Rebalancing Multi-Label Class-Incremental Learning

Kaile Du[1,2*], **Yifan Zhou**[1,2*], **Fan Lyu**[3], **Yuyang Li**[1], **Junzhou Xie**[1],
Yixi Shen[4], **Fuyuan Hu**[4,5,6], **Guangcan Liu**[1†]

[1]School of Automation, Southeast University, China
[2]Key Laboratory of Measurement and Control of CSE, Ministry of Education, China
[3]NLPR, MAIS, CASIA, China
[4]School of Electronic and Information Engineering, Suzhou University of Science and Technology, China
[5]Suzhou Key Laboratory of Intelligent Low Carbon Technology Application, China
[6]Jiangsu Industrial Intelligent Low Carbon Technology Engineering Center, China

Abstract

Multi-label class-incremental learning (MLCIL) is essential for real-world multi-label applications, allowing models to learn new labels while retaining previously learned knowledge continuously. However, recent MLCIL approaches can only achieve suboptimal performance due to the oversight of the positive-negative imbalance problem, which manifests at both the label and loss levels because of the task-level partial label issue. The imbalance at the label level arises from the substantial absence of negative labels, while the imbalance at the loss level stems from the asymmetric contributions of the positive and negative loss parts to the optimization. To address the issue above, we propose a Rebalance framework for both the Loss and Label levels (RebLL), which integrates two key modules: asymmetric knowledge distillation (AKD) and online relabeling (OR). AKD is proposed to rebalance at the loss level by emphasizing the negative label learning in classification loss and down-weighting the contribution of overconfident predictions in distillation loss. OR is designed for label rebalance, which restores the original class distribution in memory by online relabeling the missing classes. Our comprehensive experiments on the PASCAL VOC and MS-COCO datasets demonstrate that this rebalancing strategy significantly improves performance, achieving new state-of-the-art results even with a vanilla CNN backbone.

Code — https://github.com/Kaile-Du/RebLL

Introduction

Class-incremental learning (CIL) (Buzzega et al. 2020; Douillard et al. 2020; Wang et al. 2024; Yang et al. 2023) continuously learn new classes without the need to retrain on the entire dataset. However, as the model learns new classes, it may overwrite the knowledge of previous tasks, resulting in a significant decline in performance on older classes, a phenomenon known as catastrophic forgetting (McCloskey and Cohen 1989). While numerous methods have been developed for single-label class-incremental learning (SLCIL) (Sun et al. 2022; Wang et al. 2022; Lyu et al. 2023, 2024),
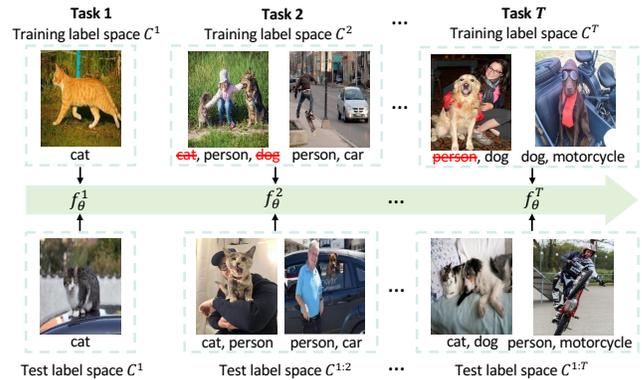
Figure 1: A diagram of multi-label class-incremental learning. Labels are trained and tested across tasks from Task 1 to Task T. Missing labels are highlighted in red. The training label space is task-specific, while the testing label space progressively expands with the addition of each new task.

there has been limited exploration into the more practical domain of multi-label class-incremental learning (MLCIL) (Dong et al. 2023; Du et al. 2024b). In MLCIL, training images for new classes are partially labeled to minimize the cost of manual annotation. Specifically, only the classes for the new task are labeled, while the past and future ones are not annotated, a phenomenon called task-level partial label (PL) (Du et al. 2024b) or category-incomplete (Dong et al. 2023) issue. For example, as shown in Figure 1, there is a training image of Task 2 containing the labels "cat", "person" and "dog", only the labels for the current task ("person") are annotated, while the labels for past ("cat") and future ("dog") tasks remain unannotated.

However, most recent MLCIL methods focus on designing new network architectures for MLCIL to fit the PL setting, ignoring the positive-negative imbalance issue. AGCN (Du et al. 2022, 2024a) utilizes pseudo-labels to construct statistical label relationships within the Graph Convolutional Network (GCN). KRT (Dong et al. 2023) is the knowledge restoration and transfer framework based on a cross-attention mechanism (CAM). CSC (Du et al. 2024b) leverages a learnable GCN to calibrate label relationships. In

fact, due to the larger number of negative labels compared to positive labels in multi-label samples (Ridnik et al. 2021a; Zhou et al. 2022; Xie et al. 2024), missing labels for both past and future tasks are prone to be labeled negative in MLCIL. A training example of task T is shown in Figure 1, it only lacks one positive label "person" (in red), while also missing many negative labels ("cat...car") from other tasks. The substantial absence of negative labels (more than 9 times that of positive) results in a positive-negative imbalance at the label level. In some cases, the model may only learn the positive labels. *Learning on such imbalance-labeled data poses the risk of collapsing to a trivial solution*, i.e., the model tends to predict each label as positive (Zhou et al. 2022; Liu et al. 2023), thus leading to numerous false positive errors during loss optimization. This positive-negative imbalance manifests at the loss level due to the asymmetric contributions of the positive and negative loss parts to the optimization objectives.

In this paper, we explore tackling the positive-negative imbalance issue in MLCIL from two key aspects: loss level and label level. The positive-negative imbalance arises at the *loss level* of recent MLCIL methods (Du et al. 2024a,b). Specifically, they utilize binary cross-entropy (BCE) loss and binary knowledge distillation (KD) loss for classification and mitigating forgetting. The loss imbalance during the optimization process fundamentally stems from the equal treatment of imbalanced positive and negative loss components by the BCE and KD losses. As a result, the contribution of positive labels to the loss is significantly greater than that of negative labels. This situation is suboptimal, as it leads to the accumulation of more loss gradients from positive labels due to the positive-negative imbalance (Ridnik et al. 2021a), resulting in insufficient learning of negative labels at the loss level. Due to the substantial absence of negative labels, the positive-negative imbalance also occurs at the *label level*. We aim to address this issue using a replay-based approach. In our further attempts to mitigate catastrophic forgetting, we observe that replay methods (Rolnick et al. 2019; Douillard et al. 2020; Buzzega et al. 2020), commonly used in single-label scenarios, exacerbate the label-level imbalance due to memory sampling of imbalance-labeled data. This exacerbation leads to an increase in false positive errors. Consequently, this label-level imbalance causes these methods to suboptimally address catastrophic forgetting.

