Overcoming Forgetting Catastrophe in Quantization-Aware Training

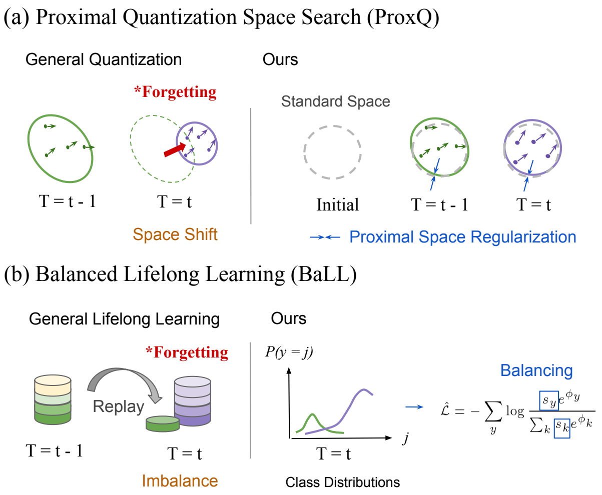
Ting-An Chen[1,2], De-Nian Yang[2,3], Ming-Syan Chen[1,3]
[1]Graduate Institute of Electrical Engineering, National Taiwan University, Taiwan
[2]Institute of Information Science, Academia Sinica, Taiwan
[3]Research Center for Information Technology Innovation, Academia Sinica, Taiwan
tachen@arbor.ee.ntu.edu.tw, dnyang@iis.sinica.edu.tw, mschen@ntu.edu.tw

Abstract

Quantization is an effective approach for memory cost reduction by compressing networks to lower bits. However, existing quantization processes learned only from the current data tend to suffer from forgetting catastrophe on streaming data, i.e., significant performance decrement on old task data after being trained on new tasks. Therefore, we propose a lifelong quantization process, LifeQuant, to address the problem. We theoretically analyze the forgetting catastrophe from the shift of quantization search space with the change of data tasks. To overcome the forgetting catastrophe, we first minimize the space shift during quantization and propose Proximal Quantization Space Search (ProxQ), for regularizing the search space during quantization to be close to a pre-defined standard space. Afterward, we exploit replay data (a subset of old task data) for retraining in new tasks to alleviate the forgetting problem. However, the limited amount of replay data usually leads to biased quantization performance toward the new tasks. To address the imbalance issue, we design a Balanced Lifelong Learning (BaLL) Loss to reweight (to increase) the influence of replay data in new task learning, by leveraging the class distributions. Experimental results show that LifeQuant achieves outstanding accuracy performance with a low forgetting rate.

1. Introduction

With increasing requirements for real-time inferences in computer vision tasks [1, 2, 3], neural networks deployed on edge devices have received increasing attention [4]. Due to the limited memory storage on edge devices, networks with a large volume of parameters are required to be compressed [5]. Accordingly, in addition to pruning [6, 7, 8, 9] and structure simplification [10, 11], *quantization* has been developed as an efficient learning technique to effectively compress networks to lower bits without a significant performance loss [12, 13, 14, 15, 16, 17].



(a) Proximal Quantization Space Search (ProxQ)

(b) Balanced Lifelong Learning (BaLL)

General Lifelong Learning
*Forgetting
Replay
T = t - 1
T = t
Imbalance
Ours
P(y = j)
j
T = t
Class Distributions
Balancing
$\hat{\mathcal{L}} = -\sum_y \log \frac{\boxed{s_y} e^{\phi_y}}{\sum_k \boxed{s_k} e^{\phi_k}}$

Figure 1: Overview of *LifeQuant*. In Fig. (a), *Proximal Quantization Space Search (ProxQ)* is proposed to overcome the forgetting catastrophe problem by minimizing space shift, i.e., regularizing the search space to be close to a pre-defined standard space, during the quantization process. In Fig. (b), previous lifelong learning (LL) research employs replay data (old task data) in new task training to alleviate the forgetting problem. However, the limited amount of replay data inhibits the efficacy of overcoming the forgetting catastrophe. Therefore, *Balanced Lifelong Learning (BaLL) loss* is designed to reweight (to increase) the influence of replay data in new tasks, by carefully examining the class data distributions, to avoid significant forgetting catastrophe.

Quantization can be categorized as *Post-Training Quantization (PTQ)* and *Quantization-Aware Training (QAT)* according to the training process [18]. PTQ compresses the pretrained full-precision model weights and activations into low bits in a deterministic way without retraining and fine-tuning, which incurs only a tiny overhead in training [12, 13, 19, 20, 21]. However, PTQ usually suffers from significant accuracy degradation since the quantization criteria are not trained with weights [18]. In contrast, QAT learns model weights during quantization, i.e., the training loss is able to be measured for the update of weights [14, 22, 15, 16, 23]. Despite additional operations in training, QAT generally achieves a better performance [18].

To the best of our knowledge, existing quantization approaches in PTQ or QAT are designed to minimize the quantization error, i.e., the discrepancy between full-precision values and low-bit values, according to the training data only in the current storage. However, in real-world applications, data collected from the edge, such as Internet of Things (IoT) products, are *streaming* [24]. Owing to limited memory storage on devices, it is infeasible to preserve all training data. In other words, the training is mainly performed on the new data while the old training data are eliminated, which induces the biased quantization result toward the new tasks and gradually forgets the prediction of the old tasks, i.e., the *forgetting catastrophe problem*. Nevertheless, it has not been investigated thoroughly. Therefore, in this paper, we make the first attempt to explore the forgetting problem in quantization on streaming data. We first demonstrate that existing quantization processes suffer from a *forgetting catastrophe* on streaming data, i.e., significant performance deterioration on old task data after being quantized on new tasks.

To overcome the forgetting catastrophe, we design a lifelong quantization process, *LifeQuant*, to robustly learn low-bit models on streaming data with a smaller forgetting rate, i.e., the accuracy degradation after learning the new tasks. Fig. 1 illustrates the motivations of LifeQuant. Fig. 1 (a) first shows that the forgetting catastrophe mainly results from the shift of search space in quantization after learning the new task data. We theoretically analyze the increment of quantization error under the change in weights to evaluate the forgetting performance. To avoid the search space biased by new tasks, we target *space shift minimization* and propose *Proximal Quantization Space Search (ProxQ)* to regularize the search space during quantization to be close to a pre-defined standard space (i.e., the gray dashed circle in Fig. 1 (a)), by leveraging the statistics (mean and variance) of the weights. Accordingly, the space shift can be effectively reduced under the change of data tasks to overcome the forgetting catastrophe.

In addition to space shift minimization, Fig. 1 (b) shows that recent lifelong learning (LL) research alleviates the forgetting problem in full-precision network training by applying replay data (training data in old tasks) to the new task learning [25]. However, in the quantization process, only a limited amount of old task data can be stored as replay data for memory efficiency. Fig. 1 (b) illustrates that a limited amount of replay data poses a challenge, *imbalance issue* [26], where the quantization performance is inclined to be biased toward the new tasks due to the majority of the new task data. To alleviate the forgetting problem induced by the *minor quantity of replay data*, we design a *Balanced Lifelong Learning (BaLL) loss* to reweight (to increase) the influence of replay data in new tasks, by leveraging the class data distributions.

