy data augmentation to create different views of each image and consider views of the same samples as "positives" and views of different ones as "negatives". Then a contrastive loss is used to pull the feature vectors of positive examples together while pushing negative ones apart. Khosla *et al*. [11] showed that this approach learns more discriminative representations with better generalisation capabilities than ones learnt with conventional cross-entropy. Moreover, in incremental learning it has been shown to learn representations that contain general knowledge useful for the classification of unseen classes which is particularly beneficial to transfer to upcoming incremental steps [2].

In the field of transfer learning, while standard knowledge distillation compares output probability distributions to transfer knowledge from a teacher to a student model, recent works showed that richer knowledge can be transferred from the features of the models [5, 6, 23]. Therefore numerous methods based on contrastive learning have been developed to transfer knowledge from models representations. In CRD [20], authors considered the representation of the teacher as a different view of the same image and maximized mutual information between image views. Recently, in SEED [7] and SSKD [22], state-of-the-art performance was achieved in transfer learning by comparing pairwise similarities between contrastive samples in the representation of the teacher and student [7, 22]. A simplified version of SEED has further been applied to incremental learning to alleviate forgetting of previous representations [2].

Overall, in [2] and [15] contrastive learning methods were successfully applied to incremental learning to learn more discriminative representations and allieviate forgetting of previous classes. However, the performances remained limited due to the fact that contrastive learning methods train only the feature extractor and have no impact on the classification layer of the model. In order to circumvent this limitation, authors used a NEM classifier in [15] and added another training step only for the classification layer in [2], but our method is the first to our knowledge that learns jointly the feature extractor and classification layer in contrastive incremental learning.

3. The problem setting

In the class incremental problem setting that we consider, a model is trained sequentially on multiple classification tasks indexed by $t \in \{1, ..., T\}$. For a given classification task at step t, a dataset $\{\mathcal{X}_t, \mathcal{Y}_t\}$ containing data belonging to C_t classes is drawn randomly from a distribution D_t, where $\mathcal{X}_t$ is a set of images and $\mathcal{Y}_t$ the associated ground-truth labels. For each task, the classes C_t are considered disjoint and the data from the previous tasks is supposed unavailable.

Let us denote by $\mathcal{L}$ the loss function used for the optimisation, θ the parameters of the model trained for the incremental learning step t, and $\mathcal{M}_\theta^t$ the function representing the model that we further decompose into $\mathcal{M}_\theta^t = \mathcal{F}_\omega^t \circ \varphi_\vartheta^t$, with φ_ϑ^t being the feature extractor, $\circ$ the composition operator and $\mathcal{F}_\omega^t$ the classifier. If we consider the task at incremental step $\mathcal{T}$ then the goal of the incremental learning process is to minimize the empirical risk of all seen tasks given limited access to data $\{\mathcal{X}_t, \mathcal{Y}_t\}$ from previous tasks $t < \mathcal{T}$ and total access to data $\{\mathcal{X}_\mathcal{T}, \mathcal{Y}_\mathcal{T}\}$ from the current

task $\mathcal{T}$:

$$L(\theta) := \frac{1}{\mathcal{T}} \sum_{t=1}^{\mathcal{T}} \mathbb{E}_{D_t}[\mathcal{L}(\mathcal{M}_\theta^t(\mathcal{X}_t), \mathcal{Y}_t)] \quad (1)$$

4. Method

Like most state-of-the-art methods for incremental learning [18,21,24], we employ rehearsal-based training with the cross-entropy loss $\mathcal{L}_{CE}$ to learn new classes during each incremental step and the distillation loss $\mathcal{L}_D$ to preserve knowledge about previously learnt classes. The usual baseline incremental loss used in most studies is :

$$\mathcal{L} = (1-\lambda)\mathcal{L}_{CE} + \lambda\mathcal{L}_D \quad (2)$$

With λ set to $\frac{C_{old}}{C_{new}+C_{old}}$ [21, 24], C_{old} representing the number of past classes and C_{new} the number of new classes.

