CLS: Continual Learning with Logit–Structure Knowledge Distillation

Qingya Sui, Lin Zhong, Zhenyu Lei, Lianbo Ma, Jiujun Cheng, and Shangce Gao

***Abstract*—Continual learning (CL) aims to acquire new tasks from non-stationary data while retaining prior knowledge. A useful strategy is knowledge distillation (KD), but conventional KD is not well suited to this setting, as distribution shifts cause logit mismatch and accumulated representation drift across tasks. To address these issues, we propose logit–structure knowledge distillation (LSKD), a distillation method tailored for CL. Logit standardization reduces the influence of changing teachers, enabling the student to focus on relative class relations rather than absolute values. Structure distillation preserves the geometric organization of predictions by matching normalized pairwise distances among samples. This joint design constrains semantic drift and maintains consistent feature geometry across tasks. We further combine LSKD with our CL framework, called CLS, which utilizes dynamically expandable representation and attention-based feature aggregation to stabilize knowledge transfer while maintaining adaptability to new tasks. Experiments on CIFAR-10, CIFAR-100, and PathMNIST show the effectiveness of CLS under multiple class-incremental learning settings.**

***Index Terms*—Continual learning, Stability-Plasticity Dilemma, Knowledge distillation, Incremental learning, Lifelong learning**

I. Introduction

Continual Learning (CL) enables Artificial Intelligence (AI) systems to incrementally acquire new skills without forgetting previously learned knowledge [1]. This ability mirrors the adaptive learning processes observed in humans and animals, allowing them to respond effectively to changing environments. Such adaptability is also increasingly vital for AI development. For CL to be effective, models must not only learn and retain knowledge but also apply it to future tasks. This is particularly important for Deep Learning (DL) systems, which are typically trained on static, independent and identically distributed datasets. However, real-world data is often dynamic and non-stationary, making traditional DL approaches insufficient for many practical scenarios [2]. As such, CL is essential for developing AI systems that can adapt and remain relevant over time [3].

This research was partially supported by the Japan Society for the Promotion of Science (JSPS) KAKENHI under Grants JP25K21298, JP25K24386, and JP25K03179, and Japan Science and Technology Agency (JST) Support for Pioneering Research Initiated by the Next Generation (SPRING) under Grant JPMJSP2145. (Corresponding authors: Zhenyu Lei, Jiujun Cheng, and Shangce Gao)

Q. Sui, L. Zhong, Z. Lei, and S. Gao are with the Faculty of Engineering, University of Toyama, Toyama-shi, 930-8555, Japan. (E-mail: qingyasui88@gmail.com; zhonglin762@gmail.com; leizg@eng.u-toyama.ac.jp; gaosc@eng.u-toyama.ac.jp)

L. Ma is with the College of Software, Northeastern University, Shenyang 110169, China. (E-mail: malb@swc.neu.edu.cn)

J. Cheng is with the Key Laboratory of Embedded System and Service Computing, Ministry of Education, Tongji University, Shanghai 200092, China. (E-mail:chengjj@tongji.edu.cn)

A major challenge in CL is catastrophic forgetting. When a new task is introduced, the model often overwrites parameters learned from previous tasks, resulting in a significant performance decrease [4]. As CL has gained increasing attention, various methods have been proposed to mitigate this issue [5]. Regularization-based methods like EWC [6] and Synaptic Intelligence (SI) [7] slow changes to important weights but often sacrifice plasticity for stability. Replay-based methods like iCaRL [8] and CLEAR [9] rehearse stored samples from previous tasks, which preserves knowledge but requires additional memory and privacy risks. Expansion-based methods, such as DER [10], add new network components for each task to improve performance. While avoiding forgetting is crucial for AI models, completely avoiding forgetting can improve the stability of the model on old tasks, but excessive bias of the model towards old classes will reduce the plasticity of the model and perform poorly on new tasks [11]. The stability–plasticity dilemma has therefore become a central research focus in continual learning [12]. Knowledge Distillation (KD) has shown strong potential in addressing this dilemma. By transferring knowledge from earlier models, KD enables the current model to integrate past experiences while learning new tasks [13]. It provides a form of regularization that prevents the new model from deviating too far from prior behavior [14]. This mechanism supports backward transfer, helping maintain performance on previous tasks, and also improves forward transfer by guiding the model toward a more generalizable representation for future tasks [15]. However, KD was originally proposed for static model compression and is not directly suited to the dynamic nature of CL [16]. It assumes a fixed teacher, a stationary dataset, and a stable model size, but CL scenarios introduce tasks one by one and the data shifts each time [17]. As the teacher model and distillation data change after each task, logits drift away from tasks. Conventional KD cannot align features across tasks with evolving distributions, leading to catastrophic forgetting [18]. Furthermore, when future task in CL is uncertain and the teacher and student models, there is a significant capacity gap between logit scale may arise, potentially degrading rather than improving performance [19].

To address these limitations of conventional KD in CL models, we improve KD in two key aspects: logits and structure. On the logit side, we first standardize both the teacher and student outputs using Z-score normalization. This process eliminates absolute scale discrepancies caused by variations in model capacity or shifts in data distributions across task, thereby ensuring that the student can consistently interpret the teacher's soft targets as both capacity and distributions evolve [20]. On the structure level, we further introduce a

structural loss designed to preserve the pairwise distances among samples. This constraint stabilizes the feature geometry, mitigates semantic drift and reduces forgetting when new tasks introduce novel classes or covariate shifts [21]. These limitations motivate our design of the logit–structure knowledge distillation (LSKD), a simple but effective method that simultaneously aligns output distributions and preserves feature geometry. By combining logit standardization with structure-aware distillation, LSKD first aligns class probabilities and preserves sample layout, directly addressing the problems of drifting logits, scale mismatch and unstable feature spaces described above. Specifically, it calibrates the teacher and student logits via Z-score normalization, thereby eliminating scale discrepancies arsing from changes in model size or data drift, which would otherwise hinder effective knowledge transfer. It then adds a lightweight relational term that matches pairwise distances among samples, stabilizing the semantic layout across tasks. To instantiate these ideas in a practical CL setting, we design a complete framework named CLS (Continual Learning with Logit–Structure). At its core lies the proposed LSKD, which transfers both class-level and structural knowledge from earlier models to the current model. We further integrate the dynamically expandable representation (DER) strategy [10], which preserves the features representations from previous tasks by freezing earlier extractors, thereby retaining historical knowledge without interference. In addition, an attention-based feature aggregation module adaptively fuses the task-agnostic features from earlier tasks with task-specific features to the current task, leveraging attention mechanisms to enhance the performance on both old and new tasks [22]. CLS effectively mitigates representation drift, preserves discriminative features from past tasks, and enhances adaptability to new tasks, achieving state-of-the-art performance. In this study, we present CLS, which offers three major contributions to CL:

1) We introduce Logit–Structure Knowledge Distillation (LSKD), which mitigates logit scale mismatch and preserves inter-sample semantic structure, addressing key limitations of conventional KD in CL.
2) We design CLS, a continual learning framework that integrates LSKD with existing representation-preserving and feature-integration strategies to balance stability and plasticity.
3) We conduct comprehensive experiments to demonstrate that LSKD effectively overcomes the distribution mismatch and semantic drift problems of conventional KD in CL, and that CLS achieves superior performance over state-of-the-art CL methods.

The remainder of this paper is organized as follows. Section II reviews continual learning with a focus on class-incremental settings and summarizes KD in CL. Section III details the proposed CLS framework, including LSKD, feature expansion for preserving prior representations, and attention-based feature aggregation. Section IV outlines the experimental settings, datasets, and evaluation metrics, followed by a comprehensive performance analysis. Finally, Section V summarizes the contributions and findings, and discusses potential directions for future research.

