OpenIncrement: A Unified Framework for Open Set Recognition and Deep Class-Incremental Learning

Jiawen Xu
Technical University Berlin
jiawen.xu@campus.tu-berlin.de

Claas Grohnfeldt
Huawei Munich Research Center
claas.grohnfeldt@huawei.com

Odej Kao
Technical University Berlin
odej.kao@tu-berlin.de

Abstract

In most works on deep incremental learning research, it is assumed that novel samples are pre-identified for neural network retraining. However, practical deep classifiers often misidentify these samples, leading to erroneous predictions. Such misclassifications can degrade model performance. Techniques like open set recognition offer a means to detect these novel samples, representing a significant area in the machine learning domain.

In this paper, we introduce a deep class-incremental learning framework integrated with open set recognition. Our approach refines class-incrementally learned features to adapt them for distance-based open set recognition. Experimental results validate that our method outperforms state-of-the-art incremental learning techniques and exhibits superior performance in open set recognition compared to baseline methods.

1. Introduction

Deep learning-based classification, underpinning advancements in areas like object recognition [48] and sentiment analysis [11], heavily relies on ample pre-collected data. In realistic scenarios, accumulating extensive training data quickly is often infeasible due to episodic and unpredictable data emergence. Thus, the appeal of deep incremental learning, which assimilates new data while retaining prior knowledge [31, 6, 40, 47]. This mode of learning eschews retraining from scratch, eliminating the need for complete datasets during successive *training sessions*. Among various continual learning paradigms, class-incremental learning, wherein models assimilate new classes and retain information about observed ones, is paramount.

A notable oversight in the realm of class-incremental learning is the prerequisite identification of novel samples or *open sets*. While numerous works have addressed the *Open Set Recognition* (OSR) challenge [2, 10, 13, 18, 26, 36, 35], only a handful have sought an integrative framework combining OSR and incremental learning. While [20] charted this terrain within object detection, their approach was restrictive in scope and application. It's imperative to establish a singular model adept at OSR that also evolves with new classes.

Our investigation aligns closely with task-free continual learning [1, 19, 42], a domain where the onus is on innate model mechanisms to discern data drifts and instigate model updates. Notably, these works diverge in their handling of new class identification and rarely evaluate this performance aspect. Our proposition centers on a unified model employing distillation-based continual learning.

In this work, we meld class-incremental learning with OSR considerations, leaning on distance-based OSR and rehearsal-infused continual learning (see 2.1 and 2.2). The central challenge is maintaining discernible inlier and outlier features post-training. We discern that feature spaces undergo *distortions* during incremental training, leading to intertwined inlier and outlier features—a probable cause of the *catastrophic forgetting* phenomenon.

Response-based knowledge distillation [16], pivotal in staving off *catastrophic forgetting* in deep incremental learning, often bypasses feature relation transfer, resulting in distortions. We counter this by leveraging relation-based knowledge distillation (RKD) and supplement with supervised contrastive learning (*SupCon*) [21] to bolster class-specific feature separation. Unlike the method in [7], our focus is a holistic framework for OSR and class-incremental learning, not merely an incremental learning strategy. Future explorations might traverse other OSR and incremental learning intersections.

Our primary contributions are:

- Pinpointing feature distortion as an aftermath of incremental learning.

- Crafting a cohesive framework for open set recognition and class-incremental learning.

- Empirical validation of our approach on public and

continual learning benchmarks, achieving commendable results for both facets.

2. Background

Since our work relates to both open set recognition and class-incremental learning, we will review the main approaches of these two domains in this section. Meanwhile, our method is closely connected to knowledge distillation. We will therefore briefly introduce it as well.

2.1. Open Set Recognition

Open set recognition is an everlasting topic for supervised classifiers while the openness of the data is usually unknown and the novel classes can always appear and should be detected. For deep neural networks, there are mainly three categories of OSR methods, namely distance-based [2][10][13][18][26][36][41][8], generative model-based [37][5][30][43], and background class-based [15][44][12].

Bendale et al. have first proposed *OpenMax* that models the *Euclidean* distances between the softmax outputs of each data sample and their closest class centers using *Weibull* distribution [2]. The outliers are then discriminated by thresholding the inference probability. Moreover, Dhamija et al. proposed the *Objectosphere* loss [10] and Hassen et al. presented their *ii-loss* [13] to learn features that are better clustered. Miller et al. proposed the *Class Anchor Clustering* loss that enforces the inlier features to be close to each other [26]. There are more works introducing similar methods, such as [36] and [35].

Generative model-based methods directly model the inliers. In [37], variational autoencoders are trained using inliers to recognize outliers. Analogously, Cao et al. applied *variational autoencoders* to learn the latent features for inliers that are used for both inlier class clustering and outlier detection [5]. Perera et al. proposed to augment the inlier samples with GANs to gain more informative features that can be more discriminative from the outliers [30].

Background class-based methods are straightforward in that one extra outlier class output is added to the original classifiers and the entire models are trained in a supervised way. Yu et al. proposed to generate pseudo outliers using GANs inspired by the principle of *adversarial training* [44]. The generated pseudo samples are then utilized altogether with the inliers to train a $(C+1)$ classifier (C is the number of inlier classes). Analogously, Ge et al. trained a $(C+1)$ classifier with synthetic outliers to calibrate the feature distances in *OpenMax* [12]. The drawback of these methods lies in that the open set classes are infinite and cannot be completely collected, especially the "*unknown unknowns*".

2.2. Class-Incremental Learning

The challenge behind deep incremental learning is *catastrophic forgetting* that the models will lose the capability for observed tasks after the new training session. To prevent catastrophic forgetting, there are two main categories of class-incremental learning approaches, namely parameter-based methods [23][45] and distillation-based methods [31][6][17][25].

