

Gradient Reweighting: Towards Imbalanced Class-Incremental Learning

Jiangpeng He

he416@purdue.edu

Elmore Family School of Electrical and Computer Engineering, Purdue University, USA

Abstract

Class-Incremental Learning (CIL) trains a model to continually recognize new classes from non-stationary data while retaining learned knowledge. A major challenge of CIL arises when applying to real-world data characterized by non-uniform distribution, which introduces a dual imbalance problem involving (i) disparities between stored exemplars of old tasks and new class data (inter-phase imbalance), and (ii) severe class imbalances within each individual task (intra-phase imbalance). We show that this dual imbalance issue causes skewed gradient updates with biased weights in FC layers, thus inducing over/under-fitting and catastrophic forgetting in CIL. Our method addresses it by reweighting the gradients towards balanced optimization and unbiased classifier learning. Additionally, we observe imbalanced forgetting where paradoxically the instance-rich classes suffer higher performance degradation during CIL due to a larger amount of training data becoming unavailable in subsequent learning phases. To tackle this, we further introduce a distribution-aware knowledge distillation loss to mitigate forgetting by aligning output logits proportionally with the distribution of lost training data. We validate our method on CIFAR-100, ImageNetSubset, and Food101 across various evaluation protocols and demonstrate consistent improvements compared to existing works, showing great potential to apply CIL in real-world scenarios with enhanced robustness and effectiveness.

1. Introduction

The ever-evolving and unpredictable nature of real-world environments drives the imperative to develop Class-Incremental Learning (CIL) systems with the capability of acquiring knowledge continuously from non-stationary data where new classes appear sequentially over time. The advantage of CIL resides in both memory and computational efficiency which eliminates the requirement of storing all

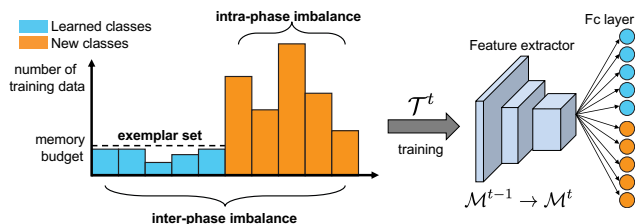


Figure 1. The illustration of imbalanced class-incremental with a dual imbalance issue including the intra-phase imbalance within each new task \mathcal{T} and inter-phase imbalance between old tasks exemplars and new task training data. \mathcal{M}^t refers to the model after learning the new task \mathcal{T}^t .

previously learned data or retraining the model from scratch entirely, making CIL applicable to various practical applications such as on-device learning [17]. However, deep neural network suffers from catastrophic forgetting [32] when learning new classes, where the performance on learned classes decreases significantly due to the unavailability of old training data. Moreover, conventional CIL methodologies typically tackle this challenge by assuming a balanced data distribution, where each class has roughly the same number of training samples. Nevertheless, this assumption is misaligned with real-world scenarios where data is usually long-tail distributed with significant disparity between instance-rich (*head*) classes and instance-rare (*tail*) classes. Such imbalance presents unique challenges that conventional CIL methods cannot adequately address, thus reducing their effectiveness and broader applicability.

As shown in Figure 1, CIL with imbalanced data introduces a dual challenge encompassing both inter-phase and intra-phase imbalances. Conventional CIL approaches primarily address the inter-phase imbalance while the intra-phase imbalance emerges distinctly in the context of imbalanced CIL. In general, the inter-phase imbalance arises from the disparities between new class data and the learned task data preserved in the exemplar set for knowledge replay [29, 37]. Such imbalances can skew the learning process of the classifier between new and old classes [11, 16] (e.g., the learned weights in the FC layer are heavily biased [46, 53]), leading to biased predictions towards the

Code is available at: https://github.com/JiangpengHe/imbalanced_cil

newer classes and thus inducing catastrophic forgetting. In the case of intra-phase imbalance, either the new classes or the stored exemplars individually present their own imbalanced distributions (the imbalance of the exemplar set arises when the number of training data is less than the memory budget for each class). This not only affects the learning of new knowledge but also exacerbates the forgetting issue by intensifying the inter-phase biases.

One of the major challenges posed by learning from imbalanced data is the biased gradient updates in FC layers that are dominated by instance-rich classes. Specifically, in gradient-based optimization such as stochastic gradient descent (SGD), the weight update step in each iteration is heavily influenced by the magnitude of gradients, which significantly depends on the data distribution of the sampled mini-batch. As a result, instance-rich classes tend to receive gradients with larger magnitudes thus skewing the optimization process to make the classifier over-fit on head classes while under-fit on tail classes. This problem becomes more challenging under the context of CIL as the training data distribution changes over time, and the learned classes suffer from catastrophic forgetting. Besides, we observe that the forgetting could also be imbalanced where the head classes usually suffer more performance degradation as a substantial portion of their training data becomes unavailable in subsequent incremental learning stages. To this end, our work aims to address the dual imbalance issue by reweighting the gradient updates in the FC layer, which recalibrates the optimization process and fosters the learning of unbiased classifiers. Furthermore, we introduce a distribution-aware knowledge distillation loss to alleviate the imbalanced catastrophic forgetting problem by considering the distribution of lost training data to impose stronger regularization effects on the head classes. The main contributions of this work are summarized in the following,

- We study CIL under realistic class-imbalanced scenarios and introduce a new end-to-end gradient reweighting framework to tackle both intra-phase and inter-phase imbalance challenges by rebalancing the optimization process in FC layers.
- A new distribution-aware knowledge distillation loss is proposed to further mitigate imbalance catastrophic forgetting caused by the uneven number of lost training data during imbalanced CIL.
- The efficacy of our proposed method is validated through extensive experiments across a variety of CIL settings, which achieves notable improvements over existing methods in both CIL and long-tailed recognition tasks.

2. Related Work

In this section, we review the existing studies that are most related to our work including class-incremental learning in Section 2.1 and long-tailed recognition in Section 2.2.

2.1. Class-Incremental Learning

Class-incremental learning (CIL) is a subarea of continual learning where the task index is not provided during the inference phase, *i.e.* a single head classifier [30] is used for all classes seen so far. Existing CIL has been studied in both online and offline scenarios where each data is observed only once by the model in the former case.

Conventional CIL approaches can be mainly categorized into three groups including regularization, memory replay, and parameter isolation. The *regularization-based* methods aim to restrict drastic alterations to model parameters that are important for learned classes. A representative technique is to apply knowledge distillation [20] based on output logits [10, 25, 37, 46] or the intermediate layers [14, 21, 40]. In addition, recent studies [1, 5, 46] reveal the biased weights in FC layer towards newer learned classes and address it by applying post-hoc biased logits correction [5, 18, 46, 53] or using separated softmax [1] and cosine normalized classifier [21]. The *replay-based* methods store a small set of exemplar data to perform knowledge rehearsal in subsequent phases. Herding algorithm [45] is widely employed in CIL [10, 37, 46] for exemplar selection based on the class mean. Later, a learning-based exemplar selection method is introduced in [27]. In the online setting, the exemplar selection is performed at the end of each training iteration [2, 3, 16, 29, 36] without knowing the distribution of training data. Finally, the *parameter isolation* related approaches gradually expand the network size or increase model parameters [28, 44, 47, 49, 50] to provide dedicated space for learning new knowledge while ensuring the previously learned information remains undisturbed.

