



Adapter Learning in Pretrained Feature Extractor for Continual Learning of Diseases

Wentao Zhang^{1,4}, Yujun Huang^{1,4}, Tong Zhang², Qingsong Zou^{1,3},
Wei-Shi Zheng^{1,4}, and Ruixuan Wang^{1,2,4}(✉)

¹ School of Computer Science and Engineering, Sun Yat-sen University,
Guangzhou, China

wangruix5@mail.sysu.edu.cn

² Peng Cheng Laboratory, Shenzhen, China

³ Guangdong Province Key Laboratory of Computational Science,
Sun Yat-sen University, Guangzhou, China

⁴ Key Laboratory of Machine Intelligence and Advanced Computing, MOE,
Guangzhou, China

Abstract. Currently intelligent diagnosis systems lack the ability of continually learning to diagnose new diseases once deployed, under the condition of preserving old disease knowledge. In particular, updating an intelligent diagnosis system with training data of new diseases would cause catastrophic forgetting of old disease knowledge. To address the catastrophic forgetting issue, an **Adapter-based Continual Learning** framework called ACL is proposed to help effectively learn a set of new diseases at each round (or task) of continual learning, without changing the shared feature extractor. The learnable lightweight task-specific adapter(s) can be flexibly designed (e.g., two convolutional layers) and then added to the pretrained and fixed feature extractor. Together with a specially designed task-specific head which absorbs all previously learned old diseases as a single ‘out-of-distribution’ category, task-specific adapter(s) can help the pretrained feature extractor more effectively extract discriminative features between diseases. In addition, a simple yet effective fine-tuning is applied to collaboratively fine-tune multiple task-specific heads such that outputs from different heads are comparable and consequently the appropriate classifier head can be more accurately selected during model inference. Extensive empirical evaluations on three image datasets demonstrate the superior performance of ACL in continual learning of new diseases. The source code is available at https://github.com/GiantJun/CL_Pytorch.

Keywords: Continual learning · Adapter · Disease diagnosis

W. Zhang and Y. Huang—Authors contributed equally.

Supplementary Information The online version contains supplementary material available at https://doi.org/10.1007/978-3-031-43895-0_7.

© The Author(s), under exclusive license to Springer Nature Switzerland AG 2023
H. Greenspan et al. (Eds.): MICCAI 2023, LNCS 14221, pp. 68–78, 2023.
https://doi.org/10.1007/978-3-031-43895-0_7

1 Introduction

Deep neural networks have shown expert-level performance in various disease diagnoses [35, 36]. In practice, a deep neural network is often limited to the diagnosis of only a few diseases, partly because it is challenging to collect enough training data of all diseases even for a specific body tissue or organ. One possible solution is to enable a deployed intelligent diagnosis system to continually learn new diseases with collected new training data later. However, if old data are not accessible due to certain reasons (e.g., challenge in data sharing), current intelligent systems will suffer from catastrophic forgetting of old knowledge when learning new diseases [17].

Multiple approaches have been proposed to alleviate the catastrophic forgetting issue. One approach aims to determine part of the model parameters which are crucial to old knowledge and tries to keep these parameters unchanged during learning new knowledge [5, 15, 18]. Another approach aims to preserve old knowledge by making the updated model imitate the behaviour (e.g., output at certain layer) of the old model particularly with the help of knowledge distillation technique [8, 17, 34]. Storing a small amount of old data or synthesizing old data relevant to old knowledge and using them together with training data of new knowledge can often help significantly alleviate forgetting of old knowledge [2, 3, 19, 22]. Although the above approaches can help the updated model keep old knowledge to some extent, they often fall into the dilemma of model plasticity (for new knowledge learning) and stability (for old knowledge preservation). In order to resolve this dilemma, new model components (e.g., neurons or layers in neural networks) can be added specifically for learning new knowledge, while old parameters are largely kept unchanged for old knowledge [20, 26, 30]. While this approach has shown state-of-the-art continual learning performance, it faces the problem of rapid model expansion and effective fusion of new model components into the existing ones. To alleviate the model expansion issue and meanwhile well preserve old knowledge, researchers have started to explore the usage of a pretrained and fixed feature extractor for the whole process of continual learning [14, 27, 28, 32], where the challenge is to discriminate between different classes of knowledge with limited learnable parameters.

