
Appendix for Learning Expandable and Adaptable Representations for Continual Learning

Anonymous Author(s)

Affiliation

Address

email

1	Contents	
2	A The Additional Information for the Related Work	2
3	B The Detailed Pseudocode	4
4	C Additional Information for the Experiment Setting	5
5	C.1 Implement details	5
6	C.2 Analysis of Multi-domain Continual learning (MDCL)	5
7	D Additional Experimental Results	7
8	D.1 The results of CIL settings	7
9	D.2 The results of TRC and analysis of forgetting effect	8
10	D.3 The impact of individual components in LEAR.	8
11	D.4 Performance comparison of different backbone fine-tuning configurations in LEAR	9
12	D.5 Visualization of the HSICBCO approach.	10
13	D.6 Visualization of the ESM’s memory distributions.	10
14	D.7 Analysis of other distance metrics in ESM	11
15	D.8 Other clarifications	11
16	E Discussion of Contributions, Limitations and Societal Impacts	12
17	E.1 Contributions and Limitations	12
18	E.2 Societal Impacts	12

19 A The Additional Information for the Related Work

20 **Rehearsal-based techniques.** The rehearsal approaches enable the network to have partial access to
21 data from historical tasks. In addition to experience replay methods which retain authentic train-
22 ing samples [4, 6, 58], numerous studies have advocated for the development of generative models,
23 such as Variational Autoencoders (VAEs) [23] or Generative Adversarial Networks (GANs) [15],
24 utilizing historical data samples to maintain historical context [1, 44, 48, 66, 22]. Compare to ex-
25 perience replay methods, generative replay methods enhance privacy and efficiency by eliminating
26 the need for direct data storage. However, the generalization capabilities of these methodologies are
27 contingent upon the quality of the generative replay samples. Furthermore, these approaches incur
28 additional training expenses relative to experience replay methods.

29 **Parameter-efficient tuning (PET) techniques.** In continual learning (CL), the adaptation of pretrained
30 models to new tasks often results in catastrophic forgetting, wherein the acquisition of new knowl-
31 edge overwrites previously learned representations [69]. Furthermore, storing fine-tuned models for
32 each task incurs substantial computational overhead and memory costs [67]. These challenges have
33 motivated the recent emergence of parameter-efficient tuning (PET)-based continual learning ap-
34 proaches as a promising research direction [55]. Nowadays, prompt-based techniques have become
35 the predominant strategy in PET-based CL, including L2P [60] (prompt retrieval), Dual-Prompt [59]
36 (task-shared and task-specific prompts), CODA-Prompt [49] (attention-based prompt combination),
37 DAP (adaptive prompt generator), S-Prompt [57] and HiDe-Prompt [54] (only task-specific prompt).
38 Furthermore, PET approaches based on Low-Rank Adaptation (LoRA) [33, 61] and adapter archi-
39 tectures [38, 14] (inject lightweight trainable units between backbone layers) have also been actively
40 explored in CL. Notably, LAE [13] and HiDe-PET [55] propose unified frameworks that systemat-
41 ically integrate these three methodologies in PET. However, prompt-based methods exhibit limi-
42 tations in fine-grained tasks [35] and self-supervised pretraining [64]. HiDe-PET [55] proves that
43 LoRA/adaptor-based PET outperform prompt-based PET within both task-specific and task-shared
44 approaches, but their evaluation does not cover model performance in domain-incremental learning
45 scenarios.

46 **Knowledge distillation (KD) techniques** were initially developed for model compression. The fun-
47 damental concept of the KD framework involves establishing a teacher-student architecture, wherein
48 a loss function is employed to align the predictions of the teacher and student models. This process
49 aims to facilitate the transfer of knowledge from the complex teacher model to the simpler student
50 model [16, 19]. KD has found extensive applications in deep learning, yielding substantial results.
51 Given its advantageous properties and performance, KD has also been utilized to mitigate network
52 forgetting in continual learning scenarios. The primary objective of integrating KD within continual
53 learning is to minimize the divergence between the predictions of the student and teacher models
54 during task learning, as outlined in Learning Without Forgetting (LWF) [32]. Moreover, rehearsal-
55 based approaches can be synergistically combined with KD to form a unified learning framework,
56 which has demonstrated enhanced model performance, as illustrated in [46]. Additionally, the self-
57 KD approach has been proposed to maintain previously acquired representations, thereby alleviating
58 network forgetting, as discussed in [5]

Expansion-based approaches. Unlike rehearsal and knowledge distillation methods, which struggle to maintain optimal performance on prior tasks, expansion-based approaches effectively circumvent network forgetting by preserving all previously acquired network weights while adaptively generating new sub-models or parameters for learning new tasks [21, 50]. Two stage method FOSTER [53] first dynamically expands new modules to fit the residuals between the target and the output of the original model, then executes effective distillation strategy to remove redundant parameters. The energy-based expansion and fusion approach BEEF [52] achieves bi-directional compatibility by training decoupled modules with forward (pf) and backward (pb) prototypes, enabling robust learning of new tasks while mitigating catastrophic forgetting through energy-based joint distribution modeling. More recently, advancements have been made utilizing pre-trained ViT models, exemplified by the approach in [38], which suggests the integration of a frozen random projection layer between the output head and the feature representations of the pre-trained model, thereby enhancing linear separability for class-prototype-based continual learning. A notable ViT-based dynamic expansion model is introduced in [11], which dynamically constructs self-attention blocks and classifiers to accommodate new tasks. A similar concept is presented in [63], which incorporates a meta-attention mechanism to capture task-specific information. Besides, a parameter-efficient training framework for vision-language models is introduced in [65], which employs a MoE-Adapters based dynamic expansion architecture for enhanced adaptability and efficiency in response to new tasks.

Dual-branch approaches. These methods are biologically grounded in the Complementary Learning Systems (CLS) theory [37] from neuroscience, which proposes two distinct but interacting memory systems: (1) a fast-learning hippocampal system for rapid encoding of new information, and (2) a slow-learning neocortical system for gradual knowledge consolidation. DualNet [43] couples a supervised fast learner with a self-supervised slow learner, they complement each other while working synchronously. CLS-ER [2] employs a dual-memory learning mechanism with interaction whereby the episodic memory stores the samples and the semantic memories build short-term and long-term memories of the learned representations of the working model. Nevertheless, existing dual-branch approaches might suffer from limited plasticity for drastically varying data domains due to their fixed network architectures, while also struggling with slow adaptation to rapid domain shifts and heavy reliance on data replay for historical knowledge retention.

The key distinctions between our method and prior approaches. The proposed LEAR fundamentally advances continual learning by synergistically addressing five critical limitations of existing paradigms:

- (1) Unlike rehearsal-based methods that depend on data replay (with inherent privacy/storage overheads) or generative models (constrained by sample quality), LEAR achieves stability through MIBPA in prediction level and KLDBFA in feature representation level, eliminating explicit data retention while preserving historical knowledge.
- (2) While PET approaches like prompt tuning struggle with domain shifts due to frozen feature extractors, LEAR’s interactively optimized collaborative backbone architecture enables adaptive feature learning across distinct data domains, enhancing model plasticity.