Parameter-based methods require no stored history exemplars and apply constraints directly on the model parameters. *Elastic Weight Consolidation* (*EWC*) [23] proposed to constrain the weights in the region of low-performance degradation for observed classes. *Synaptic Intelligence* (*SI*) [45] regulates the weights during new training sessions to keep the important weights from significant changes.

The distillation-based methods apply knowledge distillation (see 2.3) in new training sessions to directly transfer the knowledge learned by old models (i.e. the teacher) to the new models (i.e. the student). A portion of observed data (i.e., exemplars) should be stored in these methods. *Learning without Forgetting* (*LwF*) [25] has first applied response-based knowledge distillation in incremental learning. *Incremental classifier and Representation Learning* (*iCaRL*) [31] adapts a similar knowledge distillation approach but classifies the test samples with *nearest-mean-of-exemplars* approach instead of the final softmax layer because of the class imbalance.

2.3. Knowledge Distillation

In the context of deep learning, knowledge distillation (KD) was first known in [16] and [33] that enables to transfer the knowledge learned by a larger model (teacher) to a smaller one (student). There are three types of knowledge that can be transferred, namely response-based knowledge, feature-based knowledge, and relation-based knowledge.

Response-based knowledge distillation is to directly align the predictions between students and teachers. The divergence between the logit layers of teachers and students is encouraged to be minimized. In [16], the logit layer of the teacher network is adapted as *soft targets* when computing the softmax loss so that the student can be expected to give the same outputs as the teacher.

In feature-based knowledge distillation, not only the last logit layer but also the intermediate layers of the teachers are utilized to transfer knowledge. The students can therefore output similar features as the teachers. In [33], the neural activations of the first layers in the teacher model are directly used to match the student model. Kim et. al. proposed to extract the *factor maps* of teachers' and students' layers using convolutional modules and let the student factor mimic the teacher factor [22]. Similarly, Passban et. al. proposed a combinatorial technique that can merge the features of multiple layers in the teacher model using the atten-

tion mechanism [29].

Unlike the previous two categories that align the layer outputs between the teachers and students, relation-based knowledge distillation tends to preserve the relationships among the features in different layers or instances [28][39][38]. Park et. al. proposed the relational knowledge distillation that transfers the instance relations between teachers and students [28]. Tung et. al. proposed the similarity-preserving knowledge distillation, in which the similarities between the instance features can be transferred to the students [39].

3. Problem Statement

The goal of this study is to enable neural networks to learn representations that are suitable for open set recognition under class-incremental learning settings. Distance-based OSR methods are often based on two assumptions (see Assumption 1 and 2 below) [26][13][41][8], which are from the *Smoothness Assumption* [9] stating that *if two points are near, their corresponding output values cannot be arbitrarily far from each other*. The problem is then converted to how to enable the learned representations to still hold these two assumptions after repeating class-incremental training.

Assumption 1 *Data representations of the same class should be close to each other.*

Assumption 2 *Data representations of different classes should be pushed apart.*

The metrics *intra spread*, S_{intra}, in Equ. (1), and *inter spread*, S_{inter}, in Equ. (2) are introduced in [13] to evaluate how well the learned representations follow the above two assumptions respectively. In Equ. (1) and (2) and the following text, μ represents the euclidean centers of each class and $\mathbf{z}_i$ are the instance representations. The number of inlier classes is denoted using K, and C_j is the number of instances of each class. According to Assumption 1 and 2, S_{inter} should be as large as possible whereas S_{intra} is supposed to be small. However, the distances are of different scales in different models and are hard to be compared directly. We, therefore, propose to fuse these two metrics using their ratio, i.e., $R_s = \frac{S_{intra}}{S_{inter}}$. It is straightforward to understand that the smaller the R_s is, the more suitable the feature is for OSR.

$$S_{intra} = \frac{1}{N} \sum_{j=1}^{K} \sum_{i=1}^{C_j} \|\vec{\mu_i} - \mathbf{z}_i\|_2^2 \qquad (1)$$

$$S_{inter} = \min_{\substack{1 \le m \le K \\ m+1 \le n \le K}} \|\vec{\mu_m} - \vec{\mu_n}\|_2^2 \qquad (2)$$

It can be found from Tab. 1 that R_s in incremental learning settings are much larger than in normal training (line *Joint* in the table). We believe it is because the feature relations between the data samples are not preserved when the models are updated incrementally. For incremental learning methods based on rehearsal and regularization, such as *iCaRL*, the knowledge learned in previous sessions is preserved mainly through response-based knowledge distillation. As stated in [28], response-based knowledge distillation cannot transfer data-sample relations between teachers and students. Therefore, Assumption 1 and 2 can be easily broken with these incremental learning methods. For regularization-based incremental learning methods, such as *EWC* and *SI*, the critical weights for observed classes are constrained to change during retraining to prevent catastrophic forgetting. In these approaches, the newly optimized models have achieved balances between the loss functions of old and new tasks. But they ignored maintaining the relations between the features as well.

In order to prevent such a phenomenon, the main research question in this study is to search for an incremental learning method, with which the above two assumptions for data representations can still be maintained after repeating incremental training. And the newly learned representations are hence adaptable for OSR.