To address the issues above, we first propose asymmetric knowledge distillation (AKD) for loss rebalance. AKD emphasizes the learning of negative labels in the classification loss for the current model and down-weights the contribution of overconfident old task predictions in the KD process. Then, we design the online relabeling (OR) strategy for label rebalance by continuously restoring the original class distribution in memory. Specifically, we relabel the missing old labels of the memory samples using the past model, while the missing new labels in memory are labeled with the trained current model. **Reb**alancing at both the **L**oss and **L**abel levels, forming our **RebLL** framework. Our framework can mitigate the positive-negative imbalance and enhance MLCIL performance. Under the PL setting, our contributions are summarized below:

- We are the first to identify the inherent positive-negative imbalance in MLCIL under the PL setting. This imbalance, occurring at both the loss and label levels, impedes the effective learning of MLCIL.

- We propose a RebLL framework to suppress the positive-negative imbalance, consisting of AKD and OR components. AKD is designed for loss rebalance, while OR targets label rebalance. RebLL can effectively mitigate forgetting and enhance MLCIL performance by suppressing the positive-negative imbalance.

- Extensive experiments conducted across multiple MLCIL scenarios using the PASCAL VOC and MS-COCO demonstrate that RebLL achieves new SOTA results, regardless of whether we use a vanilla CNN or a more powerful network architecture as the backbone.

Related Work

Single-Label Class-Incremental Learning

Significant progress has been made in single-label class-incremental learning in recent years. Regularization-based methods (Kirkpatrick et al. 2017; Li and Hoiem 2017; Zhao et al. 2023; Mohamed et al. 2023) impose constraints on the loss function to preserve old knowledge from being overwritten by new one. For instance, EWC (Kirkpatrick et al. 2017) employs the Fisher matrix to retain the critical parameters associated with earlier tasks. LwF (Li and Hoiem 2017) employs knowledge distillation to transfer old knowledge to new tasks. Replay-based methods (Rolnick et al. 2019; De Lange and Tuytelaars 2021; Ye and Bors 2022; Cha et al. 2023; Luo et al. 2023; Lyu et al. 2021) enhance the learning of new data by incorporating a portion of old data, thereby reinforcing previously acquired knowledge. Experience replay (Rolnick et al. 2019) randomly samples a subset of sample from the old task dataset to serve as memory data. DER++ (Buzzega et al. 2020) combines replay with knowledge distillation (Jia et al. 2024; Guo et al. 2023; Zeng et al. 2024) and regularization techniques. Architecture-based approaches assign distinct parameters to each task to save the corresponding knowledge. Some architecture-based methods decompose the model into task-sharing and task-specific components (Douillard et al. 2022; Wu et al. 2021). Furthermore, L2P (Wang et al. 2022) introduces prompt-driven learning into the domain of class-incremental learning, yielding encouraging results by leveraging a pre-trained ViT model (Dosovitskiy et al. 2021).

Multi-Label Class-Incremental Learning

Multi-label class-incremental learning is an emerging field. PRS (Kim, Jeong, and Kim 2020) and OCDM (Liang and Li 2022) introduce sampling approaches designed for replay methods, aimed at alleviating the effects of multi-label long-tail class distributions. The methods above differ significantly from recent MLCIL approaches (Du et al. 2024a; Dong et al. 2023; Du et al. 2024b) in both benchmarks and settings. AGCN (Du et al. 2024a) utilizes pseudo-labels to construct statistical label relationships within the GCN to enhance MLCIL performance. Meanwhile, KRT (Dong
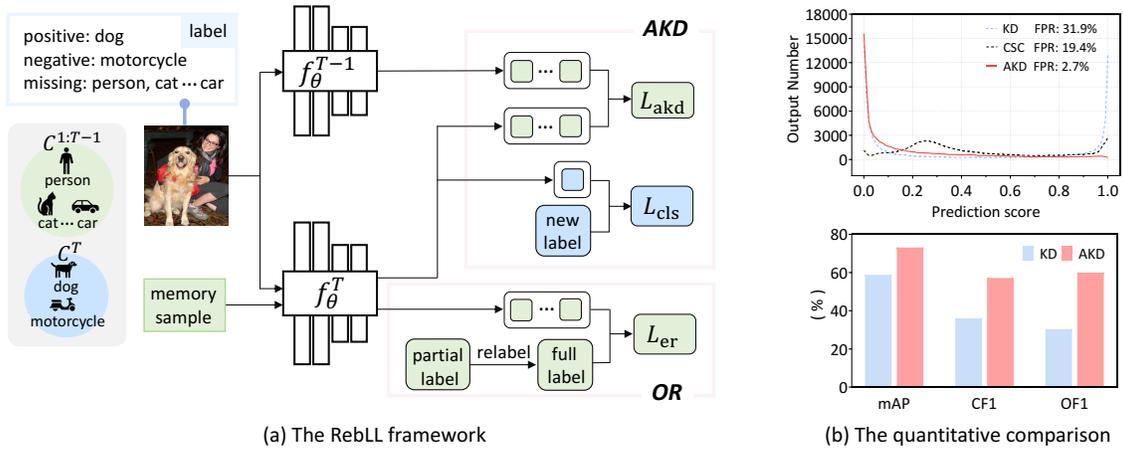
(a) The RebLL framework   (b) The quantitative comparison

Figure 2: (a) The RebLL framework consists of two modules: AKD and OR. In AKD, a training image is fed into both old (f_θ^{T-1}) and new (f_θ^T) models, where the output old task predictions (in green) are used to compute L_{akd}. The new task predictions (in blue) are compared with the new label to compute L_{cls}. In OR, the partially labeled memory sample is relabeled to form the full label, which is then used in conjunction with the predictions to compute L_{er}. (b) The quantitative comparison between KD, CSC and AKD after training on the final task in {B4-C2} of VOC 2007.

et al. 2023) is the knowledge restoration and transfer framework based on a cross-attention mechanism to address the category-incomplete issue. CSC (Du et al. 2024b) leverages a learnable GCN to calibrate label relationships and employs maximum entropy to adjust overconfident predictions. Notably, CSC is the SOTA method. It can be observed that AGCN, KRT, and CSC all utilize powerful network architectures as backbones, while our approach, by rebalancing the inherent positive-negative imbalance in MLCIL, allows us to surpass them using only a vanilla CNN.