In experiments, LifeQuant improves the state-of-the-art quantization approaches by a 7% accuracy increment and 8% forgetting rate reduction for 2-bit ResNet-20 on CIFAR-100, while by a 17% accuracy improvement and 23% forgetting rate reduction for 3-bit MobileNet-V2 on ImageCLEF.

Our contributions are summarized as follows:

1. We make the first attempt to develop a novel lifelong quantization process, *LifeQuant*, to overcome the forgetting catastrophe in quantization-aware training.

2. We theoretically analyze the forgetting problem caused by the search space shift with the change of data tasks. Thus, we propose *Proximal Quantization Space Search (ProxQ)* to regularize the shift during quantization to avoid a significant accuracy loss in old tasks.

3. We study the limited quantity of replay data that induces the biased prediction result toward the new tasks and design a *Balanced Lifelong Learning (BaLL) loss* to reweight the influence of the replay data, to alleviate the forgetting problem.

4. Experimental results demonstrate that LifeQuant achieves significant accuracy enhancement and forgetting rate reduction compared with the state-of-the-art quantization approaches.

2. Related works

Post-training Quantization (PTQ). PTQ compresses pretrained full-precision models to the low bits under deterministic quantization criteria without retraining or finetuning, which generates only a tiny overhead [12, 13, 19, 20, 27]. ACIQ [12] focused on the design of clipping bounds to discretize weights and activations with a smaller quantization error, i.e., the discrepancy between the uncompressed floating-point values and the quantized values. OMSE [13] minimized the quantization error by regularizing the L2-norm of the error. OCS [19] halved channels to remove outliers and preserve valuable information in prediction under quantization. AdaRound [20] improved the typical round-to-nearest approach by an adaptive rounding operation to reduce quantization errors. Mr. BiQ [27] targeted minimizing the reconstruction error, i.e., the difference of accumulated multiplications of convolutional layers before and after quantization. However, PTQ usually suffers from significant accuracy degradation since the quantization criteria are not trained with weights [18].

Quantization-aware Training (QAT). In contrast to PTQ, QAT learns quantization criteria with model weights, where the training loss of the quantized model is evaluated for the subsequent update of weights [14, 22, 16, 28, 23]. Although additional training iterations are required for convergence, QAT generally achieves a better performance than PTQ [18]. LSQ [14] proposed a new gradient estimation approach for the non-differentiable quantization functions and learned the scaling of activation functions such as *ReLU* [29]. LLSQ [22] learned the parameters of batch normalization layers to modify the feature distributions for quantization error reduction. Qimera [16] designed a generator to synthesize boundary samples to enhance the prediction performance under quantization. IntraQ [28] preserved the property of intra-class heterogeneity during quantization to enhance the performance. AlignQ [23] minimizes the discrepancy between training and testing data distributions to adapt the trained criterion to the non-i.i.d testing data for inference. However, the existing QAT processes are learned and validated according to only the current data while ignoring the change in data tasks, i.e., the circumstances of streaming data. In this paper, we investigate the *forgetting catastrophe* problem in QAT on streaming data, i.e., inevitably significant performance deterioration on the old tasks after being trained on new task data.

Lifelong Learning (LL). Related research developed to alleviate the forgetting problem is lifelong learning (LL). However, they target full-precision operations, instead of the quantization process, an efficient learning procedure, under the memory constraint. EWC [30], as the very beginning LL approach, proposed to penalize the important parameters with a large change in new tasks, where the importance is evaluated by layer-wise fisher information based on replay data. SI [31] introduced intelligent synapses to memorize old task information for forgetting reduction. MAS [32], inspired by SI, stored task-relevant information but targeted unsupervised learning. RWalk [33]was designed as a generalized EWC that exploited KL-divergence [34] in Riemannian Manifold [35], instead of fisher information. SCP [36] aimed to preserve the layer-wise distributions under the regularization of the Sliced Cramer distance. PFR [37] designed a loss to distill the feature information of old data tasks into new tasks. In summary, the existing LL approaches generally employ *replay data* (old task data) in new task training to alleviate the forgetting problem. However, for quantization in efficient learning, only a limited amount of data can be employed due to memory constraints. It poses an *imbalance issue* [26], where the models are inclined to be biased toward the new tasks due to the majority of the new task data, leading to the forgetting catastrophe.

3. Notations and preliminaries

The quantization criteria in QAT are learned with model weights and can be evaluated by the training loss. Therefore, QAT usually has a superior performance over PTQ under deterministic designs. In this section, we first introduce the problem formulation in quantization-aware training (QAT) and prescribe the notations.

QAT is to learn a quantization criterion Q to minimize the quantization error $\mathbb{E}(||\mathbf{w}^q - \mathbf{w}||_2)$, i.e., the discrepancy (L2-norm distance) between the floating-point weights $\mathbf{w}$ and the quantized weights $\mathbf{w}^q$ derived from $\mathbf{w}^q = Q(\mathbf{w})$. In other words, QAT finds a minimal quantization error to avoid significant performance degradation during the quantization. In this paper, we further consider a gradient term $\frac{\partial \mathscr{L}}{\partial \mathbf{w}^q}$, where $\mathscr{L}$ denotes the training loss since gradient indicates the influence of weights on the prediction loss. Our idea is to penalize the weights not only with large quantization errors but also with huge impacts on the prediction result. Therefore, we target minimizing $\mathbb{E}(||(\mathbf{w}^q - \mathbf{w}) \cdot \frac{\partial \mathscr{L}}{\partial \mathbf{w}^q}||_2)$ and prove that this new objective is equivalent to the minimization of the performance loss as follows.

Theorem 3.1. *(*Proved in Appendix A.1*) Let $(\boldsymbol{x}^T, y)$ be the (feature, label) of input data, $\mathscr{L}$ denote the training loss, $\boldsymbol{w}$ indicate the weights, and $\boldsymbol{w}^q$ represent the quantized weights. Then $\arg\min_{\boldsymbol{w}^q} \mathbb{E}(||(\boldsymbol{w}^q - \boldsymbol{w}) \cdot \frac{\partial \mathscr{L}}{\partial \boldsymbol{w}^q}||_2) \simeq \arg\min_{\boldsymbol{w}^q} |\mathscr{L}((\boldsymbol{x}^T, y; \boldsymbol{w}^q) - \mathscr{L}((\boldsymbol{x}^T, y; \boldsymbol{w})|$, where $|\mathscr{L}((\boldsymbol{x}^T, y; \boldsymbol{w}^q) - \mathscr{L}((\boldsymbol{x}^T, y; \boldsymbol{w})|$ is the performance decrement after quantization.*

According to Theorem 3.1, the quantization performance is able to be better evaluated with an additional gradient term. Therefore, we re-define the quantization error as $\mathbb{E}(||(\mathbf{w}^q - \mathbf{w}) \cdot \frac{\partial \mathscr{L}}{\partial \mathbf{w}^q}||_2)$ to measure the performance in this study.

4. Forgetting problem in QAT

Existing QAT introduced in Sec. 2 is learned and validated according to only the current data and thereby suffers from a *forgetting catastrophe* on streaming data with varying data distributions. In this section, we first theoretically prove the forgetting problem in QAT and defined quantization error in Sec. 3 and then experimentally validate the problem by evaluating the accuracy performance on real-world data.