The problem we address in this paper can be formulated as follows: the labeled observed training data is denoted using $\mathbf{D}_{obser} = (\mathbf{X}_{obser}, \mathbf{Y}_{obser}) = \{\mathbf{x}_i, y_i\}$ and the label $y_i \in \mathbf{C}_{obser} = [0, 1, ..., c_{obser} - 1]$. A deep classifier $\mathbf{F}(\mathbf{x}_i) \rightarrow \hat{y}_i$ is trained using $\mathbf{D}_{obser}$. $\hat{y}_i$ denotes the predictions and $\mathbf{F}(\mathbf{x}_i) = (\mathbf{H} \circ \mathbf{E})(\mathbf{x}_i)$, in which $\mathbf{E}$ and $\mathbf{H}$ are the deep encoder and classification header respectively. $\mathbf{z}_i = \mathbf{E}(\mathbf{x}_i)$ represents the feature maps outputted by the deep encoder. When the new classes $\mathbf{C}_{new}$ appear, then $\mathbf{D}_{obser} = \mathbf{D}_{obser} \cup \mathbf{D}_{new}$, $c_{obser} = c_{obser} + c_{new}$ and $\mathbf{F}(\cdot), \mathbf{E}(\cdot)$ and $\mathbf{H}(\cdot)$ will be updated using incremental learning approach $\mathcal{I}$ so that $\mathbf{F}'(\cdot) = \mathcal{I}(\mathbf{F}(\cdot))$, $\mathbf{H}'(\cdot) = \mathcal{I}(\mathbf{H}(\cdot))$ and $\mathbf{E}'(\cdot) = \mathcal{I}(\mathbf{E}(\cdot))$. We intend to optimize $\mathcal{I}$ in order to let R_s over $\mathbf{E}'(\mathbf{x}_{obser})$ be as small as possible.

4. Methods

This study is aimed to propose a deep class-incremental learning method in which the learned features can fulfill Assumption 1 and 2 for distance-based open set recognition. As discussed in Sec. 2 and 3, deep incremental learning using response-based knowledge distillation loss cannot transfer the instance relations to the new models and therefore fails the OSR methods. Hence, we apply relation-based knowledge distillation in this research. Furthermore, compared with cross-entropy loss, which is the most applied training strategy in classification problems, supervised contrastive learning is capable of learning features that satisfy Assumption 1 and 2 better as we will prove in the follow-

#Classes	10 (base)	20	30	40	50	60	70	80	90	100
Joint	2.54	3.38	3.76	3.95	4.06	3.98	4.07	3.65	3.85	3.82
SI [45]	2.54	5.56	6.64	6.61	6.82	7.42	7.33	7.25	9.83	7.25
EWC [23]	2.54	11.70	10.14	9.83	10.64	13.48	14.96	15.17	13.79	12.28
iCaRL [31]	2.54	3.41	3.40	6.87	5.70	5.98	6.06	6.11	7.05	7.15
LwF [25]	2.54	5.11	4.74	6.41	6.66	6.05	6.29	5.52	6.37	7.44

Table 1: R_s for joint retraining and incremental training using methods in [45][23][31][25] on *Cifar-100* dataset [24]. The dataset is split into 10 tasks with 10 novel classes each. The base 10-class models are the same for all methods. All models are trained using the best configurations given in [4].

ing text. In summary, the deep incremental learning approach proposed in this study is based on supervised contrastive learning and relation-based knowledge distillation. We name it *OpenIncrement*. The loss function is in Equ. (3), in which $\mathcal{L}_{SupCon}$ stands for supervised contrastive loss and $\mathcal{L}_{distill}$ represents the distillation loss. α is the hyperparameter to balance both.

$$\mathcal{L}_{total} = \alpha * \mathcal{L}_{SupCon} + (1 - \alpha) * \mathcal{L}_{dis} \qquad (3)$$

4.1. Supervised Contrastive Learning

In contrastive learning, the samples from the same class (positive sets) are pushed closer in feature space and vice versa. Normally, contrastive learning is applied in a self-supervised fashion, in which the positive sets are formed using data augmentation or co-occurrence. Khosla et al. developed contrastive learning to a supervised manner so that the labels can guide the selection of positive and negative sets and have gained higher accuracy in image recognition tasks than cross-entropy loss [21]. The supervised contrastive loss function is shown in Equ. (4), in which $\mathbf{z}_i$ and $\mathbf{z}_p$ are positive feature pairs belonging to positive set $P(i)$, and $A(i)$ stands for the negative sets for $\mathbf{z}_i$. τ is the temperature scaling factor that can tune the discrimination between the positive and negative sets.

$$\mathcal{L}_{SupCon} = -\sum_{i \in I} \frac{1}{|P(i)|} \sum_{p \in P(i)} \log \frac{\exp(\mathbf{z}_i \cdot \mathbf{z}_p / \tau)}{\sum_{a \in A(i)} \exp(\mathbf{z}_i \cdot \mathbf{z}_a / \tau)} \qquad (4)$$

In order to prove its superiority for satisfying Assumption 1 and 2 as cross entropy loss, we compare their R_s through image recognition task on *Cifar-100* dataset [24]. It can be found from Fig. 2 that the R_s in the models trained using supervised contrastive loss are much lower than those trained using cross entropy loss no matter how many classes are. Meanwhile, as reported in [46], the cross entropy loss has less capability in reducing intra-class feature scattering than contrastive learning that can increase S_{intra}. Considering all the above, we adopt supervised contrastive learning for our approach.

4.2. Relation-based Knowledge Distillation

As introduced above, relation-based knowledge distillation is better for preserving feature relations. Inspired by [28], we utilize a temporal angle-wise and distance-wise distillation method in this paper to transfer the structural knowledge between the models before and after incremental training.