II. Related Works

A. Class-Incremental Learning

CL aims to enable models to learn from a sequence of tasks without forgetting previously acquired knowledge. Depending on the type of changes across tasks, CL is typically categorized into three scenarios [23]: Task-Incremental Learning (TIL), where the task identity is provided at test time and models can use task-specific parameters; Domain-Incremental Learning (DIL), where the label space remains fixed but the input distribution shifts across tasks; and Class-Incremental Learning (CIL), where new classes arrive sequentially and the task identity is unknown at inference.

Among these scenarios, CIL is widely regarded as the most challenging [24]. In this setting, a single unified classifier must distinguish both old and new classes using only current-task training data, which exacerbates the stability–plasticity dilemma. Formally, let $\{D_1, D_2, \ldots, D_T\}$ denote a sequence of tasks, where each task $D_t = \{(x_i^t, y_i^t)\}_{i=1}^{n_t}$ contains samples from a disjoint set of classes C_t. At incremental step t, the model only observes D_t, but must perform prediction over the cumulative label space:

$$C_{1:t} = \bigcup_{k=1}^{t} C_k. \tag{1}$$

The learning objective is to train a classifier $f_t : \mathcal{X} \to C_{1:t}$ that achieves high accuracy on both previously seen and newly introduced classes. This scenario closely reflects many real-world applications where data arrive in a non-stationary manner and the set of classes continually expands [25].

B. KD in CL

KD was originally introduced as a technique for transferring knowledge from a large and well-trained teacher model to a student model [26]. It encourages the student to match the teacher's softened output distributions, which reveal inter-class similarities and facilitate more effective learning [27]. Traditional KD methods use a temperature-scaled softmax to soften predictions, aiding the student in capturing nuanced information beyond hard labels. In the context of CL, KD helps mitigate catastrophic forgetting and support long-term knowledge retention [28]. By leveraging outputs or internal representations from the model trained on prior tasks, KD acts as a regularization strategy, guiding the current model to maintain consistency with past behaviors [29]. For example, iCaRL [8] combines KD with exemplar rehearsal by maintaining a small memory of past examples. The model jointly minimizes classification loss on both new data and exemplars, along with a distillation loss to preserve past outputs. PODNet [30] extends this further by aligning spatial features through a pooled output distillation loss, which enhances fine-grained feature retention across tasks. Teacher Adaptation [31] adapts the teacher model by updating batch normalization statistics to match the current data distribution, leading to more effective distillation and improved retention of old knowledge.

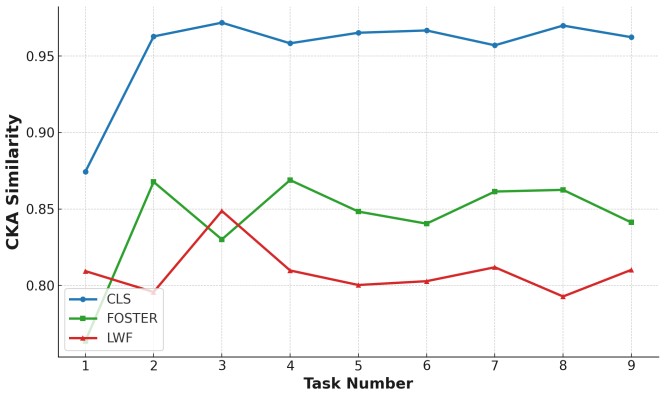


Fig. 1. Comparison of CKA similarity between consecutive task models on CIFAR-100. Traditional distillation-based CL methods (LwF, FOSTER) exhibit an early and pronounced decline with large fluctuations, revealing substantial representation drift. In contrast, the proposed CLS approach maintains consistently high similarity with minimal variance, indicating stable knowledge retention while still allowing the network to adapt to new tasks.

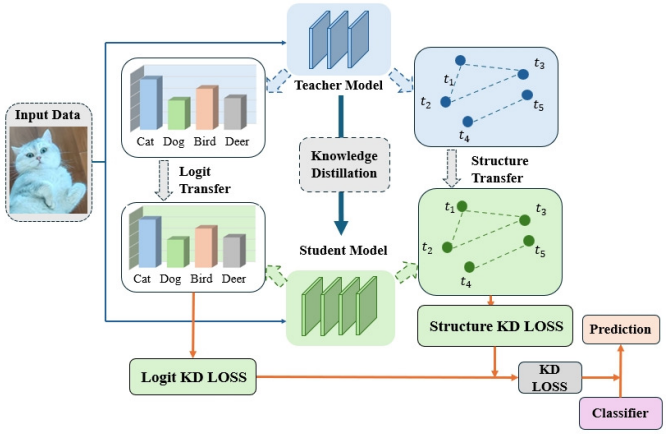


Fig. 2. Proposed logit–structure knowledge distillation (LSKD). Logit standardization distillation mitigates the magnitude-matching issue in conventional KD by enabling the student to adopt an arbitrary logit scale suited to its capacity while preserving class-level relations from the teacher. Structure distillation transfers the structural relationships among outputs, allowing the student to capture the teacher's semantic structure more effectively. In the context of continual learning, LSKD alleviates distribution mismatch and preserves semantic structure, leading to more stable knowledge retention across tasks.

KD helps CL models preserve information from earlier tasks even as new knowledge is acquired. This balance between stability and plasticity is crucial for sustaining performance across tasks. KD is a powerful technique for continual learning because it preserves prior knowledge and supports task integration, but it still suffers from several challenges. The performance of KD may be sensitive to the temperature parameter and may diminish when there is a significant capacity gap between the teacher and student. Similar issues can also arise in other adaptive learning settings with changing parameterizations. For example, LoRA freezes pretrained weights and learns only low-rank updates for downstream adaptation [32], while Auto-Encoding Neural Tucker Factorization combines low-rank Tucker decomposition with nonlinear representation learning [33]. Although these settings differ from class-incremental learning, they suggest that calibration and structure preservation may also be useful when knowledge must be transferred under evolving model parameterizations. These limitations motivate the development of CLS to solve these issues.

III. Proposed Method

A. Motivation

Given the limitations identified above, we aim to develop a KD strategy inherently adapted to the challenges of CL. Unlike traditional KD-based CL methods that suffer from distribution-induced logit mismatch and representation drift, our approach explicitly calibrates logits and preserves the geometric structure of the feature space across tasks. Recent studies have reframed forgetting not merely as a defect to be eliminated, but as a deliberate design choice in scenarios where memory is bounded and tasks are unbounded [34]. From this perspective, a well-designed continual learner should retain essential and generalizable knowledge while selectively discarding less relevant information. As shown in Fig. 1, conventional KD methods such as LwF and FOSTER exhibit an early and pronounced decline in Centered Kernel Alignment (CKA) similarity between consecutive tasks, accompanied by large oscillations, indicating substantial representation drift [35]. Motivated by these observations, we design LSKD to act at two complementary levels: logit standardization aligns class evidence across tasks, and structure-aware transfer stabilizes the sample-wise prediction geometry. In contrast, our proposed CLS maintains CKA similarity at approximately 0.95 with minimal fluctuation, achieving stable knowledge retention while preserving sufficient capacity for new task acquisition.