Imbalanced CIL remains less explored compared to conventional CIL due to the intricate challenge of addressing both catastrophic forgetting and imbalanced learning simultaneously. The earlier works focused on imbalanced CIL under multi-label [23], semi-supervised [6], and online scenarios [12]. The most recent study [26] formulated long-tailed CIL for both shuffled and ordered cases and proposed a two-stage technique to decouple representation learning and classifier learning. Later, a dynamic residual classifier is introduced in [11] to handle the imbalance between new and learned classes. In this work, we primarily focus on CIL following the shuffled long-tailed case [26] to address the imbalance issue from the perspective of balancing the gradient updates in FC layer. In addition, contrasting the findings in [23], we observed imbalanced forgetting where the head classes instead suffer more performance degradation attributed to the larger volume of lost training data in subsequent incremental phases. Therefore, we incorporate a distribution-aware knowledge distillation loss to impose a more stringent regularization on head classes to further mitigate the forgetting issue.

2.2. Long-tailed Learning

The deep long-tailed learning for visual recognition has been studied comprehensively [48, 52] over the decades due to the prevalence of class-imbalanced data obtained in real-world scenarios. The major challenge arises from the fact that the performance of deep learning models tends to be dominated by head classes, leaving the learning for tail classes significantly under-realized. In this work, we center on class-rebalanced approaches with the goal of rebalancing the influence resulting from imbalanced training samples, which mainly consists of re-sampling, and class-sensitive learning. Specifically, the *re-sampling* based methods [8, 35, 43] aim to construct a balanced training batch by increasing the probability of tail classes to be sampled. The *class-sensitive* learning seeks to adjust the value of training loss to rebalance the uneven learning effects by reweighting class influence [9, 13, 33, 34, 38]. In addition, recent studies [16, 41, 42] argue that the imbalance between positive and negative gradients severely inhibits balanced learning and they address it by adjusting the loss influence across different classes. Our proposed method stems from a similar observation but identifying the magnitude of gradient vectors diverges significantly between head and tail classes during the learning process. This discrepancy leads to uneven optimization steps, subsequently causing the model to over-fit the head classes while under-fitting the tail classes. To address this, our goal is to equilibrate the gradient updates, ensuring a more balanced learning process. Furthermore, existing long-tailed learning methods do not consider previously acquired knowledge for learned classes, making them unsuitable under the context of CIL.

3. Preliminaries

The CIL can be formulated as learning a sequence of N tasks $\{\mathcal{T}^1, \mathcal{T}^2, \dots, \mathcal{T}^N\}$. Each task $t \in [1, N]$ refers to one learning phase in CIL, which can be represented as $\mathcal{T}^t = \{\mathcal{X}^t, \mathcal{Y}^t\}$ where $\mathcal{X}^t = \{\mathbf{x}_1^t, \dots, \mathbf{x}_{|\mathcal{X}^t|}^t\}$ denotes the set of total $|\mathcal{X}^t|$ number of training data and \mathcal{Y}^t denotes the associated class labels belonging to $|\mathcal{Y}^t|$ classes. In general, we have total $\sum_{t=1}^N |\mathcal{X}^t|$ training data corresponding to $\sum_{t=1}^N |\mathcal{Y}^t|$ classes. After learning each task \mathcal{T}^t , the model is evaluated on test data belonging to all seen classes $\mathcal{Y}^{1:t}$ without knowing the task identifier t . A strong constraint in CIL when incrementally learning a new task $\mathcal{T}^t (t > 1)$ is the unavailability of previously learned tasks data $\{\mathcal{X}^1, \dots, \mathcal{X}^{t-1}\}$, which causes the catastrophic forgetting on learned tasks $\mathcal{T}^{1:t-1}$. Therefore, given a memory budget n_ϵ data per class, the replay-based methods employ an exemplar set $\mathcal{E}^t = \{\mathcal{X}_e^{1:t-1}, \mathcal{Y}_e^{1:t-1}\}$ to store both data $\mathcal{X}_e^{1:t-1}$ and their class labels $\mathcal{Y}_e^{1:t-1}$ to combine with new data $\{\mathcal{X}_e^{1:t-1} \cup \mathcal{X}^t\} \times \{\mathcal{Y}_e^{1:t-1} \cup \mathcal{Y}^t\}$ during each incremental learning phase to maintain the learned knowledge.

Imbalanced CIL refers to the case where the number

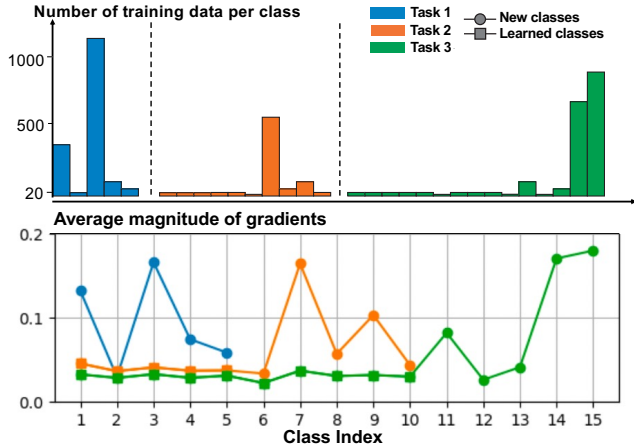


Figure 2. The average magnitudes of gradient $\|\nabla_{\mathcal{L}_{cc}}(W^j)\|$ for each class j by incrementally learning 3 tasks $\mathcal{T}^1, \mathcal{T}^2, \mathcal{T}^3$ with cross-entropy and memory budget $n_\epsilon = 20$ exemplars per class.

of training data varies a lot among different classes while the test data remain balanced. This induces a dual imbalance issue in the training phase including the intra-phase imbalance and inter-phase imbalance. Specifically, the former case represents the discrepancy within each task \mathcal{T}^t where given a head class j and a tail class k with $j, k \in \mathcal{Y}^t$, we have the corresponding training data $n_j \gg n_k$. Furthermore, the exemplar set $|\mathcal{E}^t|$ may also exhibit this class-imbalance issue especially given a larger memory budget n_ϵ as more training from head classes will be retained compared to tail classes, thereby amplifying this challenge. The latter case refers to the discrepancy between new task data and stored exemplars when $t > 1$ where $|\mathcal{X}^t| \gg |\mathcal{X}_e^{1:t-1}|$. In general, the intra-phase imbalance can restrict the model’s capacity to learn new knowledge while the inter-phase imbalance can lead to catastrophic forgetting as the model might exhibit a bias towards new classes.