In this study, inspired by recent advances in transfer learning in natural language processing [7, 12], we propose adding a light-weight learnable module called adapter to a pretrained and fixed convolutional neural network (CNN) for effective continual learning of new knowledge. For each round of continual learning, the CNN model will be updated to learn a set of new classes (hereinafter also called learning a new task). The learnable task-specific adapters are added between consecutive convolutional stages to help the pretrained CNN feature extractor more effectively extract discriminative features of new diseases. To the best of our knowledge, it is the first time to apply the idea of CNN adapter in the continual learning field. In addition, to keep extracted features discriminative between different tasks, a special task-specific classifier head is added when learning each new task, in which all previously learned old classes are considered as the ‘out-of-distribution’ (OOD) class and correspond to an additional

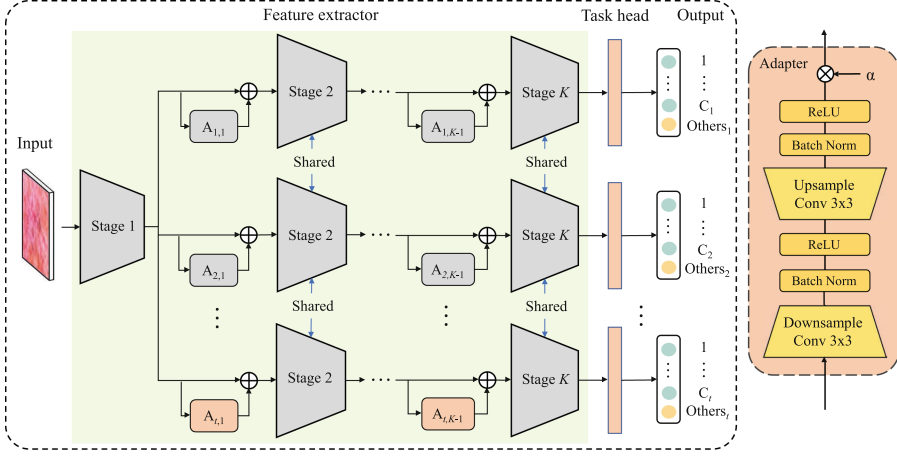


Fig. 1. The proposed framework for continual learning of new diseases. Left: task-specific adapters ($\{A_{t,1}, A_{t,2}, \dots, A_{t,K-1}\}$ in orange) between consecutive convolutional stages are added and learned for a new task. After learning the new task-specific adapters, all tasks' classifier heads (orange rectangles) are fine-tuned with balanced training data. The pretrained feature extractor is fixed during the continual learning process. Right: the structure of each adapter, with α representing the global scaling. (Color figure online)

output neuron in each task-specific classifier head. A simple yet effective fine-tuning strategy is applied to calibrate outputs between multiple task-specific heads. Extensive empirical evaluations on three image datasets show that the proposed method outperforms existing continual learning methods by a large margin, consistently supporting the effectiveness of the proposed method.

2 Method

This study aims to improve continual learning performance of an intelligent diagnosis system. At each learning round, following previous studies [2, 3] which show that rehearsal with old samples can significantly improve continual learning performance, the system will be updated based on the training data of new diseases and preserved small subset for each previously learned disease. During inference, the system is expected to accurately diagnose all learned diseases, without knowing which round (i.e., task) the class of any test input is from.