99 (3) Compared to KD techniques that assume stable decision boundaries and often require exemplars,
100 LEAR preserves learned task-shared (auxiliary model $F_{\hat{\theta}^g}$) and task-specific knowledge (auxiliary
101 model $F_{\hat{\theta}^l}$) at the end phase of each task, employing MIBPA and KLDBFA to maintain model sta-
102 bility during subsequent task training. LEAR also enforces representation disentanglement between
103 global backbone and local backbone through HSICBCO, ensuring complementary feature learning.

104 (4) While other dynamic network methods suffer from parameter redundancy and limited histori-
105 cal knowledge transfer, LEAR’s ESM simultaneously preserves knowledge through expert-specific
106 memory distributions \mathcal{N}_j and achieves parameter-efficiency by reusing the most relevant expert’s
107 parameters based on feature-level similarity in Eq.(12).

108 (5) Compared to existing dual-branch approaches’ fixed network architectures, dependence on data
109 replay and slow adaptation to rapid domain shifts. LEAR simultaneously achieving plasticity (adap-
110 tive backbones and parameter-efficient network expansion) and stability (historical knowledge align-
111 ment constraints) without data replay.

112 B The Detailed Pseudocode

113 We provide the detailed pseudocode in **Algorithm 1**. The equation numbers here are taken from the
114 main paper.

Algorithm 1 Learning Process of LEAR

Input: Task sequence $\{\mathcal{T}_1, \dots, \mathcal{T}_n\}$, datasets $\{\mathcal{D}_1^S, \dots, \mathcal{D}_n^S\}$, training epoch n'
Output: Trained model parameters $\{\theta^g, \theta^l, \{\varphi_j^f, \varphi_j^c\}_{j=1}^n\}$

- 1: **Step 1: Collaborative backbone initialization.**
- 2: Initialize global backbone F_{θ^g} and local backbone F_{θ^l} with pre-trained ViT and freeze all layers except the last three layers.
- 3: **for** each task \mathcal{T}_j in sequence **do**
- 4: **Step 2: Dynamic expert creation and selection**
- 5: **if** $j > 1$ **then**
- 6: Sample $\{\mathbf{x}_l\}_{l=1}^{m'}$ from \mathcal{D}_j^S and extract features $\{\mathbf{z}_l'' = F_{\theta^f}(\mathbf{x}_l)\}_{l=1}^{m'}$
- 7: For l -th representation \mathbf{z}_l'' , employ $\{F_{\varphi_c^f}\}_{c=1}^{j-1}$ to generate transformed features $\{\mathbf{z}_l^c = F_{\varphi_c^f}(\mathbf{z}_l'')\}_{c=1}^{j-1}$
- 8: Compute Mahalanobis distances between $\{\mathbf{z}_l^c = F_{\varphi_c^f}(\mathbf{z}_l'')\}_{c=1}^{j-1}$ and the corresponding memory distributions $\{\mathcal{N}_c = \mathcal{N}(\boldsymbol{\mu}_c, \boldsymbol{\Sigma}_c)\}_{c=1}^{j-1}$ to select the most relevant expert \mathcal{E}_{c^*} using Eq.(12)
- 9: Initialize new expert \mathcal{E}_j with $\varphi_j^f \leftarrow \varphi_{c^*}^f, \varphi_j^c \leftarrow \varphi_{c^*}^c$
- 10: **else**
- 11: Randomly initialize the first expert \mathcal{E}_1
- 12: **end if**
- 13: **Step 3: Interactive optimization with alignment constraints**
- 14: **while** current epoch number $t \leq n'$ **do**
- 15: Given the data batch $\mathbf{X} = \{\mathbf{x}_1, \dots, \mathbf{x}_b\}$ from \mathcal{D}_j^S , compute representations $\mathbf{z}_g = F_{\theta^g}(\mathbf{X})$, $\mathbf{z}_l = F_{\theta^l}(\mathbf{X})$, then compute predictions $\mathbf{Y} = F_{\varphi_j^c}(\mathbf{z}_g \oplus F_{\varphi_j^f}(\mathbf{z}_l))$
- 16: Update parameters $\{\theta^g, \theta^l, \varphi_j^f, \varphi_j^c\}$ using $\mathcal{L}_{\text{final}}$ in Eq.(13):
- 17: **end while**
- 18: Duplicate and frozen the last three layers of F_{θ^g} and F_{θ^l} to auxiliary model $F_{\hat{\theta}^g}$ and $F_{\hat{\theta}^l}$ respectively.
- 19: Frozen the expert \mathcal{E}_j and construct memory distribution \mathcal{N}_j for \mathcal{E}_j using Eq.(11)
- 20: **end for**

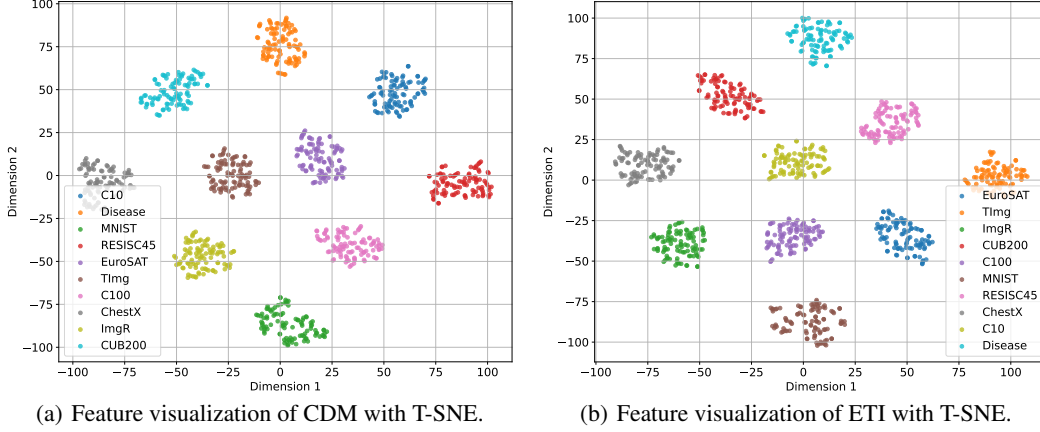


Figure 1: The visualizations of feature representations derived from the $F_{\varphi_j^f}$ of each expert $j, \forall j = 1, \dots, 10$ after learning 10 tasks in CDM and ETI.

C Additional Information for the Experiment Setting

C.1 Implement details

Device Configurations. All experiments were conducted on the same hardware environment running Ubuntu 22.04.2 LTS, with 256 GB of RAM and Intel Xeon Silver4320. A single NVIDIA A100 GPU provides the computing acceleration in experiments.