B. Logit-Structure KD for CL

Traditional KD methods enforce direct matching between teacher and student logits. This includes both their absolute scale and variance. In continual learning, model capacity often changes across tasks and data distributions shift. Such hard alignment can block effective knowledge transfer. To address these issues, we propose logit-structure knowledge distillation for CL (LSKD), a novel KD strategy that integrates logit standardization [20] with structure distillation. Compared with prior logit standardization and relational knowledge distillation methods for conventional distillation, LSKD is designed for continual learning with evolving teachers and data distributions [21]. Standard KD enforces direct matching between the output logits of the teacher and student, including their absolute scale and variance. This constraint can be especially restrictive in continual learning, as model capacity and data distributions evolve. Fig. 2 shows the mechanism of LSKD. LSKD resolves this mismatch by applying Z-score normalization to both the teacher and student logits.

$$\bar{z}_s = \frac{z_s - \mu(z_s)}{\sigma(z_s)}, \tag{2}$$

$$\bar{z}_t = \frac{z_t - \mu(z_t)}{\sigma(z_t)}, \tag{3}$$

where z_s and z_t denote the logits from the student and teacher, while $\mu(\cdot)$ and $\sigma(\cdot)$ represent their mean and standard devia-

tion. This normalization removes scale discrepancies, ensuring the student focuses on relative class relations rather than absolute values. The softened distributions are then computed as:

$$p_s = \mathrm{Softmax}\left(\frac{\bar{z}_s}{\tau}\right), \quad p_t = \mathrm{Softmax}\left(\frac{\bar{z}_t}{\tau}\right), \tag{4}$$

using a temperature parameter τ. The distillation loss based on Kullback-Leibler divergence is expressed as:

$$\mathcal{L}_{logit} = \tau^2 \cdot \mathrm{KL}(p_t \,\|\, p_s). \tag{5}$$

This normalization is applied only to the distillation term, while the classification loss is still computed from the original student outputs, so useful class-specific confidence can still be learned from the task labels. By focusing on relative rankings instead of absolute values, this process encourages more robust transfer of class relations, reducing the negative impact of distributional shift. Logit-based distillation primarily conveys inter-class relations: the student is trained to match the teacher's relative ordering of class evidence. Directly matching raw logits across tasks is fragile because the teacher's magnitude changes with distribution shift and model capacity. By applying Z-score standardization, we remove this magnitude constraint and let the student focus on relative class relations, which reduces the burden on a smaller model. However, logit calibration alone does not regulate how samples are arranged with respect to one another as tasks accumulate, and the geometry among predictions can still drift. To address this, we add a structure-aware term that preserves inter-sample relations within a batch. The combined objective transfers knowledge at two complementary levels to encode new classes. We further introduce a structure distillation loss that matches the pairwise relationships among logits, ensuring the student preserves the relative geometry of the teacher's predictions. To strengthen structural consistency, this loss explicitly maintains the geometric structure of predictions across samples. Let $Z_t = \{\bar{z}_{t,1}, \ldots, \bar{z}_{t,B}\}$ and $Z_s = \{\bar{z}_{s,1}, \ldots, \bar{z}_{s,B}\}$ denote the standardized logits from the teacher and student for a mini-batch of size B. For any two samples i and j, the normalized pairwise distance is defined as

$$\psi_D(\bar{z}_{t,i}, \bar{z}_{t,j}) = \frac{\|\bar{z}_{t,i} - \bar{z}_{t,j}\|_2^2}{\mu_t}, \tag{6}$$

$$\psi_D(\bar{z}_{s,i}, \bar{z}_{s,j}) = \frac{\|\bar{z}_{s,i} - \bar{z}_{s,j}\|_2^2}{\mu_s}, \tag{7}$$

where

$$\mu_t = \mathbb{E}_{i\neq j}\Big[\|\bar{z}_{t,i} - \bar{z}_{t,j}\|_2^2\Big], \quad \mu_s = \mathbb{E}_{i\neq j}\Big[\|\bar{z}_{s,i} - \bar{z}_{s,j}\|_2^2\Big], \tag{8}$$

and $\mathbb{E}_{i\neq j}$ denotes the mean of off-diagonal pairwise distances within a batch. We denote by $\phi_D(Z)$ the distance geometry operator that produces the normalized pairwise distance matrix from a logit set Z. Since the logits $z = Wh + b$ are a linear transformation of the hidden representation h, the Euclidean distance $\|z_i - z_j\|_2^2$ corresponds to a Mahalanobis distance in feature space with metric $M = W^\top W$, thus capturing the relative geometry among features. By enforcing alignment of $\phi_D(Z_t)$ and $\phi_D(Z_s)$, the student is encouraged to preserve the teacher's semantic layout of samples. The structure loss is calculated as:

$$\mathcal{L}_{st} = \frac{1}{B^2} \sum_{i\neq j} \ell_\delta\Big(\psi_D(\bar{z}_{t,i}, \bar{z}_{t,j}),\, \psi_D(\bar{z}_{s,i}, \bar{z}_{s,j})\Big), \tag{9}$$

where B is the batch size, ψ_D computes normalized pairwise distances, and ℓ_δ is the Huber loss:

$$\ell_\delta(x, y) = \begin{cases} \frac{1}{2}(x-y)^2, & \text{if } |x-y| \le 1, \\ |x-y| - \frac{1}{2}, & \text{otherwise.} \end{cases} \tag{10}$$

Therefore, the overall KD loss of LSKD is defined as:

$$\mathcal{L}_{\mathcal{LSKD}} = \mathcal{L}_{logit} + \lambda_{st}\mathcal{L}_{st}, \tag{11}$$

where λ_{st} is balancing coefficient.

LSKD mitigates the effects of distribution mismatch and preserves the semantic relationships within the data. As a result, the proposed method enables more reliable knowledge transfer and stronger retention of previously acquired knowledge during continual learning.

C. Feature Expansion

To further mitigate catastrophic forgetting and maintain the representation of prior tasks, we adopt the dynamically expandable representation (DER) strategy [10]. When a new task T arrives, a task-specific feature extractor $\Psi_T(x)$ is added, and the representation becomes:

$$\Omega_T(x) = [\Omega_{T-1}(x), \Psi_T(x)], \tag{12}$$

where $\Omega_{T-1}(x)$ is the concatenated feature from previous tasks and $\Psi_T(x)$ encodes the new task. During training, only the newly added extractor is updated, while all previous extractors remain frozen, effectively preserving historical knowledge. DER employs a classifier that is adapted to handle the expanded representation, allowing the model to accommodate new categories introduced by incremental tasks. This approach ensures that knowledge from previous tasks is preserved and readily accessible, while providing the capacity for the model to learn new concepts without interference.

D. Attention-based Feature Aggregation

After feature expansion, CLS applies an attention-based feature aggregation module to adaptively fuse task-agnostic and task-specific features. This mitigates feature collision and improves cross-task generalization [22]. We flatten the previous-task feature map F_T^{ts} and current-task feature map F_T^{ta} along the spatial axes and stack the channels:

$$F_T^{\mathrm{ts}} = \mathrm{Flatten}(\Omega_{T-1}(x)) \in \mathbb{R}^{N\times C_o}, \tag{13}$$

$$F_T^{\mathrm{ta}} = \mathrm{Flatten}(\Psi_T(x)) \in \mathbb{R}^{N\times C_o}. \tag{14}$$

where C_o is the channel number, N denotes the number of spatial locations after flattening the feature map. We project both features to a shared embedding dimension D, and form the queries Q, keys K_ts and K_ta, and values V_ts and V_ta:

$$Q = F_t^{\mathrm{ts}} W_q, \quad K_{\mathrm{ts}} = F_T^{\mathrm{ts}} W_k, \quad V_{\mathrm{ts}} = F_T^{\mathrm{ts}} W_v, \tag{15}$$

$$K_{\mathrm{ta}} = F_T^{\mathrm{ta}} W_k, \quad V_{\mathrm{ta}} = F_T^{\mathrm{ta}} W_v, \tag{16}$$

where $W_q, W_k, W_v \in \mathbb{R}^{C_o \times D}$. The attention weights A_t are computed as:

$$A_t = \text{Softmax}\left(\frac{Q[K_{\text{ts}}, K_{\text{ta}}]^\top}{\sqrt{D}}\right)[V_{\text{ts}}, V_{\text{ta}}]. \tag{17}$$

The merge attention enables the model to adaptively aggregate information from task-agnostic and task-specific sources. This selective fusion helps mitigate feature collision, improves feature diversity, and facilitates robust transfer of generic knowledge across tasks, which is especially crucial as the number of tasks increases.