2.1 Overall Framework

We propose an **Adapter-based Continual Learning** framework called ACL with a multi-head training strategy. With the motivation to make full use of readily available pretrained CNN models and slow down the speed of model expansion that appears in some state-of-the-art continual learning methods (e.g.,

DER [30]), and inspired by a recently developed transfer learning strategy Delta tuning for downstream tasks in natural language process [7], we propose adding a learnable light-weight adapter between consecutive convolutional stages in a pretrained and fixed CNN model when learning new classes of diseases at each learning round (Fig. 1). Each round of continual learning as a unique task is associated with task-specific adapters and a task-specific classifier head. Model update at each round of continual learning is to find optimal parameters in the newly added task-specific adapters and classifier head. During inference, since multiple classifier heads exist, the correct head containing the class of a given input is expected to be selected. In order to establish the potential connections between tasks and further boost the continual learning performance, a two-stage multi-head learning strategy was proposed by including the idea of out-of-distribution (OOD) detection.

2.2 Task-Specific Adapters

State-of-the-art continual learning methods try to preserve old knowledge by either combining old fixed feature extractors with the newly learned feature extractor [30], or by using a shared and fixed pretrained feature extractor [32]. However, simply combining feature extractors over rounds of continual learning would rapidly expand the model, while using a shared and fixed feature extractor could largely limit model ability of learning new knowledge because only the model head can be tuned to discriminate between classes. To resolve this dilemma, we propose adding a light-weight task-specific module called adapter into a single pretrained and fixed CNN feature extractor (e.g., with the ResNet backbone), such that the model expands very slowly and the adapter-tuned feature extractor for each old task is fixed when learning a new task. In this way, old knowledge largely stored in the feature extractor and associated task-specific adapters will be well preserved when the model learns new knowledge at subsequent rounds of continual learning. Formally, suppose the pretrained CNN feature extractor contains K stages of convolutional layers (e.g., 5 stages in ResNet), and the output feature maps from the k -th stage is denoted by $\mathbf{z}_k, k \in \{1, \dots, K-1\}$. Then, when the model learns a new task at the t -th round of continual learning, a task-specific adapter $A_{t,k}$ is added between the k -th and $(k+1)$ -th stages as follows,

$$\hat{\mathbf{z}}_{t,k} = A_{t,k}(\mathbf{z}_k) + \mathbf{z}_k, \quad (1)$$

where the adapter-tuned output $\hat{\mathbf{z}}_{t,k}$ will be used as input to the $(k+1)$ -th stage. The light-weight adapter can be flexibly designed. In this study, a simple two-layer convolution module followed by a global scaling is designed as the adapter (Fig. 1, Right). The input feature maps to the adapter are spatially downsampled by the first convolutional layer and then upsampled by the second layer. The global scaling factor α is learned together with the two convolutional layers. The proposed task-specific adapter for continual learning is inspired by Delta tuning [7] which adds learnable 2-layer perceptron(s) into a pretrained and fixed

Transformer model in natural language processing. Different from Delta tuning which is used as a transfer learning strategy to adapt a pretrained model for any individual downstream task, the proposed task-specific adapter is used as a continual learning strategy to help a model continually learn new knowledge over multiple rounds (i.e., multiple tasks) in image processing, with each round corresponding to a specific set of adapters.

Also note that the proposed adapter differs from existing adapters in CLIP-Adapter (CA) [9] and Tip-Adapter (TA) [33]. First, in structure, CA and TA use 2-layer MLP or cache model, while ours uses a 2-layer convnet with a global scaling factor. Second, the number and locations of adapters in model are different. CA and TA use adapter only at output of the last layer, while ours appears between each two consecutive CNN stages. Third, the roles of adapters are different. Existing adapters are for few-shot classification, while ours is for continual learning. It is also different from current prompt tuning. Prompts appear as part of input to the first or/and intermediate layer(s) of model, often in the form of learnable tokens for Transformer or image regions for CNNs. In contrast, our adapter appears as an embedded neural module for each two consecutive CNN stages, in the form of sub-network.