Method

Preliminary

MLCIL. Following previous works (Du et al. 2024a; Dong et al. 2023; Du et al. 2024b), given T multi-label learning tasks, and the corresponding training sets for these tasks are denoted as $\{\mathcal{D}_{trn}^1, \cdots, \mathcal{D}_{trn}^T\}$, and the testing sets are represented as $\{\mathcal{D}_{tst}^1, \cdots, \mathcal{D}_{tst}^T\}$. For each incremental state t, the training dataset is defined as $\mathcal{D}_{trn}^t$. $\mathcal{C}^t$ denotes the current task-specific class set, while the past class collection is denoted by $\mathcal{C}^{1:t-1} = \bigcup_{i=1}^{t-1} \mathcal{C}^i$. Moreover, $\mathcal{C}^{1:t} = \mathcal{C}^{1:t-1} \cup \mathcal{C}^t$, and it holds that $\mathcal{C}^{1:t-1} \cap \mathcal{C}^t = \varnothing$. The goal of MLCIL is for the model to recognize all previously learned labels in a multi-label image after sequentially training task-specific labels. Hence, for the t-th task, the training label space $\mathcal{Y}_{trn}^t = \mathcal{C}^t$, the testing label space $\mathcal{Y}_{tst}^t = \mathcal{C}^{1:t}$.

As shown in Figure 2 (a), similar to distillation-based MLCIL methods (Du et al. 2024a; Dong et al. 2023; Du et al. 2024b), we maintain a fixed old model f_θ^{T-1} and a trainable new model f_θ^T. In MLCIL, the inherent positive-negative imbalance manifests at loss and label levels. To address this issue, we propose the RebLL framework, which includes asymmetric knowledge distillation (AKD) for loss rebalance and online relabeling (OR) for label rebalance.

Imbalance in Loss. Recent MLCIL (Du et al. 2024a,b) methods utilize BCE and KD for classification and mitigating forgetting. The BCE loss decouples the multi-label classification tasks into multiple binary issues, formulated as:

$$L_{bce} = \sum_{c \in \mathcal{C}^t} -y_c^t \log(\hat{y}_c^t) - (1 - y_c^t) \log(1 - \hat{y}_c^t), \quad (1)$$

where $\hat{y}_c^t$ and y_c^t represents the predcitions and ground-truth for class c and task t. KD loss uses the last task prediction $\hat{y}_c^{t-1}$ as supervisory information to guide the learning of current task to suppress catastrophic forgetting, formulated as:

$$L_{kd} = \sum_{c \in \mathcal{C}^{1:t-1}} -\hat{y}_c^{t-1} \log(\hat{y}_c^t) - (1 - \hat{y}_c^{t-1}) \log(1 - \hat{y}_c^t). \quad (2)$$

To clarify how positive and negative labels are operated in L_{bce}, a general form of Eq. (1) can be given by: $L_{bce} = \sum -y_c^t L_+ - (1 - y_c^t) L_-$, L_+ and L_- can be regarded positive and negative loss parts, L_{kd} follows the same format. Both L_{bce} and L_{kd} treat positive and negative loss parts equivalently, making it unsuitable for MLCIL where negative labels are insufficiently learned using the imbalance-labeled data. This approach is suboptimal because it results in the accumulation of excessive loss gradients from positive samples under positive-negative imbalance (Ridnik et al. 2021a). At the loss level, as there is an asymmetric learning of positive and negative labels in MLCIL, the contribution of the negative loss part should be emphasized, while the positive loss part should be down-weighted.

Imbalance in Label. As shown in Figure 2 (a), given a multi-label image, its ground truth labels include a positive label "dog" and a negative label "motorcycle". In the PL setting, it is missing a positive label "person", while it lacks numerous negative labels, such as "cat...car". This asymmetric missing of positive and negative labels leads to a label-level imbalance in MLCIL. We aim to address this issue using a

Classical Replay (FPR: 6.7%)   **Online Relabeling (FPR: 2.4%)**

Task 1   Task 2   Task 3

■ Current task labels   ⬚ Missing labels   ▨ Past model relabel   ▦ Current model relabel
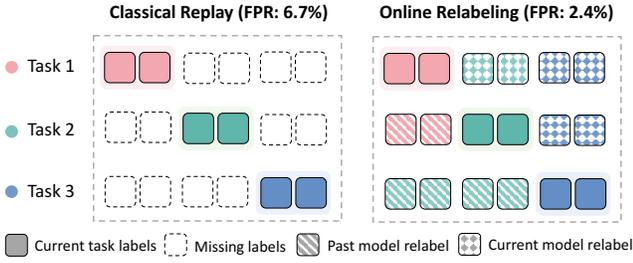
Figure 3: Online relabeling. For the label block matrix, missing new task labels above the main diagonal are relabeled using the trained current model, while the missing old task labels below the main diagonal are relabeled using the past model. This process reduces the FPR from 6.7% to 2.4%.

replay-based approach. Figure 3 illustrates that many future and past labels are missing in the memory data sampled by the classical replay method in SLCIL. This exacerbates the imbalance in label during subsequent replay. When applying the classical replay method with AKD, the false positive rate (FPR) increased from 2.7% to 6.7%.

AKD for Loss Rebalance

To rebalance the positive-negative at the loss level, we propose asymmetric knowledge distillation (AKD), which down-weights the contribution of overconfident predictions while emphasizing the learning of negative labels. AKD consists of two components, L_{cls} and L_{akd}, which are enhancements of the BCE loss L_{bce} and KD loss L_{kd}.