Training details of LEAR. We initialize both the global backbone and the local backbone using a ViT-B/16 model pre-trained on ImageNet-21K. Throughout the training phase, the dual backbone structure remains frozen except the last three layers of the each backbone are activated. We employ the Adam optimizer to optimize our network parameters. The learning rate is set to 0.03, and the batch size is set to 32. The model is trained for 10 epochs on each data domain. The calculation of HSIC employs a standard Gaussian kernel $K(\mathbf{x}_1, \mathbf{x}_2) = \exp(-\|\mathbf{x}_1 - \mathbf{x}_2\|^2 / \sigma^2)$, where we adopt $\sigma = 1$ in the experiments. Each expert is composed of two distinct modules: the first is a fully connected layer comprising 500 hidden units while the second is a linear classifier realized through a fully connected layer with 1,268 hidden units. The hyper-parameter configuration $(\lambda_1, \lambda_2, \lambda_3) = (2, 1, 1)$ was empirically determined to optimize model performance. The sample number m and m' in ESM are set to 100 and 50 (before multiplying by the batch size), respectively. Furthermore, We regularize the covariance matrix Σ_j by adding ηI ($\eta = 10^{-6}$) to ensure positive definiteness.

C.2 Analysis of Multi-domain Continual learning (MDCL)

Most current researches in continual learning predominantly emphasize Class-Incremental Learning (CIL) within a singular data domain [10, 4, 60, 5], which fails to capture the complex dynamics of real-world scenarios where learning occurs across multiple domains, which called Domain-Incremental Learning (DIL). S-Prompt [57] perform experiments on three standard DIL benchmark datasets: CDDb [30], CORE50 [34] and DomainNet [42]. ZSCL [68] proposes a benchmark called Multi-domain Task Incremental Learning (MTIL), comprising 11 domain tasks: Aircraft [36], Cal-

tech101 [12], CIFAR100 [26], DTD [8], EuroSAT [17], Flowers [40], Food [3], MNIST [29], OxfordPet [41], StanfordCars [24] and SUN397 [62]. CoLeCLIP [31] also employ this benchmark to evaluate its performance. DAP [20] conducts experiments across 7 data domains in 3 fields: natural domains include CIFAR100 and OxfordPet; aerial domains comprise EuroSAT and RESISC45 [7]; medical domains consist of CropDiseases [39], ISIC2018 [9] and ChestX [56]. DIL represents a more authentic learning scenario than CIL, as it inherently assumes new samples will originate from previously unseen data domains.

Although the aforementioned methods employ task sequences with significant domain shifts, which seemingly increases the difficulty of continual learning for the model, we observe **two key limitations**:

- (1) The pre-trained Vision Transformers (ViT) have already achieved over 90% classification accuracy on certain data domains (e.g., Aircraft, MNIST, EuroSAT), suggesting that testing solely on these relatively simple datasets may not adequately reflect the model’s continual learning capability.
- (2) Due to the substantial semantic differences across domains, prompt-based continual learning approaches or dynamic network methods can easily distinguish between different types of data domains, enabling the training of task-specific modules for more accurate predictions. This characteristic paradoxically reduces the overall complexity of model training.

Therefore, we propose this challenging Multi-domain Continual learning (MDCL) scenario to address these two limitations. Building upon the domain categorization in DAP, we further introduce three challenging domains: TinyImageNet [28], ImageNet-R [18], and CUB200 [51]. Moreover, compared to traditional DIL scenarios, certain domains in MDCL exhibit higher semantic similarity (e.g., CIFAR10, CIFAR100 and TinyImageNet), making the learning task more demanding due to the increased domain overlap and discrimination difficulty. Nevertheless, MDCL still exhibits substantial domain discrepancies across the entire task sequence (e.g., from EuroSAT to TinyImageNet), which maintain a significant level of challenge for DIL. Specifically, we employ three MDCL task sequences in different order, each comprising 10 data domains from natural, medical and aerial image classification: **CDM** (C10, Disease, MNIST, RESISC45, EuroSAT, TImg, C100, ChestX, ImgR, CUB200), **ETI** (EuroSAT, TImg, ImgR, CUB200, C100, MNIST, RESISC45, ChestX, C10, Disease) and **TRI** (TImg, RESISC45, CUB200, ChestX, ImgR, EuroSAT, MNIST, C10, Disease, C100). We explore various combinations of domain orders to assess the generalization performance of different models under varying domain configurations. Additionally, MDCL can be further extended by introducing new types of data domains or allowing existing domains to reappear, thereby increasing the complexity of the continual learning scenario and making it more reflective of real-world applications. This also represents a direction for our future research.

Datasets. We consider 10 widely used benchmarks to assess the proposed LEAR, natural domains including CIFAR-10 [25], MNIST [29], CIFAR-100[26], CUB200[51], TinyImageNet[28] and ImageNet-R[18]; aerial domains consist of RESISC45 [7] and EuroSAT [17]; medical domains comprise CropDiseases [39] and ChestX [56]. The CIFAR-10 dataset consists of 60,000 32x32 color images, evenly distributed across 10 distinct categories. The CIFAR-100 dataset also contains 60,000 32x32 color images but divided into 100 fine-grained classes, and these 100 classes

Table 1: Performance Comparison of LEAR and baselines on CIL settings

Method	CIFAR100	CUB200	ImageNetR	TinyImageNet
RanPAC	92.23	90.32	78.11	72.89
MoE	85.21	82.26	76.77	80.23
L2P	82.76	79.23	73.73	76.37
DualPrompt	84.12	83.21	78.47	81.38
CODAPrompt	86.33	83.36	74.45	82.80
LEAR	95.80	88.38	77.67	85.86

are further grouped into 20 superclasses, providing both fine and coarse labels for each image. The CUB200 dataset is a fine-grained dataset focused on bird species, containing 11,788 images across 200 classes. TinyImageNet is a simplified version of the ImageNet dataset[27], designed for efficient experimentation with limited computational resources. It includes 200 classes, each with 500 training images, 50 validation images, and 50 test images, all resized to 64x64 pixels. ImageNet-R (Rendering) is a subset of the ImageNet dataset, specifically designed to evaluate the robustness of models to various image transformations. It includes images from 200 ImageNet classes, with each class containing multiple renditions, such as cartoons, sketches, and paintings, to test the generalization capabilities of models across different visual styles. The MNIST dataset contains 70,000 grayscale images of handwritten digits (0-9), each 28x28 pixels, divided into 60,000 training and 10,000 test images. It serves as a foundational benchmark for digit recognition tasks. The RESISC45 dataset includes 31,500 high-resolution (256x256 pixels) remote sensing images across 45 scene categories, such as “airplane” and “desert,” offering diverse challenges for aerial domain classification. The EuroSAT dataset consists of 27,000 geo-referenced RGB satellite images (64x64 pixels) from 10 land use and land cover classes, like “highway” and “river”, designed to assess model performance on real-world satellite imagery. The CropDiseases dataset includes 54,306 images of diseased and healthy plant leaves from 14 crop species and 26 disease types, it supports the training of deep learning models for accurate plant disease diagnosis. The ChestX dataset comprises over 108,948 frontal-view chest X-ray images annotated with up to 8 thoracic disease labels, enabling the training and evaluation of deep learning models for thoracic disease diagnosis.

D Additional Experimental Results

In this section, we provide additional experimental results to further analyze the performance of the proposed LEAR.