2.3 Task-Specific Head

Task-specific head is proposed to alleviate the potential feature fusion issue in current state-of-the-art methods [30] which combine task-specific features by a unified classifier head. In particular, if feature outputs from multiple task-specific feature extractors are simply fused by concatenating or averaging followed by a unified 1- or 2-layer perceptron (as in [30]), discriminative feature information appearing only in those classes of a specific task could become less salient after fusion with multiple (possibly less discriminative) features from other task-specific feature extractors. As a result, current state-of-the-art methods often require storing relatively more old data to help train a discriminative unified classifier head between different classes. To avoid the possible reduction in feature discriminability, we propose not fusing features from multiple feature extractors, but a task-specific classifier head for each task. Each task-specific head consists of one fully connected layer followed by a softmax operator. Specially, for a task containing C new classes, one additional class absorbing all previously learned old classes (‘others’ output neuron in Fig. 1) is also included, and therefore the number of output elements from the softmax will be $C + 1$. The ‘others’ output is used to predict the probability of the input image being from certain class of any other task rather than from the current task. In other words, each task-specific head has the ability of out-of-distribution (OOD) ability with the help of the ‘others’ output neuron. At the t -round of continual learning (i.e., for the t -th task learning), the task-specific adapters and the task-specific classifier head can be directly optimized, e.g., by cross-entropy loss, with the C new classes of training data and the ‘others’ class of all preserved old data.

However, training the task-specific classifier head without considering its relationship with existing classifier heads of previously learned tasks may cause the

head selection issue during model inference. For example, a previously learned old classifier head may consider an input of latterly learned class as one of the old classes (correspondingly the ‘others’ output from the old head will be low). In other words, the ‘others’ outputs from multiple classifier heads cannot not be reliably compared (i.e., not calibrated) with one another if each classifier head is trained individually. In this case, if all classifier heads consider a test input as ‘others’ class with high confidence or multiple classifier heads consider a test input as one of their classes, it would become difficult to choose an appropriate classifier head for final prediction. To resolve the head selection issue, after initial training of the current task’s adapters and classifier head, all the tasks’ heads are fine-tuned together such that all ‘others’ outputs from the multiple heads are comparable. In short, at the t -th round of continual learning, the t task-specific classifier heads can be fine-tuned by minimizing the loss function L ,

$$L = \frac{1}{t} \sum_{s=1}^t L_s^c, \quad (2)$$

where L_s^c is the cross-entropy loss for the s -th classifier head. Following the fine-tuning step in previous continual learning studies [4, 30], training data of the current t -th task are sub-sampled such that training data in the fine-tuning step are balanced across all learned classes so far. Note that for each input image, multiple runs of feature extraction are performed, with each run adding adapters of a different task to the original feature extractor and extracting the feature vector for the corresponding task-specific head. Also note that in the fine-tuning step, adapters of all tasks are fixed and not tuned. Compared to training the adapters of each task with all training data of the corresponding task, fine-tuning these adapters would likely cause over-fitting of the adapters to the sub-sampled data and therefore is avoided in the fine-tuning step.

Once the multi-head classifier is fine-tuned at the t -th round of continual learning, the classifier can be applied to predict any test data as one of all the learned classes so far. First, the task head with the smallest ‘others’ output probability (among all t ‘others’ outputs) is selected, and then the class with the highest output from the selected task head is selected as the final prediction result. Although unlikely selected, the ‘others’ class in the selected task head is excluded for the final prediction.

3 Experiment Results

3.1 Experimental Setup

Four datasets were used to evaluate the proposed ACL (Table 1). Among them, Skin8 is imbalanced across classes and from the public challenge organized by the International Skin Imaging Collaboration (ISIC) [24]. Path16 is a subset of publicly released histopathology images collated from multiple publicly available datasets [1, 6, 13, 25, 29, 35], including eleven diseases and five normal classes (see Supplementary Material for more details about dataset generation). These data

Table 1. Statistics of three datasets. ‘[600, 1024]’: the range of image width and height.