Firstly, we modify L_{bce} in Eq. (1) to L_{cls}. We emphasize the learning of negative samples by down-weighting the positive ones in the classification loss for the current model:

$$L_{\text{cls}} = \sum_{c \in \mathcal{C}^t} -y_c^t L_+ - (1 - y_c^t) \log(1 - \hat{y}_c^t). \quad (3)$$

Inspired by focal loss (Lin et al. 2017), L_+ is formulated as:

$$L_+ = (1 - \hat{y}_c^t)^{\alpha \log |\mathcal{C}^{1:t}|} \log(\hat{y}_c^t), c \in \mathcal{C}^t, \alpha > 0, \quad (4)$$

where L_+ is the positive loss part, $\alpha \log |\mathcal{C}^{1:t}|$ is an adaptive exponential decay factor related to the class number $|\mathcal{C}^{1:t}|$ and α is a hyperparameter. As the number of learned classes increases, the absence of negative labels becomes more severe, exacerbating the positive-negative imbalance and leading to more false positive errors. The exponential decay factor adaptively controls the intensity of down-weighting, with $\log(\cdot)$ smoothing parameter adjustments.

We then design L_{akd} based on L_{kd} to down-weight the contribution of overconfident old task predictions:

$$L_{\text{akd}} = \sum_{c \in \mathcal{C}^{1:t-1}} -\hat{y}_c^{t-1} L'_+ - (1 - \hat{y}_c^{t-1}) \log(1 - \hat{y}_c^t), \quad (5)$$

$$L'_+ = (1 - \hat{y}_c^t)^{\alpha \log |\mathcal{C}^{1:t}|} \log(\hat{y}_c^t), c \in \mathcal{C}^{1:t-1}, \alpha > 0. \quad (6)$$

From a qualitative perspective, when a training image in Figure 2 (a) contains only "person" and "dog" labels, the classical KD method may output high prediction scores

for non-existent labels like "car" and "cat". AKD can reduce such false positive errors by down-weighting overconfident predictions, thereby rebalancing in loss. Figure 2 (b) presents a quantitative comparison: 1) it shows the prediction score statistics of KD, CSC (Du et al. 2024b) and AKD. Due to the imbalance, KD and CSC produce numerous erroneous and overconfident predictions. Compared to the baseline KD, our method significantly reduces the FPR from 31.9% to 2.7%. Additionally, in comparison to the SOTA method CSC, our approach also demonstrates significant superiority in FPR (2.7% vs 19.4%); 2) AKD also significantly improves the performance on mAP, CF1, and OF1 metrics, indirectly indicating that our method can more effectively mitigate catastrophic forgetting than the baseline KD.

OR for Label Rebalance

To mitigate the imbalance at the label level, we propose an approach based on reservoir sampling strategy (Rolnick et al. 2019) called online relabeling (OR). This method completes the labels of memory samples using both past and current models in an online manner. OR consists of two steps:

First, after completing training for task t, we utilize the trained current model f_θ^t to relabel new classes $\mathcal{C}^t$ for the memory data sampled from old task $1 : t - 1$. Simultaneously, we employ the past model f_θ^{t-1} to relabel old classes $\mathcal{C}^{1:t-1}$ for the memory data sampled from the current task t. For instance, as shown in Figure 3, after completing training for task 3, we use f_θ^3 to relabel the missing new classes $\mathcal{C}^3$ (in blue) for the memory data from tasks $1 : 2$. Next, we apply f_θ^2 to relabel the missing old classes $\mathcal{C}^{1:2}$ (in green) for task 3 memory samples. This label augmentation process continues until all tasks are trained, ensuring that all labels in memory are fully annotated online. We relabel the missing labels as positive or negative by setting a threshold n. A memory sample x' is input into the comparatively reliable model (f_θ^{t-1} or f_θ^t) used for relabeling. If the model outputs a probability bigger than n for class i, the i-th label is set to 1 (positive); otherwise, it is set to 0 (negative), formulated as $y_i' = \mathbb{I}(\hat{y}_i' > n)$, where $\mathbb{I}$ is the indicator function that returns 1 if the condition inside is true and 0 otherwise. In the second step, when training task t, after relabeling in the first step, all memory samples are fully annotated. We then sample (x', y') randomly from the memory buffer. Given that the first step introduces many negative labels, we use a loss function similar to L_{cls} to down-weight these negative labels, rather than using L_{bce}. The loss function is formulated as:

$$L_{\text{er}} = \sum_{c \in \mathcal{C}^{1:t-1}} -y_c' \log(\hat{y}_c') - (1 - y_c') L_-, \quad (7)$$

$$L_- = (\hat{y}_c')^{\beta \log |\mathcal{C}^{1:t}|} \log(1 - \hat{y}_c'), c \in \mathcal{C}^{1:t-1}, \beta > 0, \quad (8)$$

where $\beta \log |\mathcal{C}^{1:t}|$ represents the adaptive exponential decay factor, ensuring balanced learning of positive and negative labels. OR completes the label matrix in memory, reducing the FPR from 6.7% to 2.4%, mitigating label-level imbalance and catastrophic forgetting.

Method	Backbone	Buffer Size	VOC B4-C2				VOC B5-C3			
			Last Acc			Avg.Acc	Last Acc			Avg.Acc
			mAP	CF1	OF1	mAP	mAP	CF1	OF1	mAP
Joint	CNN	-	92.6	86.7	89.2	-	92.6	86.7	89.2	-
Fine-Tuning	CNN		22.6	14.2	21.4	52.8	48.7	29.1	40.3	73.4
LwF (Li and Hoiem 2017)	CNN		50.4	32.8	30.9	73.4	74.1	50.5	45.6	84.8
AGCN (Du et al. 2024a)	GCN		50.2	35.5	33.5	71.2	71.6	55.2	51.1	83.1
KRT (Dong et al. 2023)	CAM	0	43.6	13.7	31.0	71.0	74.6	39.3	46.6	86.2
CSC (Du et al. 2024b)	GCN		73.8	51.5	43.1	81.9	81.3	63.0	56.6	87.3
RebLL (AKD)	CNN		73.1	57.2	60.0	84.6	79.4	64.6	66.9	87.7
ER (Rolnick et al. 2019)	CNN		47.1	34.7	33.1	69.9	62.8	50.6	47.8	78.3
PODNet (Douillard et al. 2020)	CNN	2/class	60.4	45.3	38.5	71.1	70.3	47.7	43.3	81.4
DER++ (Buzzega et al. 2020)	CNN		61.6	33.4	29.9	77.0	68.1	53.3	51.4	78.0
RebLL (AKD + OR)	CNN	2/class	**81.0**	**71.0**	**70.3**	**88.5**	**84.0**	**73.9**	**72.5**	**90.3**

Table 1: Class-incremental results on PASCAL VOC dataset. A buffer size of 0 means no replay is applied, same as below.

