Low-Rank Mixture-of-Experts for Continual Medical Image Segmentation

Qian Chen[1,2,3,4], Lei Zhu[1,2,3,4], Hangzhou He[1,2,3,4], Xinliang Zhang[1,2,3,4], Shuang Zeng[1,2,3,4], Qiushi Ren[1,2,3,4], and Yanye Lu[1,2,3,4]

[1] Department of Biomedical Engineering, Peking University, Beijing, China
[2] Institute of Medical Technology, Peking University, Beijing, China
[3] Institute of Biomedical Engineering, Peking University Shenzhen Graduate School, Shenzhen, China
[4] National Biomedical Imaging Center, Peking University, Beijing, China

Abstract. The primary goal of continual learning (CL) task in medical image segmentation field is to solve the "catastrophic forgetting" problem, where the model totally forgets previously learned features when it is extended to new categories (class-level) or tasks (task-level). Due to the privacy protection, the historical data labels are inaccessible. Prevalent continual learning methods primarily focus on generating pseudo-labels for old datasets to force the model to memorize the learned features. However, the incorrect pseudo-labels may corrupt the learned feature and lead to a new problem that the better the model is trained on the old task, the poorer the model performs on the new tasks. To avoid this problem, we propose a network by introducing the data-specific Mixture of Experts (MoE) structure to handle the new tasks or categories, ensuring that the network parameters of previous tasks are unaffected or only minimally impacted. To further overcome the tremendous memory costs caused by introducing additional structures, we propose a Low-Rank strategy which significantly reduces memory cost. We validate our method on both class-level and task-level continual learning challenges. Extensive experiments on multiple datasets show our model outperforms all other methods.

Keywords: Continual Learning · Mixture of Experts · Multi-Organ Segmentation · Medical image segmentation.

1 Introduction

Medical image segmentation (MIS) has been extensively studied and is crucial for quantitative disease analysis [10], computer-aided diagnosis [24], and cancer radiotherapy planning [12]. The advancement of deep learning technologies has boosted the field. However, the current setting of deep learning scenarios has limitations compared to the practical deployment in clinical medical environments. Present mainstream models can only segment single-modal datasets of lesions or organs, yet actual clinical practitioners require dynamic extending to identify targets across multi-modal datasets. Additionally, in the scenario of multi-organ segmentation, there is also an expectation that segmentation models can dynamically segment new organs without accessing old datasets. This anticipated clinical scenario can be understood as a continual learning (CL) problem in MIS. Models are easily prone to forgetting old data while learning new knowledge. This problem, known as "catastrophic forgetting" [15] in CL, is an urgent issue that needs to be addressed. It is worth noting that in CL, data arrives at the model in sequence, and when the model learns new knowledge, it has no access to the old dataset.

Current research also explores the issue of CL in MIS, where the most critical problem is how to prevent catastrophic forgetting. Common approaches are regularization-based methods, which primarily use pseudo-labeling for old classes. These methods [28,19] train on datasets from both old and new classes together to achieve test results on both. The issue with this approach is that the accuracy of pseudo-labels is not high, and since the model is mainly trained on new datasets, the performance on both old and new classes is not satisfactory. Architecture-based methods are dedicated to dynamically adding dataset-specific partial networks [6,9] or expanding the network by fixing the parameters of old tasks and adding new parameters for new tasks [25,16]. Although the issue of catastrophic forgetting is addressed, it leads to tremendous memory costs for model parameters.

We summarize the ongoing issues in CL for MIS. (1) Can we design networks that avoid the problem of catastrophic forgetting? (2) There is a clear trade-off in the CL capability when handling new and old tasks, the better the model performs on old datasets, the worse it tends to perform on new datasets [11,17]. Can the model maintain the ability to learn continuously over multiple learning steps? (3) Whether it's applying pseudo-labels or adding extra parameters, both introduce significant costs. Can the model have the advantages of being both lightweight and cost-effective?