4.1. Theoretical analysis

Sec. 3 has indicated that a large quantization error generated in QAT induces a significant performance loss. Accordingly, to evaluate the forgetting problem, in this subsection, we first investigate if the quantization error on old data increases after the new task learning. The quantization error in Definition 4.1 is measured on the current data, i.e., a single data task. In the following, we define the quantization error on streaming data with multiple data tasks (in different data distributions) and then evaluate the change in quantization error across multiple data tasks.

Definition 4.1. (***Multi-task quantization error.***) *Quantization error on the s-th task data $(\mathbf{x}_s^T, y_s)$ of the low-bit model learned from the t-th task data $\mathbf{w}_t^q$ is denoted as $\xi(\mathbf{x}_s^T; \mathbf{w}_t^q)$*

which is defined as $\mathbb{E}(||(\mathbf{w}_t^q - \mathbf{w}_t) \cdot \frac{\partial \mathscr{L}(\mathbf{x}_s;\mathbf{w}_t^q)}{\partial \mathbf{w}_t^q}||_2)$, where $\frac{\partial \mathscr{L}(\mathbf{x}_s;\mathbf{w}_t^q)}{\partial \mathbf{w}_t^q}$ represents the gradient $\frac{\partial \mathscr{L}(\mathbf{x}_s^T, y_s;\mathbf{w}_t^q)}{\partial \mathbf{w}_t^q}$, $\forall s \leq t$.

After the definition of quantization error across tasks, we evaluate the forgetting problem under the demonstration for streaming data as follows.

Proposition 1. The quantization error on old task data increases after the low-bit model learns the new task data. (proved in Theorem 4.1).

Theorem 4.1. (Proved in Appendix A.2) *Based on Definition 4.1, and denote* $\boldsymbol{w}_s^{q*}$ *as the optimal solution on task* s*, then* $\xi(\boldsymbol{x}_s^T; \boldsymbol{w}_t^{q*}) \geq \xi(\boldsymbol{x}_s^T; \boldsymbol{w}_s^{q*}), \forall s \leq t$.

Theorem 4.1 proves that the quantization error increases on the old task after the new task learning, since the solution $\mathbf{w}_s^{q*}$ is learned from the data in task s $\mathbf{x}_s^T$. However, the weights are usually converged to another solution other than the optimum when data change. Thus, a larger quantization error of a new model on old task data is obtained, i.e., the forgetting problem. We only show the existence of the forgetting problem in quantization here and will derive the upper bound of the forgetting performance in Sec. 5.1.

4.2. Exploration analysis

In this subsection, we investigate the forgetting problem in real-world data. Table 1 presents the performances before and after learning new tasks under (8, 4, 2)-bit uniform quantization for ResNet-20 [38] on split CIFAR-100 [39][1]. The experimental results validate a more significant accuracy loss on task 1 (the old data task), which is consistent with the forgetting problem analyzed in Sec. 4.1. Moreover, the forgetting problem is particularly notable under low-bit quantization. For instance, the 2-bit model obtains as much as 69.86% accuracy degradation on task 1 after learning tasks 2 and 3. Therefore, it is imperative to design a robust quantization process to overcome the forgetting catastrophe.

5. LifeQuant

In this section, we study two fundamental issues that induce the forgetting catastrophe in quantization and propose *LifeQuant* to overcome the problem. In Sec. 5.1, we first derive the upper bound of the increment of the quantization error for the problem due to the shift of the search space during quantization. Afterward, in Sec. 5.2, we propose *Proximal Quantization Space Search (ProxQ)* to impose a regularization on the space shift to overcome the forgetting problem. In Sec. 5.3, we further investigate the imbalance issue where the influence of old task data in new task learning is underestimated. Accordingly, we design a *Balanced Lifelong Learning (BaLL) loss* in Sec. 5.4 to reweight data losses to alleviate the forgetting problem.

Table 1: Accuracy (%) of low-bit ResNet-20 [38] on CIFAR-100 [39] under uniform quantization-aware training. *Curr. Task* represents the performance of the current task before learning new tasks. *Multi. Tasks* manifests the performance after learning multiple tasks. The learning order is Task 1 → Task 2 → Task 3.

Bits	Accuracy (%)	Task 1	Task 2	Task 3	Avg.
8	Curr. Task	79.61	76.88	66.20	74.23
	Multi. Tasks	32.81	46.93	61.97	47.24
	Drop Rate	**-58.79**	-38.96	-6.39	-34.71
4	Curr. Task	73.65	65.86	60.06	66.52
	Multi. Tasks	25.14	39.49	46.20	36.94
	Drop Rate	**-65.87**	-40.04	-23.08	-42.99
2	Curr. Task	68.96	60.51	53.90	61.12
	Multi. Tasks	21.02	36.55	45.40	34.32
	Drop Rate	**-69.52**	-39.60	-15.77	-41.63

5.1. Space shift issue

In this subsection, we investigate the forgetting problem by analyzing the upper bound of the increment of the quantization error based on Theorem 4.1. We demonstrate that *the shift of the search space* during quantization increases the quantization error in the following proposition.

Proposition 2. The increment of the quantization error mainly results from the shift of the search space during quantization under the change of data tasks. (proved in Theorem 5.1).

Theorem 5.1. (Proved in Appendix A.3) *Based on Definition 4.1 and Theorem 4.1, the increment of the quantization error is* $\xi(\boldsymbol{x}_s^T; \boldsymbol{w}_t^q) - \xi(\boldsymbol{x}_s^T; \boldsymbol{w}_s^q)$ *which has an upper bound* $\mathbb{E}(||(\boldsymbol{w}_s^q - \boldsymbol{w}_s) \cdot \frac{\partial \mathscr{L}(\boldsymbol{x}_s;\boldsymbol{w}_t^q)}{\partial \boldsymbol{w}_t^q}||_2) + \mathbb{E}(||(\boldsymbol{w}_t^q - \boldsymbol{w}_s^q) \cdot \frac{\partial \mathscr{L}(\boldsymbol{x}_s;\boldsymbol{w}_t^q)}{\partial \boldsymbol{w}_t^q}||_2)$*,* $\forall s \leq t$.

Theorem 5.1 derives the upper bound of increment of the quantization error. The first term includes $||\mathbf{w}_s^q - \mathbf{w}_s||_2$, which is the within-task quantization error that existing QAT aims to minimize (see preliminaries in Sec. 3). In other words, the first term of the bound can be minimized by general QAT processes., and the forgetting problem mainly originates from the second term, where $||\mathbf{w}_t^q - \mathbf{w}_s^q||_2$ represents the shift of the search space under the change of data tasks (from task s to t) (illustrated in Fig. 1 (a)). Therefore, we aim to minimize the space shift to effectively reduce the increasing quantization error to overcome the forgetting catastrophe.