D.1 The results of CIL settings

As MDCL is indeed more challenging than standard CIL, comparing with existing methods under the CIL setting would further demonstrate and strengthen the superiority of LEAR. Therefore, we have further supplemented the comparison of LEAR with relevant methods on CIL performance over CIFAR-100, CUB-200, ImageNet-R, and TinyImageNet, each partitioned into 10 tasks. The results in Table 1 demonstrate LEAR’s excellent adaptability to CIL settings, where it achieves competitive performance among baseline methods.

Table 2: The classification accuracy of all testing datasets after learning the **TRC** task sequence.

Methods	TImg	RESISC45	CUB200	ChestX	ImgR	EuroSAT	MNIST	C10	Disease	C100	Avg
DER++(Re)	7.79	17.68	5.52	14.99	2.01	16.70	33.12	53.72	74.20	78.14	30.39
CLS-ER	9.73	26.43	15.61	16.34	15.94	22.62	22.15	57.09	77.20	76.65	33.98
RanPAC	71.28	83.85	56.56	40.27	44.18	92.66	87.31	86.85	96.74	51.99	71.17
MoE	0.63	49.60	19.85	31.11	59.74	63.91	97.85	96.92	78.14	88.42	58.62
L2P	2.98	4.75	6.35	10.44	4.98	12.46	18.25	54.15	84.78	88.48	28.76
DAP	0.87	4.85	2.80	15.84	9.83	27.46	31.81	40.73	64.12	88.85	28.71
D-Prompt	4.26	8.83	11.58	15.63	6.60	23.58	40.50	58.16	77.30	89.66	33.61
C-Prompt	1.96	3.76	7.13	12.93	2.87	16.76	14.86	38.10	56.20	85.78	24.04
LEAR	80.30	92.41	83.49	44.67	69.28	96.62	98.53	95.84	99.16	86.22	84.65

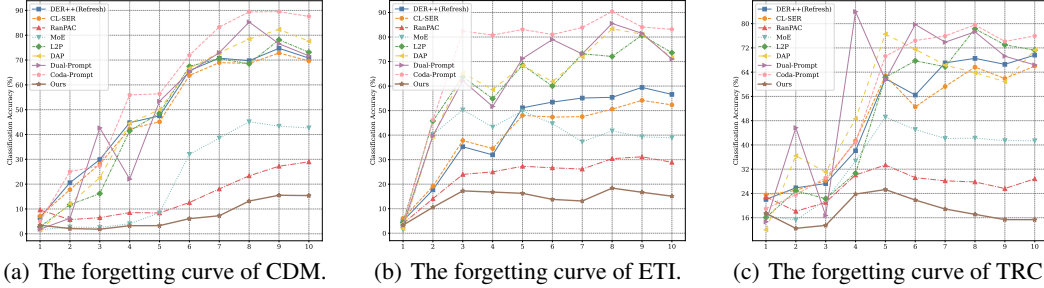


Figure 2: The comparison of the forgetting rates between LEAR and other baseline methods.

D.2 The results of TRC and analysis of forgetting effect

Table 2 demonstrates the additional results in the TRC scenario, LEAR achieves a remarkable average accuracy of 84.65%, significantly outperforming the performance of other methods such as dual-learner approach CLS-ER (33.98%), prompt-based method DualPrompt (33.61%). The Figure 2 provides a comprehensive comparison of the forgetting effect between LEAR and other approaches under three MDCL scenarios. The results indicate that LEAR maintains stable robustness and generalization capabilities compared to other methods across distinct task sequences.

D.3 The impact of individual components in LEAR.

Table 3 clearly demonstrates the contribution of each module or their combinations in LEAR in TRC and we present the following analysis :

- (1) “UpperBound” denotes the results in a single-domain setting, which closely approximates LEAR’s performance upper bound in MDCL;
- (2) “CB” denotes using only the collaborative backbone with a single shared expert network across all datasets;
- (3) “CBE” extends “CB” with task-specific expert network expansion and ESM expert selection. The results demonstrate that the dynamic expert network expansion and selection (“CB”→“CBE”) significantly enhance model performance in TRC;
- (4) “SBE” denotes the configuration where “CBE”’s dual-backbone architecture is replaced with a single backbone. The dual-backbone design yields about a 4% performance gain over the single-backbone counterpart (“SBE”→“CBE”);

Table 3: The classification accuracy of LEAR and its variants in three distinct MDCL scenarios.

Methods	TImg	RE45	CUB	ChestX	ImgR	ES	MNIST	C10	Disease	C100	Avg
UpperBound	82.77	93.72	85.08	50.14	72.81	96.71	98.97	96.49	99.35	86.39	86.24
CB	8.43	22.67	39.45	23.51	25.67	42.74	60.95	64.50	93.45	84.87	46.62
SBE	72.58	81.94	70.12	32.93	57.85	91.19	89.80	89.66	93.96	84.59	76.46
CBE	79.01	89.72	80.68	30.26	64.06	92.34	94.15	94.44	97.05	86.19	80.79
CBE+MI	81.24	90.76	82.51	40.13	67.98	95.19	97.68	95.64	98.03	85.77	83.49
CBE+KL	78.77	90.36	80.22	40.21	66.52	94.67	97.01	95.27	98.40	87.02	82.85
CBE+HSIC	79.14	90.21	80.54	42.12	66.21	93.26	94.24	94.48	97.25	85.41	82.29
CBE+MI+KL	81.24	90.39	82.60	44.11	68.58	96.01	98.37	96.22	98.86	86.17	84.25
CBE+MI+HSIC	82.21	90.43	83.05	44.74	68.41	95.18	95.86	96.09	98.91	85.85	84.07
CBE+KL+HSIC	79.67	89.88	81.44	42.47	66.27	95.07	96.23	95.54	98.91	86.74	83.22
LEAR	80.30	92.41	83.49	44.67	69.28	96.62	98.53	95.84	99.16	86.22	84.65
LEAR w/o ESM	4.27	4.83	68.49	9.42	3.76	14.59	1.98	6.33	1.64	9.25	12.46

Table 4: The classification accuracy (%) of LEAR and its variants in ETI and TRC.

Methods	ES	TImg	ImgR	CUB	C100	MNIST	RE45	ChestX	C10	Disease	Avg	TrainParms
LEAR+FT1	80.42	90.02	82.68	37.14	65.76	95.21	97.17	95.62	98.54	84.17	82.67	14.26M
LEAR+FT2	81.06	90.54	82.53	38.92	67.63	95.37	98.17	95.94	98.90	85.41	83.45	28.43M
LEAR+FT3	80.30	92.41	83.49	44.67	69.28	96.62	98.53	95.84	99.16	86.22	84.65	42.54M
LEAR+FT4	81.65	91.76	82.87	44.89	69.94	96.18	97.63	96.43	99.14	86.80	84.73	56.80M

Methods	TImg	RE45	CUB	ChestX	ImgR	ES	MNIST	C10	Disease	C100	Avg	TrainParms
LEAR+FT1	94.52	79.83	65.42	80.92	82.89	97.29	90.51	40.90	95.37	98.29	82.60	14.26M
LEAR+FT2	94.90	80.94	67.05	83.21	84.67	98.16	90.63	44.31	95.78	98.82	83.85	28.43M
LEAR+FT3	95.89	81.25	69.57	84.12	85.30	98.56	92.92	45.45	96.60	99.30	84.90	42.54M
LEAR+FT4	96.01	81.96	69.44	83.80	86.05	98.49	92.98	47.86	96.34	99.22	85.22	56.80M

(5) “CBE+MI/KL/HSIC” represents “CBE” augmented with individual components or combination of components. Each regularization component and its respective combinations yield effective performance enhancements relative to “CBE” in TRC;

(6) “LEAR” denotes the complete framework, while “LEAR w/o ESM” represents the variant where experts are randomly selected during both the initialization and testing phases of each task. This configuration explains the observed significant performance degradation (“LEAR”→“LEAR w/o ESM”).