The angle-wise similarity, ψ_A, is illustrated in Equ. (5), in which $\mathbf{z}_i, \mathbf{z}_j, \mathbf{z}_k$ represent a triplet of features. Let $\psi_A(\mathbf{z}_i^{t-1}, \mathbf{z}_j^{t-1}, \mathbf{z}_k^{t-1})$ and $\psi_A(\mathbf{z}_i^t, \mathbf{z}_j^t, \mathbf{z}_k^t)$ represent the angle-wise feature similarities given by the models in training session $t-1$ and t respectively (see Equ. (5)), the temporal angle-wise distillation loss is in Equ. (6). N stands for the number of exemplars (see 4.3.1 for exemplar management). The angle-wise distillation loss is finally composed using the sum over the L_2 norms of the angle-wise similarity changes of all triples between the training sessions of all the exemplars.

$$\psi_A(\mathbf{z}_i, \mathbf{z}_j, \mathbf{z}_k) = \cos \angle \mathbf{z}_i \mathbf{z}_j \mathbf{z}_k = \langle \mathbf{e}^{ij}, \mathbf{e}^{kj} \rangle,$$
$$\mathbf{e}^{ij} = \frac{\mathbf{z}_i - \mathbf{z}_j}{\|\mathbf{z}_i - \mathbf{z}_j\|_2}, \mathbf{e}^{kj} = \frac{\mathbf{z}_k - \mathbf{z}_j}{\|\mathbf{z}_k - \mathbf{z}_j\|_2} \qquad (5)$$

$$\mathcal{L}_{dis-A} = \sum_{i,j,k \in N} \|\psi_A(\mathbf{z}_i^{t-1}, \mathbf{z}_j^{t-1}, \mathbf{z}_k^{t-1}) - \psi_A(\mathbf{z}_i^t, \mathbf{z}_j^t, \mathbf{z}_k^t)\|_2 \qquad (6)$$

Besides, distance-wise distillation in Equ. (7) is also applied to enforce the transfer of structural knowledge that aims to penalize distance chances between the corresponding sample pairs. ψ_D is *Euclidean* distance in our settings.

$$\mathcal{L}_{dis-D} = \sum_{i,j \in N} \|\psi_D(\mathbf{z}_i^{t-1}, \mathbf{z}_j^{t-1}), \psi_D(\mathbf{z}_i^t, \mathbf{z}_j^t))\|_2 \qquad (7)$$

The distillation loss is the sum of $\mathcal{L}_{dis-A}$ and $\mathcal{L}_{dis-D}$ as in Equ. (8). λ_{dis} stands for a hyperparameter to balance $\mathcal{L}_{dis-A}$ and $\mathcal{L}_{dis-D}$.

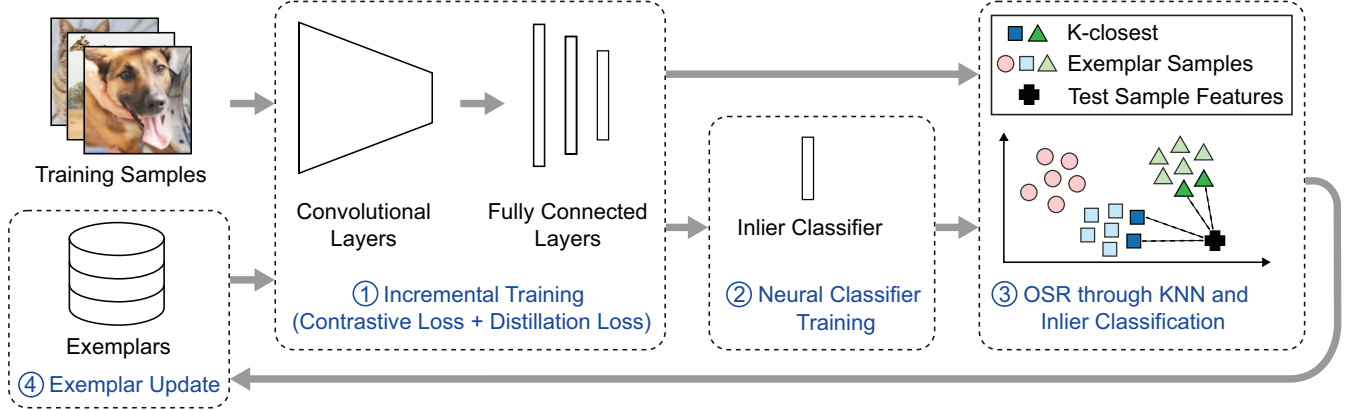$$\mathcal{L}_{dis} = \mathcal{L}_{dis-A} + \lambda_{dis} \cdot \mathcal{L}_{dis-D} \qquad (8)$$

Figure 1: Illustration of our proposed framework. 1. A CNN encoder (backbone) is continuously trained with supervised contrastive learning and relation-based knowledge distillation. 2. A neural classifier after that is then trained with cross-entropy loss. 3. During testing, the features of the testing samples and exemplars given by the backbone are compared and outliers are detected using K nearest neighbor method. The inliers are classified using the neural classifier. 4. Exemplars are updated after each training session. Better viewed in color.
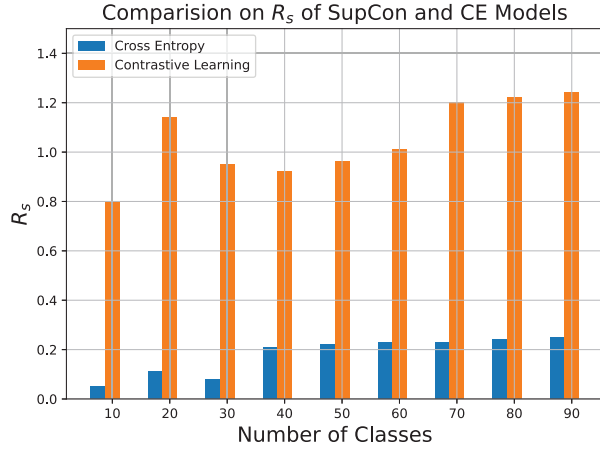

Figure 2: R_s of the feature maps extracted from *ResNet-18* trained on *Cifar-100* using supervised contrastive loss and cross entropy loss respectively.