5.2. Proximal Quantization Space Search (ProxQ)

To minimize the space shift during quantization, we propose *Proximal Quantization Space Search (ProxQ)* to im-

[1] CIFAR-100 is split to three tasks in Table 1 under the setting of the parameter $\gamma = 25$ which indicates 25% of class data change between tasks. The details will be described in Sec. 6.

pose regularization on the search space after the update of weights by Gaussian projection and penalty imposed on the increasing quantization error derived from Theorem 5.1. To efficiently minimize the shift, our idea in Fig. 1 (a) is to individually regularize the search spaces in all tasks to be close to a pre-defined standard space with the bounds $[-\alpha, \alpha]$, where $\alpha > 0$[2].

5.2.1 Pre-definition of the standard space

According to the normality of converged quantized weights demonstrated in previous research [12, 15, 23], we adopt *Gaussian* space $N(\mathbf{0}, \Sigma)$ as the standard space for regularizing the search space during quantization, where the covariance matrix $\Sigma \simeq (\frac{\alpha}{3})^2 I$ since nearly 99.7% weights are located in the range $[-3 \cdot diag(\Sigma^{-\frac{1}{2}}), 3 \cdot diag(\Sigma^{-\frac{1}{2}})]$ according to the $(68 - 95 - 99.7)$-*rule* in probability theory [40]. The notation $diag$ represents the diagonal elements, i.e., $diag(\Sigma^{-\frac{1}{2}})$ indicates a vector of the standard deviations of the weights. Accordingly, after the search space during quantization is projected to $N(\mathbf{0}, \frac{\alpha^2}{9} I)$, the weights are guaranteed located within $[-\alpha, \alpha]$ at 99.7% confidence level.

5.2.2 Proximal space regularization and quantization

During training, after the weights in the t-th task $\mathbf{w}_t$ are updated, we project the space to the standard Gaussian space $N(\mathbf{0}, \frac{\alpha^2}{9} I)$ (defined in Sec. 5.2.1) by *Proximal Space Regularization*, $Prox$, which is formulated as:

$$\mathbf{w}_t^{std} = Prox(\mathbf{w}_t) = \frac{\alpha}{3} \Sigma_t^{-\frac{1}{2}} (\mathbf{w}_t - \mathbf{m}_t), \forall t \qquad (1)$$

where $\mathbf{m}_t$ is the mean vector of the weights $\mathbf{w}_t$ in task t, Σ_t represents the covariance matrix of $\mathbf{w}_t$, and $\mathbf{w}_t^{std}$ denotes the weights regularized on the standard space.

After the regularization in Eq. (1), the quantized weights in separate tasks are restricted within the range $[-\alpha, \alpha]$ at a high confidence level. We then quantize the weights $\mathbf{w}_t^{std}$ under the uniform quantization scheme Q to the discrete values $\{-\alpha, -\alpha+\Delta, -\alpha+2\Delta, ..., \alpha-2\Delta, \alpha-\Delta, \alpha\}$. The interval Δ is $\frac{2\alpha}{2^b-1}$, where b represents the b-bit quantization. In other words, the quantized weights can be derived from

$$\mathbf{w}_t^{proxq} = Q(\mathbf{w}_t^{std}) = \frac{round(\tau \cdot \mathbf{w}_t^{std})}{\tau}, \forall t \qquad (2)$$

where $round$ manifests the rounding operation, and τ represents the total number of quantization intervals $(2^b - 1)$.

5.2.3 Fine-grained regularization in backward process

According to Sec. 5.2.2, in the forward process, the quantized weights in separate tasks under ProxQ are projected to the same standard space. Thus, we are able to effectively reduce the space shift to $||(\mathbf{w}_t^{proxq} - \mathbf{w}_s^{proxq})||_2$. However, the increment of quantization error derived from Theorem 5.1 is $\mathbb{E}(||(\mathbf{w}_t^{proxq} - \mathbf{w}_s^{proxq}) \cdot \frac{\partial \mathscr{L}(\mathbf{x}_s;\mathbf{w}_t^q)}{\partial \mathbf{w}_t^q}||_2)$ with an additional gradient term. Therefore, we further regularize the error by a loss in the backward process formulated as follows.

$$\mathscr{L}_{prox} = \sum_{\forall s<t} ||(\mathbf{w}_t^{proxq} - \mathbf{w}_s^{proxq}) \cdot \frac{\partial \mathscr{L}(\mathbf{x}_s; \mathbf{w}_t^q)}{\partial \mathbf{w}_t^q}||_2. \qquad (3)$$

5.3. Imbalance data issue

In Sec. 5.2, we introduce ProxQ to address the space shift issue (analyzed in Sec. 5.1) to overcome the forgetting catastrophe. In addition to space shift minimization, reducing the changes in data distribution in separate tasks has been demonstrated as an effective approach to the forgetting problem. As presented in Fig. 1 (b) and introduced in Sec. 2, recent lifelong learning (LL) research is designed for full-precision model operations to address the forgetting issue on old tasks by employing replay data (training data in old tasks) retrained in new tasks for accuracy enhancement.

However, under the quantization process for efficient learning, only a limited amount of old task data can be stored as replay data due to memory constraints. Accordingly, it poses a challenge, the *imbalance issue* [26]. In the following, we demonstrate that the prediction result with the limited quantity of replay data is biased toward the new tasks, by leveraging the class distributions as described in Fig. 1 (b). In other words, the forgetting problem still exists.

Proposition 3. The prediction result with the limited quantity of replay data for retraining is biased toward the new tasks. (analyzed in Theorem 5.2).

Theorem 5.2. (Proved in Appendix A.4) *Let* $\pi_j^{t|s} := P_{Y_t|\{X_t, X_s^{replay}\}}(y_t = j|x_t)$ *stand for the prediction probability of the* t*-th task data* x_t *on the* j*-th class, incorporated with the training of replay data from the* s*-th task* X_s^{replay}*, where* $s < t$*, and* $\pi_j^t := P_{Y_t|X_t}(y_t = j|x_t)$ *represent the prediction probability without replay data. Denote* n_j^t *as the sample size of* X_t *on the* j*-th class and* r_j^s *as the sample size of* X_s^{replay} *on the* j*-th class. If* $r_j^s \leq \delta_j n_j^t$*,* $\forall j$*, where* $\delta_j \in [0, 1]$ *is the replay ratio on the* j*-th class and* r_j^s*, then* $|\pi_j^{t|s} - \pi_j^t| < \delta_j \cdot (1 + \frac{\sum_{i=1}^K \delta_i n_i^t}{\sum_{i=1}^K n_i^t})$*,* $\forall j, \forall s < t$*.*

Theorem 5.2 demonstrates that when a limited amount of replay data is employed for retraining, i.e., replay ratio of δ_j is small, $\forall j$, then the prediction probability based on the replay data $\pi_j^{t|s}$ is close to the result only based on new task data π_j^t. In other words, the prediction result is biased toward new tasks. Accordingly, the forgetting problem is not solved by the limited replay data.

[2] The setting of the hyper-parameter α will be compared in Appendix C.

5.4. Balanced Lifelong Learning (BaLL) loss

According to Sec. 5.3, the limited amount of replay data inhibits the efficacy of overcoming the forgetting problem. Thus, to strengthen the influence of the minority replay data in new task learning, in this subsection, we aim to reweight the losses of data in prediction as illustrated in Fig 1 (b).