| Dataset | Classes | Train set | Test set | Number of tasks | Size |
|---------------|---------|-----------|----------|-----------------|------------------|
| Skin8 [24] | 8 | 3,555 | 705 | 4 | [600, 1024] |
| Path16 | 16 | 12,808 | 1,607 | 7 | 224×224 |
| CIFAR100 [16] | 100 | 50,000 | 10,000 | 5, 10 | 32×32 |
| MedMNIST [31] | 36 | 302,002 | 75,659 | 4 | 28×28 |

are divided into seven tasks based on source of images, including Oral cavity (OR, 2 classes), Lymph node (LY, 2 classes), Breast (BR, 2 classes), Colon (CO, 2 classes), Lung (LU, 2 classes), Stomach (ST, 4 classes), and Colorectal polyp (CP, 2 classes). In training, each image is randomly rotated and then resized to 224×224 pixels.

In all experiments, publicly released CNN models which are pretrained on the Imagenet-1K dataset were used for the fixed feature extractor. During continual learning, the stochastic gradient descent optimizer was used for task-specific adapter learning, with batch size 32, weight decay 0.0005, and momentum 0.9. The initial learning rate was 0.01 and decayed by a factor of 10 at the 70th, 100th and 130th epoch, respectively. The adapters were trained for up to 200 epochs with consistently observed convergence. For fine-tuning classifier heads, the Adam optimizer was adopted, with initial learning rate 0.001 which decayed by a factor of 10 at the 55th, and 80th, respectively. The classifier heads were fine-tuned for 100 epochs with convergence observed. Unless otherwise mentioned, ResNet18 was used as the backbone, the size of memory for storing old images was 40 on Skin8, 80 on Path16, 2000 on CIFAR100 and 200 on MedMNIST.

In continual learning, the classifier sequentially learned multiple tasks, with each task a small number of new classes (e.g., 2, 10, 20). After learning each task, the mean class recall (MCR) over all classes learned so far is used to measure the classifier’s performance. Note that MCR is equivalent to classification accuracy for class-balanced test set. For each experiment, the order of classes is fixed, and all methods were executed three times with different initialization. The mean and standard deviation of MCRs over three runs were reported.

3.2 Result Analysis

Effectiveness Evaluation: In this section, we compare ACL against state-of-the-art baselines, including iCaRL [19], DynaER [30], DER++ [3], WA [34], PODNet [8], and UCIR [11]. In addition, an upper-bound result (from a classifier which was trained with all classes of training data) is also reported. Similar amount of effort was taken in tuning each baseline method. As shown in Fig. 2, our method outperforms all strong baselines in almost all settings, no matter whether the classifier learn continually 2 classes each time on Skin8 (Fig. 2, first column), in two different task orders on Path16 (Fig. 2, second column), in 10 or

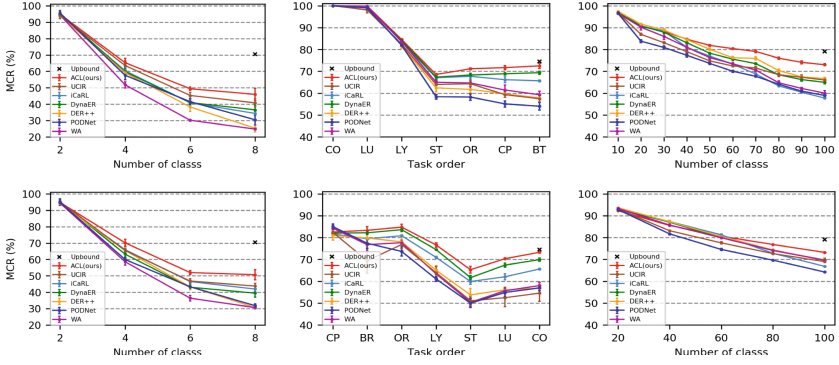


Fig. 2. Performance of continual learning on the Skin8, Path16 and CIFAR100 dataset, respectively. First column: 2 new classes each time on Skin8 respectively with memory size 16 and 40. Second column: continual learning on Path16 in different task orders. Last column: respectively learning 10 and 20 new classes each time on CIFAR100.