Using this loss as a basis we add a contrastive learning version of both $\mathcal{L}_{CE}$ and $\mathcal{L}_D$ in order to focus on the features extracted by the model to learn better representations for new classes, and extract and preserve richer knowledge about the previous tasks. We therefore introduce the following total loss function :

$$\mathcal{L} = (1-\lambda)(\frac{\mathcal{L}_{CE}+\mathcal{L}_{con}}{2}) + \lambda(\frac{\mathcal{L}_D+\mathcal{L}_{Dcon}}{2}) \quad (3)$$

With $\mathcal{L}_{con}$ being the contrastive loss described in section 4.2 learning better representations for new classes, and $\mathcal{L}_{Dcon}$ the contrastive distillation loss described in section 4.3 preserving the representation of past classes. During each incremental step we then optimize the parameters of the model in order to minimize the loss function described in equation (3) over the incremental dataset containing the data from new classes and the rehearsal memory. The overall pipeline of our framework is shown in Figure 1 and explained in detail in the following subsections.

4.1. Baseline incremental learning method

In this section we will describe the incremental learning scheme using knowledge distillation and data rehearsal on which is based our method. Let us consider the classification task at time t with C_t classes comprising C_{new} new classes and C_{old} past classes, the task dataset $\{\mathcal{X}_t, \mathcal{Y}_t\}$ contains all the available data about the C_{new} classes and only the rehearsal samples stored about the previous C_{old} classes. The model parameters θ are initialized with the values obtained at the previous incremental step $t-1$, and C_{new} new randomly initialized output nodes are added for the new classes. Then the model is trained with the cross entropy loss $\mathcal{L}_{CE}$ to learn knowledge about new classes and with the knowledge distillation loss $\mathcal{L}_D$ to alleviate forgetting of the previous classes :

$$\mathcal{L}_{CE}(x,y) = \sum_{c=1}^{C_t} -\delta_{c=y}\ \log(p_c(x)) \quad (4)$$

$$\mathcal{L}_D(x) = \sum_{c=1}^{C_{old}} -q_c^{t-1}(x)\ \log(q_c^t(x)) \quad (5)$$

where $\delta_{c=y}$ is the indicator function, $p_c(x)$ the output softmax probability for the c^{th} class, $q_c^t(x) = \frac{e^{o_c(x)/\tau}}{\sum_{i=1}^{C_{old}} e^{o_i(x)/\tau}}$ is the softened softmax probability obtained from output node o_c of the model, τ is a temperature parameter, and $q_c^{t-1}(x)$ is the same softened softmax probability but obtained from the outputs of the model from task $t-1$.

Furthermore, following each incremental learning step we adopt the weight aligning bias correction method from [24] that proved to be very effective in removing the classification layer bias while requiring negligible computation time.

4.2. Supervised contrastive representation learning

In order to learn good representations for new classes with contrastive learning, we use a setup similar to CO2L [2]. First, when a batch of N samples $\{(x_i, y_i)\}_1^N$ is drawn from the dataset we use heavy data augmentation to generate 2 augmentations $\{(\hat{x}_i, y_i)\}_1^{2N}$ of each image. Then, considering the augmented minibatch $\{(\tilde{x}_i, y_i)\}_1^{3N}$, we extract the features $\phi_i = \varphi_\vartheta^t(\tilde{x}_i)$. Following [2,7,20,22], a projection map Γ_ψ^t parametrized by ψ is used to project features onto a d-dimensional unit hypersphere : $\tilde{z}_i = \frac{\Gamma_\psi^t(\phi_i)}{\|\Gamma_\psi^t(\phi_i)\|}$, and the parameters ψ are optimized together with the model parameters to minimize our overall loss described in eq. 3. Finally, the contrastive features $\tilde{z}_i$ of the augmented batch are used to minimize the asymmetric supervised contrastive loss introduced in [2] :

$$\mathcal{L}_{con} = \sum_{i\in S} \frac{-1}{|\mathcal{P}_i|} \sum_{j\in\mathcal{P}_i} \log(\frac{\exp(\tilde{z}_i \cdot \tilde{z}_j/\tau)}{\sum_{\substack{k\in I \\ k\neq i}} \exp(\tilde{z}_i \cdot \tilde{z}_k/\tau)}) \quad (6)$$

where I is the augmented minibatch, S is the subset of I containing only samples of new classes, $\mathcal{P}_i$ the subset of S containing the positives of sample i, i.e. all the samples and augmented samples of the same label, and τ is a temperature hyperparameter.

Since the $\tilde{z}_i$ and $\tilde{z}_j$ represent the projection of the features onto a d-dimensional unit hypersphere, the dot product is equivalent to a cosine similarity. Therefore, this loss can be seen as maximizing similarity between new classes samples and their positives while minimizing similarity with negatives. Thus "pulling" together new classes samples and their positives in the representation while "pushing" away all other samples.