Based on the analysis in Theorem 5.2, the prediction result depends on the class distributions. The imbalanced class distributions lead to biased prediction performance. To rebalance the influence of replay data in new tasks, as presented in Fig 1 (b), we leverage the class distributions and reweight the prediction loss from the original $\mathscr{L}_{pred} = -\sum_y \log \frac{e^{\phi_y}}{\sum_k e^{\phi_k}}$[3] to the balanced one, named *Balanced Lifelong Learning (BaLL)* loss,

$$\mathscr{L}_{BaLL} = -\sum_y \log \frac{s_y e^{\phi_y}}{\sum_k s_k e^{\phi_k}}, \tag{4}$$

where ϕ_y represents the prediction result (not normalized) on the class y, and s_y is the *rebalancing factor* on the class y derived as follows.

Proposition 4. The factor s_y in $\mathscr{L}_{BaLL}$ which enforces the prediction result rebalanced is derived in Theorem 5.3.

Theorem 5.3. (Proved in Appendix A.5) *Based on Theorem 5.2, suppose there are total K classes in $\{X_t, X_s^{replay}\}$. Let the original prediction loss be $\mathscr{L}_{pred} = -\sum_{j=1}^{K} \log \pi_j = -\sum_{j=1}^{K} \log \frac{e^{\phi_j}}{\sum_{k=1}^{K} e^{\phi_k}}$, where ϕ_j is the prediction result on the j-th class. Assume that ϕ_j under imbalanced class distribution p_j approximates to the balanced result ϕ_j^* after adding a rebalancing term $\log(s_j)$, i.e., $\phi_j + \log(s_j) = \phi_j^*$. Then the balanced loss is $\mathscr{L}_{BaLL} = -\sum_{j=1}^{K} \log \frac{s_j e^{\phi_j}}{\sum_{k=1}^{K} s_k e^{\phi_k}}$, where $s_j = e^{\phi_j(\frac{1}{Kp_j}-1)}, \forall j$.*

Theorem 5.3 demonstrates the rebalanced prediction result under the balanced prediction loss. The rebalancing factor $s_j = e^{\phi_j(\frac{1}{Kp_j}-1)}$ grows when p_j is small, i.e, few data samples. Therefore, the influence of the replay data, especially the classes rarely shown in the new tasks, is increased. Accordingly, the performance no longer mainly depends on the new task data. The forgetting problem is therefore able to be effectively alleviated.

In summary, the training loss of LifeQuant consists of two parts: 1) $\mathscr{L}_{Prox}$: the fine-grained regularization on the increasing quantization error in cross-task learning (see Sec. 5.2.3), and 2) $\mathscr{L}_{BaLL}$: the balanced prediction loss for reweighting the influence of replay data in new tasks (see Eq. 4), i.e., $\mathscr{L}_{LifeQuant} = \mathscr{L}_{Prox} + \mathscr{L}_{BaLL}$. Note that the gradient term in $\mathscr{L}_{Prox}$ (see Eq. 3) is derived from the backward propagation of $\mathscr{L}_{BaLL}$.

6. Experiments

6.1. Experiment settings

Datasets. We evaluate LifeQuant on three datasets, CIFAR-100 [39], Office-31 [41], and ImageCLEF [42]. Following [43, 33, 37], we split CIFAR-100 into three tasks by class with a parameter γ, representing the ratio of class changes when switching to the next task. In contrast to CIFAR-100 with a single domain, the other two benchmark datasets contain multiple domains (tasks) under different class and feature distributions. Office-31 has three domains, Amazon (A), DSLR (D), and WebCam (W), while ImageCLEF is combined with the benchmarks, Bing (B) [44], Caltech-256 (C) [45], ImageNet ILSVRC 2012 (I) [46] and Pascal VOC 2012 (P) [47].

Architectures. We adopt the widely applied CNN architectures ResNets (ResNet-20 and ResNet-50) [38], and MobileNet-V2 [10] with efficient structure designs for quantization to validate LifeQuant.

Evaluation metrics. Two metrics are evaluated: 1) *Accuracy*: mean accuracy of tasks, and 2) *Forgetting (Rate)* [25]: mean accuracy drop across tasks, as shown in Table 1. Higher accuracy and a lower forgetting rate indicate better performance in overcoming the forgetting problem.

Training. We implement our approaches on NVIDIA Tesla V100 GPU and a GTX 2080Ti[4]. The maximum number of epochs is 50 for each task. The batch size is set to 128 for CIFAR-100 and 32 to 256 for Office-31 and ImageCLEF domains. The learning rate decays from 0.04 to 0.001.

6.2. Comparison results

Table 2 compares LifeQuant with baseline and state-of-the-art QAT research [14, 22, 16, 28, 23]. We also compare the BaLL design of LifeQuant with existing lifelong learning research. The results are presented and discussed in Sec. 7.4.

Class-based multi-task quantization (CIFAR-100). The first four columns of Table 2 show the quantization results for 4-bit and 2-bit ResNet-20 on CIFAR-100 under two settings of γ. For instance, $\gamma = 25$ indicates the change in 25% classes when the task switches. First, we observe that the low-bit quantization reduces the prediction accuracy. In addition, the significant change in task data, i.e., a large γ increases the performance loss and forgetting rate. The results manifest that the process of LifeQuant under a huge data change and low-precision training achieves outstanding performances compared with the prior works. For example, LifeQuant-based 2-bit ResNet-20 with $\gamma = 50$ enhances the accuracy by 6% to 26% and reduces the forgetting rate by 9% to 40%. The outstanding performances validate the effectiveness of LifeQuant on accuracy loss reduction in learning new tasks, which is mainly due to 1) the space shift

[3] The negative log-likelihood based on the softmax is a typical form of prediction loss in classification [29].

[4] Code is available at `https://github.com/tinganchen/LifeQuant.git`.

Table 2: Quantization results of ResNets and MobileNet-V2 on CIFAR-100, Office-31 and ImageCLEF. γ represents the ratio of class data changes when the task switches. Both model weights and activations are quantized to low bits. The symbol * indicates failed prediction. The improvements over 5% (10%) are presented in **blue** (**red**).

Metrics	**Methods**	ResNet-20 on CIFAR-100				ResNet-50 on Office-31				MobileNet-V2 on ImageCLEF			
		$\gamma = 25$		$\gamma = 50$		A $\rightarrow$ D $\rightarrow$ W		W $\rightarrow$ D $\rightarrow$ A		I $\rightarrow$ P $\rightarrow$ C		B $\rightarrow$ C $\rightarrow$ I	
		4 bit	2 bit	4 bit	2 bit	4 bit	2 bit	4 bit	2 bit	4 bit	3 bit	4 bit	3 bit
Accuracy (%)	LSQ [14]	35.44	35.00	24.08	23.30	32.71	28.03	20.51	15.43	23.61	23.89	20.83	19.72
	LLSQ [22]	36.11	*	24.49	*	42.30	40.28	17.95	15.83	24.44	7.78	24.44	17.22
	Qimera [16]	47.52	11.37	29.84	9.70	*	*	*	*	*	*	*	*
	IntraQ [28]	34.18	*	19.03	*	*	*	11.09	12.15	7.22	8.61	11.39	8.61
	AlignQ [23]	47.85	42.58	31.16	27.97	46.10	39.06	22.37	16.06	67.78	57.22	64.17	57.50
	LifeQuant (Ours)	**50.15**	**46.21**	**36.94**	**34.32**	**48.54**	**46.54**	**25.65**	**17.23**	**77.50**	**76.38**	**76.39**	**76.11**
Forgetting (%)	LSQ [14]	44.58	39.23	61.24	58.52	58.88	65.15	70.29	64.37	72.86	71.33	72.81	73.69
	LLSQ [22]	43.60	*	60.60	*	47.27	49.84	71.02	65.20	71.89	90.57	68.61	76.52
	Qimera [16]	25.67	80.28	51.94	82.69	*	*	*	*	*	*	*	*
	IntraQ [28]	46.61	*	69.38	*	*	*	84.38	73.59	91.66	89.52	84.49	88.05
	AlignQ [23]	25.18	26.07	49.83	50.24	41.46	51.61	64.91	64.24	21.88	31.59	18.46	24.78
	LifeQuant (Ours)	**22.38**	**16.08**	**42.99**	**41.63**	**38.96**	**42.59**	**58.63**	**63.90**	**10.46**	**7.71**	**2.30**	**1.63**