4.3. OpenIncrement Framework

The incremental learning method proposed in this paper belongs to rehearsal-based approaches. Exemplars from the observed classes should be selected and stored. We name our method *OpenIncrementNN* (*OpenIncrement Nearest Neighbor*).

4.3.1 Exemplar Management

In order to select exemplars that are representative for each class, we propose an *Isometric Sampling* method as shown in Alg. 1[1]. When the new task T comes, a portion of the exemplars of old classes are randomly removed if the memory is fixed. For the features $\mathbf{F}_c$ belonging to the newly observed classes, their centers $\mu_{\mathbf{c}}$ are firstly derived. For each feature point $\mathbf{z}_n$ in $\mathbf{F}_c$, their Euclidean distances to $\mathbf{z}_n$, $\mathfrak{E}(\mu_c, \mathbf{z}_n)$, are computed and sorted. Then the samples are selected isometrically according to the distances to their class centers.

4.3.2 Unified Classification for In- and Outliers

The testing phase consists of two parts, namely open set recognition and inlier classification as illustrated in Fig. 1. The set of features of the exemplars in each observed class c extracted by the backbone, $\mathbf{Z}_{exem}^{C} = \{\mathbf{z}_{exem,0}^c, \mathbf{z}_{exem,1}^c, ..., \mathbf{z}_{exem,N_c}^c\}$, is first stored. For each testing sample feature $\mathbf{z}_i$, we search for its K-nearest neighbors, $\mathbf{Z}_K^{c,i}$, in $\mathbf{Z}_{exem}^c$ with cosine similarity, which is denoted using $S(\cdot)$. We use $\mathbf{S}_K^{c,i}$ to denote the set of K similarities between $\mathbf{z}_i$ and $\mathbf{Z}_{exem}^c$. The overall set of K similarities between $\mathbf{z}_i$ and exemplar sets of all classes is $\{\mathbf{S}_K^{0,i}, \mathbf{S}_K^{1,i}, ..., \mathbf{S}_K^{C,i}\}$. The score, sc_{osr}, for discriminating outliers is given in Equ. (9), in which each $\mathbf{S}_K^{c,i}$ is first normalized and its maximum is selected as sc_{osr}.

$$sc_{osr} = \arg\max_c \frac{\mathbf{S}_K^{c,i}}{\sum_c \mathbf{S}_K^{c,i}} \qquad (9)$$

sc_{osr} is then compared with a pre-defined threshold τ_{osr}. If sc_{osr} is smaller than τ_{osr}, it is an outlier, otherwise inlier.

We train a neural classifier inputted by the backbone features, $\mathbf{Z}_{exem}^C$, for inlier classification. The classifier consists

---

[1]All the pseudo-code semantically follows *python* and *Numpy* style in this paper

Algorithm 1 Isometric Sampling for Selecting Exemplars

1: **Input**: Task T with C_{new} new classes and C_{old} old classes, class centers of the old classes $\{\mathfrak{c}_1, \mathfrak{c}_1, ..., \mathfrak{c}_{C_{old}}\}$, samples in new classes $\mathbf{X}_{new}$ and their features $\mathbf{F}_{new}$, rehearsal memory size R, exemplar set for old classes $\mathbf{E} = [\mathbf{E}_1, \mathbf{E}_2, ..., \mathbf{E}_{C_{old}}]$ and their features $\mathbf{F}_{old}$.

2: **Output**: Updated exemplars set $\mathbf{E}'$ for C_{new} and C_{old}.

3: **Initialize**: $r = R/C_{old}$, $r' = R/(C_{new} + C_{old})$, $\mathbf{E}' \leftarrow \emptyset$.

Update exemplars for old classes for fixed memory:
4: **for** $c = 1, 2, ..., C_{old}$ **do**
5:    $\mathbf{I} = random.sample(range(r), r')$
6:    $\mathbf{E}' \leftarrow \mathbf{E}' \cup \mathbf{E}_c[\mathbf{I}]$
7: **end for**

Select exemplars for new classes:
8: Isometric sampling the indices $\mathbf{i} = range(0, N, r')$
9: **for** $c = 1, 2, ..., C_{new}$ **do**
10:    Read features belonging to class c with N instances, $\mathbf{F}_c \subset \mathbf{F}_{new}$, $\mathbf{F}_c = [\mathbf{z}_1, \mathbf{z}_2, ..., \mathbf{z}_N]$
11:    Class center $\mu_{\mathfrak{c}} = \frac{1}{N} \sum_{i \in N} \mathbf{z}_n$
12:    Initialize distance set $\mathbf{D} \leftarrow \emptyset$
13:    **for** $\mathbf{z}_n \in \mathbf{F}_c$ **do**
14:       $\mathbf{D} \leftarrow \mathbf{D} \cup \mathfrak{E}(\mu_{\mathfrak{c}}, \mathbf{z}_n)$
15:    **end for**
16:    Sort $\mathbf{D}$ in ascending order and get the indices $\mathbf{I} \leftarrow \mathbf{argsort}(\mathbf{D})$
17:    $\mathbf{I} \leftarrow \mathbf{I}[:, :, N//r']$
18:    $\mathbf{E}' \leftarrow \mathbf{E}' \cup \mathbf{X}_{new}[\mathbf{I}]$
19: **end for**

---

of one fully-connected layer. In order to prevent data imbalance between old and new classes, we use exemplars of the new classes instead of the full dataset to train the classifier.

5. Experiments

We test our method for incremental learning and open set recognition on image recognition tasks. The code is open source online [2].