3.1. Imbalanced Gradients

Imbalanced gradient updates pose a significant challenge when learning from imbalanced data as studied in [15]. Generally, consider the weights of a classifier $W \in \mathbb{R}^{d_f \times c}$ with input feature dimension d_f and output classes c . Given a loss function \mathcal{L} , the gradient-based weight update process is formulated as

$$W_{i+1} = W_i - \eta \nabla_{\mathcal{L}}(W_i) \quad (1)$$

where i denotes the index of iteration and η is the learning rate. The gradient matrix $\nabla_{\mathcal{L}}(W_i) \in \mathbb{R}^{d_f \times c}$ determines the magnitude of the update $\eta \|\nabla_{\mathcal{L}}(W_i)\|$ and the direction of the update $\frac{\nabla_{\mathcal{L}}(W_i)}{\|\nabla_{\mathcal{L}}(W_i)\|}$ where $\|\cdot\|$ denotes the l_2 norm. In general, the magnitude of the gradient is proportional $\|\nabla_{\mathcal{L}}(W_i)\| \propto \mathcal{L}$ to the loss value which is usually computed as the average loss over all data samples. However, challenges arise when the gradient is estimated

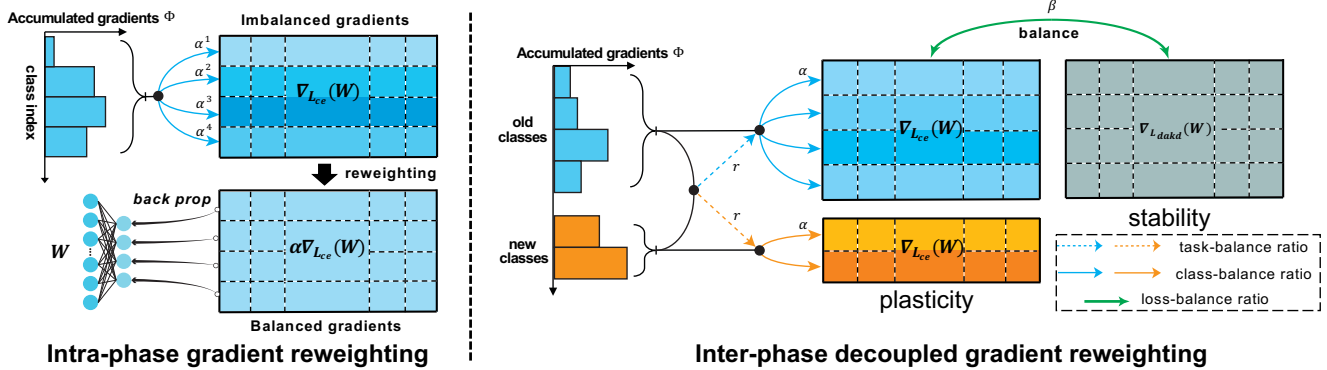


Figure 3. The overview of gradient reweighting under imbalanced CIL. Given the classifier W , the intra-phase gradient weighting is guided by scaling the gradient matrix $\nabla_{L_{ce}}(W)$ with class balance ratios α derived from the cumulative gradients Φ over iterations. Concurrently, the inter-phase Decomposed Gradient Reweighting (DGR) balances the plasticity learning by separately adjusting gradients with class-balance ratios α and task-balance ratios r . Followed by tuning the stability-plasticity trade-off with a loss balance ratio β .

from a mini-batch containing class-imbalanced data, *e.g.*, $n_j \gg n_k$ given a head class j and tail class k . In this case, the gradient magnitude for the head class tends to be significantly larger than that for the tail class $\|\nabla_{\mathcal{L}}(W_i^j)\| \gg \|\nabla_{\mathcal{L}}(W_i^k)\|$. In Figure 2, we highlight the gradient imbalances across different classes by calculating the average gradient magnitude at the end of each task by incrementally learning 3 new tasks (with cross-entropy loss) on ImageNetSubset-LT [26] with each task contains 5 classes. This disparity in gradient magnitudes leads to an imbalanced optimization process where larger gradients from head classes cause larger optimization steps in weight updates, thus potentially leading to over-fitting. Conversely, smaller updates for tail classes may result in under-fitting. Furthermore, the larger gradient magnitudes can also induce biases in the norm of weight vectors as $\|W_{i+1} - W_i\| \propto \|\nabla_{\mathcal{L}}(W_i)\|$, resulting in biased prediction towards instance-rich or newly learned classes under CIL. In the following section, we illustrate our proposed method to address the aforementioned issues by re-weighting the biased gradients to facilitate learning of a balanced classifier.

4. Method

Overview. As illustrated in Figure 3, we aim to reweight the imbalanced gradient matrix by classwisely multiplying it with a balance vector $\alpha = [\alpha^1, \alpha^2, \dots, \alpha^c]$ for total c classes seen so far. Therefore, the gradient update process can be modified as

$$W_{i+1}^j = W_i^j - \eta \alpha^j \nabla_{\mathcal{L}}(W_i^j) \quad (2)$$

where W_i^j refers to the weight vector in FC layer for class j and $\alpha^j \nabla_{\mathcal{L}}(W_i^j)$ denotes the corresponding reweighted balanced gradient. However, it is non-trivial to obtain the appropriate balance vector due to the intricate data-dependent optimization process. In addition, when learning a new task $\mathcal{T}^t (t > 1)$ with the present of learned classes

belonging to $\mathcal{Y}^{1:t-1}$, it poses two additional challenges. First, the training distribution, denoted as \mathcal{D} , changed for learned classes during CIL where $\mathcal{D}(\mathcal{X}_e^{1:t-1}, \mathcal{Y}_e^{1:t-1}) \neq \mathcal{D}(\mathcal{X}^{1:t-1}, \mathcal{Y}^{1:t-1})$ as only limited exemplars are stored for training in subsequent learning phases after they were initially observed. Therefore, simply balancing all gradients without accounting for this will result in knowledge forgetting for learned classes, which could be imbalanced as well due to the uneven number of lost training data between head and tail classes. Second, the training objective varies between new and old tasks where the learned tasks $\mathcal{T}^{1:t-1}$ build upon knowledge accumulated from previous learning phases, while the new task \mathcal{T}^t starts from scratch without any prior knowledge. Consequently, balancing the gradient updates for all classes may not be sufficient due to the intrinsic imbalance in optimization, which requires a more adaptive and flexible approach to ensure the model can effectively learn new tasks without forgetting the knowledge from previous ones.

In the following sections, we illustrate how to effectively determine the balance vector α under CIL for intra-phase scenario in Section 4.1, and inter-phase case to tackle both biased gradients and imbalanced forgetting in Section 4.2.

4.1. Intra-Phase Class-Imbalance

Consider the case of learning the first task $\mathcal{T}^1 = \{\mathcal{X}^1, \mathcal{Y}^1\}$ from a scratch model $\mathcal{M}^0(f_\theta, W)$ where f_θ and W refers to the feature extractor and classifier, respectively. The cross-entropy loss is formulated as

$$\mathcal{L}_{ce} = -\frac{1}{|\mathcal{X}^1|} \sum_{k=1}^{|\mathcal{X}^1|} \sum_{j=1}^{|\mathcal{Y}^1|} \hat{y}_j \log(p_j) \quad (3)$$

where p_j is the Softmax of the j th output logit $z_j = [W^T f_\theta(\mathbf{x}_k)]_j$ and \hat{y}_j denotes the class label. The goal is to determine the class-wise balance vector $\alpha = [\alpha^1, \dots, \alpha^{|\mathcal{Y}^1|}]$.