Table 2. Ablation study of ACL on Skin8 (with 2 new class per time) and on CIFAR100 (with 10 new classes per time). ‘T.S.H.’: inclusion of task-specific heads; ‘Others’: inclusion of the ‘others’ output neuron in each head. ‘Avg’: average of MCRs over all rounds of continual learning; ‘Last’: MCR at the last round.

| Components | | | | Skin8 | | CIFAR100 | |
|------------|---------|--------|-----------|----------------------------------|----------------------------------|----------------------------------|----------------------------------|
| T.S.H | Adapter | Others | Fine-tune | Avg | Last | Avg | Last |
| ✓ | | | | 50.91 \pm 0.18 | 27.47 \pm 0.32 | 41.68 \pm 0.04 | 18.64 \pm 0.14 |
| ✓ | ✓ | | | 60.89 \pm 0.56 | 35.4 \pm 1.20 | 47.22 \pm 0.09 | 21.20 \pm 0.13 |
| ✓ | ✓ | ✓ | | 60.90 \pm 1.97 | 42.18 \pm 2.65 | 58.72 \pm 0.07 | 46.44 \pm 0.42 |
| ✓ | ✓ | ✓ | ✓ | 66.44\pm0.90 | 50.38\pm0.31 | 82.50\pm0.39 | 73.02\pm0.47 |
| | ✓ | | ✓ | 64.80 \pm 0.87 | 46.77 \pm 1.58 | 81.67 \pm 0.39 | 70.72 \pm 0.33 |

20 classes each time on CIFAR100 (Fig. 2, last column), or in 4 different domains on MedMNIST [31] (Fig. 1 in Supplementary Material). Note that performance of most methods does not decrease (or even increase) at the last two or three learning rounds on Path16, probably because most methods perform much better on these tasks than on previous rounds of tasks.

Ablation Study: An ablation study was performed to evaluate the performance gain of each proposed component in ACL. Table 2 (first four rows) shows that the continual learning performance is gradually improved while more components are included, confirming the effectiveness of each proposed component. In addition, when fusing all the task features with a unified classifier head (Table 2, last row), the continual learning performance is clearly decreased compared to that from the proposed method (fourth row), confirming the effectiveness of task-specific classifier heads for class-incremental learning.

Table 3. Continual learning performance with different CNN backbones. Two new classes and ten new classes were learned each time on Skin 8 and CIFAR100, respectively. The range of standard deviation is [0.06, 3.57].

| Backbones | | ResNet18 | | | EfficientNet-B0 | | | MobileNetV2 | | |
|-----------|-------------|----------|--------|--------------|-----------------|--------|--------------|-------------|--------|--------------|
| Methods | | iCaRL | DynaER | ACL(ours) | iCaRL | DynaER | ACL(ours) | iCaRL | DynaER | ACL(ours) |
| Skin8 | Avg | 62.16 | 60.24 | 66.44 | 61.17 | 60.52 | 66.86 | 64.58 | 62.16 | 66.08 |
| | Last | 41.94 | 39.47 | 50.38 | 42.60 | 40.17 | 48.50 | 42.52 | 41.49 | 48.83 |
| CIFAR100 | Avg | 75.74 | 78.59 | 82.50 | 73.98 | 81.98 | 84.56 | 73.47 | 77.04 | 81.04 |
| | Last | 57.75 | 64.97 | 73.02 | 53.40 | 70.13 | 75.55 | 55.00 | 64.30 | 70.88 |

Generalizability Study: The pretrained feature extractor with different CNN backbones were used to evaluate the generalization of ACL. As shown in Table 3, ACL consistently outperforms representative strong baselines with each CNN backbone (ResNet18 [10], EfficientNet-B0 [23] and MobileNetV2 [21]) on Skin8, supporting the generalizability of our method.