Final Loss

The final loss of RebLL is formulated as follows:

$$L = \lambda_{akd} L_{cls} + (1 - \lambda_{akd}) L_{akd} + \lambda_{er} L_{er}, \qquad (9)$$

where balancing parameters λ_{akd} and λ_{er} control the strength of distillation loss L_{akd} and replay loss L_{er}.

Experiments

Datasets and Experimental Setting

Datasets. We adopt the experimental framework from (Dong et al. 2023; Du et al. 2024b), utilizing the MS-COCO 2014 (Lin et al. 2014) and PASCAL VOC 2007 (Everingham et al. 2010) datasets to validate the efficacy of our method. MS-COCO, a widely recognized benchmark for multi-label classification, includes 82,081 training images and 40,504 validation images across 80 classes, with an average of 2.9 labels per image. The PASCAL VOC 2007 dataset comprises 20 classes, with 5011 images in the training set and 4952 images in the test set, averaging 2.4 labels per image.

Evaluation Metrics. Similar to (Dong et al. 2023; Du et al. 2024b), we employ the widely used metrics of average accuracy (Avg.Acc) and last accuracy (Last Acc) for evaluating CIL tasks. We also use mean average precision (mAP), per-class F1 score (CF1), and overall F1 score (OF1) to assess the performance. The mAP is further divided into average mAP and last mAP, representing the mean mAP across all tasks and the mAP for the final task, respectively. Notably, in the last task, the entire test set is used to evaluate all classes.

Experiments Setup. Following the conventions of prior CIL research (Douillard et al. 2020; Cha et al. 2021; Dong et al. 2023; Du et al. 2024b), we define various MLCIL scenarios using the format {Bx-Cy}, where "x" denotes the number of trained classes in the base task and "y" indicates the number of trained classes in each subsequent incremental task. For the VOC 2007 dataset, we use two challenging scenarios {B4-C2 and B5-C3}. Similarly, for the MS-COCO dataset, we assess RebLL with two challenging scenarios {B20-C4 and B0-C5}. There are more incremental tasks in these challenging scenarios than in (Dong et al. 2023). The CIL process adheres to the lexicographical order of class names, as described in (Dong et al. 2023; Du et al. 2024b).

Implementation Details. We adhere to the experimental settings in (Dong et al. 2023; Du et al. 2024b) to ensure a fair comparison. The training is conducted with a batch size of 64 for 20 epochs on both the PASCAL VOC and MS-COCO datasets. Similar to CSC (Du et al. 2024b) and KRT (Dong et al. 2023), we use TResNetM (Ridnik et al. 2021b) pre-trained on ImageNet-21k as the feature extractor. However, unlike them, we only use the vanilla CNN as our backbone. We optimize the network using the Adam optimizer (Kingma and Ba 2015) with a weight decay of 1e-4. We apply a consistent learning rate of 4e-5 across all tasks for the PASCAL VOC dataset. For the MS-COCO dataset, we use an initial learning rate of 3e-5 for the base task and 5e-5 for subsequent tasks. We employ the same data augmentation methods as detailed in (Du et al. 2024b; Dong et al. 2023).

Overall Performance

We compare eight representative class-incremental methods, including the regularization-based methods LwF (Li and Hoiem 2017), AGCN (Du et al. 2024a), KRT (Dong et al. 2023) and CSC (Du et al. 2024b), the replay-based methods such as ER (Rolnick et al. 2019), PODNet (Douillard et al. 2020) and DER++ (Buzzega et al. 2020), and the prompt-based method L2P (Wang et al. 2022). For benchmarking, we use Fine-Tuning and Joint as the lower and upper bounds for our method, respectively, as done by KRT and CSC. Notably, AGCN, KRT, and CSC are currently the best-performing MLCIL methods, utilizing the powerful graph convolutional network (GCN) or cross-attention mechanism (CAM) as the backbone, with CSC being the current SOTA method. These MLCIL methods, however, do not address the positive-negative imbalance issue. To address this inherent imbalance in MLCIL, we propose the RebLL framework, which includes AKD and OR. Next, we will demonstrate their effects.

PASCAL VOC. The results for PASCAL VOC in {B4-C2 and B5-C3} are presented in Table 1. The RebLL outperforms other representative methods in all scenarios on PASCAL VOC. We have the following observations: 1) When the buffer size is set to 0, even with a vanilla CNN as the backbone and using AKD alone (without OR), we outper-

Method	Backbone	Buffer Size	MS-COCO B20-C4				MS-COCO B0-C5			
			Last Acc			Avg.Acc	Last Acc			Avg.Acc
			mAP	CF1	OF1	mAP	mAP	CF1	OF1	mAP
Joint	CNN	-	81.8	76.4	79.4	-	81.8	76.4	79.4	-
Fine-Tuning	CNN		19.4	10.9	13.4	36.5	22.5	15.0	23.6	48.1
LwF (Li and Hoiem 2017)	CNN		34.6	17.3	31.8	55.4	50.6	36.3	41.1	66.2
AGCN (Du et al. 2024a)	GCN		55.6	44.2	39.6	65.7	53.0	43.2	41.1	64.4
KRT (Dong et al. 2023)	CAM	0	45.2	17.6	33.0	64.0	44.5	22.6	37.5	63.1
CSC (Du et al. 2024b)	GCN		60.6	44.5	43.0	69.8	63.4	50.7	50.1	71.1
RebLL (AKD)	CNN		60.1	51.3	49.2	69.2	63.5	53.5	51.9	71.7
ER (Rolnick et al. 2019)	CNN		41.9	32.9	29.8	53.0	40.1	32.9	32.3	54.6
PODNet (Douillard et al. 2020)	CNN	5/class	58.4	44.0	39.1	67.7	58.2	45.1	40.8	67.2
DER++ (Buzzega et al. 2020)	CNN		57.3	41.4	35.5	65.5	57.9	43.6	39.2	68.2
RebLL (AKD + OR)	CNN	5/class	**62.8**	**53.3**	**53.0**	**71.2**	**65.5**	**56.1**	**54.0**	**72.5**

Table 2: Class-incremental results on MS-COCO dataset.