Recognizing that a static balance vector may not sufficiently capture the biases inherent in the gradients matrix that evolve as the learning progresses, we propose a simple yet effective solution in this work to determine the balance vector adaptively by leveraging historical accumulated gradients. In detail, we calculate the class balance ratio α_i^j at iteration i based on the accumulated magnitude of gradients corresponding to each class j as

$$\alpha_i^j = \frac{\min_{m \in \mathcal{Y}^1} \Phi_i^m}{\Phi_i^j}, \quad \Phi_i^j = \sum_{n=1}^i \|\nabla_{\mathcal{L}_{ce}}(W_n^j)\| \quad (4)$$

By doing so, we dynamically recalibrate the weight updates, ensuring that both majority and minority classes contribute equally to the learning process and address the biased issue.

Regularized softmax cross-entropy. While the gradient update adjustments help in emphasizing tail classes, it also inadvertently leads to increased gradients towards head classes due to the decrease of output logit z_j of head classes, making the cross-entropy loss put more effort into increasing its output logits. Motivated by [33], we alleviate this side effect by applying regularized cross-entropy using a modified softmax equation

$$p_j = \frac{\exp(z_j + \log \pi_j)}{\sum_{m=1}^{|\mathcal{Y}^1|} \exp(z_m + \log \pi_m)} \quad (5)$$

where π_j denotes the estimates of class priors and $\log \pi_j$ is the per-label offsets added to the original output logit. In practice, the calibrated class distribution with the number of data for each seen class is widely used to determine π_j . Generally, the instance-rich classes have larger π_j with an increase of softmax output p_j to compensate for the side effect of down-weighting the gradients during the training process.

4.2. Inter-Phase Class-Imbalance

We extend our intra-phase method to address the inter-phase imbalance in CIL by adding two modules: (i) a Distribution-Aware Knowledge Distillation (DAKD) loss to maintain the information from the original training distribution while mitigating the imbalanced forgetting (Section 4.2.1), and (ii) a Decoupled Gradient Reweighting (DGR) framework to separately manage the gradients between new and already learned tasks to ensure a balance between the stability of past knowledge and the plasticity required for learning new classes (Section 4.2.2).

4.2.1 Distribution-Aware Knowledge Distillation

Similar to existing bias correction methods [1, 46, 53], we consider adding the knowledge distillation loss [20] to maintain the learned knowledge due to the changing of training distribution for learned classes. The original distillation loss for learning a new task t can be expressed as

$$\mathcal{L}_{kd}(z, \hat{z}) = -\frac{1}{|\mathcal{X}^t + \mathcal{X}_e^{1:t-1}|} \sum_{k=1}^{|\mathcal{X}^t + \mathcal{X}_e^{1:t-1}|} \sum_{j=1}^{|\mathcal{Y}^{1:t-1}|} \hat{p}_j^\tau \log(p_j^\tau) \quad (6)$$

where $\hat{p}_j^\tau = \text{Softmax}(\hat{z}_j/\tau)$ is softened ($\tau = 2$) output probability from the teacher model \hat{M}^{t-1} with frozen parameters obtained in the last learning phase $t - 1$. However, as mentioned earlier in Section 1, catastrophic forgetting can be imbalanced as a larger volume of instances from the head class become unavailable in comparison to the tail class in the subsequent incremental learning phases due to the fixed memory budget. Though the most intuitive way to address this issue is storing more data for head classes and less data for tail classes, it poses another nontrivial challenge arising from a class-imbalanced exemplar set that potentially intensifies both over-fitting and under-fitting issues. Therefore, we introduce the Distribution-Aware Knowledge Distillation (DAKD) to maintain the knowledge by taking into account the distribution of the lost training data. Specifically, we obtain the lost training data distribution \mathbf{s} by calculating $s_j = |\mathcal{X}^j| - |\mathcal{X}_e^j|$ for each class $j \in \mathcal{Y}^{1:t-1}$ where $|\mathcal{X}^j|$ and $|\mathcal{X}_e^j|$ denote the number of original training data and stored exemplars for class j , respectively. Motivated by [54], we decouple the original distillation loss into a weighted sum of two parts using a ratio $\sigma \in [0, 1]$ (1 indicates balanced distribution) measured by the entropy of \mathbf{s} . The DAKD is then formulated as

$$\mathcal{L}_{daked}(z, \hat{z}|\mathbf{s}) = \sigma \mathcal{L}_{kd}(z, \hat{z}) + (1 - \sigma) \mathcal{L}_{kd}^{imb}(\tilde{z}, \hat{z}) \quad (7)$$

where \mathcal{L}_{kd} denotes the balanced part calculated using the original output logit z , and \mathcal{L}_{kd}^{imb} denotes the imbalanced part with adjusted output logits \tilde{z} determined by

$$\tilde{z}_j = \frac{s_j}{\sum_m^{|\mathcal{Y}^{1:t-1}|} s_m} z_j + (1 - \frac{s_j}{\sum_m^{|\mathcal{Y}^{1:t-1}|} s_m}) \hat{z}_j \quad (8)$$

where this calibrated distribution of output logits \tilde{z} demonstrates larger discrepancy ($|\tilde{z}_j - \hat{z}_j|$) for class j with a higher volume of lost training data (*i.e.*, head classes), thereby assigning those classes with more efforts for knowledge distillation compared to those with less data lost (*i.e.*, tail classes) which only require a subtler distillation intervention to maintain the learned knowledge. This tailored logit adjustment ensures that the extent of knowledge distillation is appropriately aligned with the level of data lost experienced by each class. The overall loss function for learning a new task \mathcal{T}^t , $t > 1$ can be expressed as

$$\mathcal{L} = \mathcal{L}_{ce} + \lambda \mathcal{L}_{daked} \quad (9)$$

We use $\lambda = \lambda_b \sqrt{|\mathcal{X}_{old}|/|\mathcal{X}_{new}|}$ to adjust the influence of knowledge distillation, which increases as more data have been observed. $|\mathcal{X}_{old}|$ and $|\mathcal{X}_{new}|$ denote the number of old and new classes training data and λ_b is a fixed scalar.

4.2.2 Decoupled Gradient Reweighting

The overview of Decoupled Gradient Reweighting (DGR) is shown in Figure 3, which addresses the inter-phase imbalance issue by striking a balance between stability (maintain past knowledge) and plasticity (learn new classes). Generally, the gradients from cross-entropy loss ($\nabla_{\mathcal{L}_{ce}}$) in CIL represent the plasticity that enables the model to adapt to new training data distribution by incorporating both exemplars of old classes and data from new classes. In contrast, the stability is introduced by the gradients from knowledge distillation loss ($\nabla_{\mathcal{L}_{dakkd}}$) to guide the model towards a solution that aligns with the training distribution from the previous learning phases. Therefore, the DGR address inter-phase imbalance in two folds by first reweighting gradient $\nabla_{\mathcal{L}_{ce}}$ from \mathcal{L}_{ce} to ensure unbiased plasticity and then modulating the interaction between $\nabla_{\mathcal{L}_{ce}}$ and $\nabla_{\mathcal{L}_{dakkd}}$ to attain a balanced equilibrium between plasticity and stability.