E. Summary

The CLS framework integrates three major components to address the stability–plasticity dilemma in continual learning. The first is LSKD, which calibrates logits to alleviate distribution-induced mismatch while preserving the geometric structure of the feature space to mitigate representation drift across tasks. The second is feature expansion, which assigns task-specific subnetworks, thereby retaining knowledge from previous tasks while providing additional capacity for learning new tasks. The third is attention-based feature aggregation used to unify features in the incremental tasks. The overall loss of our framework is defined as:

$$\mathcal{L} = \mathcal{L}_{cls} + \mathcal{L}_{\mathcal{LSKD}} + \mathcal{L}_{aux}, \tag{18}$$

where $\mathcal{L}_{cls}$ is the classification loss, $\mathcal{L}_{aux}$ is the auxiliary loss, which aligns the outputs of auxiliary classifiers in the student network with those from the frozen teacher network of the previous task, thereby stabilizing intermediate feature representations. These three components work in concert: LSKD constrains semantic drift at the output level, feature expansion preserves prior knowledge while providing space for new learning, and attention-based aggregation integrates multi-task features for knowledge transfer. As a result, CLS effectively mitigates both distribution-induced logit mismatch and representation drift, enabling the model to retain essential and generalizable knowledge and adapt efficiently to new information.

IV. Experiments

A. Experimental Setup

1) Datasets and Data Split: To comprehensively evaluate the effectiveness of the proposed CL framework, we conduct experiments on three benchmark datasets: CIFAR-10, CIFAR-100, and PathMNIST.

CIFAR-10 [36] is an image classification dataset consisting of 60,000 32×32 color images across 10 classes. In our experiments, CIFAR-10 is split into 5 sequential tasks, each comprising 2 disjoint classes. At every incremental step, the model is trained on a new subset of two classes, while being evaluated on all classes observed up to the current stage.

CIFAR-100 [36] includes 100 classes with 600 images per class. Two incremental settings are adopted: a 10-task scenario with 10 classes per task, and a 5-task scenario with 20 classes per task. For both configurations, the model learns classes in sequence, and in each task, only samples from the current subset are accessible for training.

PathMNIST [37] is a medical image dataset consisting of 107,180 RGB images belonging to 9 tissue categories. In our experiments, PathMNIST is divided into 8 incremental tasks: the first task contains 2 classes, and each of the following 7 tasks introduces one new class. This scenario presents a realistic and challenging setting for class-incremental medical image classification.

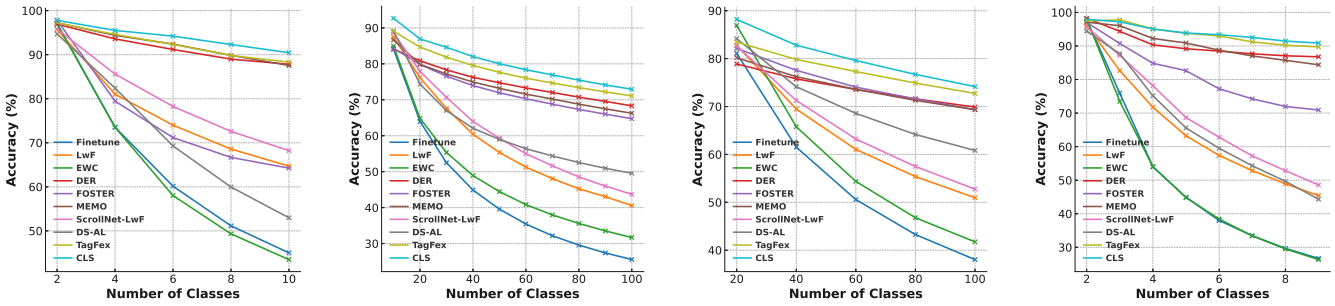
2) Baselines: To provide a comprehensive comparison, we consider representative class-incremental learning methods. Regularization-based methods mitigate forgetting by constraining model parameters to retain previous knowledge. Replay and architecture based methods address forgetting by storing and replaying samples from previous tasks. KD-based approaches utilize the outputs of previous models to guide the learning of new tasks, thus preserving prior knowledge. We also include the naive Finetune baseline for reference.

- **Finetune** [38]: A baseline without any specific mechanisms.
- **LwF** [13]: Learning without Forgetting constrains the outputs of the previous model via KD.
- **EWC** [39]: Elastic Weight Consolidation introduces a regularization term based on the Fisher information matrix to prevent significant changes to important parameters for previously learned tasks.
- **DER** [10]: Dynamically Expandable Representation trains a new feature extractor for each task and aggregates features for joint classification, alleviating representation bias in incremental learning.
- **FOSTER** [40]: This approach boosts feature diversity and compresses redundant representations by combining feature boosting with network compression for efficient incremental learning.
- **MEMO** [41]: A memory-efficient method that adaptively switches between storing exemplars and model parameters to balance memory usage and performance.
- **ScrollNet-LwF** [42]: This method dynamically allocates parameter importance using a learned importance score and incorporates KD to mitigate forgetting.
- **DS-AL** [43]: A dual-stream analytic learning framework designed for exemplar-free class-incremental learning, which maintains model stability and adaptability without relying on rehearsal data.
- **TagFex** [22]: Task-Agnostic Guided Feature Expansion captures task-agnostic features through adaptively merges them with task-specific features to enhance feature diversity and reduce feature collision.

3) Details: We adopt ResNet-32 as the backbone model for all experiments. The model is trained for 200 epochs on the initial task and 170 epochs for each subsequent task. For all compared methods, hyperparameters and implementation details strictly follow the original papers or the official PyCIL [44] codebase to ensure fair comparison. To improve the statistical reliability of our results, all experiments are repeated three times with different random seeds, and following the CIL setting. For all datasets, classes are ordered randomly but fixed across runs to ensure reproducibility. We set $\lambda_{\text{st}} = 0.07$ based

TABLE I
COMPARISON OF THE FINAL ACC AND LAST WITH DIFFERENT DATASETS.