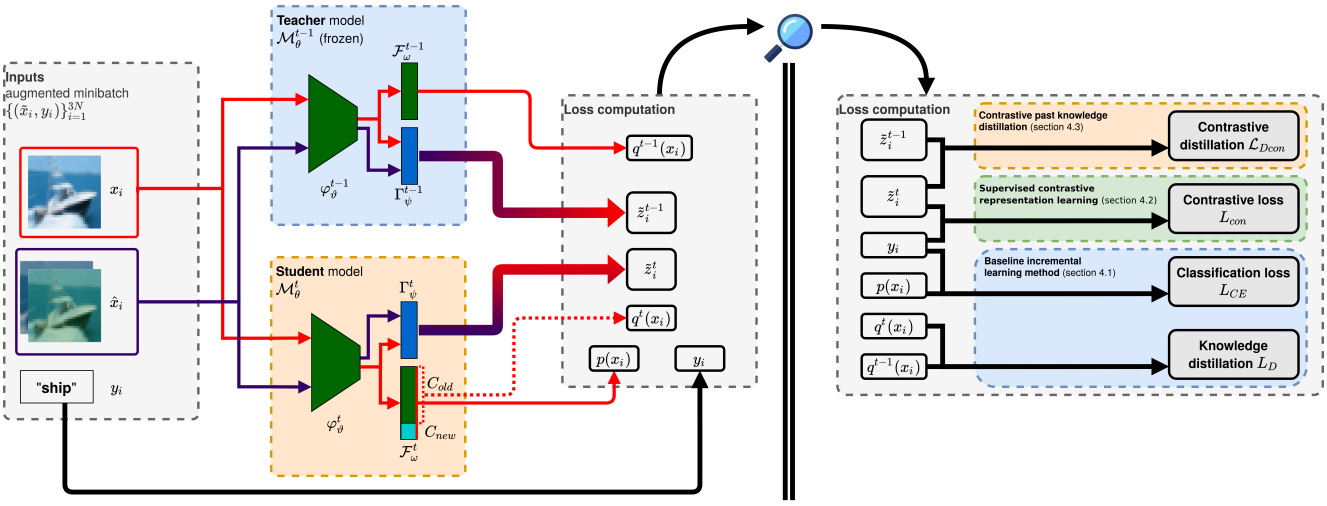


Figure 1. Pipeline of the proposed approach. During each incremental step, images are sampled in minibatches from the incremental dataset containing new data and rehearsal data in order to compute the four losses $\mathcal{L}_{CE}$, $\mathcal{L}_{con}$, $\mathcal{L}_{D}$, and $\mathcal{L}_{Dcon}$ used in eq. 3.

4.3. Contrastive past knowledge distillation

Using the asymmetric supervised contrastive loss allows the model to learn better representations for new classes but not for past classes. In order to preserve the good representation learnt previously for those classes we introduce a new supervised contrastive distillation loss inspired from [22]. The general goal of this loss is to allow the representation of the model to change to extract discriminative features for new classes but to ensure the new features produce the same similarities between rehearsal samples than the previous features. This way the representation is allowed to adapt to new classes but the underlying information about past classes is preserved.

Since the model $\mathcal{M}_\theta^{t-1}$ from the previous incremental step has not been trained on data about the new classes, the representation obtained for the $\{(x_i, y_i)\}_{y_i \in C_{new}}$ is not necessarily a very discriminative one. We therefore ignore samples from new classes when computing this distillation loss in order to focus on preserving similarities between representations of samples kept in the rehearsal memory.

Similarly to the loss $\mathcal{L}_{con}$ described in the previous section, using $\{(x_i, y_i), y_i \in C_{old}\}$ and $\{(\hat{x}_i, y_i), y_i \in C_{old}\}$, we compute z_i^t, and $\hat{z}_i^t$, but also z_i^{t-1} and $\hat{z}_i^{t-1}$, to obtain the contrastive representation produced by $\mathcal{M}_\theta^t$ and $\mathcal{M}_\theta^{t-1}$ for each image of past classes from the minibatch and the augmented versions of these images. We then compute the pairwise similarities between z_i and $\hat{z}_i$ for each model and organize them into matrices B^t and B^{t-1} with :

$$B_{i,j}^t = \frac{z_i^t \cdot \hat{z}_j^t}{\tau} \tag{7}$$

where $B_{i,j}^t$ contains the similarity between the contrastive representation of M_t for x_i and $\hat{x_j}$, and τ is another temperature hyperparameter. We then apply softmax to each row of the matrices B^t and B^{t-1} to obtain probability distributions, and in analogy to the distillation process described in eq. 5, we minimise the divergence between those two probability matrices :

$$\mathcal{L}_{Dcon} = -\tau^2 \sum_{i,j} B_{i,j}^{t-1} \log(B_{i,j}^t) \tag{8}$$