For plasticity, the DGR first separately reweights the gradients $\nabla_{\mathcal{L}_{ce}}$ for learned classes $j \in \mathcal{Y}^{1:t-1}$ and new classes for $j \in \mathcal{Y}^t$ by calculating class-balance ratios α_i^j respectively as specified in Equation 4. Following this, a task-balance ratio r_i^j is introduced for tuning between new and learned tasks as

$$r_i^j = \begin{cases} \min\{1, \frac{1}{r_{\Phi_i}}\} & j \in \mathcal{Y}^{1:t-1} \\ \min\{1, r_{\Phi_i} \times \exp(-\gamma \frac{|\mathcal{X}^{1:t-1}|}{|\mathcal{X}^{1:t}|})\} & j \in \mathcal{Y}^t \end{cases} \quad (10)$$

where $\exp(-\gamma \frac{|\mathcal{X}^{1:t-1}|}{|\mathcal{X}^{1:t}|})$ is the attenuation factor and $\gamma > 0$ is a hyper-parameter to adjust its magnitude. $r_{\Phi_i} = \frac{\bar{\Phi}_i^{j \in \mathcal{Y}^{1:t-1}}}{\bar{\Phi}_i^{j \in \mathcal{Y}^t}}$ is the ratio of mean accumulated gradients for learned classes $\bar{\Phi}_i^{j \in \mathcal{Y}^{1:t-1}}$ to new classes $\bar{\Phi}_i^{j \in \mathcal{Y}^t}$.

The advantage of DGR with an attenuation factor lies in promoting a more equitable optimization based on the fact that the learned classes are built upon the knowledge learned in prior learning phases, whereas the new classes are being trained from scratch. Therefore, merely balancing the gradient contributions between new and old classes could inadvertently lead to the under-fitting of new classes as they may not receive adequate training emphasis. Recognizing this, the attenuation factor adjusts the gradient reweighting ratio in favor of new classes proportional to the volume of data that has already been observed so far $|\mathcal{X}^{1:t-1}|$, thereby providing a calibrated boost in support of the new classes to mitigate the learning disparities.

For tuning between stability and plasticity, we adjust the magnitude of gradients from distillation $\|\nabla_{\mathcal{L}_{dakkd}}(W_i)\|$ to make it balance with reweighted cross-entropy by including a loss balance ratio β_i at iteration i as

$$\beta_i = \frac{\|\alpha_i r_i \nabla_{\mathcal{L}_{ce}}(W_i)\|}{\|\nabla_{\mathcal{L}_{dakkd}}(W_i)\|} \quad (11)$$

The overall DGR can be formulated as

$$W_{i+1}^j = W_i^j - \underbrace{\eta(\alpha_i^j r_i^j \nabla_{\mathcal{L}_{ce}}(W_i^j))}_{j \in \mathcal{Y}^t} + \beta_i \nabla_{\mathcal{L}_{dakkd}}(W_i^j) \quad (12)$$

Note that in this work, we intentionally avoid classwisely reweighting the gradients from knowledge distillation loss as it contains instrumental information obtained from previous training distribution to help maintain the discrimination of learned classes during CIL as studied in [53]. Later in Section 5.3, we will show that reweighting the gradients of knowledge distillation could harm the overall performance.

5. Experiments

5.1. Experimental Setups

Evaluation protocol. In this work, we adopt two widely used protocols for CIL including (1) learning from scratch (*LFS*) [37] and (2) learning from half (*LFH*) [14, 21]. The *LFS* equally split all the classes into N tasks to incrementally train a model from scratch. The *LFH* first trains the model with the initial half of the classes and then equally divides the remaining half of the classes into N tasks. For both cases, the model is evaluated on all classes seen so far after learning each task without knowing the task identifier. We apply *growing memory* with n_ϵ exemplars per class selected by Herding [37] for both *LFS* and *LFH*. The discussion about *fixed memory* setup can be found in *Appendix*.

Datasets. We evaluate our method on three public datasets including CIFAR100 [24], ImageNetSubset with 100 classes from ImageNet [39] and Food101 [7]. Specifically, we first follow [26] to construct CIFAR100-LT, ImageNetSubset-LT and Food101-LT, which are the long-tailed versions of the original balanced datasets by removing training samples with an imbalance factor [13] $\rho = \frac{n_{max}}{n_{min}} = 100$ where n_{max} and n_{min} denote the maximum and minimum number of training data per class, respectively. For all three datasets, we use $N = \{10, 20\}$ tasks for *LFS* and $N = \{5, 10\}$ tasks for *LFH*. The test sets remain unchanged with the original class-balanced distributions.

Implementation details. We utilize ResNet-32 [19] for CIFAR100-LT and ResNet-18 [19] for ImageNetSubset-LT and Food101-LT. Both feature extractor and classifiers are trained from scratch using Equation 9 in an end-to-end fashion. To ensure a fair comparison, we adopt the same class order and setups from [26] where we train 160 epochs for CIFAR100-LT with an initial learning rate of 0.1 and then reduced by a factor of 10 at the 80th and 120th epochs. For ImageNetSubset-LT and Food101-LT, we train 90 epochs with an initial learning rate of 0.1 which is reduced by 10 at the 30th and 60th epoch. A consistent batch size of 128 with an SGD optimizer is used across all experiments. For simplicity, we set our hyper-parameters with $\gamma = 1$ and $\lambda_b = 1$

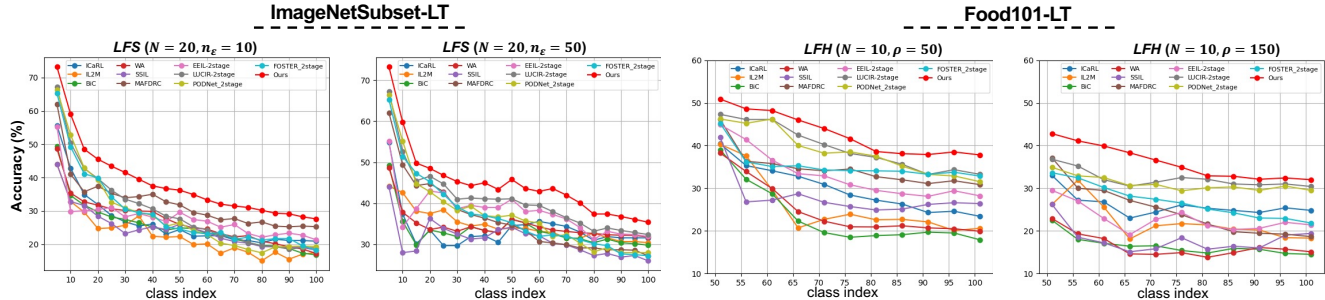


Figure 4. The classification accuracy (%) on test data belonging to all classes seen so far at each incremental step by varying the memory budget $n_\epsilon \in \{10, 50\}$ on ImageNetSubset-LT and imbalance factor $\rho \in \{50, 150\}$ on Food101-LT.