Method	CIFAR-10		CIFAR-100 (10 Task)		CIFAR-100 (5 Task)		PathMNIST	
	Last	ACC	Last	ACC	Last	ACC	Last	ACC
Finetune	20.467 ± 0.531	45.013 ± 0.069	9.037 ± 0.067	25.552 ± 0.519	17.137 ± 0.131	38.039 ± 0.439	6.415 ± 1.445	26.693 ± 2.378
LwF	49.150 ± 2.400	64.697 ± 1.216	21.470 ± 0.118	40.555 ± 0.174	32.450 ± 0.090	50.945 ± 1.432	20.763 ± 5.747	45.414 ± 5.619
EWC	20.033 ± 2.177	43.477 ± 2.062	15.713 ± 0.732	31.678 ± 0.167	22.770 ± 1.226	41.741 ± 1.968	5.907 ± 1.657	26.358 ± 1.114
DER	84.617 ± 1.378	87.819 ± 0.707	59.257 ± 1.331	68.323 ± 1.310	62.997 ± 1.269	69.888 ± 1.043	84.807 ± 0.841	86.768 ± 1.226
FOSTER	54.503 ± 3.194	64.246 ± 2.266	49.343 ± 1.943	64.748 ± 3.166	58.203 ± 0.509	69.339 ± 0.263	64.167 ± 1.495	70.921 ± 2.067
MEMO	81.723 ± 0.845	87.551 ± 0.026	55.187 ± 2.039	66.258 ± 2.538	22.927 ± 0.135	69.338 ± 0.598	78.293 ± 0.713	84.387 ± 1.297
ScrollNet-LwF	52.077 ± 1.274	68.215 ± 1.058	24.080 ± 1.202	43.695 ± 2.760	35.587 ± 1.122	52.755 ± 1.332	27.070 ± 1.058	48.587 ± 5.528
DS-AL	54.793 ± 1.057	67.642 ± 1.036	37.517 ± 0.799	49.552 ± 1.004	46.167 ± 0.905	60.830 ± 1.035	21.430 ± 5.728	44.344 ± 0.510
TagFex	83.010 ± 0.670	88.311 ± 0.691	60.877 ± 0.057	71.087 ± 1.233	**64.003 ± 0.451**	72.726 ± 1.096	86.047 ± 1.957	89.713 ± 0.993
CLS	**84.747 ± 1.745**	**90.353 ± 0.411**	**62.223 ± 0.162**	**72.940 ± 0.067**	63.537 ± 0.118	**74.154 ± 0.045**	**86.487 ± 1.218**	**90.865 ± 0.410**



(a) CIFAR-10. (b) CIFAR-100 (10 Task). (c) CIFAR-100 (5 Task). (d) PathMNIST.

Fig. 3. Average accuracy after each task with different datasets. In Task 1, the model is trained from scratch, and the remaining classes are given evenly in the later tasks.

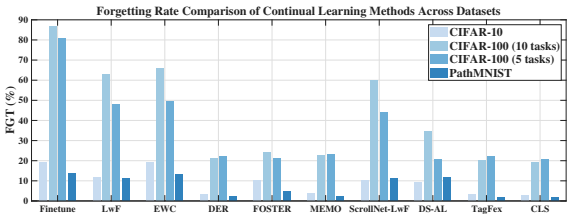


Fig. 4. FGT Comparison Across Datasets.

on preliminary validation experiments to balance structure preservation and flexibility for new task learning.

4) Evaluation Metrics: To comprehensively assess the performance of continual learning methods, we employ three standard metrics: Last, ACC, and FGT.

Last accuracy measures the classification accuracy on the test set of the most recently introduced task after the completion of the final incremental step. This metric reflects the model's ability to learn and retain knowledge relevant to the most recent task.

ACC accuracy computes the average classification accuracy over all observed tasks after each incremental step, with the final value being the average across all tasks at the end of the learning sequence. This metric evaluates the model's overall capability to preserve knowledge acquired from earlier tasks while adapting to new tasks. ACC is computed as:

$$\text{ACC} = \frac{1}{T}\sum_{j=1}^{T} A_{T,j} \tag{19}$$

where $A_{i,j}$ denotes the accuracy on task j after training up to task i.

FGT evaluates a model's robustness against catastrophic forgetting in the continual learning setting. This metric quantifies the average degradation in performance on previously seen tasks after the model has been sequentially trained on a complete series of tasks. FGT is calculated as:

$$\text{FGT} = \frac{1}{T-1}\sum_{i=1}^{T-1} F_i \tag{20}$$

where F_i denotes the forgetting on task i. A lower FGT value signifies superior knowledge retention. The FGT provides a comprehensive assessment of a model's ability to balance plasticity for new knowledge acquisition with stability for preserving existing competencies.

B. Results

As shown in Table I, CLS achieves the highest ACC in all evaluated settings and remains competitive in LAST. Although its LAST accuracy on CIFAR-100 (5 Task) is slightly lower than that of TagFex, CLS still achieves the best ACC in this setting. These results indicate that CLS provides a favorable balance between knowledge retention and adaptation to new tasks. High Last accuracy reflects that the model retains sufficient plasticity throughout continual learning. That is, CLS can effectively incorporate new knowledge without being overly constrained by past tasks. The LSKD prevents overfitting to old knowledge and supports continual adaptation to new tasks, which is critical for CL in dynamic data streams.

CLS also achieves the highest ACC across all datasets. Particularly, on PathMNIST, which features class imbalance

TABLE II
ABLATION STUDY RESULT OF CLS ON CIFAR-100 WITH 10 TASKS

Method	LAST	ACC
Without logit KD/ structure KD/ feature aggregation	60.710	68.449
Without logit KD	60.720	71.917
Without structure KD	61.090	72.701
Without feature aggregation	61.860	70.529
CLS	**62.310**	**73.003**

and higher inter-class similarity, CLS outperforms all baselines by a clear margin. This result suggests that CLS remains effective on this medical image benchmark. Fig. 3 illustrates the evolution of average accuracy as the number of classes increases in the incremental learning process. CLS consistently maintains the highest accuracy across all stages and datasets, regardless of the number of introduced classes. This trend is stable from the early Task to the final task in different data environments. This ability is crucial for practical CL problems, where the number of target classes is often unknown in advance and may change over time. The ability of CLS to maintain top-level performance under varying task and dataset types highlights its generalizability. It demonstrates that CLS can flexibly adapt to evolving environments while preserving previously acquired knowledge, showing its capability to generalize across various continual learning settings.

Fig. 4 reports the forgetting rate (FGT) on four datasets. Finetune, LwF, and EWC all show very high FGT, indicating severe forgetting of previous knowledge. Expansion-based methods such as DER, FOSTER, and MEMO substantially reduce forgetting, but still leave room for improvement. TagFex achieves further reduction in FGT, benefiting from its task-agnostic feature transfer. Our proposed CLS method achieves the lowest FGT across all datasets, demonstrating its superior ability to preserve prior knowledge while learning new tasks. In contrast, our proposed CLS consistently attains the lowest FGT in all datasets and settings, reflecting its ability to effectively retain knowledge from earlier tasks while adapting to new ones. CLS mitigates scale mismatch and preserves inter-class structural information. By jointly addressing these two key factors, CLS maintains more stable representations over time and mitigates decision boundary shifts, resulting in robust long-term performance in CL scenarios.

C. Ablation Study

Table II reports the results of the ablation study on CIFAR-100 with 10 tasks, to evaluate the effect of each key component in CLS. We compare the CLS model with four variants. All ablation experiments share the same training configuration as CLS. The result shows that standardizing logits helps retain recent task performance. Fig. 5 further illustrates the similarity between teacher and student models during distillation, measured by centered kernel alignment (CKA). The solid lines represent the average CKA for each task, while the shaded regions denote the standard deviation across batches. A higher mean CKA indicates stronger alignment of representations, and a narrower band suggests a more stable distillation process. The results show that the model without Z-score exhibits lower average similarity and larger fluctuations.

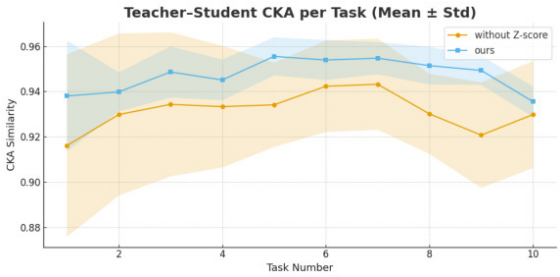


Fig. 5. Teacher–Student CKA Comparison With and Without Z-score.