regularization (see ProxQ in Sec. 5.3) to effectively reduce the increasing quantization error (analyzed in Sec. 5.1), and 2) data influence rebalancing (see BaLL loss design in Sec. 5.4), under the multi-task quantization.

Domain-based multi-task quantization (Office-31 and ImageCLEF). We further evaluate LifeQuant on Office-31 and ImageCLEF containing separate domains not only with the change in class distributions but with the varying feature distributions. Table 2 presents that the 2-bit ResNet-50 on Office-31 under the process of LifeQuant in case A $\rightarrow$ D $\rightarrow$ W receives 6% to 16% accuracy improvements and 7% to 23% forgetting rate reduction. In addition to ResNets, we validate LifeQuant on MobileNet-V2, which is a lightweight architecture widely applied for memory cost reduction [48, 49, 50]. Table 2 reveals that LifeQuant-based 2-bit MobileNet-V2, on ImageCLEF (e.g., B $\rightarrow$ C $\rightarrow$ I) with 76.11% accuracy and 1.63% forgetting rate, which improves the state-of-the-art by more than 20% accuracy improvements and 23% forgetting rate reduction. The results demonstrate LifeQuant with a smaller increasing quantization error on old task data under multi-task quantization, by space shift minimization (see ProxQ in Sec. 5.2) and class data rebalancing (see BaLL loss design in Sec. 5.4). More comparisons including MobileNet-V2 on Office-31 and ResNet-50 on ImageCLEF are presented in Appendix B.

7. Ablation study

7.1. Performances of LifeQuant components

In the following, we evaluate the effectiveness of each component of LifeQuant. Table 3 shows the performances for 4-bit ResNet-20 on CIFAR-100 and 3-bit MobileNet-V2 on ImageCLEF. First, the results on CIFAR-100 manifest that the LifeQuant process without the BaLL loss, i.e., the case **Ours (w/o BaLL)**, suffers from much more accuracy degradation (4% to 6%) and forgets more prediction results on old tasks (5.5% to 10%), which validates the efficacy of BaLL loss to alleviate the forgetting problem by rebalancing the influences of data on prediction loss (replay data vs. new task data) (detailed in Sec. 5.4). On the other hand, LifeQuant on ImageCLEF without ProxQ, i.e., **Ours (w/o ProxQ)**, suffers from 30% accuracy loss and 37% forgetting rate increment, demonstrating the importance of space shift minimization on quantization error reduction under the change in data domains (see ProxQ in Sec. 5.2).

Table 3: Performances of LifeQuant components. The case with the best performance is presented in **bold text**. The case with the most significant performance degradation is presented with the upper script †.

Metrics	**Methods**	4-bit ResNet-20 on CIAFAR-100		3-bit MobileNet-V2 on ImageCLEF	
		$\gamma = 25$	$\gamma = 50$	I $\rightarrow$ P $\rightarrow$ C	B $\rightarrow$ C $\rightarrow$ I
Accuracy (%)	Ours (w/o BaLL)	47.73	31.01	43.61†	45.28†
	Ours (w/o Prox)	46.17†	30.00†	71.94	72.78
	Ours (LifeQuant)	**50.15**	**36.94**	**76.38**	**76.11**
Forgetting (%)	Ours (w/o Prox)	26.00	51.77	47.43†	39.35†
	Ours (w/o BaLL)	27.81†	52.74†	13.16	2.64
	Ours (LifeQuant)	**22.38**	**42.99**	**7.71**	**1.63**

7.2. Effectiveness of ProxQ on space shift reduction

After examining the individual performances of LifeQuant components, we evaluate ProxQ on space shift reduction in this subsection. Table 4 demonstrates the ef-

Table 4: Effectiveness of ProxQ on space shift reduction. The space shift is measured by the RMSE (root mean square error) of quantized weights under the change of tasks.

Methods	ResNet-20 on CIAFAR-100 ($\gamma = 25$)		ResNet-20 on CIAFAR-100 ($\gamma = 50$)		ResNet-50 on ImageCLEF (B $\rightarrow$ C $\rightarrow$ I)	
	4 bit	2 bit	4 bit	2 bit	4 bit	2 bit
Ours (w/o ProxQ)	0.0030	0.0034	0.0039	0.0039	0.0005	0.0007
Ours (w/. ProxQ)	**0.0026**	**0.0032**	**0.0028**	**0.0032**	**0.0001**	**0.0001**
Reduction (%)	**-13.33**	**-5.88**	**-28.21**	**-17.95**	**-80.00**	**-85.71**

Table 5: Effectiveness of BaLL on imbalanced task data evaluated with MobileNet-V2 on ImageCLEF (B $\rightarrow$ C $\rightarrow$ I). δ represents the replay ratio. *BaLL w/o rebal.* indicates the BaLL loss without rebalancing (see Sec. 5.4).

Metrics	Methods	4 bit			3 bit		
		$\delta = 10$	20	35	$\delta = 10$	20	35
Accuracy (%)	BaLL w/o rebal.	70.84	71.39	71.39	46.11	47.22	49.72
	Ours (BaLL)	**72.50**	**78.34**	**77.22**	**50.55**	**72.78**	**70.83**
	Increment (%)	**+1.66**	**+6.95**	**+5.83**	**+4.44**	**+25.56**	**+21.11**
Forgetting (%)	BaLL w/o rebal.	8.08	10.35	9.95	21.99	34.88	34.37
	Ours (BaLL)	**3.18**	**0.93**	**2.05**	**12.57**	**2.64**	**3.64**
	Decrement (%)	**-4.90**	**-9.42**	**-7.90**	**-9.42**	**-32.24**	**-30.73**

fectiveness of ProxQ on reducing the RMSE of quantized weights under the change of tasks, by regularizing the search space to a defined Gaussian space[5] during quantization (see Sec. 5.2). The reduction of shift validates that the ProxQ process achieves a smaller quantization error increment based on Theorem 5.1, and it can thus avoid a significant performance loss according to Theorem 3.1, which is consistent with the experimental results as shown in Table 3.