Datasets Evaluation protocol Total tasks N	CIFAR100-LT				ImageNetSubset-LT				Food101-LT			
	LFS		LFH		LFS		LFH		LFS		LFH	
	10	20	5	10	10	20	5	10	10	20	5	10
iCaRL [37]	21.83	24.28	28.68	28.33	33.75	29.71	41.82	40.21	18.13	12.50	21.83	21.31
IL2M [5]	31.37	29.99	34.90	33.42	31.70	25.20	40.75	39.08	16.11	16.27	23.93	22.48
BiC [46]	28.89	20.10	25.68	25.95	33.31	30.86	33.18	29.23	16.94	16.81	22.80	20.75
WA [53]	27.63	23.48	32.07	26.85	32.58	29.03	32.62	28.10	16.58	15.99	18.45	19.45
SSIL [1]	26.07	26.15	30.72	29.21	30.38	25.99	38.97	35.18	16.86	15.65	21.65	19.03
FOSTER [44]	30.43	29.96	37.25	37.91	34.38	29.75	46.51	43.88	24.27	20.45	32.39	31.46
MAFDRC [11]	32.67	31.95	37.94	38.51	40.01	34.48	48.23	44.12	26.93	19.21	34.22	30.91
EEIL-2stage [10, 26]	33.64	32.25	36.40	34.91	36.84	30.39	43.62	41.49	19.75	20.02	22.65	22.83
LUCIR-2stage [21, 26]	31.09	31.03	38.47	37.86	39.87	34.79	48.97	47.39	27.65	24.68	36.05	35.06
PODNet-2stage [14, 26]	30.41	30.37	38.38	38.45	35.47	31.71	48.02	47.74	23.78	21.13	35.42	35.22
FOSTER-2stage [26, 44]	31.27	30.68	40.26	39.43	36.47	33.95	48.89	46.93	25.82	22.28	35.69	33.48
Ours	35.66	34.35	40.18	39.11	45.12	40.79	50.57	49.13	29.05	26.42	36.84	36.19

Table 1. Results of average accuracy (%) on CIFAR100-LT, ImageNetSubset-LT and Food101-LT with imbalance factor $\rho = 100$, memory budget $n_\epsilon = 20$ evaluated under Learning From Scratch (LFS) and Learning From Half (LFH). **Best** and **Second Best** results are marked.

for all experiments. Each experiments are conducted three times to report average performance. All results are obtained by reproducing the original methods under the same setting based on the framework in [26, 31].

5.2. Experimental Results

Table 1 summarizes the Average Accuracy (ACC) [29] on CIFAR100-LT, ImageNetSubset-LT, and Food101-LT with imbalance factor $\rho = 100$, memory budget $n_\epsilon = 20$ per class. We observe the LFS is more challenging for CIL as the performance under LFH protocol consistently higher than LFS, underscoring the advantage of pre-training with a larger portion of classes in the initial incremental step. The 2-stage module [26] shows the effectiveness in addressing imbalanced data due to its additional learnable layer to mitigate bias. Notably, LUCIR-2stage [21], which employs a cosine classifier with normalized weights, performs well in handling imbalanced data. Our method achieves promising results under LFH and significant improvements under LFS even without necessitating extra training stages and parameters. This demonstrates the potential of our end-to-end approach for more efficient training methodologies in CIL, especially as the task complexity scales up to learn an increasing number of classes from a scratch model.

In Figure 4, we visualize the classification accuracy on

all classes seen so far at each incremental step by varying the imbalance factor $\rho \in \{50, 150\}$ and the memory budget $n_\epsilon \in \{10, 50\}$. While generally increasing the memory budget results in better performance, the improvements could be marginal for some methods such as SSIL [1] and MAFDRC [11]. This paradox is attributed to the exacerbation of exemplar set imbalance by using a larger memory budget n_ϵ where the instance-rich classes retain more exemplars than instance-rare classes. On the other hand, the increase of imbalance factor ρ results in a noticeable degradation in performance. Despite these challenges, our method is able to manage diverse learning environments posed by both memory budget and imbalance factors to consistently achieve the best performance at each incremental step.

Additional results for (i) conventional CIL setup ($\rho = 1$), (ii) long-tail recognition, and (iii) computation and memory evaluations can be found in Appendix.

5.3. Ablation Study

We evaluate each of our components including (i) the addition of our DAKD loss, (ii) the Decomposed Gradient Reweighting (DGR) to separately balance the gradients for new and learned tasks instead of treating all classes together (GR) for intra-phase imbalance, and (iii) the sig-

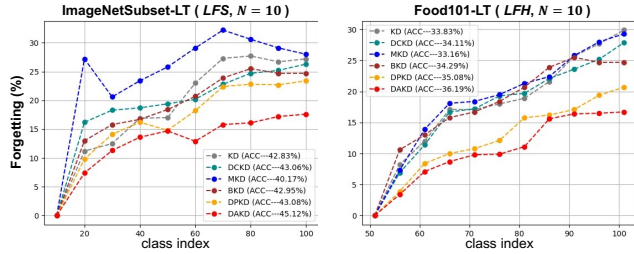


Figure 5. The Forgetting [29] (%) at each incremental step by comparing our proposed DAKD with variants of distillation. The average classification accuracy (ACC) is shown in the legend (●).

$\rho = 100$ $n_\varepsilon = 20$			Food101-LT		ImageNetSubset-LT	
DAKD	DGR	$\Delta(\text{DAKD})$	LFS $\bar{N} = 10$	LFH $\bar{N} = 5$	LFS $\bar{N} = 20$	LFH $\bar{N} = 10$
✓	(GR)		27.77	24.62	47.42	45.17
✓	✓		28.45	25.47	48.64	46.76
✓	(GR)		28.02	26.08	48.72	46.53
✓	(GR)	✓	28.96	27.14	49.59	47.37
✓	✓	✓	29.05	26.42	50.57	49.13

Table 2. Ablation study on Food101-LT and ImageNetSubset-LT

nificance of maintaining the discrimination of gradients from knowledge distillation loss ($\Delta(\text{DAKD})$) as described in Section 4.2.2. Specifically, we consider GR and original knowledge distillation (KD) as the baseline to evaluate the contributions by adding each component from our method. The results are summarized in Table 2, where we observe performance improvements contributed by each component. Notably, the GR exhibits better performance compared to DGR due to the potential underfitting of new classes which received insufficient emphasis during CIL. We also observe improved performance by maintaining the discriminativeness of the gradient from DAKD to capture and leverage information from the original training distribution, thus effectively preserving learned knowledge.

Variants of knowledge distillation loss: Additionally, we assess the effectiveness of DAKD to mitigate forgetting under CIL by replacing it with existing variants of logit-based knowledge distillation loss including (i) the balanced knowledge distillation (BKD) [51], (ii) multi-level logit distillation (MKD) [22], (iii) decoupled knowledge distillation (DCKD) [54], and (iv) decomposed knowledge distillation (DPKD) [4]. The results are visualized in Figure 5 ($\rho = 100, n_\varepsilon = 20$) where we measure the forgetting rate [29] at each incremental learning phase. Remarkably, we notice that the MKD, DCKD, and BKD achieve comparable or even worse results compared to the original KD loss as they assume the student and teacher model observe the same training distribution, which does not hold under CIL. While DPKD marks some improvements, its effectiveness remains hampered by imbalanced data. Overall, our DAKD achieves the lowest forgetting and best ACC by factoring lost training data distribution.