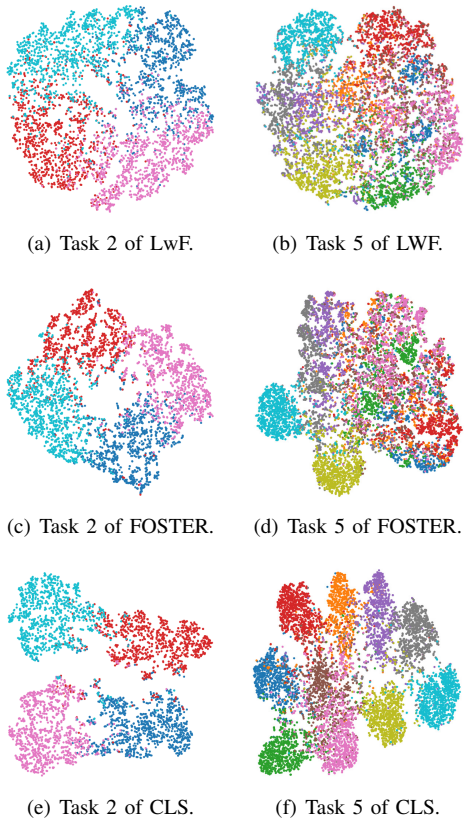
In contrast, the use of Z-score leads to consistently higher and more stable CKA across tasks. This result reflects the limitation of conventional KD in CIL, as tasks and data distributions change, the student model is forced to mimic the absolute numerical range of the teacher's logits. In contrast, Z-score normalization focuses on the relative relationships among classes and helps student models retain the teacher's semantic structure. Removing structure-aware distillation leads to a larger drop in ACC. This indicates that preserving inter-sample relationships improves overall knowledge retention. Without feature aggregation, the model also performs worse, especially in LAST. This suggests that attention-based fusion helps the model adapt to new features while keeping past ones. CLS achieves the best performance in both metrics. This confirms that all components contribute to improving the stability of old tasks and knowledge transfer in this setting.

D. Analysis

1) Visualization Analysis: Fig. 6 shows the t-SNE projections of the feature representations after Task 2 (the first incremental task) and Task 5 (the final task) for LwF, FOSTER, and CLS on CIFAR-10 with 5 tasks.

At Task 1, all methods form well-delineated and non-overlapping regions in the feature space, suggesting that the initial classes representations are discriminative. However, after Task 5, the differences among methods become apparent. For LwF, the representation regions of different categories become substantially intermixed, with category boundaries largely obscured. This phenomenon indicates that LwF struggles to retain distinct decision regions as new tasks are learned, resulting in increased confusion among similar categories. FOSTER remains a significant region overlap, indicating partial resistance to interference but limited capability to maintain inter-class discrimination. In contrast, CLS can preserve clear decision boundaries and minimize mixing between categories, reflecting enhanced stability for old task knowledge while adapting to new task information.

2) Confusion Matrix Analysis: We further evaluate the classification performance of different methods by analyzing the confusion matrices on the CIFAR-10 dataset after the final incremental learning step. As shown in Fig. 7, LwF and FOSTER both exhibit noticeable off-diagonal elements, indicating frequent classification error between similar categories. In particular, the confusion between visually related



(a) Task 2 of LwF. (b) Task 5 of LWF.

(c) Task 2 of FOSTER. (d) Task 5 of FOSTER.

(e) Task 2 of CLS. (f) Task 5 of CLS.

Fig. 6. Visualization of class representation in feature space.

classes such as cat and dog, or automobile and truck, remains pronounced.

In contrast, our method produces a confusion matrix with much stronger diagonal dominance, reflecting higher classification accuracy and clearer category discrimination. Most samples are correctly assigned to their true classes, and confusion between categories is greatly reduced. These results confirm that our approach more effectively preserves the model's ability to distinguish between categories throughout the continual learning process, thereby mitigating the impact of catastrophic forgetting.

3) Grad-CAM Analysis: To further illustrate the evolution of attention over CL tasks. We visualize the class activation maps (CAMs) for a sample belonging to the "bed" category in Fig. 8, which was introduced in the first task [45]. At task 1, the model attends correctly to the most salient regions of the bed, indicating that CLS effectively learns discriminative features upon first exposure. As the model continues to learn new tasks, these attention patterns remain well-preserved. Although some diffusion in attention is observed, the core focus consistently remains on the object-relevant regions. This suggests that CLS maintains stable representations of previously learned categories even as new classes are introduced.

Additionally, beginning from task 3, CLS starts to attend to additional informative regions that were not activated during the initial task, such as the headboard. This change suggests that CLS improves the structure of old task features over time. The logit-structure distillation helps transfer useful patterns from new tasks to earlier ones, making the learned feature representations more complete. The CAMs demonstrate that CLS can retain what it has learned in the past while still acquiring new tasks. Its design helps reduce forgetting and enhances old knowledge with new learning.

TABLE III
RESULTS OF DIFFERENT λ_{st} VALUES OF CLS ON CIFAR-100 WITH 10 STEPS.

λ_{st}	LAST	ACC
0.03	61.830	72.743
0.05	62.310	73.003
0.07	**62.390**	**73.100**
0.09	62.120	72.743

4) Parameter Analysis: In CLS, the logit distillation weight is fixed to 1.0 to ensure that class-level relations are consistently preserved across tasks. We vary the structure distillation weight $\mathcal{L}_{st}$ to investigate its influence on overall performance. The best results are achieved when $\mathcal{L}_{st}$ is set to 0.07, which provides a balanced trade-off between retaining the structural relationships from previous tasks and adapting to new task-specific representations. A smaller $\mathcal{L}_{st}$ weakens the constraint on structural preservation, leading to increased forgetting. In contrast, a larger value may overemphasize past structures, reducing flexibility for learning novel classes. This analysis confirms that an appropriate structural weight is crucial for maximizing the benefits of LSKD.

V. CONCLUSION

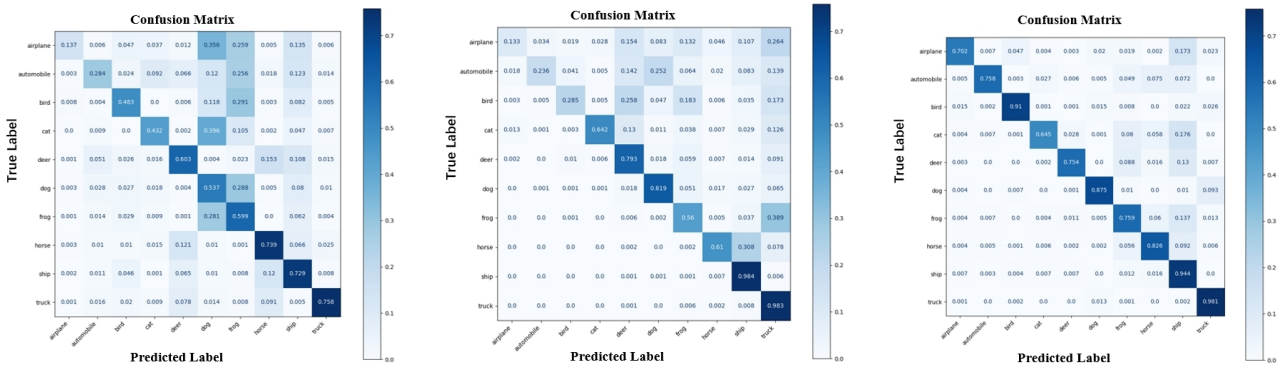
In this paper, we propose Logit–Structure Knowledge Distillation (LSKD) to address the limitations of traditional KD in class-incremental learning, namely logit scale mismatch and representation drift. LSKD standardizes logits to mitigate scale variation across tasks and preserves the structural relation among predictions to maintain the semantic consistency of the feature space. We integrated LSKD into the CLS framework, which also incorporates representation preservation and attention-based feature aggregation, ensuring stable knowledge retention while maintaining adaptability to new tasks. Experiments on CIFAR-10, CIFAR-100, and PathMNIST demonstrate the effectiveness of CLS under multiple CIL settings. While feature expansion increases model capacity over time, CLS maintains competitive performance for continual learning applications. In future work, we will further evaluate CLS in more realistic continual learning scenarios and explore its application to real-world problems, such as medical image analysis and other dynamic recognition tasks.