D.4 Performance comparison of different backbone fine-tuning configurations in LEAR

The rationale for selecting the final three layers lies in the fact that high-level representation layers capture semantically enriched features, which are advantageous for a variety of downstream applications [47]. [45] also find that deeper layers are disproportionately the source of forgetting. Empirical evidence presented in Table 4 further substantiates that utilizing the last three trainable layers of the global backbone yields robust performance with limited parameter growth compared to other choices, where “FTX” denotes that we activate the final “X” layers of the ViT backbone.

In addition, we clarify that both DER++(Rresh) [58] and CLS-ER [2] were originally implemented using a single ResNet-18 backbone without frozen parameters. To ensure fair comparison when migrating these baselines to ViT, we adopt the consistent backbone configurations between LEAR (keep last three layers of both backbones trainable) and these two approaches. This results in comparable scales of trainable parameters across all three methods. In practice, we have also conducted comparative experiments by activating the last three layers of ViT backbones for other baselines.

Table 5: The comparison of LEAR and baselines in ETI under identical fine-tuning settings.

Methods	EuroSAT	TImg	ImgR	CUB200	C100	MNIST	RESISC45	ChestX	C10	Disease	Avg
DER++(Re)+FT3	51.82	36.25	2.32	6.20	23.08	65.97	40.45	27.41	82.69	97.77	43.40
CLS-ER+FT3	45.76	29.33	16.19	33.08	30.74	67.10	46.44	30.26	80.29	97.62	47.68
RanPAC+FT3	97.83	44.49	21.44	16.76	22.05	87.94	78.94	40.06	74.41	95.84	57.97
L2P+FT3	11.15	2.06	1.50	1.90	14.97	21.12	31.94	11.72	89.01	98.32	28.37
DAP+FT3	14.86	0.83	2.02	4.66	18.48	30.66	27.34	15.91	90.01	98.91	30.37
D-Prompt+FT3	12.69	0.69	0.30	0.69	0.93	14.53	3.09	15.84	25.55	96.72	17.10
C-Prompt+FT3	14.16	0.51	1.02	0.64	2.30	13.02	8.80	9.52	39.79	95.61	18.54
LEAR+FT3	95.89	81.25	69.57	84.12	85.30	98.56	92.92	45.45	96.60	99.30	84.90

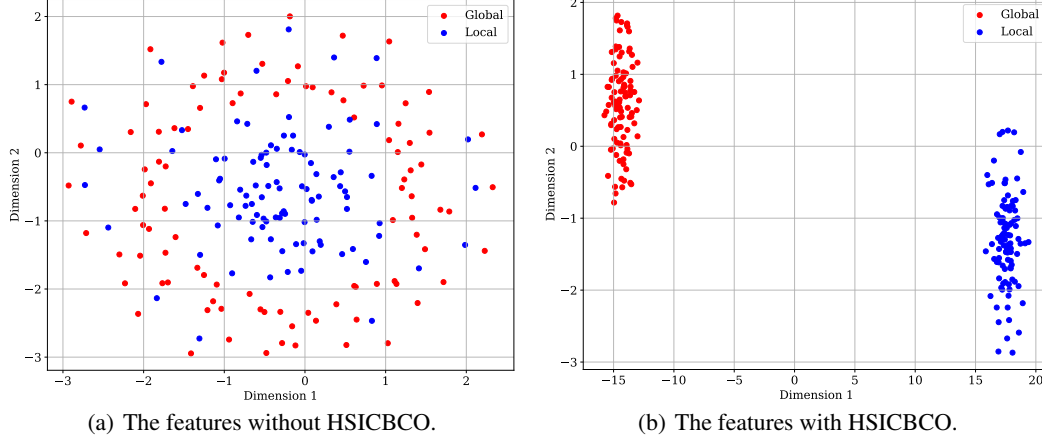


Figure 3: The comparison of backbone features between “W/O HSIC” and LEAR.

As shown in Table 5, the results indicate that merely scaling up model parameters cannot guarantee performance gains in MDCL scenario, which necessitates the incorporation of regularization constraints during backbone adaptation to mitigate adverse parameter optimization caused by domain shift.

D.5 Visualization of the HSICBCO approach.

Both the global and local backbones are initialized with identical pretrained weights. Under the guidance of MIBPA and KLDBFA, they learn task-general and task-specific representations, respectively. However, their feature representations still exhibit strong correlations. At the end phase of the task CUB-200 in the TRC scenario, we randomly sampled 100 training images from CUB-200 and fed them into both the global and local backbones to extract features. These features were then visualized using T-SNE dimensionality reduction, resulting in Figure 3. The proposed HSICBCO module effectively decouples these representations, demonstrating its capability to promote distinct and complementary feature learning.

D.6 Visualization of the ESM’s memory distributions.

The file “ETI_distributions.html” in “SupplementaryMaterial.zip” provides an interactive 3D visualization of all memory distributions generated by ESM after learning the ETI task sequence. These distributions are projected into a 3D space using Principal Component Analysis (PCA), with diamond marker representing the mean vector of a distribution and ellipsoidal surface encodes the

corresponding covariance matrix. ESM computes Mahalanobis distances between stored distributions and incoming task samples to select experts for either network expansion or test-time inference.

D.7 Analysis of other distance metrics in ESM

Since the model cannot access other test samples while processing a given test sample during inference, it is unable to generate a gaussian distribution for the test samples as done at the end phase of training. Consequently, metrics such as Maximum Mean Discrepancy (MMD) and Wasserstein distance between the test sample distribution and memory distributions cannot be computed to select the appropriate expert. Therefore, we consider calculate the normalized entropy for each resulting distribution from N trained experts when processing testing data \mathbf{x}_j , and mitigate class-imbalance bias in entropy computation by incorporating the class number \mathbf{C}_i saved from the expert \mathcal{E}_i 's training domain :

$$c^* = \arg \min_{t=1, \dots, N} \left(\frac{-\sum_{k=1}^{\mathbf{C}_i} p_k^{(t,j)} \log p_k^{(t,j)}}{\log \mathbf{C}_i} \right), \quad (1)$$

where the expert \mathcal{E}_{c^*} is chosen for the prediction and $p_k^{(t,j)}$ denotes the prediction on the j -th sample made by the t -th expert. This method is capable of estimating the uncertainty of each expert with respect to the current test sample. However, in practice, we observe that it performs well only for experts trained on relatively easy datasets (e.g., CIFAR10, CropDiseases), while its stability remains limited on data domains with higher classification difficulty (e.g., ChestX or ImageNet-R). Therefore, in the MDCL scenarios, computing the Mahalanobis distance between test samples and memory distributions is the most appropriate way to select expert for LEAR. Specifically, we perform expert selection only during the inference of a small subset of test samples (e.g., the first 50 samples of each task), while the remaining samples are processed without further selection, thereby reducing computational overhead.