5.1. Settings

5.1.1 Datasets

We test our method on two open source datasets for image recognition, namely *Cifar-100* [24] and *Tiny ImageNet* [34]. *Cifar-100* contains 100 classes of natural color images belonging to 20 sub-classes and there are 600 images

Datasets	♯ Classes per Session/♯ Sessions
CIFAR-100	10/10
Tiny ImageNet	20/10

Table 2: Class split settings of each dataset for class-incremental learning

in each class. *Tiny ImageNet* is the subset of *ImageNet* [34] and consists of 200 classes of natural images. We split the above two datasets into multiple non-overlapping sessions for different continual learning tasks and to act as inliers and outliers. The number of classes in each session is shown in Tab. 2. The classes in the last session of each dataset are outliers. Datasets are augmented before training. The details of data augmentation are in Appendix A.2.

5.1.2 Network Architecture

For all experiments, we use *ResNet-18* [14] (residual neural network) as the backbone model. The last fully-connected layer that outputs class probability is replaced by two fully connected layers, which are the *head* in [21]. *ResNet-18* is a convolutional neural network and was originally used for image recognition. The output dimension of the last layer is 512. As introduced above, the neural classifier for inliers is one fully-connected layer.

5.1.3 Training

We conduct all experiments under the class-incremental learning settings. Following the class split protocol introduced in 5.1.1, the models were trained in a sequential way, i.e., session after session. The backbones are trained using *Adam* optimizer and the classifiers are trained with stochastic gradient descent (SGD) strategy. The model hyperparameters were selected using *grid search* and their search space and final configurations are shown in Appendix A.3. The baselines in 5.1.4 are trained using the configurations in the original paper if they are available.

5.1.4 Baselines

We select the state-of-the-art incremental learning methods as our baselines for inlier classification, namely *SI*[45], *EWC*[23], *LwF*[25], *ER*[32], *iCaRL*[31], *LUCIR*[17], and *Der*[4]. As mentioned in 2.2, *SI*, *EWC*, and *LwF* are parameter-based methods. *ER*, *iCaRL*, *LUCIR*, and *Der* are replay-based methods.

For open set recognition, since there is no existing work or benchmarks on the same topic in literature, we take the three settings listed in Tab. 5 as baselines.

Methods	CIFAR-100		Tiny-ImageNet	
M_{size}	500	2000	2000	5000
Joint	74.44		53.55	
SI [45]	17.26		6.58	
EWC [23]	23.1		7.58	
LwF [25]	16.22		8.46	
ER [32]	22.10	38.58	12.14	27.2
iCaRL [31]	46.52	49.82	13.38	13.98
LUCIR [17]	40.59	41.73	14.97	17.61
Der [4]	36.60	51.89	34.75	36.73
Supcon Full	75.46		55.34	
CE+ResKD	18.63	41.91	12.3	14.5
CE+RKD	32.57	44.37	14.22	15.23
DeepIncrement (**ours**)	**49.95**	**63.73**	**33.65**	**37.76**

Table 3: Average accuracy (in %) for inlier classification of class-incremental learning. Some results are from [4] and [3]. Bold indicates the best results except *Joint*.

Methods	CIFAR-100		Tiny-ImageNet	
M_{size}	500	2000	2000	5000
Supcon Full	74.51		62.54	
CE+ResKD	55.05	60.64	50.23	51.57
CE+RKD	56.55	59.7	50.98	51.58
DeepIncrement	**63.09**	**72.23**	**56.88**	**59.39**

Table 4: AUROC for open set recognition on *CIFAR-100* and *Tiny-ImageNet* datasets. M_{size} stands for the memory size. Bold indicates the best results except *Joint*.

Settings	Details
CE+ResKD	Cross-entropy loss with response-based knowledge distillation in [16]
CE+RKD	Cross-entropy loss with RKD
Supcon Full	Supervised contrastive learning trained in joint fashion

Table 5: Baselines for open set recognition

5.1.5 Evaluation Metrics

As in most works, we evaluate the incremental learning method for inlier classification using *accuracy* defined in Equ. 10. $C_{inliers}$ is the number of correctly classified inliers whereas $N_{inliers}$ is the total number of inlier testing samples.

$$A_{incremental} = \frac{C_{inliers}}{N_{inliers}} \qquad (10)$$

Since the approach for OSR in 4.3.2 requires setting the threshold manually, a direct result comparison with different thresholds is not reasonable. A threshold-independent metric, the *area under the receiver operating characteristic* (AUROC) curve [27] is taken as the metric to evaluate the OSR method. In the AUROC curve, the true positive rate is plotted against the false positive rate by varying the threshold. The higher AUROC is, the better the method can detect outliers (if outliers are assumed positive). When the AUROC value is 0.5, it is basically equivalent to tossing coins.

5.2. Results

5.2.1 Class-incremental learning

The experiment results for class-incremental learning are shown in Tab. 3. *Joint* stands for normal offline training with all class samples using *cross entropy* loss. Our method surpasses most of the state-of-the-art baselines. Compared with the three baselines proposed by us, besides *Supcon Full*, our method is much superior to the rest two. The re-

sults can also suggest that supervised contrastive learning is more robust in continual learning than cross entropy.

5.2.2 Open Set Recognition

The results of open set recognition are in Tab. 4. Similar to the results of the inlier classification, our method shows better performance than the two baselines. It should be noted that we use the exemplars to detect outliers even for *Supcon Full*. We have observed during conducting the experiments that the OSR performance also drops with the increase of training sessions, which is the same as for inlier classification (see the plots in Appendix B).