Bias correction effects: Finally, we examine the effective-

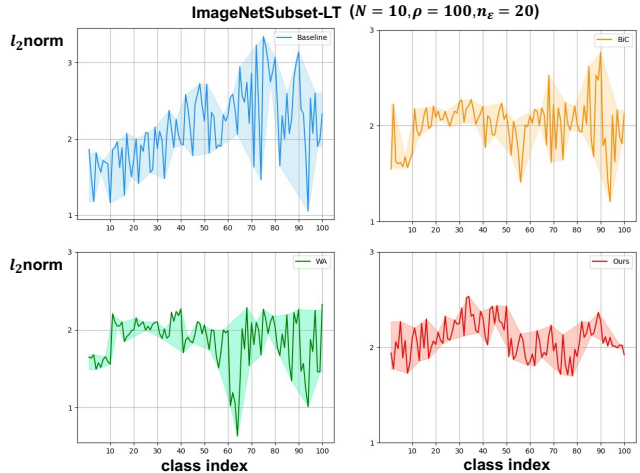


Figure 6. The l_2 norm of learned weight vectors after incrementally learning $N = 10$ tasks on ImageNetSubset-LT under LFS protocol. The shaded area shows the variations of each curve.

ness of our method in rectifying the biased weights in the fully connected layers under CIL. Specifically, we compare with Baseline (fine-tuning), BiC [46] and WA [53] to show the variance of l_2 norm for weight vectors corresponding to each class during CIL as shown in Figure 6. The Baseline method shows significant variation both within and between tasks. While BiC and WA address the variation between tasks through post-hoc bias correction, they do not effectively resolve the variation within individual tasks. Our method, in contrast, achieves more uniform weight vectors across both intra-task and inter-task learning, contributing to our best overall performance in imbalanced CIL.

6. Conclusion

In this work, we study class-incremental learning (CIL) when applied to imbalanced data to address both intra-phase and inter-phase imbalances by reweighting the gradients in FC layer to foster a balanced optimization and learn unbiased classifiers. Additionally, we introduce a distribution-aware knowledge distillation loss that dynamically modulates the loss intensity in proportion to the extent of training data attrition to further mitigate imbalanced catastrophic forgetting. Our method shows consistent improvements under CIL and proves effective in long-tailed recognition. Overall, our findings underscore the importance of addressing data imbalance in CIL and pave the way for more robust and equitable class-incremental learning models.

Acknowledgement

This project was supported by the National Institutes of Health (Grant No. U24CA268228). Special thanks are extended to Professor Fengqing Zhu for invaluable comments and advice on the manuscript, with a regrettable oversight of not being included in the author list.

References

- [1] Hongjoon Ahn, Jihwan Kwak, Subin Lim, Hyeonsu Bang, Hyojun Kim, and Taesup Moon. Ss-il: Separated softmax for incremental learning. *Proceedings of the IEEE/CVF International conference on computer vision*, pages 844–853, 2021. [2](#), [5](#), [7](#)
- [2] Rahaf Aljundi, Eugene Belilovsky, Tinne Tuytelaars, Laurent Charlin, Massimo Caccia, Min Lin, and Lucas Page-Caccia. Online continual learning with maximal interfered retrieval. *Advances in Neural Information Processing Systems*, 32, 2019. [2](#)
- [3] Rahaf Aljundi, Min Lin, Baptiste Goujaud, and Yoshua Bengio. Gradient based sample selection for online continual learning. *Advances in Neural Information Processing Systems*, 32, 2019. [2](#)
- [4] Donghyeon Baek, Youngmin Oh, Sanghoon Lee, Junghyup Lee, and Bumsuh Ham. Decomposed knowledge distillation for class-incremental semantic segmentation. *Advances in Neural Information Processing Systems*, 35:10380–10392, 2022. [8](#)
- [5] Eden Belouadah and Adrian Popescu. Il2m: Class incremental learning with dual memory. *Proceedings of the IEEE International Conference on Computer Vision*, pages 583–592, 2019. [2](#), [7](#)
- [6] Eden Belouadah, Adrian Popescu, Umang Aggarwal, and Léo Saci. Active class incremental learning for imbalanced datasets. *European Conference on Computer Vision*, pages 146–162, 2020. [2](#)
- [7] Lukas Bossard, Matthieu Guillaumin, and Luc Van Gool. Food-101 – mining discriminative components with random forests. *Proceedings of the European Conference on Computer Vision*, 2014. [6](#)
- [8] Mateusz Buda, Atsuto Maki, and Maciej A Mazurowski. A systematic study of the class imbalance problem in convolutional neural networks. *Neural networks*, 106:249–259, 2018. [3](#)
- [9] Kaidi Cao, Colin Wei, Adrien Gaidon, Nikos Arachiga, and Tengyu Ma. Learning imbalanced datasets with label-distribution-aware margin loss. *Advances in neural information processing systems*, 32, 2019. [3](#)
- [10] Francisco M. Castro, Manuel J. Marin-Jimenez, Nicolas Guil, Cordelia Schmid, and Karteek Alahari. End-to-end incremental learning. *Proceedings of the European Conference on Computer Vision*, 2018. [2](#), [7](#)
- [11] Xiuwei Chen and Xiaobin Chang. Dynamic residual classifier for class incremental learning. *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 18743–18752, 2023. [1](#), [2](#), [7](#)
- [12] Aristotelis Chrysakis and Marie-Francine Moens. Online continual learning from imbalanced data. *International Conference on Machine Learning*, pages 1952–1961, 2020. [2](#)
- [13] Yin Cui, Menglin Jia, Tsung-Yi Lin, Yang Song, and Serge Belongie. Class-balanced loss based on effective number of samples. *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 9268–9277, 2019. [3](#), [6](#)
- [14] Arthur Douillard, Matthieu Cord, Charles Ollion, Thomas Robert, and Eduardo Valle. Podnet: Pooled outputs distillation for small-tasks incremental learning. *Proceedings of the European Conference on Computer Vision*, pages 86–102, 2020. [2](#), [6](#), [7](#)
- [15] Emanuele Francazi, Marco Baity-Jesi, and Aurelien Lucchi. A theoretical analysis of the learning dynamics under class imbalance. *International Conference on Machine Learning*, pages 10285–10322, 2023. [3](#)
- [16] Yiduo Guo, Bing Liu, and Dongyan Zhao. Dealing with cross-task class discrimination in online continual learning. *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 11878–11887, 2023. [1](#), [2](#), [3](#)
- [17] Tyler L. Hayes and Christopher Kanan. Online continual learning for embedded devices. *Conference on Lifelong Learning Agents*, 2022. [1](#)
- [18] Jiangpeng He and Fengqing Zhu. Online continual learning via candidates voting. *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*, pages 3154–3163, 2022. [2](#)
- [19] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 770–778, 2016. [6](#)
- [20] Geoffrey Hinton, Oriol Vinyals, and Jeffrey Dean. Distilling the knowledge in a neural network. *Proceedings of the NIPS Deep Learning and Representation Learning Workshop*, 2015. [2](#), [5](#)
- [21] Saihui Hou, Xinyu Pan, Chen Change Loy, Zilei Wang, and Dahua Lin. Learning a unified classifier incrementally via rebalancing. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 831–839, 2019. [2](#), [6](#), [7](#)
- [22] Ying Jin, Jiaqi Wang, and Dahua Lin. Multi-level logit distillation. *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 24276–24285, 2023. [8](#)
- [23] Chris Dongjoo Kim, Jinseo Jeong, and Gunhee Kim. Imbalanced continual learning with partitioning reservoir sampling. *European Conference on Computer Vision*, pages 411–428, 2020. [2](#)
- [24] Alex Krizhevsky, Geoffrey Hinton, et al. Learning multiple layers of features from tiny images. *Technical Report*, 2009. [6](#)
- [25] Zhizhong Li and Derek Hoiem. Learning without forgetting. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 40(12):2935–2947, 2017. [2](#)
- [26] Xialei Liu, Yu-Song Hu, Xu-Sheng Cao, Andrew D Bagdanov, Ke Li, and Ming-Ming Cheng. Long-tailed class incremental learning. *European Conference on Computer Vision*, pages 495–512, 2022. [2](#), [4](#), [6](#), [7](#)
- [27] Yaoyao Liu, Yuting Su, An-An Liu, Bernt Schiele, and Qianru Sun. Mnemonics training: Multi-class incremental learning without forgetting. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 12245–12254, 2020. [2](#)