Overall, this loss allows the representation of the model to adapt to new classes but ensures that the representation of samples from previous incremental steps produce the same similarities than in the representation of the previous model.

5. Experiments

In the following sections we will describe the details of our algorithm and the general incremental setup we used, compare our algorithm to other state-of-the-art methods and conduct ablation studies to validate the effectiveness of our method.

5.1. Experimental setup

We evaluate our algorithm and other methods on two datasets that are widely used in incremental learning [18,21, 24], Cifar-100 and ImageNet-100. Cifar-100 [12] contains 32 $\times$ 32 pixel color images of 100 classes, with 500 images per class for training and 100 images per class for validation. ImageNet-100 on the other hand contains images of 64 $\times$ 64 pixels and represents a subset of 100 random classes from the ImageNet ILSVRC 2012 [19] dataset containing 1000 classes. Imagenet-100 contains 500 images per class for training and 50 images per class for validation.

We employed PyTorch in our implementation and following [18, 21, 24] we chose the 32-layer Resnet model for

	Cifar-100		ImageNet-100	
	last Acc	avg Acc	last Acc	avg Acc
Finetuning	8.81 ±0.38%	18.76 ±0.14%	8.66 ±0.28%	17.58 ±0.43%
iCarL_CNN	39.47 ±0.75%	54.78 ±1.24%	40.65 ±1.17%	53.58 ±1.66%
iCarL_NEM	47.80 ±0.73%	59.51 ±1.28%	**47.82 ±1.07%**	57.83 ±1.60%
CO2L	32.15 ±0.18%	47.27 ±0.10%	33.51 ±0.20%	50.01 ±1.03%
MDFCIL	50.43 ±0.71%	62.02 ±2.14%	46.29 ±1.84%	56.08 ±1.75%
Ours	**50.81 ±0.59%**	**64.13 ±0.52%**	47.64 ±1.32%	**59.13 ±1.59%**
Joint Training	69.39 ±0.26%	69.39 ±0.26%	67.24 ±0.78%	67.24 ±0.78%

Table 1. Class incremental learning performance on Cifar-100 and ImageNet-100 with 10 incremental steps and 10 classes added per step. The top-1 average accuracy over all the incremental steps aswell as the accuracy after the last one are reported. For each method we report the mean over 10 runs with random class orderings for fair comparison. For the method iCarL we report performances using the model classification layer (iCarL_CNN) and using their nearest exemplar mean classifier (iCarL_NEM).

Cifar-100 dataset and 18-layer Resnet [9] for ImageNet-100. We used the optimizer SGD with a momentum of 0.9, a batch size of 128, and a weight decay of 0.0002. We trained our models for 250 epochs during each incremental step, the learning rate starts at 0.1 and is divided by ten after 150, 180, and 210 epochs. The data augmentation applied to training images consists in random cropping, horizontal flip and normalization. The temperature parameter τ was set to 2 in $\mathcal{L}_D$ and 0.2 for the contrastive losses $\mathcal{L}_{con}$ and $\mathcal{L}_{Dcon}$. Moreover, for the contrastive losses we create 2 images with the same data augmentation than the initial image with the addition of color jitter and random color dropping similarly to [2]. Following other contrastive learning methods [22] we use a 2-layer MLP to project features onto the contrastive unit hypersphere. We separate each dataset in 10 incremental steps, starting initial learning with 10 classes and adding 10 classes per step. Following [18] we use a rehearsal memory of 2000 images and use the herding sampling strategy.

5.2. Comparison to other methods

We compare our method to several other rehearsal based competitive incremental learning algorithms :

Incremental Classifier And Representation Learning (iCarL). [18] This algorithm uses a nearest-exemplar-mean (NEM) classifier to remove new classes bias during evaluation time, trains the model using a binary cross-entropy based classification and distillation loss, and uses data rehearsal with the herding selection strategy.