Method	Backbone	MS-COCO B0-C5				MS-COCO B20-C4			
		Last Acc			Avg.Acc	Last Acc			Avg.Acc
		mAP	CF1	OF1	mAP	mAP	CF1	OF1	mAP
L2P (Wang et al. 2022)	ViT-B/16	61.6	47.0	40.2	67.8	62.1	47.3	39.8	66.6
CSC (Du et al. 2024b)	GCN	63.4	50.7	50.1	71.1	60.6	44.5	43.0	69.8
RebLL	CNN	65.5	56.1	54.0	72.5	62.8	53.3	53.0	71.2
RebLL	GCN	**71.7**	**62.7**	**65.3**	**76.9**	**70.6**	**60.1**	**63.1**	**75.5**

Table 3: Comparative experimental results under different backbones.

form the GCN-based SOTA method CSC on many metrics. Specifically, in {B4-C2}, we achieve an improvement of **5.7%** (51.5%→57.2%) in CF1, **16.9%** (43.1%→60.0%) in OF1, and **2.7%** (81.9%→84.6%) in average mAP; 2) When the buffer size is configured to 2/class, RebLL (AKD+OR) with a vanilla CNN demonstrates significant superiority over all other approaches across every scenario and metric. In {B4-C2}, it improves by **7.2%** (73.8%→81.0%) in final mAP, **19.5%** (51.5%→71.0%) in CF1, **27.2%** (43.1%→70.3%) in OF1, and **6.6%** (81.9%→88.5%) in average mAP. Notably, this significant improvement is primarily attributed to the AKD and OR components of our method, with the addition of buffer playing a minimal role. Subsequent ablation experiments will further support this claim. Similar trends are observed in the {B5-C3} scenario.

MS-COCO. The results for MS-COCO in {B20-C4 and B0-C5} are presented in Table 2. Similar to the results on PASCAL VOC, RebLL with a vanilla CNN as backbone surpasses the SOTA in all scenarios. For example, in {B20-C4}, it improves by **2.2%** (60.6%→62.8%) in final mAP, **8.8%** (44.5%→53.3%) in CF1, **10.0%** (43.0%→53.0%) in OF1, and **1.4%** (69.8%→71.2%) in average mAP.

Comparison of Different Backbones. Table 3 illustrates the comparison between RebLL, CSC and L2P (Wang et al. 2022). While L2P utilizes a pre-trained ViT-B/16, CSC employs a GCN backbone, and our method uses a vanilla CNN. Remarkably, RebLL achieves SOTA results even with a vanilla CNN. Furthermore, when we switch to the same GCN backbone as CSC, our performance surpasses that of CSC by a considerable margin in both {B0-C5 and B20-C4} of MS-COCO. For example, in {B0-C5}, we im-
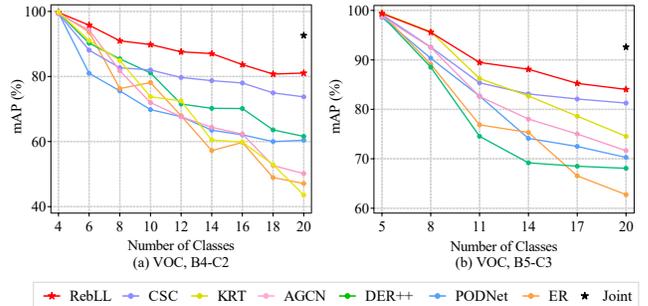

Figure 4: Incremental results on PASCAL VOC in challenging scenarios. There are more tasks in these scenarios.

proved by **8.3%** (63.4%→71.7%) in final mAP, **12.0%** (50.7%→62.7%) in CF1, **15.2%** (50.1%→65.3%) in OF1 and **5.8%** (71.1%→76.9%) in average mAP.

The incremental results for the challenging scenarios {B4-C2 and B5-C3} of PASCAL VOC are shown in Figure 4. These mAP curves illustrate the substantial superiority of RebLL throughout the CIL process. Additionally, RebLL closely aligns with the upper bound (Joint), showcasing its effectiveness even in long-term class-incremental settings.

Ablation Study

Effectiveness of AKD. In Table 4, we use Fine-Tuning (L_{bce}) and KD ($L_{bce} + L_{kd}$) as baselines to validate the effectiveness of AKD in scenario {B4-C2} of PASCAL VOC. AKD (L_{cls}) and AKD ($L_{cls} + L_{akd}$) represent the respective improvements made to Fine-Tuning and KD. 1) AKD

Method	Last Acc			Avg.Acc
	mAP	CF1	OF1	mAP
Fine-Tuning (L_{bce})	22.6	14.2	21.4	52.8
AKD (L_{cls})	36.3	23.3	31.6	56.5
KD ($L_{bce} + L_{kd}$)	58.2	35.3	31.0	77.5
AKD ($L_{cls} + L_{akd}$)	**73.1**	**57.2**	**60.0**	**84.6**

Table 4: Quantitative ablation studies for variants of AKD.

Method	Last Acc			Avg.Acc
	mAP	CF1	OF1	mAP
AKD ($L_{cls} + L_{akd}$)	73.1	57.2	60.0	84.6
+ Replay (L_{bce})	74.2	63.8	63.9	85.5
+ OR (L_{bce})	77.3	67.4	68.3	87.2
+ OR (L_{er})	**81.0**	**71.0**	**70.3**	**88.5**

Table 5: Quantitative ablation studies for variants of OR.

(L_{cls}) outperforms Fine-Tuning, highlighting the effectiveness of emphasizing the contribution of negative labels. 2) From the KD to the AKD ($L_{cls} + L_{akd}$), the final mAP, CF1 and OF1 are improved by **14.9%** (58.2%→73.1%), **21.9%** (35.3%→57.2%) and **29.0%** (31.0%→60.0%). This indicates the effectiveness of AKD in down-weighting overconfident old task predictions. These observations demonstrate that AKD can perform significantly better by calibrating asymmetric contributions of the positive and negative loss parts to the optimization objectives.

Effectiveness of OR. As shown in Table 5, we conduct ablation of OR based on AKD ($L_{cls} + L_{akd}$) in scenario {B4-C2} of PASCAL VOC. The improvement is minimal when adding the baseline Replay (L_{bce}) on top of AKD. OR (L_{bce}) represents the first step of online relabeling, while OR (L_{er}) means combining the first step with the second step. AKD ($L_{cls} + L_{akd}$) + OR (L_{er}) constitute our RebLL framework. We observe that OR can effectively mitigate label-level imbalance and catastrophic forgetting, thereby enhancing model performance.