D.8 Other clarifications

Storage at the end of the task. Following the network's input-output pipeline, we preserve the following components per task: (1) The last three layers of both global and local backbones, which are saved and frozen as Frozen-Global and Frozen-Local backbones (overwrite their corresponding components saved from the previous task); (2) Task-specific mean vectors and covariance matrices which computed from the features extracted by the Static backbone for a random subset of training samples.

Total parameter count. The total parameter count reaches approximately 241.7M, with the following detailed composition: (1) Two complete ViT backbones (Global and Local) comprise 172M (86M*2) parameters; (2) Three backbone variants (Frozen-Global, Frozen-Local, and Static) collectively contribute 63.6M (21.2M*3) parameters; (3) 10 task-specific expert networks account for about 4.8M (0.38M*10 for 10 fc layers + 1M for all classifiers) parameters; and (4) Mean vectors and covariance matrices for 10 tasks sum to 1.3M (0.13M* 10 for 10 mu & sigma) parameters.

E Discussion of Contributions, Limitations and Societal Impacts

E.1 Contributions and Limitations

In this paper, we propose LEAR, a novel framework for Multi-Domain Continual Learning that simultaneously addresses stability and plasticity. Specifically, built on a collaborative backbone structure, we introduce MIBPA and KLDBFA to maintain history prediction consistency and task-specific feature alignment during model updates, while HSICBCO ensures disentangling representations between the global and local backbone. Additionally, ESM dynamically selects relevant experts for efficient network expansion and test evaluation. The empirical results demonstrate the effectiveness of the proposed approach. The primary limitation of this paper is that the proposed approach would contain a considerable number of parameters due to fine-tuning the backbones. To address this issue, we will further propose a novel expert merging technology with teacher-student distillation for effective model compression as our future work.

E.2 Societal Impacts

The proposed LEAR framework demonstrates significant societal benefits by advancing multi-domain continual learning (MDCL) in critical areas. Its dynamic expansion mechanism and parameter-efficient design enable rapid adaptation to novel tasks (e.g., medical diagnosis on ChestX and CropDiseases datasets, environmental monitoring via EuroSAT and RESISC45, while maintaining high accuracy across various domains. By reducing computational overhead (lowest GPU/CPU utilization among baselines), the framework can be easily deployed on resource-constrained edge devices, thereby lowering energy consumption. Furthermore, the method’s emphasis on stability-plasticity trade-offs ensures robust performance in evolving real-world scenarios like autonomous systems and personalized education, enabling the development of algorithms that actively leverage (rather than passively adapting to) domain structures like MDCL.

References

- [1] A. Achille, T. Eccles, L. Matthey, C. Burgess, N. Watters, A. Lerchner, and I. Higgins. Life-long disentangled representation learning with cross-domain latent homologies. In *Advances in Neural Information Processing Systems (NeurIPS)*, pages 9873–9883, 2018. 2
- [2] Elahe Arani, Fahad Sarfraz, and Bahram Zonooz. Learning fast, learning slow: A general continual learning method based on complementary learning system. *arXiv preprint arXiv:2201.12604*, 2022. 3, 9
- [3] Lukas Bossard, Matthieu Guillaumin, and Luc Van Gool. Food-101—mining discriminative components with random forests. In *Computer vision—ECCV 2014: 13th European conference, zurich, Switzerland, September 6-12, 2014, proceedings, part VI 13*, pages 446–461. Springer, 2014. 6
- [4] Pietro Buzzega, Matteo Boschini, Angelo Porrello, Davide Abati, and Simone Calderara. Dark experience for general continual learning: a strong, simple baseline. *Advances in neural information processing systems*, 33:15920–15930, 2020. 2, 5

- 338 [5] Hyuntak Cha, Jaeho Lee, and Jinwoo Shin. Co2l: Contrastive continual learning. In *Proceed-*
339 *ings of the IEEE/CVF International Conference on Computer Vision*, pages 9516–9525, 2021.
340 2, 5
- 341 [6] Arslan Chaudhry, Albert Gordo, Puneet Dokania, Philip Torr, and David Lopez-Paz. Using
342 hindsight to anchor past knowledge in continual learning. In *Proceedings of the AAAI confer-*
343 *ence on artificial intelligence*, volume 35, pages 6993–7001, 2021. 2
- 344 [7] Gong Cheng, Junwei Han, and Xiaoqiang Lu. Remote sensing image scene classification:
345 Benchmark and state of the art. *Proceedings of the IEEE*, 105(10):1865–1883, 2017. 6
- 346 [8] Mircea Cimpoi, Subhransu Maji, Iasonas Kokkinos, Sammy Mohamed, and Andrea Vedaldi.
347 Describing textures in the wild. In *Proceedings of the IEEE conference on computer vision*
348 *and pattern recognition*, pages 3606–3613, 2014. 6
- 349 [9] Noel Codella, Veronica Rotemberg, Philipp Tschandl, M Emre Celebi, Stephen Dusza, David
350 Gutman, Brian Helba, Aadi Kalloo, Konstantinos Liopyris, Michael Marchetti, et al. Skin
351 lesion analysis toward melanoma detection 2018: A challenge hosted by the international skin
352 imaging collaboration (isic). *arXiv preprint arXiv:1902.03368*, 2019. 6
- 353 [10] Matthias De Lange and Tinne Tuytelaars. Continual prototype evolution: Learning online from
354 non-stationary data streams. In *Proc. of the IEEE/CVF International Conference on Computer*
355 *Vision*, pages 8250–8259, 2021. 5
- 356 [11] Arthur Douillard, Alexandre Ramé, Guillaume Couairon, and Matthieu Cord. Dytox: Trans-
357 formers for continual learning with dynamic token expansion. In *Proceedings of the IEEE/CVF*
358 *Conference on Computer Vision and Pattern Recognition*, pages 9285–9295, 2022. 3
- 359 [12] Li Fei-Fei, Rob Fergus, and Pietro Perona. Learning generative visual models from few train-
360 ing examples: An incremental bayesian approach tested on 101 object categories. In *IEEE*
361 *CVPR-workshop*, pages 1–9, 2004. 6
- 362 [13] Qiankun Gao, Chen Zhao, Yifan Sun, Teng Xi, Gang Zhang, Bernard Ghanem, and Jian Zhang.
363 A unified continual learning framework with general parameter-efficient tuning. In *Proceed-*
364 *ings of the IEEE/CVF International Conference on Computer Vision*, pages 11483–11493,
365 2023. 2
- 366 [14] Xinyuan Gao, Songlin Dong, Yuhang He, Qiang Wang, and Yihong Gong. Beyond prompt
367 learning: Continual adapter for efficient rehearsal-free continual learning. In *European Con-*
368 *ference on Computer Vision*, pages 89–106. Springer, 2024. 2
- 369 [15] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville,
370 and Y. Bengio. Generative adversarial nets. In *Proc. Advances in Neural Inf. Proc. Systems*
371 *(NIPS)*, pages 2672–2680, 2014. 2
- 372 [16] Jianping Gou, Baosheng Yu, Stephen J Maybank, and Dacheng Tao. Knowledge distillation:
373 A survey. *International Journal of Computer Vision*, 129:1789–1819, 2021. 2