Maintaining Discrimination and Fairness in Class Incremental Learning (MDFCIL). [24] This method differentiates itself from iCarL by using the conventional cross-entropy for the classification and distillation losses and adding the weight alignment step that we used in our algorithm after each incremental training step to remove bias from the classification layer of the model.

Contrastive Continual Learning (CO2L). [2] This method trains a feature extractor using contrastive versions of the classification and distillation losses used in incremental learning. Compared to the contrastive losses used in our method the main difference is the equation of their contrastive distillation loss and its computation on all samples from the minibatches instead of just samples coming from the rehearsal memory. Since their method trains only a feature extractor they further add a second training step to train a classifier with the conventional cross-entropy.

Finetuning. Finetuning represents the lower bound of performance achievable in incremental learning. Finetuning is a simple training setup with only the conventional cross entropy applied to finetune the model with each incremental dataset and no other incremental learning parts.

Joint-training. Joint-training in the other hand represents the upper bound of performances. It corresponds to training a model from scratch with conventional cross-entropy during each incremental steps with the total dataset containing all data about new and past classes.

For thorough comparison of our method to state-of-the-art ones we run each algorithm 10 times on the two datasets considered with random class orderings. For fair comparison, we use the same models for each algorithms, so 32-layer Resnet for the dataset Cifar-100 and 18-layer Resnet for the dataset Imagenet-100. Performances of contrastive methods are positively correlated with large batch sizes [3, 20], but it is not the case for incremental methods. Therefore we run all algorithms with a batch size of 128 on both datasets for an unbiased comparison and coherence with other incremental non-contrastive studies. We report in table 1 the top-1 accuracy obtained after the last incremental step and the average incremental accuracy over all incremental steps, ignoring the accuracy of the initial non-incremental learning step. We further provide in figure 2 the accuracy of each method on Cifar-100 as a function of the number of classes seen during the incremental process.

As can be seen in table 1 and in figure 2 our method slightly surpass all other methods on both datasets, ver-

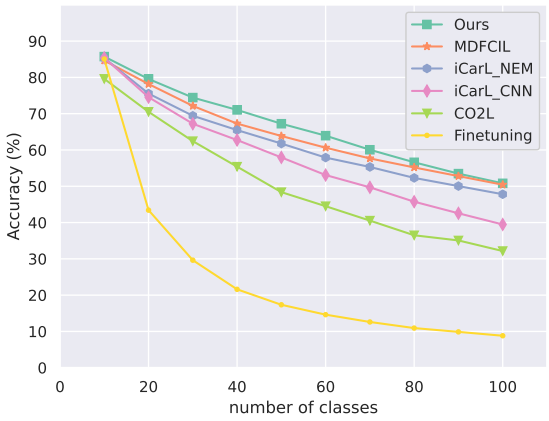


Figure 2. Evolution of the accuracy as a function of the number of classes learnt incrementally. The mean performance obtained on Cifar-100 over 10 training trials with random class orderings is shown, excluding standard deviations for clarity.

ifying our idea that contrastive methods can be used together with current state-of-the-art incremental methods to improve the representation of the model which in turn improves downstream classification accuracy. Besides, we can also observe on the figure 2 that the performances of the other contrastive learning method CO2L are quite low on Cifar-100 which can be explained by the batch size used relatively low for the dataset compared to usual contrastive learning batch sizes. However, since our method uses contrastive losses jointly with incremental losses we can see that it is much more robust to small batch sizes.

5.3. Ablation Study

In order to validate the effectiveness of our method we performed the following ablation studies :

- **Ablation A.** Ablation of $\mathcal{L}_{Dcon}$. We remove the contrastive distillation loss from the optimization process to evaluate the impact of this new distillation.

- **Ablation B.** Ablation of all contrastive losses. Removing only $\mathcal{L}_{con}$ would also impact $\mathcal{L}_{Dcon}$ because Γ_{ψ}^{t-1} would not have been trained by $\mathcal{L}_{con}$ during the previous incremental step. Therefore we instead perform an ablation of both $\mathcal{L}_{con}$ and $\mathcal{L}_{Dcon}$ to see the added benefit of contrastive losses and compare to ablation A to see the added benefit of individual contrastive losses.

- **Ablation C.** Ablation of non-contrastive losses. In order to observe the impact of the incremental losses we perform an ablation of $\mathcal{L}_{CE}$ and $\mathcal{L}_{D}$ and keep only $\mathcal{L}_{con}$ and $\mathcal{L}_{Dcon}$.