REFERENCES

[1] D. Kudithipudi, M. Aguilar-Simon, J. Babb, M. Bazhenov, D. Blackiston, J. Bongard, A. P. Brna, S. Chakravarthi Raja, N. Cheney, J. Clune *et al.*, "Biological underpinnings for lifelong learning machines," *Nature Machine Intelligence*, vol. 4, no. 3, pp. 196–210, 2022.

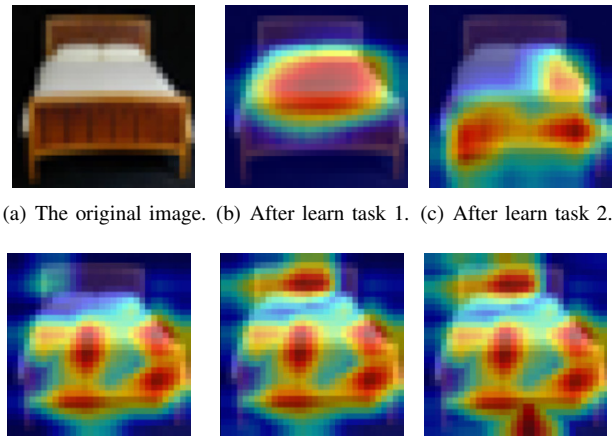
[2] R. Hadsell, D. Rao, A. A. Rusu, and R. Pascanu, "Embracing change: Continual learning in deep neural networks," *Trends in cognitive sciences*, vol. 24, no. 12, pp. 1028–1040, 2020.

[3] O. Friha, M. Amine Ferrag, B. Kantarci, B. Cakmak, A. Ozgun, and N. Ghoualmi-Zine, "LLM-based edge intelligence: A comprehensive survey on architectures, applications, security and trustworthiness," *IEEE Open Journal of the Communications Society*, vol. 5, pp. 5799–5856, 2024.



(a) Confusion matrix of LwF (b) Confusion matrix of FOSTER (c) Confusion matrix of CLS

Fig. 7. Confusion matrix for LwF, FOSTER, and CLS after continual learning on the CIFAR-10 dataset, with two classes introduced per step over five tasks.



(a) The original image. (b) After learn task 1. (c) After learn task 2.

(d) After learn task 3. (e) After learn task 4. (f) After learn task 5.

Fig. 8. Grad-CAM visualizations of the "bed" class after each incremental task.

[4] E. Verwimp, R. Aljundi, S. Ben-David, M. Bethge, A. Cossu, A. Gepperth, T. L. Hayes, E. Hüllermeier, C. Kanan, D. Kudithipudi, C. H. Lampert, M. Mundt, R. Pascanu, A. Popescu, A. S. Tolias, J. van de Weijer, B. Liu, V. Lomonaco, T. Tuytelaars, and G. M. van de Ven, "Continual learning: Applications and the road forward," *Transactions on Machine Learning Research*, 2024.

[5] G. Bai, C. Ling, Y. Gao, and L. Zhao, "Saliency-augmented memory completion for continual learning," in *Proceedings of the 2023 SIAM International Conference on Data Mining (SDM)*. SIAM, 2023, pp. 244–252.

[6] J. Kirkpatrick, R. Pascanu, N. Rabinowitz, J. Veness, G. Desjardins, A. A. Rusu, K. Milan, J. Quan, T. Ramalho, A. Grabska-Barwinska *et al.*, "Overcoming catastrophic forgetting in neural networks," *Proceedings of the National Academy of Sciences*, vol. 114, no. 13, pp. 3521–3526, 2017.

[7] F. Zenke, B. Poole, and S. Ganguli, "Continual learning through synaptic intelligence," in *Proceedings of the 34th International Conference on Machine Learning*, vol. 70. Proceedings of Machine Learning Research, 2017, pp. 3987–3995.

[8] S.-A. Rebuffi, A. Kolesnikov, G. Sperl, and C. H. Lampert, "iCaRL: Incremental classifier and representation learning," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, July 2017, pp. 2001–2010.

[9] D. Rolnick, A. Ahuja, J. Schwarz, T. Lillicrap, and G. Wayne, "Experience replay for continual learning," in *Advances in Neural Information Processing Systems*, vol. 32. Curran Associates, Inc., 2019, pp. 350–360.

[10] S. Yan, J. Xie, and X. He, "DER: Dynamically expandable representation for class incremental learning," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2021, pp. 3014–3023.

[11] S. Dohare, J. F. Hernandez-Garcia, Q. Lan, P. Rahman, A. R. Mahmood, and R. S. Sutton, "Loss of plasticity in deep continual learning," *Nature*, vol. 632, no. 8026, pp. 768–774, 2024.

[12] D. Kim and B. Han, "On the stability-plasticity dilemma of class-incremental learning," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2023, pp. 20 196–20 204.

[13] Z. Li and D. Hoiem, "Learning without forgetting," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 40, no. 12, pp. 2935–2947, 2018.

[14] C. Shen, X. Wang, Y. Yin, J. Song, S. Luo, and M. Song, "Progressive network grafting for few-shot knowledge distillation," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 35, no. 3, 2021, pp. 2541–2549.

[15] M. H. Phan, T.-A. Ta, S. L. Phung, L. Tran-Thanh, and A. Bouzerdoum, "Class similarity weighted knowledge distillation for continual semantic segmentation," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2022, pp. 16 866–16 875.

[16] J. Gou, B. Yu, S. J. Maybank, and D. Tao, "Knowledge distillation: A survey," *International Journal of Computer Vision*, vol. 129, no. 6, pp. 1789–1819, 2021.

[17] Z. Yang, T. Pang, H. Feng, H. Wang, W. Chen, M. Zhu, and Q. Liu, "Self-distillation bridges distribution gap in language model fine-tuning," in *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics*, vol. 1, August 2024, pp. 1028–1043.

[18] A. Daram and D. Kudithipudi, "Does alignment help continual learning?" in *Proceedings of The Workshop on Classifier Learning from Difficult Data*, vol. 263. PMLR, 19–20 Oct 2024, pp. 48–55.

[19] Y. Li, Y. Liu, X. Cheng, Z. Zhu, H. Li, B. Yang, and Z. Huang, "Kc-prompt: End-to-end knowledge-complementary prompting for rehearsal-free continual learning," in *ICASSP 2024 - 2024 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2024, pp. 1–5.

[20] S. Sun, W. Ren, J. Li, R. Wang, and X. Cao, "Logit standardization in knowledge distillation," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2024, pp. 15 731–15 740.

[21] W. Park, D. Kim, Y. Lu, and M. Cho, "Relational knowledge distillation," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2019, pp. 3967–3976.

[22] B. Zheng, D.-W. Zhou, H.-J. Ye, and D.-C. Zhan, "Task-agnostic guided feature expansion for class-incremental learning," in *Proceedings of the Computer Vision and Pattern Recognition Conference (CVPR)*, June 2025, pp. 10 099–10 109.