- [17] Patrick Helber, Benjamin Bischke, Andreas Dengel, and Damian Borth. Eurosat: A novel dataset and deep learning benchmark for land use and land cover classification. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, 12(7):2217–2226, 2019. 6
- [18] Dan Hendrycks, Steven Basart, Norman Mu, Saurav Kadavath, Frank Wang, Evan Dorundo, Rahul Desai, Tyler Zhu, Samyak Parajuli, Mike Guo, Dawn Song, Jacob Steinhardt, and Justin Gilmer. The many faces of robustness: A critical analysis of out-of-distribution generalization. *ICCV*, 2021. 6
- [19] G. Hinton, O. Vinyals, and J. Dean. Distilling the knowledge in a neural network. In *Proc. NIPS Deep Learning Workshop, arXiv preprint arXiv:1503.02531*, 2014. 2
- [20] Dahuin Jung, Dongyoon Han, Jihwan Bang, and Hwanjun Song. Generating instance-level prompts for rehearsal-free continual learning. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 11847–11857, 2023. 6
- [21] Haeyong Kang, Rusty John Lloyd Mina, Sultan Rizky Hikmawan Madjid, Jaehong Yoon, Mark Hasegawa-Johnson, Sung Ju Hwang, and Chang D Yoo. Forget-free continual learning with winning subnetworks. In *International Conference on Machine Learning*, pages 10734–10750. PMLR, 2022. 3
- [22] Junsu Kim, Hoseong Cho, Jihyeon Kim, Yihalem Yimolal Tiruneh, and Seungryul Baek. Sd-dgr: Stable diffusion-based deep generative replay for class incremental object detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 28772–28781, 2024. 2
- [23] D. P. Kingma and M. Welling. Auto-encoding variational Bayes. *arXiv preprint arXiv:1312.6114*, 2013. 2
- [24] Jonathan Krause, Michael Stark, Jia Deng, and Li Fei-Fei. 3d object representations for fine-grained categorization. In *Proceedings of the IEEE international conference on computer vision workshops*, pages 554–561, 2013. 6
- [25] Alex Krizhevsky and Geoffrey Hinton. Learning multiple layers of features from tiny images. Technical report, Univ. of Toronto, 2009. 6
- [26] Alex Krizhevsky, Geoffrey Hinton, et al. Learning multiple layers of features from tiny images. 2009. 6
- [27] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E. Hinton. Imagenet classification with deep convolutional neural networks. In *Advances in Neural Inf. Proc. Systems (NIPS)*, pages 1097–1105, 2012. 7
- [28] Ya Le and Xuan Yang. Tiny imageNet visual recognition challenge. Technical report, Univ. of Stanford, 2015. 6
- [29] Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner. Gradient-based learning applied to document recognition. *Proc. of the IEEE*, 86(11):2278–2324, 1998. 6

- [30] Chuqiao Li, Zhiwu Huang, Danda Pani Paudel, Yabin Wang, Mohamad Shahbazi, Xiaopeng Hong, and Luc Van Gool. A continual deepfake detection benchmark: Dataset, methods, and essentials. In *Proceedings of the IEEE/CVF winter conference on applications of computer vision*, pages 1339–1349, 2023. 5
- [31] Yukun Li, Guansong Pang, Wei Suo, Chenchen Jing, Yuling Xi, Lingqiao Liu, Hao Chen, Guoqiang Liang, and Peng Wang. Coleclip: Open-domain continual learning via joint task prompt and vocabulary learning. *IEEE Transactions on Neural Networks and Learning Systems*, 2025. 6
- [32] Z. Li and D. Hoiem. Learning without forgetting. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 40(12):2935–2947, 2017. 2
- [33] Yan-Shuo Liang and Wu-Jun Li. Inflora: Interference-free low-rank adaptation for continual learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 23638–23647, 2024. 2
- [34] Vincenzo Lomonaco and Davide Maltoni. Core50: a new dataset and benchmark for continuous object recognition. In *Conference on robot learning*, pages 17–26. PMLR, 2017. 5
- [35] Xinhong Ma, Yiming Wang, Hao Liu, Tianyu Guo, and Yunhe Wang. When visual prompt tuning meets source-free domain adaptive semantic segmentation. *Advances in Neural Information Processing Systems*, 36:6690–6702, 2023. 2
- [36] Subhransu Maji, Esa Rahtu, Juho Kannala, Matthew Blaschko, and Andrea Vedaldi. Fine-grained visual classification of aircraft. *arXiv preprint arXiv:1306.5151*, 2013. 5
- [37] James L McClelland, Bruce L McNaughton, and Randall C O’Reilly. Why there are complementary learning systems in the hippocampus and neocortex: insights from the successes and failures of connectionist models of learning and memory. *Psychological review*, 102(3):419, 1995. 3
- [38] Mark D McDonnell, Dong Gong, Amin Parvaneh, Ehsan Abbasnejad, and Anton van den Hengel. Ranpac: Random projections and pre-trained models for continual learning. *Advances in Neural Information Processing Systems*, 36, 2024. 2, 3
- [39] Sharada P Mohanty, David P Hughes, and Marcel Salathé. Using deep learning for image-based plant disease detection. *Frontiers in plant science*, 7:215232, 2016. 6
- [40] Maria-Elena Nilsback and Andrew Zisserman. Automated flower classification over a large number of classes. In *2008 Sixth Indian conference on computer vision, graphics & image processing*, pages 722–729. IEEE, 2008. 6
- [41] Omkar M Parkhi, Andrea Vedaldi, Andrew Zisserman, and CV Jawahar. Cats and dogs. In *2012 IEEE conference on computer vision and pattern recognition*, pages 3498–3505. IEEE, 2012. 6