For a more straightforward comparison and since the focus of this ablation study is to evaluate the impact of each part of our method and not to evaluate differences between datasets, we compared performances only on the dataset Cifar-100. We can see that the performances of our method in table 1 and 2 are similar but not exactly the same, this is the case because of different class orderings, therefore for fair comparison we used the same random class orderings for each ablation. We report in table 2 the top-1 average incremental accuracy obtained with the model classifier and with the nearest exemplar mean (NEM) classifier from [18].

The NEM classifier allows us to evaluate the representations of models without being impacted by the classification layer. This classifier first computes the mean feature vector of each class using the incremental training set and the rehearsal memory after each incremental step. Then, at test time, images are classified to the closest mean vector, therefore classifying images directly within the feature space without the use of the classification layer or any other parameters. Since all incremental losses are removed in ablation C, the classification layer of the model is not trained at all, therefore we make use of the NEM classifier to compare its performances to the other ablations.

By comparing the full method and ablation C in table 2, we can clearly see that adding incremental losses to the contrastive ones improves the method. Indeed, in ablation C the top-1 accuracy is not provided because contrastive losses only train the representation of the neural network and not the classifier. This is the most straightforward benefit of using them with incremental losses, the classifier is trained jointly with the representation. Moreover, comparing NEM accuracies we can see that incremental losses also improve the representation of the model which is mainly due to $\mathcal{L}_{D}$

	Top-1 accuracy	Top-1 NEM accuracy
Full method	**63.91** ±0.96%	**63.10** ±1.10%
Ablation A	63.33 ±1.18%	62.71 ±1.28%
Ablation B	62.65 ±1.32%	61.39 ±1.18%
Ablation C	-	58.51 ±1.05%

Table 2. Ablation study done on CIFAR100 with 10 incremental steps. We report the top-1 average incremental accuracy and NEM accuracy. Each method was run 10 times with random class orders but with the same ones for each ablation for fair comparison. Ablation C does not train a classification layer therefore top-1 accuracy can not reported and NEM accuracy is used instead to compare performances.

that can extract knowledge about past classes from images of new classes where $\mathcal{L}_{Dcon}$ uses only the rehearsal memory to extract knowledge about past classes.

On the other hand, comparing ablations A and B to the full method shows that incremental losses also benefit from contrastive ones as the addition of each contrastive loss slightly improves accuracies. Indeed, in ablation A where only $\mathcal{L}_{con}$ is added compared to ablation B, the accuracies are slightly higher which can be explained by the representation of new classes during each incremental step that is improved. And the same observation can be done when comparing the full method to ablation A, the addition of $\mathcal{L}_{Dcon}$ further improves the representation of the model by alleviating catastrophic of the features from previous classes therefore improving accuracy.

Overall, the ablation results show that the removal of the incremental and contrastive losses both decrease performances, therefore validating our hypothesis that both the standard distillation and the contrastive distillation alleviate forgetting and that the model benefits from using both. Performance gains from contrastive losses however remain moderate, we therefore believe it would be interesting for subsequent works to increase overall importance of contrastive losses compared to incremental ones during training and validate the method on large scale datasets.

6. Conclusion

In this work we adapted contrastive learning concepts to the incremental learning problem. In particular, we showed that while conventional incremental learning methods are effective in alleviating catastrophic forgetting, they can further benefit from contrastive learning losses both for learning more general knowledge from new classes and remembering better the representation of past classes. This allowed our proposed approach using both incremental and contrastive learning concepts to jointly train the representation and classifier of the model and attain noticeable improvements over other state-of-the-art methods in the incremental learning baseline scenario on two different datasets.

References

[1] Francisco M. Castro, Manuel J. Marín-Jiménez, Nicolás Guil, Cordelia Schmid, and Karteek Alahari. End-to-end incremental learning. In Vittorio Ferrari, Martial Hebert, Cristian Sminchisescu, and Yair Weiss, editors, *European Conference on Computer Vision (ECCV)*, 2018. 2

[2] Hyuntak Cha, Jaeho Lee, and Jinwoo Shin. Co2l: Contrastive continual learning. In *IEEE/CVF International Conference on Computer Vision (ICCV)*, 2021. 2, 3, 5

[3] Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. A Simple Framework for Contrastive Learning of Visual Representations. In *International Conference on Machine Learning*. PMLR, 2020. 1, 2, 5