7.3. Effectiveness of BaLL on imbalanced task data

In experiments from Sec. 6 to Sec. 7.2, we adopted 20% old data as replay data for retraining. Here, we evaluate the BaLL loss with different replay ratios. Table 5 presents three settings $\delta = 10, 20$, and 35, e.g., $\delta = 10$ indicates 10% old task data as replay data, which manifests that the prediction accuracy is improved, and the forgetting rate is reduced when the replay ratio increases from 10% to 20% or to 35%. Thus, the efficacy of the employment of replay data is validated. Furthermore, it shows that the BaLL loss has superior performances over the baseline without rebalancing. For example, the 3-bit model, at a replay ratio of 20% under BaLL, achieves an accuracy of 72.78% and only obtains a forgetting rate of 2.64%, which improves the baseline by 25.56% accuracy gain and 32.24% forgetting rate reduction. The enhancement demonstrates that the BaLL loss can effectively alleviate the forgetting problem by reweighting the influence of old data in new tasks (see Sec. 5.4).

Table 6: Performance of MobileNet-V2 on ImageCLEF (B $\rightarrow$ C $\rightarrow$ I) compared with lifelong learning research. The improvements over 5% (10%) are presented in **blue** (**red**).

Metrics	Methods	4 bit			3 bit		
		$\delta = 10$	20	35	$\delta = 10$	20	35
Accuracy (%)	EWC [30]	66.94	69.44	69.17	46.66	47.50	49.44
	SI [31]	70.28	66.39	73.73	48.05	48.33	48.89
	MAS [32]	68.06	72.22	72.78	42.78	45.83	50.83
	RWalk [33]	69.72	71.95	71.17	43.06	46.38	46.67
	SCP [36]	70.83	70.55	72.22	40.56	45.28	48.33
	PFR [37]	72.22	69.72	73.33	45.00	44.44	46.95
	Ours (BaLL)	**72.50**	**78.34**	**77.22**	**50.55**	**72.78**	**70.83**
Forgetting (%)	EWC [30]	4.76	13.23	12.15	20.92	38.71	32.91
	SI [31]	3.12	16.31	6.87	25.78	40.97	34.07
	MAS [32]	5.98	8.88	13.44	22.44	39.74	30.01
	RWalk [33]	4.15	9.87	9.82	20.14	43.25	35.88
	SCP [36]	2.58	8.30	8.54	21.24	34.88	38.85
	PFR [37]	3.19	8.88	7.34	21.75	39.04	34.37
	Ours (BaLL)	**3.18**	**0.93**	**2.05**	**12.57**	**2.64**	**3.64**

7.4. Comparisons with recent LL losses

We further compare the BaLL loss with recent lifelong learning (LL) research. Table 6 shows that the recent LL approaches suffer from notable performance losses under 3-bit quantization. By contrast, EWC, MAS, RWalk, and SCP are able to receive a 3% to 8% accuracy increment by exploiting more replay data. However, the improvements are limited due to the imbalance data issue studied in Sec. 5.3 with the underestimated influence on replay data. Accordingly, the BaLL loss is designed to reweight the data influences (detailed in Sec. 5.4). Table 6 manifests that BaLL under the rebalancing strategy achieves 72.78% accuracy with only 2.64% forgetting at the replay ratio of 20%, which is close to the performances of previous LL approaches under 4-bit quantization at the replay ratio of 35%. Thus, the efficiency of the BaLL loss is demonstrated since models can be compressed to lower bits with less memory storage for replay data without a forgetting catastrophe.

8. Conclusion

In this paper, we propose LifeQuant to overcome the forgetting catastrophe in quantization-aware training. We prove that the space shift in multi-task quantization increases quantization error. Thus, we propose ProxQ to regularize the search space for space shift minimization during quantization. Moreover, we investigate that the limited amount of replay data in new task learning incurs a biased prediction result and the forgetting problem. Therefore, we design a BaLL loss to reweight (to increase) the influences on task data under the theoretical guarantee to approximate the prediction result on balanced data. Experimental results manifest that LifeQuant outperforms the state-of-the-art on multi-task datasets.

[5] The settings of the regularized space will be studied in Appendix C.

Acknowledgment

This work was supported in part by the Ministry of Science and Technology, Taiwan, under grant MOST 112-2223-E-002-015, and by the Ministry of Education, Taiwan, under grant MOE 112L9009 and MOE 112L900901.

References

[1] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. Imagenet classification with deep convolutional neural networks. In *Advances in neural information processing systems*, pages 1097–1105, 2012.

[2] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. In *International Conference on Medical image computing and computer-assisted intervention*, pages 234–241. Springer, 2015.

[3] Joseph Redmon, Santosh Divvala, Ross Girshick, and Ali Farhadi. You only look once: Unified, real-time object detection. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 779–788, 2016.

[4] Keyan Cao, Yefan Liu, Gongjie Meng, and Qimeng Sun. An overview on edge computing research. *IEEE access*, 8:85714–85728, 2020.

[5] Cristian Buciluǎ, Rich Caruana, and Alexandru Niculescu-Mizil. Model compression. In *Proceedings of the 12th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 535–541, 2006.

[6] Hao Li, Asim Kadav, Igor Durdanovic, Hanan Samet, and Hans Peter Graf. Pruning filters for efficient convnets. *arXiv preprint arXiv:1608.08710*, 2016.

[7] Yihui He, Xiangyu Zhang, and Jian Sun. Channel pruning for accelerating very deep neural networks. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 1389–1397, 2017.

[8] Yang He, Ping Liu, Ziwei Wang, Zhilan Hu, and Yi Yang. Filter pruning via geometric median for deep convolutional neural networks acceleration. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4340–4349, 2019.

[9] Shaohui Lin, Rongrong Ji, Chenqian Yan, Baochang Zhang, Liujuan Cao, Qixiang Ye, Feiyue Huang, and David Doermann. Towards optimal structured cnn pruning via generative adversarial learning. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2790–2799, 2019.

[10] Mark Sandler, Andrew Howard, Menglong Zhu, Andrey Zhmoginov, and Liang-Chieh Chen. Mobilenetv2: Inverted residuals and linear bottlenecks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4510–4520, 2018.

[11] François Chollet. Xception: Deep learning with depthwise separable convolutions. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1251–1258, 2017.

[12] Ron Banner, Yury Nahshan, Elad Hoffer, and Daniel Soudry. Aciq: Analytical clipping for integer quantization of neural networks. 2018.

[13] Yoni Choukroun, Eli Kravchik, Fan Yang, and Pavel Kisilev. Low-bit quantization of neural networks for efficient inference. In *2019 IEEE/CVF International Conference on Computer Vision Workshop (ICCVW)*, pages 3009–3018. IEEE, 2019.

[14] Steven K Esser, Jeffrey L McKinstry, Deepika Bablani, Rathinakumar Appuswamy, and Dharmendra S Modha. Learned step size quantization. *arXiv preprint arXiv:1902.08153*, 2019.

[15] Yuhang Li, Xin Dong, and Wei Wang. Additive powers-of-two quantization: An efficient non-uniform discretization for neural networks. In *International Conference on Learning Representations*, 2020.

[16] Kanghyun Choi, Deokki Hong, Noseong Park, Youngsok Kim, and Jinho Lee. Qimera: Data-free quantization with synthetic boundary supporting samples. *Advances in Neural Information Processing Systems*, 34, 2021.