4 Conclusion

Here we propose a new adapter-based strategy for class-incremental learning of new diseases. The learnable light-weight and task-specific adapters, together with the pretrained and fixed feature extractor, can effectively learn new knowledge of diseases and meanwhile keep old knowledge from catastrophic forgetting. The task-specific heads with the special ‘out-of-distribution’ output neuron within each head helps keep extracted features discriminative between different tasks. Empirical evaluations on multiple medical image datasets confirm the efficacy of the proposed method. We expect such adapter-based strategy can be extended to other continual learning tasks including lesion detection and segmentation.

Acknowledgement. This work is supported in part by the Major Key Project of PCL (grant No. PCL2023AS7-1), the National Natural Science Foundation of China (grant No. 62071502 & No. 12071496), Guangdong Excellent Youth Team Program (grant No. 2023B1515040025), and the Guangdong Provincial Natural Science Fund (grant No. 2023A1515012097).

References

1. Borkowski, A.A., Bui, M.M., Thomas, L.B., Wilson, C.P., DeLand, L.A., Mastorides, S.M.: Lung and colon cancer histopathological image dataset (LC25000). arXiv preprint [arXiv:1912.12142](https://arxiv.org/abs/1912.12142) (2019)
2. Boschini, M., et al.: Transfer without forgetting. In: Avidan, S., Brostow, G., Cissé, M., Farinella, G.M., Hassner, T. (eds.) ECCV 2022. LNCS, vol. 13683, pp. 692–709. Springer, Cham (2022). https://doi.org/10.1007/978-3-031-20050-2_40
3. Buzzega, P., Boschini, M., Porrello, A., Abati, D., Calderara, S.: Dark experience for general continual learning: a strong, simple baseline. In: NeurIPS (2020)

4. Castro, F.M., Marín-Jiménez, M.J., Guil, N., Schmid, C., Alahari, K.: End-to-end incremental learning. In: ECCV (2018)
5. Chaudhry, A., Ranzato, M., Rohrbach, M., Elhoseiny, M.: Efficient lifelong learning with a-gem. arXiv preprint [arXiv:1812.00420](https://arxiv.org/abs/1812.00420) (2018)
6. Cruz-Roa, A., et al.: Automatic detection of invasive ductal carcinoma in whole slide images with convolutional neural networks. In: Medical Imaging 2014: Digital Pathology (2014)
7. Ding, N., et al.: Delta tuning: a comprehensive study of parameter efficient methods for pre-trained language models. arXiv preprint [arXiv:2203.06904](https://arxiv.org/abs/2203.06904) (2022)
8. Douillard, A., Cord, M., Ollion, C., Robert, T., Valle, E.: PODNet: pooled outputs distillation for small-tasks incremental learning. In: Vedaldi, A., Bischof, H., Brox, T., Frahm, J.-M. (eds.) ECCV 2020. LNCS, vol. 12365, pp. 86–102. Springer, Cham (2020). https://doi.org/10.1007/978-3-030-58565-5_6
9. Gao, P., et al.: Clip-adapter: better vision-language models with feature adapters. arXiv preprint [arXiv:2110.04544](https://arxiv.org/abs/2110.04544) (2021)
10. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. In: CVPR (2016)
11. Hou, S., Pan, X., Loy, C.C., Wang, Z., Lin, D.: Lifelong learning via progressive distillation and retrospection. In: ECCV (2018)
12. Housby, N., et al.: Parameter-efficient transfer learning for NLP. In: ICML (2019)
13. Kebede, A.F.: Oral cancer dataset, version 1 (2021). <https://www.kaggle.com/datasets/ashenafasilkebede/dataset>
14. Kim, G., Liu, B., Ke, Z.: A multi-head model for continual learning via out-of-distribution replay. In: Conference on Lifelong Learning Agents (2022)
15. Kirkpatrick, J., et al.: Overcoming catastrophic forgetting in neural networks. In: PNAS (2017)
16. Krizhevsky, A., Hinton, G., et al.: Learning multiple layers of features from tiny images (2009)
17. Li, Z., Hoiem, D.: Learning without forgetting. TPAMI **40**(12), 2935–2947 (2017)
18. Lopez-Paz, D., Ranzato, M.: Gradient episodic memory for continual learning. In: NeurIPS (2017)
19. Rebuffi, S.A., Kolesnikov, A., Sperl, G., Lampert, C.H.: iCaRL: incremental classifier and representation learning. In: CVPR (2017)
20. Rusu, A.A., et al.: Progressive neural networks. In: NeurIPS (2016)
21. Sandler, M., Howard, A., Zhu, M., Zhmoginov, A., Chen, L.C.: Mobilenetv 2: inverted residuals and linear bottlenecks. In: CVPR (2018)
22. Shin, H., Lee, J.K., Kim, J., Kim, J.: Continual learning with deep generative replay. In: NeurIPS (2017)
23. Tan, M., Le, Q.: Efficientnet: rethinking model scaling for convolutional neural networks. In: ICML (2019)
24. Tschandl, P., Rosendahl, C., Kittler, H.: The ham10000 dataset, a large collection of multi-source dermatoscopic images of common pigmented skin lesions. Sci. Data **5**(1), 1–9 (2018)
25. Veeling, B.S., Linmans, J., Winkens, J., Cohen, T., Welling, M.: Rotation equivariant CNNs for digital pathology. In: Frangi, A.F., Schnabel, J.A., Davatzikos, C., Alberola-López, C., Fichtinger, G. (eds.) MICCAI 2018. LNCS, vol. 11071, pp. 210–218. Springer, Cham (2018). https://doi.org/10.1007/978-3-030-00934-2_24
26. Verma, V.K., Liang, K.J., Mehta, N., Rai, P., Carin, L.: Efficient feature transformations for discriminative and generative continual learning. In: CVPR (2021)