6. Conclusion & Future Work

In this work, we presented a unified framework for class-incremental learning and open set recognition (OSR). Our method achieves state-of-the-art inlier classification accuracy and excels over baseline models in OSR, yet there remains room for improvement.

The field is in its infancy, prompting several pivotal questions for further exploration:

While catastrophic forgetting seems to induce feature map distortion post-continual learning, the specific nature of forgotten class-specific features causing this distortion warrants investigation. It remains to be determined whether the features vital for OSR and inlier classification are congruent, and how continual learning impacts feature retention. Enhancing other continual learning methodologies for outlier detection is essential. Concurrently, strategies to embed continual learning within established OSR techniques, ensuring retained efficiency, should be explored. Despite the inherent synergy between open set recognition and continual learning, research intersections remain sparse. We urge a focused exploration in this domain, given its significance in evolving adaptive and autonomous machine learning systems.

References

[1] Aljundi, R.; Kelchtermans, K.; Tuytelaars, T. Task-free continual learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR'19)*, pages 11254–11263, Long Beach, USA, Jun. 2019.

[2] Bendale, A.; Boult, T.E. Towards open set deep networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR'16)*, pages 1563–1572, Las Vegas, USA, Jun. 2016.

[3] Boschini, M.; Bonicelli, L.; Buzzega, P.; Porrello, A.; Calderara, S. Class-incremental continual learning into the extended der-verse. *arXiv preprint arXiv:2201.00766*, 2022.

[4] Buzzega, P.; Boschini, M.; Porrello, A.; Abati, D.; Calderara, S. Dark experience for general continual learning: a strong, simple baseline. In Larochelle, H.; Ranzato, M.; Hadsell, R.;

[5] Cao, A.; Luo, Y.; Klabjan, D. Open-set recognition with gaussian mixture variational autoencoders. In *Thirty-Fifth AAAI Conference on Artificial Intelligence (AAAI'21)*, Virtual, Feb. 2021.

[6] Castro, F.M.; Marín-Jiménez, M.J.; Guil, N.; Schmid, C.; Alahari, K. End-to-end incremental learning. In *Proceedings of the European Conference on Computer Vision (ECCV'18)*, pages 233–248, Munich,Germany, Sep, 2018.

[7] Cha, H.; Lee, J.; Shin, J. Co2l: Contrastive continual learning. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (CVPR'21)*, pages 9516–9525, Virtual, Jun. 2021.

[8] Chen, G.; Peng, P.; Wang, X.; Tian, Y. Adversarial reciprocal points learning for open set recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 44(11):8065–8081, 2022.

[9] Cohen, A. *Numerical analysis of wavelet methods*. Elsevier, 2003.

[10] Dhamija, A.R.; Günther, M.; Boult, T.E. Reducing network agnostophobia. In *Thirty-second Annual Conference on Neural Information Processing Systems (NIPS'18)*, Montréal, Cananda, Dec 2018.

[11] Do, H.H.; Prasad, P.; Maag, A.; Alsadoon, A. Deep learning for aspect-based sentiment analysis: a comparative review. *Expert systems with applications*, 118:272–299, 2019.

[12] Ge, Z.; Demyanov, S.; Garnavi, R. Generative openmax for multi-class open set classification. In *28th British Machine Vision Conference (BMVC'17)*, London, UK, Sep. 2017.

[13] Hassen, M.; Chan, P.K. Learning a neural-network-based representation for open set recognition. In *Proceedings of the 2020 SIAM International Conference on Data Mining (ICDM'20)*, pages 154–162, Sorrento, Italy, Nov 2020.

[14] He, K.; Zhang, X.; Ren, S.; Sun, J. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, Las Vegas, US, Jun. 2016.

[15] Hendrycks, D.; Mazeika, M.; Dietterich, T. Deep anomaly detection with outlier exposure. *Proceedings of the International Conference on Learning Representations (ICLR'19)*, May 2019.

[16] Hinton, G.; Vinyals, O.; Dean, J. Distilling the knowledge in a neural network. *arXiv preprint arXiv:1503.02531*, 2015.

[17] Hou, S.; Pan, X.; Loy, C.C.; Wang, Z.; Lin, D. Learning a unified classifier incrementally via rebalancing. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR'19)*, Long Bench, US, Jun. 2019.

[18] Jia, J.; Chan, P.K. Mmf: A loss extension for feature learning in open set recognition. In *In Proceeding 30th International Conference on Artificial Neural Networks (ICANN'21)*, pages 319–331, Bratislava, Slovak, Sep 2021.

[19] Jin, X.; Sadhu, A.; Du, J.; Ren, X. Gradient-based editing of memory examples for online task-free continual learning. *Advances in Neural Information Processing Systems (NIPS'21)*, pages 29193–29205, Dec 2021.

Balcan, M.F.; Lin, H., editors, *Advances in Neural Information Processing Systems (NIPS'20)*, pages 15920–15930, Virtual, Dec 2020.

[20] Joseph, K.J.; Khan, S.; Khan, F.S.; Balasubramanian, V.N. Towards open world object detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR'21)*, Virtual, Jun, 2021.

[21] Khosla, P.; Teterwak, P.; Wang, C.; Sarna, A.; Tian, Y.; Isola, P.; Maschinot, A.; Liu, C.; Krishnan, D. Supervised contrastive learning. *arXiv preprint arXiv:2004.11362*, 2020.

[22] Kim, J.; Park, S.; Kwak, N. Paraphrasing complex network: Network compression via factor transfer. *arXiv preprint arXiv:1802.04977*, 2018.

[23] Kirkpatrick, J.; Pascanu, R.; Rabinowitz, N.; Veness, J.; Desjardins, G.; Rusu, A.A.; Milan, K.; Quan, J.; Ramalho, T.; Grabska-Barwinska, A.; et al. Overcoming catastrophic forgetting in neural networks. *Proceedings of the national academy of sciences*, 114(13):3521–3526, 2017.