- [42] Xingchao Peng, Qinxun Bai, Xide Xia, Zijun Huang, Kate Saenko, and Bo Wang. Moment matching for multi-source domain adaptation. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 1406–1415, 2019. 5
- [43] Quang Pham, Chenghao Liu, and Steven Hoi. Dualnet: Continual learning, fast and slow. *Advances in Neural Information Processing Systems*, 34, 2021. 3
- [44] J. Ramapuram, M. Gregorova, and A. Kalousis. Lifelong generative modeling. In *Proc. Int. Conf. on Learning Representations (ICLR)*, *arXiv preprint arXiv:1705.09847*, 2017. 2
- [45] Vinay V Ramasesh, Ethan Dyer, and Maithra Raghu. Anatomy of catastrophic forgetting: Hidden representations and task semantics. *arXiv preprint arXiv:2007.07400*, 2020. 9
- [46] Sylvestre-Alvise Rebuffi, Alexander Kolesnikov, Georg Sperl, and Christoph H Lampert. iCaRL: Incremental classifier and representation learning. In *Proc. of the IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, pages 2001–2010, 2017. 2
- [47] Sreemanananth Sadanand and Jason J Corso. Action bank: A high-level representation of activity in video. In *2012 IEEE Conference on computer vision and pattern recognition*, pages 1234–1241. IEEE, 2012. 9
- [48] H. Shin, J. K. Lee, J. Kim, and J. Kim. Continual learning with deep generative replay. In *Advances in Neural Inf. Proc. Systems (NIPS)*, pages 2990–2999, 2017. 2
- [49] James Seale Smith, Leonid Karlinsky, Vyshnavi Gutta, Paola Cascante-Bonilla, Donghyun Kim, Assaf Arbelle, Rameswar Panda, Rogerio Feris, and Zsolt Kira. Coda-prompt: Continual decomposed attention-based prompting for rehearsal-free continual learning. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 11909–11919, 2023. 2
- [50] Vinay Kumar Verma, Kevin J Liang, Nikhil Mehta, Piyush Rai, and Lawrence Carin. Efficient feature transformations for discriminative and generative continual learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 13865–13875, 2021. 3
- [51] Catherine Wah, Steve Branson, Peter Welinder, Pietro Perona, and Serge Belongie. The caltech-ucsd birds-200-2011 dataset. Technical Report CNS-TR-2011-001, California Institute of Technology, 2011. 6
- [52] Fu-Yun Wang, Da-Wei Zhou, Liu Liu, Han-Jia Ye, Yatao Bian, De-Chuan Zhan, and Peilin Zhao. Beef: Bi-compatible class-incremental learning via energy-based expansion and fusion. In *The eleventh international conference on learning representations*, 2022. 3
- [53] Fu-Yun Wang, Da-Wei Zhou, Han-Jia Ye, and De-Chuan Zhan. Foster: Feature boosting and compression for class-incremental learning. In *European conference on computer vision*, pages 398–414. Springer, 2022. 3

- [54] Liyuan Wang, Jingyi Xie, Xingxing Zhang, Mingyi Huang, Hang Su, and Jun Zhu. Hierarchical decomposition of prompt-based continual learning: Rethinking obscured sub-optimality. *Advances in Neural Information Processing Systems*, 36:69054–69076, 2023. 2
- [55] Liyuan Wang, Jingyi Xie, Xingxing Zhang, Hang Su, and Jun Zhu. Hide-pet: continual learning via hierarchical decomposition of parameter-efficient tuning. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2025. 2
- [56] Xiaosong Wang, Yifan Peng, Le Lu, Zhiyong Lu, Mohammadhadi Bagheri, and Ronald M Summers. Chestx-ray8: Hospital-scale chest x-ray database and benchmarks on weakly-supervised classification and localization of common thorax diseases. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2097–2106, 2017. 6
- [57] Yabin Wang, Zhiwu Huang, and Xiaopeng Hong. S-prompts learning with pre-trained transformers: An occam’s razor for domain incremental learning. *Advances in Neural Information Processing Systems*, 35:5682–5695, 2022. 2, 5
- [58] Zhenyi Wang, Yan Li, Li Shen, and Heng Huang. A unified and general framework for continual learning. *arXiv preprint arXiv:2403.13249*, 2024. 2, 9
- [59] Zifeng Wang, Zizhao Zhang, Sayna Ebrahimi, Ruoxi Sun, Han Zhang, Chen-Yu Lee, Xiaoqi Ren, Guolong Su, Vincent Perot, Jennifer Dy, et al. Dualprompt: Complementary prompting for rehearsal-free continual learning. In *European conference on computer vision*, pages 631–648. Springer, 2022. 2
- [60] Zifeng Wang, Zizhao Zhang, Chen-Yu Lee, Han Zhang, Ruoxi Sun, Xiaoqi Ren, Guolong Su, Vincent Perot, Jennifer Dy, and Tomas Pfister. Learning to prompt for continual learning. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 139–149, 2022. 2, 5
- [61] Yichen Wu, Hongming Piao, Long-Kai Huang, Renzhen Wang, Wanhua Li, Hanspeter Pfister, Deyu Meng, Kede Ma, and Ying Wei. Sd-lora: Scalable decoupled low-rank adaptation for class incremental learning. In *The Thirteenth International Conference on Learning Representations*, 2025. 2
- [62] Jianxiong Xiao, James Hays, Krista A Ehinger, Aude Oliva, and Antonio Torralba. Sun database: Large-scale scene recognition from abbey to zoo. In *2010 IEEE computer society conference on computer vision and pattern recognition*, pages 3485–3492. IEEE, 2010. 6
- [63] Mengqi Xue, Haofei Zhang, Jie Song, and Mingli Song. Meta-attention for vit-backed continual learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 150–159, 2022. 3
- [64] Seungryong Yoo, Eunji Kim, Dahuin Jung, Jungbeom Lee, and Sungroh Yoon. Improving visual prompt tuning for self-supervised vision transformers. In *International Conference on Machine Learning*, pages 40075–40092. PMLR, 2023. 2

- 518 [65] Jiazuo Yu, Yunzhi Zhuge, Lu Zhang, Ping Hu, Dong Wang, Huchuan Lu, and You He. Boosting
519 continual learning of vision-language models via mixture-of-experts adapters. In *Proceedings*
520 *of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 23219–
521 23230, 2024. 3
- 522 [66] M. Zhai, L. Chen, F. Tung, J He, M. Nawhal, and G. Mori. Lifelong GAN: Continual learning
523 for conditional image generation. In *Proc. of the IEEE/CVF Int. Conf. on Computer Vision*
524 *(ICCV)*, pages 2759–2768, 2019. 2
- 525 [67] Gengwei Zhang, Liyuan Wang, Guoliang Kang, Ling Chen, and Yunchao Wei. Slca: Slow
526 learner with classifier alignment for continual learning on a pre-trained model. In *Proceedings*
527 *of the IEEE/CVF International Conference on Computer Vision*, pages 19148–19158, 2023. 2
- 528 [68] Zangwei Zheng, Mingyuan Ma, Kai Wang, Ziheng Qin, Xiangyu Yue, and Yang You. Prevent-
529 ing zero-shot transfer degradation in continual learning of vision-language models. In *Pro-*
530 *ceedings of the IEEE/CVF international conference on computer vision*, pages 19125–19136,
531 2023. 5
- 532 [69] Da-Wei Zhou, Hai-Long Sun, Jingyi Ning, Han-Jia Ye, and De-Chuan Zhan. Continual learn-
533 ing with pre-trained models: A survey. *arXiv preprint arXiv:2401.16386*, 2024. 2