Hyperparameter Analysis. Figure 5 presents an analysis of hyperparameters α and β in {B4-C2} of PASCAL VOC. When discussing α without including the OR method, AKD reaches its optimum when α is set to 1.2. When OR is incorporated, the best performance is achieved with β set to 0.7. Table 6 compares fixed and adaptive exponential decay factor at the optimal α. The results demonstrate that a fixed decay factor fails to adapt to the evolving positive-negative imbalance in MLCIL, whereas the adaptive factor significantly outperforms it. The balancing parameters λ_{akd} and λ_{er} are set to 0.15 and 0.30, respectively.

Decay Factor	mAP ↑	CF1 ↑	OF1 ↑	Avg.mAP ↑	FPR ↓
α ($\alpha = 2.0$)	72.0	52.8	57.9	83.5	7.2
$\alpha \log |\mathcal{C}^{1:t}|$ ($\alpha = 1.2$)	**73.1**	**57.2**	**60.0**	**84.6**	**2.7**

Table 6: The comparison of fixed and adaptive exponential decay factor in scenario {B4-C2} of PASCAL VOC.
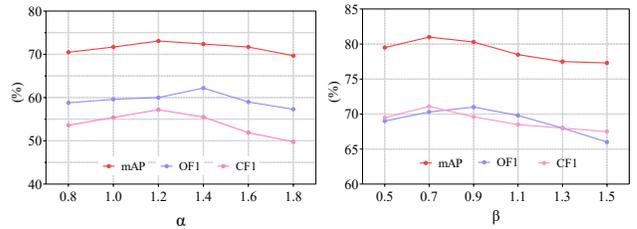

Figure 5: Analysis of α and β for exponential decay factor.
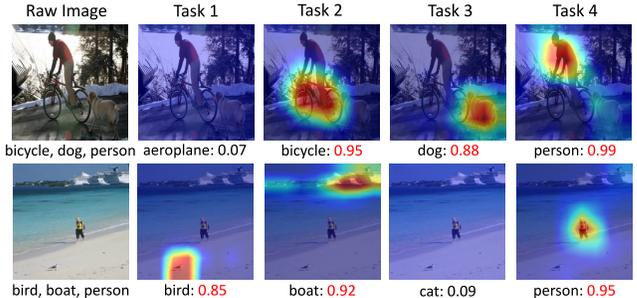

Figure 6: Visualization of RebLL using the final task model.

Visualization

As shown in Figure 6, we visualize raw test images alongside their corresponding class activation maps. Given that certain labels are trained sequentially across different tasks, each row presents the class activation map and prediction score for each class using the final task model. We observe that our model effectively highlights label semantic regions, with low activation for non-existent labels. For example, in the first row, the activation maps accurately reflect the semantic regions for old task labels such as "bicycle", "dog" and "person". The prediction scores further demonstrate that class-specific representations are highly discriminative. These findings suggest that our model exhibits robust anti-forgetting capabilities, allowing it to precisely capture the semantic regions of labels while demonstrating adequate discernment of positive and negative labels.

Conclusion

In this paper, we identify and rebalance the inherent positive-negative imbalance problem in MLCIL under the PL setting. We propose the RebLL framework consisting of two key components: AKD and OR. AKD rebalances at the loss level by emphasizing the negative label learning and down-weighting the contribution of overconfident predictions. OR is designed for label rebalance, which restores the original class distribution in memory by online relabeling the missing classes. Our method is demonstrated to effectively mitigate the positive-negative imbalance, thereby serving as a more efficient anti-forgetting strategy to enhance the ML-CIL performance. We achieve new state-of-the-art results across various challenging MLCIL scenarios on the PASCAL VOC and MS-COCO datasets, regardless of whether we use a vanilla CNN or a more powerful network architecture as the backbone.

Acknowledgments

References

Buzzega, P.; Boschini, M.; Porrello, A.; Abati, D.; and Calderara, S. 2020. Dark experience for general continual learning: a strong, simple baseline. In *Proceedings of the Advances in Neural Information Processing Systems*, 15920–15930.

Cha, S.; Cho, S.; Hwang, D.; Hong, S.; Lee, M.; and Moon, T. 2023. Rebalancing Batch Normalization for Exemplar-based Class-Incremental Learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 20127–20136.

Cha, S.; Yoo, Y.; Moon, T.; et al. 2021. SSUL: Semantic segmentation with unknown label for exemplar-based class-incremental learning. In *Proceedings of the Advances in Neural Information Processing Systems*, 10919–10930.

De Lange, M.; and Tuytelaars, T. 2021. Continual prototype evolution: Learning online from non-stationary data streams. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 8250–8259.

Dong, S.; Luo, H.; He, Y.; Wei, X.; Cheng, J.; and Gong, Y. 2023. Knowledge Restore and Transfer for Multi-Label Class-Incremental Learning. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 18711–18720.

Dosovitskiy, A.; Beyer, L.; Kolesnikov, A.; Weissenborn, D.; Zhai, X.; Unterthiner, T.; Dehghani, M.; Minderer, M.; Heigold, G.; Gelly, S.; et al. 2021. An image is worth 16x16 words: Transformers for image recognition at scale. In *Proceedings of the International Conference on Learning Representations*.

Douillard, A.; Cord, M.; Ollion, C.; Robert, T.; and Valle, E. 2020. Podnet: Pooled outputs distillation for small-tasks incremental learning. In *Proceedings of the European Conference on Computer Vision*, 86–102.

Douillard, A.; Ramé, A.; Couairon, G.; and Cord, M. 2022. Dytox: Transformers for continual learning with dynamic token expansion. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 9285–9295.

Du, K.; Lyu, F.; Hu, F.; Li, L.; Feng, W.; Xu, F.; and Fu, Q. 2022. AGCN: Augmented Graph Convolutional Network for Lifelong Multi-Label Image Recognition. In *IEEE International Conference on Multimedia and Expo*, 01–06.

Du, K.; Lyu, F.; Li, L.; Hu, F.; Feng, W.; Xu, F.; Xi, X.; and Cheng, H. 2024a. Multi-Label Continual Learning Using Augmented Graph Convolutional Network. *IEEE Transactions on Multimedia*, 26: 2978–2992.

Du, K.; Zhou, Y.; Lyu, F.; Li, Y.; Lu, C.; and Liu, G. 2024b. Confidence Self-Calibration for Multi-Label Class-Incremental Learning. In *Proceedings of the European Conference on Computer Vision*, 234–252.