[23] G. M. Van de Ven, T. Tuytelaars, and A. S. Tolias, "Three types of incremental learning," *Nature Machine Intelligence*, vol. 4, no. 12, pp. 1185–1197, 2022.

[24] G. I. Parisi, R. Kemker, J. L. Part, C. Kanan, and S. Wermter, "Continual lifelong learning with neural networks: A review," *Neural Networks*, vol. 113, pp. 54–71, 2019.

[25] M. De Lange, R. Aljundi, M. Masana, S. Parisot, X. Jia, A. Leonardis, G. Slabaugh, and T. Tuytelaars, "A continual learning survey: Defying

forgetting in classification tasks," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 44, no. 7, pp. 3366–3385, 2022.
[26] Y. Xia, Y. Tong, J. Yang, X. Sun, Y. Zhang, H. Wang, and L. Yun, "Multi-Label prototype-aware structured contrastive distillation," *Tsinghua Science and Technology*, vol. 30, no. 4, pp. 1808–1830, 2025.
[27] J. H. Cho and B. Hariharan, "On the Efficacy of Knowledge Distillation," in *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, October 2019, pp. 4794–4802.
[28] X. Wang, Q. Zhao, L. Wang, and W. Liu, "Causality-based contrastive incremental learning framework for domain generalization," *Tsinghua Science and Technology*, vol. 30, no. 4, pp. 1636–1647, 2025.
[29] S. Li, T. Su, X.-Y. Zhang, and Z. Wang, "Continual learning with knowledge distillation: A survey," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 36, no. 6, pp. 9798–9818, 2025.
[30] A. Douillard, M. Cord, C. Ollion, T. Robert, and E. Valle, "Podnet: Pooled outputs distillation for small-tasks incremental learning," in *Computer vision–ECCV 2020*. Springer, 2020, pp. 86–102.
[31] F. Szatkowski, M. Pyla, M. Przewięźlikowski, S. Cygert, B. Twardowski, and T. Trzciński, "Adapt your teacher: Improving knowledge distillation for exemplar-free continual learning," in *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*, January 2024, pp. 1977–1987.
[32] E. J. Hu, Y. Shen, P. Wallis, Z. Allen-Zhu, Y. Li, S. Wang, L. Wang, and W. Chen, "LoRA: Low-rank adaptation of large language models," in *International Conference on Learning Representations*, 2022.
[33] P. Tang, X. Luo, and J. Woodcock, "Auto-encoding neural tucker factorization," *IEEE Transactions on Knowledge and Data Engineering*, vol. 37, no. 10, pp. 5795–5807, 2025.
[34] G. Bai, C. Ling, Y. Gao, and L. Zhao, *Saliency-Augmented Memory Completion for Continual Learning*, 2023, pp. 244–252.
[35] S. Kornblith, M. Norouzi, H. Lee, and G. Hinton, "Similarity of neural network representations revisited," in *Proceedings of the 36th International Conference on Machine Learning*, vol. 97. PMLR, 09–15 Jun 2019, pp. 3519–3529.
[36] A. Krizhevsky and G. Hinton, "Learning multiple layers of features from tiny images," *Technical Report TR-2009, University of Tront*, 2009.
[37] J. Yang, R. Shi, D. Wei, Z. Liu, L. Zhao, B. Ke, H. Pfister, and B. Ni, "Medmnist v2-a large-scale lightweight benchmark for 2d and 3d biomedical image classification," *Scientific Data*, vol. 10, no. 1, p. 41, 2023.
[38] D.-W. Zhou, Q.-W. Wang, Z.-H. Qi, H.-J. Ye, D.-C. Zhan, and Z. Liu, "Class-incremental learning: A survey," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 46, no. 12, pp. 9851–9873, 2024.
[39] J. Kirkpatrick, R. Pascanu, N. Rabinowitz, J. Veness, G. Desjardins, A. A. Rusu, K. Milan, J. Quan, T. Ramalho, A. Grabska-Barwinska *et al.*, "Overcoming catastrophic forgetting in neural networks," *Proceedings of the National Academy of Sciences*, vol. 114, no. 13, pp. 3521–3526, 2017.
[40] F.-Y. Wang, D.-W. Zhou, H.-J. Ye, and D.-C. Zhan, "Foster: Feature boosting and compression for class-incremental learning," in *European conference on computer vision –ECCV 2022*. Springer, 2022, pp. 398–414.
[41] D.-W. Zhou, Q.-W. Wang, H.-J. Ye, and D.-C. Zhan, "A model or 603 exemplars: Towards memory-efficient class-incremental learning," in *ICLR*, 2023, pp. 1124–1133.
[42] F. Yang, K. Wang, and J. van de Weijer, "Scrollnet: Dynamic weight importance for continual learning," in *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV) Workshops*, October 2023, pp. 3345–3355.
[43] H. Zhuang, R. He, K. Tong, Z. Zeng, C. Chen, and Z. Lin, "DS-AL: A dual-stream analytic learning for exemplar-free class-incremental learning," vol. 38, pp. 17 237–17 244, March 2024.
[44] D.-W. Zhou, F.-Y. Wang, H.-J. Ye, and D.-C. Zhan, "Pycil: a python toolbox for class-incremental learning," *SCIENCE CHINA Information Sciences*, vol. 66, no. 9, p. 197101, 2023.
[45] R. R. Selvaraju, M. Cogswell, A. Das, R. Vedantam, D. Parikh, and D. Batra, "Grad-CAM: Visual explanations from deep networks via gradient-based localization," in *2017 IEEE International Conference on Computer Vision (ICCV)*, 2017, pp. 618–626.

Qingya Sui received her B.S. degree at the Jilin University, Jilin, China in 2019, and the M.E. degree from the University of Toyama, Toyama, Japan, in 2024. She is currently pursuing her Ph.D. degree at the University of Toyama, Toyama, Japan. Her current research interests are in computational intelligence.

Lin Zhong received his B.S. degree at the Shanghai Jiao Tong University, Shanghai, China in 2018, and the M.E. degree from the University of Toyama, Toyama, Japan, in 2024. He is currently pursuing his Ph.D. degree at the University of Toyama, Toyama, Japan. His current research interests are in computational intelligence.

Zhenyu Lei (Member, IEEE) received the Ph.D. degree in Science and Engineering from the University of Toyama, Toyama, Japan, in 2023. He is currently an Assistant Professor with the Faculty of Engineering, University of Toyama, Japan. His current research interests include evolutionary computation, machine learning, and neural network for real-world applications and optimization problems.

Lianbo Ma (Member, IEEE) received the B.Sc. degree in Communication Engineering and M.Sc. degree in Communication and Information System from Northeastern University, Shenyang, China, in 2004 and 2007 respectively, and the Ph.D. degree from University of Chinese Academy of Sciences, China, in 2015. He is currently a Professor of Northeastern University, China. His current research interests include computational intelligence and machine learning.

Jiujun Cheng received the Ph.D. degree from the Beijing University of Posts and Telecommunications, in 2006. In 2009, he was a Visiting Professor with Aalto University, Espoo, Finland. He is currently a Professor with Tongji University, Shanghai, China. He has over 80 publications including conference and journal articles. His current research interests include mobile computing, complex networks, Internet of Vehicles, and autonomous cars.

Shangce Gao (Senior Member, IEEE) received his Ph.D. degree in Innovative Life Science from the University of Toyama, Toyama, Japan, in 2011. He is currently a Professor with the Faculty of Engineering at the University of Toyama. His research interests focus on brain-inspired neural networks and their applications in medical diagnosis and drug discovery. He serves as an Associate Editor for several international journals, including IEEE Transactions on Neural Networks and Learning Systems and the IEEE/CAA Journal of Automatica Sinica.