[24] Krizhevsky, A.; Nair, V.; Hinton, G. Cifar-100 (canadian institute for advanced research).

[25] Li, Z.; Hoiem, D. Learning without forgetting. *IEEE transactions on pattern analysis and machine intelligence*, 40(12):2935–2947, 2017.

[26] Miller, D.; Sunderhauf, N.; Milford, M.; Dayoub, F. Class anchor clustering: A loss for distance-based open set recognition. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision (WACV'21)*, pages 3570–3578, Virtual, Jan. 2021.

[27] Narkhede, S. Understanding auc-roc curve. *Towards Data Science*, 26(1):220–227, 2018.

[28] Park, W.; Kim, D.; Lu, Y.; Cho, M. Relational knowledge distillation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR'19)*, pages 3967–3976, Long Beach, USA, Jun. 2019.

[29] Passban, P.; Wu, Y.; Rezagholizadeh, M.; Liu, Q. Alp-kd: Attention-based layer projection for knowledge distillation. *arXiv preprint arXiv:2012.14022*, 2020.

[30] Perera, P.; Morariu, V.I.; Jain, R.; Manjunatha, V.; Wigington, C.; Ordonez, V.; Patel, V.M. Generative-discriminative feature representations for open-set recognition. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR'20)*, pages 11814–11823, Virtual, Jun. 2020.

[31] Rebuffi, S.A.; Kolesnikov, A.; Sperl, G.; Lampert, C.H. icarl: Incremental classifier and representation learning. In *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition (CVPR'17)*, pages 2001–2010, Hawaii, Jun. 2017.

[32] Rolnick, D.; Ahuja, A.; Schwarz, J.; Lillicrap, T.; Wayne, G. Experience replay for continual learning. In *Advances in Neural Information Processing Systems (NIPS'19)*, Vancouver, Canada, Dec 2019.

[33] Romero, A.; Ballas, N.; Kahou, S.E.; Chassang, A.; Gatta, C.; Bengio, Y. Fitnets: Hints for thin deep nets. *arXiv preprint arXiv:1412.6550*, 2014.

[34] Russakovsky, O.; Deng, J.; Su, H.; Krause, J.; Satheesh, S.; Ma, S.; Huang, Z.; Karpathy, A.; Khosla, A.; Bernstein, M.; et al. ImageNet Large Scale Visual Recognition Challenge. *International Journal of Computer Vision (IJCV)*, 115(3):211–252, 2015.

[35] Saito, K.; Kim, D.; Saenko, K. Openmatch: Open-set consistency regularization for semi-supervised learning with outliers. *ArXiv*, abs/2105.14148, 2021.

[36] Sun, X.; Ding, H.; Zhang, C.; Lin, G.; Ling, K.V. M2iosr: Maximal mutual information open set recognition. *arXiv preprint arXiv:2108.02373*, 2021.

[37] Sun, X.; Zhang, C.; Lin, G.; Ling, K.V. Open set recognition with conditional probabilistic generative models. *arXiv preprint arXiv:2008.05129*, 2020.

[38] Tian, Y.; Krishnan, D.; Isola, P. Contrastive representation distillation. *arXiv preprint arXiv:1910.10699*, 2019.

[39] Tung, F.; Mori, G. Similarity-preserving knowledge distillation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (CVPR'19)*, pages 1365–1374, Long Bench, US, Jun. 2019.

[40] Wu, Y.; Chen, Y.; Wang, L.; Ye, Y.; Liu, Z.; Guo, Y.; Fu, Y. Large scale incremental learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR'19)*, pages 374–382, Long Bench, US, Jun. 2019.

[41] Yang, H.M.; Zhang, X.Y.; Yin, F.; Yang, Q.; Liu, C.L. Convolutional prototype network for open set recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, 44(5):2358–2370, 2020.

[42] Ye, F.; Bors, A.G. Task-free continual learning via online discrepancy distance learning. *Advances in Neural Information Processing Systems (NIPS'22)*, pages 23675–23688, Dec. 2022.

[43] Yoshihashi, R.; Shao, W.; Kawakami, R.; You, S.; Iida, M.; Naemura, T. Classification-reconstruction learning for open-set recognition. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR'19)*, pages 4016–4025, Long Bench, US, Jun. 2019.

[44] Yu, Y.; Qu, W.; Li, N.; Guo, Z. Open-category classification by adversarial sample generation. In *26th International Joint Conference on Artificial Intelligence (IJCAI'17)*, Melbourne, Australia, Aug. 2017.

[45] Zenke, F.; Poole, B.; Ganguli, S. Continual learning through synaptic intelligence. In *Proceedings of the International Conference on Machine Learning (ICML'17)*, pages 3987–3995, Sydney, Australia, Aug 2017.

[46] Zhang, Y.; Hooi, B.; Hu, D.; Liang, J.; Feng, J. Unleashing the power of contrastive self-supervised visual models via contrast-regularized fine-tuning. In *Proceedings of the Advances in Neural Information Processing Systems (NIPS'21)*, pages 29848–29860, Virtual, Dec. 2021.

[47] Zhao, B.; Xiao, X.; Gan, G.; Zhang, B.; Xia, S.T. Maintaining discrimination and fairness in class incremental learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR'20)*, pages 13208–13217, Virtual, Jun. 2020.

[48] Zhao, Z.Q.; Zheng, P.; Xu, S.t.; Wu, X. Object detection with deep learning: A review. *IEEE Transactions on Neural Networks and Learning Systems*, 30(11):3212–3232, 2019.