Everingham, M.; Van Gool, L.; Williams, C. K.; Winn, J.; and Zisserman, A. 2010. The pascal visual object classes (voc) challenge. *International Journal of Computer Vision*.

Guo, G.; Han, L.; Wang, L.; Zhang, D.; and Han, J. 2023. Semantic-aware knowledge distillation with parameter-free feature uniformization. *Visual Intelligence*, 1(1): 6.

Jia, Z.; Sun, S.; Liu, G.; and Liu, B. 2024. MSSD: multi-scale self-distillation for object detection. *Visual Intelligence*, 2(1): 8.

Kim, C. D.; Jeong, J.; and Kim, G. 2020. Imbalanced continual learning with partitioning reservoir sampling. In *Proceedings of the European Conference on Computer Vision*, 411–428.

Kingma, D. P.; and Ba, J. 2015. Adam: A method for stochastic optimization. In *Proceedings of the International Conference on Learning Representations*.

Kirkpatrick, J.; Pascanu, R.; Rabinowitz, N.; Veness, J.; Desjardins, G.; Rusu, A. A.; Milan, K.; Quan, J.; Ramalho, T.; Grabska-Barwinska, A.; et al. 2017. Overcoming catastrophic forgetting in neural networks. *National Academy of Sciences*, 114(13): 3521–3526.

Li, Z.; and Hoiem, D. 2017. Learning without forgetting. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 40(12): 2935–2947.

Liang, Y.-S.; and Li, W.-J. 2022. Optimizing Class Distribution in Memory for Multi-Label Online Continual Learning. *arXiv preprint arXiv:2209.11469*.

Lin, T.-Y.; Goyal, P.; Girshick, R.; He, K.; and Dollár, P. 2017. Focal loss for dense object detection. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2980–2988.

Lin, T.-Y.; Maire, M.; Belongie, S.; Hays, J.; Perona, P.; Ramanan, D.; Dollár, P.; and Zitnick, C. L. 2014. Microsoft coco: Common objects in context. In *Proceedings of the European Conference on Computer Vision*, 740–755.

Liu, B.; Xu, N.; Lv, J.; and Geng, X. 2023. Revisiting pseudo-label for single-positive multi-label learning. In *International Conference on Machine Learning*, 22249–22265. PMLR.

Luo, Z.; Liu, Y.; Schiele, B.; and Sun, Q. 2023. Class-incremental exemplar compression for class-incremental learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 11371–11380.

Lyu, F.; Du, K.; Li, Y.; Zhao, H.; Zhang, Z.; Liu, G.; and Wang, L. 2024. Variational Continual Test-Time Adaptation. *arXiv preprint arXiv:2402.08182*.

Lyu, F.; Sun, Q.; Shang, F.; Wan, L.; and Feng, W. 2023. Measuring asymmetric gradient discrepancy in parallel continual learning. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 11411–11420.

Lyu, F.; Wang, S.; Feng, W.; Ye, Z.; Hu, F.; and Wang, S. 2021. Multi-domain multi-task rehearsal for lifelong learning. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 35, 8819–8827.

McCloskey, M.; and Cohen, N. J. 1989. Catastrophic interference in connectionist networks: The sequential learning problem. In *Psychology of Learning and Motivation*, volume 24, 109–165.

Mohamed, A.; Grandhe, R.; Joseph, K.; Khan, S.; and Khan, F. 2023. D3Former: Debiased Dual Distilled Transformer for Incremental Learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2420–2429.

Ridnik, T.; Ben-Baruch, E.; Zamir, N.; Noy, A.; Friedman, I.; Protter, M.; and Zelnik-Manor, L. 2021a. Asymmetric loss for multi-label classification. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 82–91.

Ridnik, T.; Lawen, H.; Noy, A.; Ben Baruch, E.; Sharir, G.; and Friedman, I. 2021b. Tresnet: High performance gpu-dedicated architecture. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, 1400–1409.

Rolnick, D.; Ahuja, A.; Schwarz, J.; Lillicrap, T.; and Wayne, G. 2019. Experience replay for continual learning. In *Proceedings of the Advances in Neural Information Processing Systems*, 350–360.

Sun, Q.; Lyu, F.; Shang, F.; Feng, W.; and Wan, L. 2022. Exploring example influence in continual learning. In *Proceedings of the Advances in Neural Information Processing Systems*, volume 35, 27075–27086.

Wang, L.; Zhang, X.; Su, H.; and Zhu, J. 2024. A Comprehensive Survey of Continual Learning: Theory, Method and Application. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 46(8): 5362–5383.

Wang, Z.; Zhang, Z.; Lee, C.-Y.; Zhang, H.; Sun, R.; Ren, X.; Su, G.; Perot, V.; Dy, J.; and Pfister, T. 2022. Learning to prompt for continual learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 139–149.

Wu, Z.; Baek, C.; You, C.; and Ma, Y. 2021. Incremental learning via rate reduction. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 1125–1133.

Xie, M.-K.; Xiao, J.; Liu, H.-Z.; Niu, G.; Sugiyama, M.; and Huang, S.-J. 2024. Class-distribution-aware pseudo-labeling for semi-supervised multi-label learning. *Advances in Neural Information Processing Systems*, 36.

Yang, Y.; Cui, Z.; Xu, J.; Zhong, C.; Zheng, W.-S.; and Wang, R. 2023. Continual learning with Bayesian model based on a fixed pre-trained feature extractor. *Visual Intelligence*, 1(1): 5.

Ye, F.; and Bors, A. G. 2022. Lifelong Teacher-Student Network Learning. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 44(10): 6280–6296.

Zeng, Z.; Yang, F.; Liu, H.; and Satoh, S. 2024. Improving deep metric learning via self-distillation and online batch diffusion process. *Visual Intelligence*, 2(1): 18.

Zhao, L.; Lu, J.; Xu, Y.; Cheng, Z.; Guo, D.; Niu, Y.; and Fang, X. 2023. Few-Shot Class-Incremental Learning via Class-Aware Bilateral Distillation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 11838–11847.

Zhou, D.; Chen, P.; Wang, Q.; Chen, G.; and Heng, P.-A. 2022. Acknowledging the unknown for multi-label learning with single positive labels. In *Proceedings of the European Conference on Computer Vision*, 423–440.