[4] 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. 2

[5] Prithviraj Dhar, Rajat Vikram Singh, Kuan-Chuan Peng, Ziyan Wu, and Rama Chellappa. Learning without memorizing. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019. 1, 2

[6] Arthur Douillard, Matthieu Cord, Charles Ollion, Thomas Robert, and Eduardo Valle. Podnet: Pooled outputs distillation for small-tasks incremental learning. In *European Conference on Computer Vision (ECCV)*, 2020. 1, 2

[7] Zhiyuan Fang, Jianfeng Wang, Lijuan Wang, Lei Zhang, Yezhou Yang, and Zicheng Liu. Seed: Self-supervised distillation for visual representation. *International Conference on Learning Representations (ICLR)*, 2021. 2, 3

[8] Kaiming He, Haoqi Fan, Yuxin Wu, Saining Xie, and Ross Girshick. Momentum contrast for unsupervised visual representation learning. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020. 1

[9] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep Residual Learning for Image Recognition. 2015. arXiv: 1512.03385. 1, 5

[10] Saihui Hou, Xinyu Pan, Chen Change Loy, Zilei Wang, and Dahua Lin. Learning a unified classifier incrementally via rebalancing. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019. 1

[11] Prannay Khosla, Piotr Teterwak, Chen Wang, Aaron Sarna, Yonglong Tian, Phillip Isola, Aaron Maschinot, Ce Liu, and Dilip Krishnan. Supervised Contrastive Learning. 2021. arXiv: 2004.11362. 1, 2

[12] Alex Krizhevsky, Geoffrey Hinton, et al. Learning multiple layers of features from tiny images. 2009. 4

[13] Zhizhong Li and Derek Hoiem. Learning without Forgetting. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2018. 2

[14] Yaoyao Liu, Yuting Su, An-An Liu, Bernt Schiele, and Qianru Sun. Mnemonics training: Multi-class incremental learning without forgetting. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020. 1

[15] 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

[16] Marc Masana, Xialei Liu, Bartlomiej Twardowski, Mikel Menta, Andrew D Bagdanov, and Joost van de Weijer. Class-incremental learning: survey and performance evaluation. *arXiv:2010.15277*, 2020. 1, 2

[17] Michael McCloskey and Neal J. Cohen. Catastrophic Interference in Connectionist Networks: The Sequential Learning Problem. In *Psychology of Learning and Motivation*. Academic Press, 1989. 1

[18] Sylvestre-Alvise Rebuffi, Alexander Kolesnikov, Georg Sperl, and Christoph H. Lampert. icarl: Incremental classifier and representation learning. In *IEEE Conference on*

Computer Vision and Pattern Recognition (CVPR), 2017. 1, 2, 3, 4, 5, 6

[19] Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael S. Bernstein, Alexander C. Berg, and Li Fei-Fei. Imagenet large scale visual recognition challenge. *International Journal of Computer Vision*, 2015. 1, 4

[20] Yonglong Tian, Dilip Krishnan, and Phillip Isola. Contrastive multiview coding. *arXiv:1906.05849*, 2019. 2, 3, 5

[21] Yue Wu, Yinpeng Chen, Lijuan Wang, Yuancheng Ye, Zicheng Liu, Yandong Guo, and Yun Fu. Large scale incremental learning. In *IEEE Conference on Computer Vision and Pattern Recognition (ICCV)*, 2019. 1, 2, 3, 4

[22] Guodong Xu, Ziwei Liu, Xiaoxiao Li, and Chen Change Loy. Knowledge distillation meets self-supervision. In *European Conference on Computer Vision (ECCV)*, 2020. 2, 3, 4, 5

[23] Sergey Zagoruyko and Nikos Komodakis. Paying More Attention to Attention: Improving the Performance of Convolutional Neural Networks via Attention Transfer. 2017. arXiv: 1612.03928. 2

[24] Bowen Zhao, Xi Xiao, Guojun Gan, Bin Zhang, and Shu-Tao Xia. Maintaining discrimination and fairness in class incremental learning. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020. 1, 2, 3, 4, 5

[25] Peng Zhou, Long Mai, Jianming Zhang, Ning Xu, Zuxuan Wu, and Larry S. Davis. M2KD: Multi-model and Multi-level Knowledge Distillation for Incremental Learning. 2019. arXiv:1904.01769 [cs]. 1