27. Wang, Z., et al.: Dualprompt: complementary prompting for rehearsal-free continual learning. In: Avidan, S., Brostow, G., Cissé, M., Farinella, G.M., Hassner, T. (eds.) ECCV 2022. LNCS, vol. 13686, pp. 631–648. Springer, Cham (2022). https://doi.org/10.1007/978-3-031-19809-0_36
28. Wang, Z., et al.: Learning to prompt for continual learning. In: CVPR (2022)
29. Wei, J., et al.: A petri dish for histopathology image analysis. In: Artificial Intelligence in Medicine (2021)
30. Yan, S., Xie, J., He, X.: Der: Dynamically expandable representation for class incremental learning. In: CVPR (2021)
31. Yang, J., Shi, R., Ni, B.: Medmnist classification decathlon: a lightweight automl benchmark for medical image analysis. In: ISBI (2021)
32. Yang, Y., Cui, Z., Xu, J., Zhong, C., Wang, R., Zheng, W.-S.: Continual learning with Bayesian model based on a fixed pre-trained feature extractor. In: de Bruijne, M., et al. (eds.) MICCAI 2021. LNCS, vol. 12905, pp. 397–406. Springer, Cham (2021). https://doi.org/10.1007/978-3-030-87240-3_38
33. Zhang, R., et al.: Tip-adapter: training-free adaption of clip for few-shot classification. In: Avidan, S., Brostow, G., Cissé, M., Farinella, G.M., Hassner, T. (eds.) ECCV 2022. LNCS, vol. 13695, pp. 493–510. Springer, Cham (2022). https://doi.org/10.1007/978-3-031-19833-5_29
34. Zhao, B., Xiao, X., Gan, G., Zhang, B., Xia, S.T.: Maintaining discrimination and fairness in class incremental learning. In: CVPR (2020)
35. Zheng, X., et al.: A deep learning model and human-machine fusion for prediction of EBV-associated gastric cancer from histopathology. *Nature Commun.* **13**(1), 2790 (2022)
36. Zhou, W., et al.: Ensembled deep learning model outperforms human experts in diagnosing biliary atresia from sonographic gallbladder images. *Nat. Commun.* **12**(1), 1259 (2021)