- [28] Yaoyao Liu, Bernt Schiele, and Qianru Sun. Adaptive aggregation networks for class-incremental learning. *Proceedings of the IEEE/CVF conference on Computer Vision and Pattern Recognition*, pages 2544–2553, 2021. 2
- [29] David Lopez-Paz and Marc’Aurelio Ranzato. Gradient episodic memory for continual learning. *Advances in neural information processing systems*, pages 6467–6476, 2017. 1, 2, 7, 8
- [30] Davide Maltoni and Vincenzo Lomonaco. Continuous learning in single-incremental-task scenarios. *Neural Networks*, 116:56–73, 2019. 2
- [31] Marc Masana, Xialei Liu, Bartłomiej Twardowski, Mikel Menta, Andrew D Bagdanov, and Joost Van De Weijer. Class-incremental learning: survey and performance evaluation on image classification. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 45(5):5513–5533, 2022. 7
- [32] Michael McCloskey and Neal J Cohen. Catastrophic interference in connectionist networks: The sequential learning problem. pages 109–165. Elsevier, 1989. 1
- [33] Aditya Krishna Menon, Sadeep Jayasumana, Ankit Singh Rawat, Himanshu Jain, Andreas Veit, and Sanjiv Kumar. Long-tail learning via logit adjustment. *International Conference on Learning Representations*, 2021. 3, 5
- [34] Seulki Park, Jongin Lim, Younghan Jeon, and Jin Young Choi. Influence-balanced loss for imbalanced visual classification. *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 735–744, 2021. 3
- [35] Seulki Park, Youngkyu Hong, Byeongho Heo, Sangdoon Yun, and Jin Young Choi. The majority can help the minority: Context-rich minority oversampling for long-tailed classification. *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6887–6896, 2022. 3
- [36] Ameya Prabhu, Philip HS Torr, and Puneet K Dokania. Gdumb: A simple approach that questions our progress in continual learning. *Proceedings of the European Conference on Computer Vision*, pages 524–540, 2020. 2
- [37] Sylvestre-Alvise Rebuffi, Alexander Kolesnikov, Georg Sperl, and Christoph H. Lampert. iCaRL: Incremental classifier and representation learning. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2017. 1, 2, 6, 7
- [38] Jiawei Ren, Cunjun Yu, Xiao Ma, Haiyu Zhao, Shuai Yi, et al. Balanced meta-softmax for long-tailed visual recognition. *Advances in neural information processing systems*, 33:4175–4186, 2020. 3
- [39] Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, Alexander C. Berg, and Li Fei-Fei. ImageNet Large Scale Visual Recognition Challenge. *International Journal of Computer Vision*, 115(3): 211–252, 2015. 6
- [40] Christian Simon, Piotr Koniusz, and Mehrtash Harandi. On learning the geodesic path for incremental learning. *Proceedings of the IEEE/CVF conference on Computer Vision and Pattern Recognition*, pages 1591–1600, 2021. 2
- [41] Jingru Tan, Changbao Wang, Buyu Li, Quanquan Li, Wanli Ouyang, Changqing Yin, and Junjie Yan. Equalization loss for long-tailed object recognition. *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 11662–11671, 2020. 3
- [42] Jingru Tan, Xin Lu, Gang Zhang, Changqing Yin, and Quanquan Li. Equalization loss v2: A new gradient balance approach for long-tailed object detection. pages 1685–1694, 2021. 3
- [43] Jason Van Hulse, Taghi M Khoshgoftaar, and Amri Napolitano. Experimental perspectives on learning from imbalanced data. *Proceedings of the 24th international conference on Machine learning*, pages 935–942, 2007. 3
- [44] Fu-Yun Wang, Da-Wei Zhou, Han-Jia Ye, and De-Chuan Zhan. Foster: Feature boosting and compression for class-incremental learning. *Proceedings of the European Conference on Computer Vision*, pages 398–414, 2022. 2, 7
- [45] Max Welling. Herding dynamical weights to learn. *Proceedings of the International Conference on Machine Learning*, pages 1121–1128, 2009. 2
- [46] Yue Wu, Yinpeng Chen, Lijuan Wang, Yuancheng Ye, Zicheng Liu, Yandong Guo, and Yun Fu. Large scale incremental learning. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2019. 1, 2, 5, 7, 8
- [47] Shipeng Yan, Jiangwei Xie, and Xuming He. Der: Dynamically expandable representation for class incremental learning. *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3014–3023, 2021. 2
- [48] Lu Yang, He Jiang, Qing Song, and Jun Guo. A survey on long-tailed visual recognition. *International Journal of Computer Vision*, 130(7):1837–1872, 2022. 3
- [49] Jaehong Yoon, Eunho Yang, Jeongtae Lee, and Sung Ju Hwang. Lifelong learning with dynamically expandable networks. *International Conference on Learning Representations*, 2018. 2
- [50] Friedemann Zenke, Ben Poole, and Surya Ganguli. Continual learning through synaptic intelligence. *International Conference on Machine Learning*, pages 3987–3995, 2017. 2
- [51] Shaoyu Zhang, Chen Chen, Xiyuan Hu, and Silong Peng. Balanced knowledge distillation for long-tailed learning. *Neurocomputing*, 527:36–46, 2023. 8
- [52] Yifan Zhang, Bingyi Kang, Bryan Hooi, Shuicheng Yan, and Jiashi Feng. Deep long-tailed learning: A survey. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2023. 3
- [53] Bowen Zhao, Xi Xiao, Guojun Gan, Bin Zhang, and Shu-Tao Xia. Maintaining discrimination and fairness in class incremental learning. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 13208–13217, 2020. 1, 2, 5, 6, 7, 8
- [54] Borui Zhao, Quan Cui, Renjie Song, Yiyu Qiu, and Jiajun Liang. Decoupled knowledge distillation. *Proceedings of the IEEE/CVF Conference on computer vision and pattern recognition*, pages 11953–11962, 2022. 5, 8