[17] Ting-An Chen, De-Nian Yang, and Ming-Syan Chen. Climbq: Class imbalanced quantization enabling robustness on efficient inferences. In *Advances in Neural Information Processing Systems*.

[18] Amir Gholami, Sehoon Kim, Zhen Dong, Zhewei Yao, Michael W Mahoney, and Kurt Keutzer. A survey of quantization methods for efficient neural network inference. *arXiv preprint arXiv:2103.13630*, 2021.

[19] Ritchie Zhao, Yuwei Hu, Jordan Dotzel, Chris De Sa, and Zhiru Zhang. Improving neural network quantization without retraining using outlier channel splitting. In *International conference on machine learning*, pages 7543–7552. PMLR, 2019.

[20] Markus Nagel, Rana Ali Amjad, Mart Van Baalen, Christos Louizos, and Tijmen Blankevoort. Up or down? adaptive rounding for post-training quantization. In *International Conference on Machine Learning*, pages 7197–7206. PMLR, 2020.

[21] Itay Hubara, Yury Nahshan, Yair Hanani, Ron Banner, and Daniel Soudry. Improving post training neural quantization: Layer-wise calibration and integer programming. *arXiv preprint arXiv:2006.10518*, 2020.

[22] Xiandong Zhao, Ying Wang, Xuyi Cai, Cheng Liu, and Lei Zhang. Linear symmetric quantization of neural networks for low-precision integer hardware. In *International Conference on Learning Representations*, 2020.

[23] Ting-An Chen, De-Nian Yang, and Ming-Syan Chen. Alignq: Alignment quantization with admm-based correlation preservation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 12538–12547, 2022.

[24] Duvindu Piyasena, Miyuru Thathsara, Sathursan Kanagarajah, Siew Kei Lam, and Meiqing Wu. Dynamically growing neural network architecture for lifelong deep learning on the edge. In

2020 30th International Conference on Field-Programmable Logic and Applications (FPL), pages 262–268. IEEE, 2020.

[25] German I Parisi, Ronald Kemker, Jose L Part, Christopher Kanan, and Stefan Wermter. Continual lifelong learning with neural networks: A review. *Neural networks*, 113:54–71, 2019.

[26] Haibo He and Edwardo A Garcia. Learning from imbalanced data. *IEEE Transactions on knowledge and data engineering*, 21(9):1263–1284, 2009.

[27] Yongkweon Jeon, Chungman Lee, Eulrang Cho, and Yeonju Ro. Mr. biq: Post-training non-uniform quantization based on minimizing the reconstruction error. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 12329–12338, 2022.

[28] Yunshan Zhong, Mingbao Lin, Gongrui Nan, Jianzhuang Liu, Baochang Zhang, Yonghong Tian, and Rongrong Ji. Intraq: Learning synthetic images with intra-class heterogeneity for zero-shot network quantization. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 12339–12348, 2022.

[29] Shai Shalev-Shwartz and Shai Ben-David. *Understanding machine learning: From theory to algorithms*. Cambridge university press, 2014.

[30] James Kirkpatrick, Razvan Pascanu, Neil Rabinowitz, Joel Veness, Guillaume Desjardins, Andrei A Rusu, Kieran Milan, John Quan, Tiago Ramalho, Agnieszka Grabska-Barwinska, et al. Overcoming catastrophic forgetting in neural networks. *Proceedings of the national academy of sciences*, 114(13):3521–3526, 2017.

[31] Friedemann Zenke, Ben Poole, and Surya Ganguli. Continual learning through synaptic intelligence. In *International conference on machine learning*, pages 3987–3995. PMLR, 2017.

[32] Rahaf Aljundi, Francesca Babiloni, Mohamed Elhoseiny, Marcus Rohrbach, and Tinne Tuytelaars. Memory aware synapses: Learning what (not) to forget. In *Proceedings of the European conference on computer vision (ECCV)*, pages 139–154, 2018.

[33] Arslan Chaudhry, Puneet K Dokania, Thalaiyasingam Ajanthan, and Philip HS Torr. Riemannian walk for incremental learning: Understanding forgetting and intransigence. In *Proceedings of the European conference on computer vision (ECCV)*, pages 532–547, 2018.

[34] Solomon Kullback and Richard A Leibler. On information and sufficiency. *The annals of mathematical statistics*, 22(1):79–86, 1951.

[35] John M Lee. *Riemannian manifolds: an introduction to curvature*, volume 176. Springer Science & Business Media, 2006.

[36] Soheil Kolouri, Nicholas A Ketz, Praveen K Pilly, and Andrea Soltoggio. Sliced cramer synaptic consolidation for preserving deeply learned representations. 2020.

[37] Alex Gomez-Villa, Bartlomiej Twardowski, Lu Yu, Andrew D Bagdanov, and Joost van de Weijer. Continually learning self-supervised representations with projected functional regularization. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3867–3877, 2022.

[38] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016.

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

[40] James G Kalbfleisch. *Probability and statistical inference*. Springer Science & Business Media, 2012.

[41] Kate Saenko, Brian Kulis, Mario Fritz, and Trevor Darrell. Adapting visual category models to new domains. In *European conference on computer vision*, pages 213–226. Springer, 2010.

[42] Barbara Caputo, Henning Müller, Jesus Martinez-Gomez, Mauricio Villegas, Burak Acar, Novi Patricia, Neda Marvasti, Suzan Üsküdarlı, Roberto Paredes, Miguel Cazorla, et al. Imageclef 2014: Overview and analysis of the results. In *Information Access Evaluation. Multilinguality, Multimodality, and Interaction: 5th International Conference of the CLEF Initiative, CLEF 2014, Sheffield, UK, September 15-18, 2014. Proceedings 5*, pages 192–211. Springer, 2014.

[43] Pravendra Singh, Vinay Kumar Verma, Pratik Mazumder, Lawrence Carin, and Piyush Rai. Calibrating cnns for lifelong learning. *Advances in Neural Information Processing Systems*, 33:15579–15590, 2020.

[44] Alessandro Bergamo and Lorenzo Torresani. Exploiting weakly-labeled web images to improve object classification: a domain adaptation approach. *Advances in neural information processing systems*, 23, 2010.

[45] Gregory Griffin, Alex Holub, and Pietro Perona. Caltech-256 object category dataset. 2007.

[46] Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, et al. Imagenet large scale visual recognition challenge. *International journal of computer vision*, 115(3):211–252, 2015.

[47] Derek Hoiem, Santosh K Divvala, and James H Hays. Pascal voc 2008 challenge. *World Literature Today*, 24, 2009.

[48] Yunhui Guo, Yandong Li, Liqiang Wang, and Tajana Rosing. Depthwise convolution is all you need for learning multiple visual domains. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pages 8368–8375, 2019.

[49] Hongmin Gao, Yao Yang, Chenming Li, Lianru Gao, and Bing Zhang. Multiscale residual network with mixed depthwise convolution for hyperspectral image classification. *IEEE Transactions on Geoscience and Remote Sensing*, 59(4):3396–3408, 2020.

[50] Krishna Kant Singh and Akansha Singh. Diagnosis of covid-19 from chest x-ray images using wavelets-based depthwise convolution network. *Big Data Mining and Analytics*, 4(2):84–93, 2021.