To address the aforementioned problems, we propose a novel network structure for CL of MIS, named the Low-rank Mixture of Experts (MoE) architecture. This is a Transformer-based network structure, where an MoE layer consists of E feed-forward networks $\mathrm{FFN}_1 ... \mathrm{FFN}_E$. First, during each training session, we only update the dataset-specific low-rank expert and fix all other parts of the network. As new datasets arrive, we append another expert layer and proceed to train solely that particular expert. In doing so, the knowledge acquired from previous tasks is not lost since the parameters within their respective experts are already fixed. This approach effectively resolves the issue of catastrophic forgetting since the knowledge learned on old tasks and new tasks is preserved by different experts. Therefore, the model can maintain its ability for CL across different steps, achieving high accuracy in training on both old and new tasks. It also presents a significant drawback despite the clear advantages of this multi-expert model: incorporating multiple dataset-specific experts into the network substantially increases computational and parameter costs. To mitigate this, we adopt a low-rank strategy. The weights within the MoE FFN are decomposed into a dimension-reducing matrix B and a dimension-increasing matrix

A. Throughout training, we maintain the original pretrained weights W_0 and update only the parameters within A and B.

To validate the effectiveness of the Low-Rank MoE structure, we conduct two series of experiments in both task-level CL settings (cross-modal multi-dataset medical image segmentation task) and class-level CL settings (multi-organ segmentation task). Our proposed method are evaluated on five datasets: ACDC [1], ISIC [3], COVID-19 Segmentation dataset [5], BTCV [13], and LiTS [2]. In the task-level CL setting, we find that the low-rank experts from earlier tasks can not only assist subsequent tasks but also resolve the catastrophic forgetting problem. In the class-level CL setting, we introduce a language-guided gating function that successfully achieves CL of multi-organ segmentation across new and old classes. Moreover, we compare our method with several popular regularization-based approaches [17,22,4,11]. The thorough results demonstrate the effectiveness of our proposed Low-Rank MoE method in achieving CL for cross-modal multi-dataset segmentation task and multi-organ segmentation task.

2 Methodology

Let $\mathcal{X}$ be the input image space and $\mathcal{Y}$ be the label space. In the CL for MIS setting, the training procedure is arranged into multiple steps, and each learning step t will involve novel class $\mathcal{C}^t$, constructing a new label set $\mathcal{Y}^t = \mathcal{Y}^{t-1} \cup \mathcal{C}^t$. When training on the $t-$th dataset D_t, the previous datasets of $\{D_1, ..., D_{t-1}\}$ are not seen. The model is required to predict the accumulated labels for all seen datasets $\{D_1, ..., D_t\}$:

$$y_i = argmax_{c \in \mathcal{C}^t} P(y_i = c|\mathcal{X}), \mathcal{C}^t = \cup_{r \leq t} \mathcal{C}^r \tag{1}$$

where P is the probability function that the model learns and y_i is the output mask.

2.1 Low-Rank Mixture of Experts

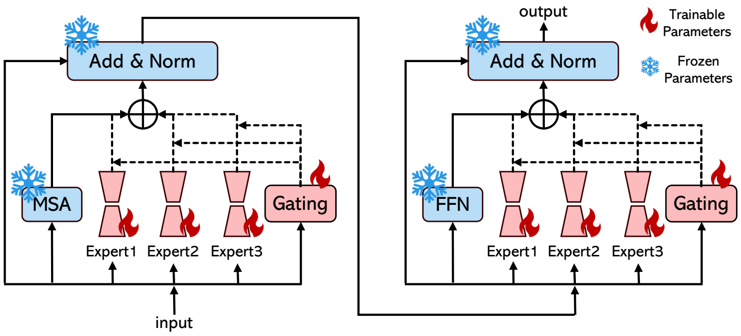

Fig. 1. An overview of Low-Rank MoE architecture. MSA means multi-head self-attention module.

Low-Rank Mixture of Experts Layers To facilitate CL in MIS, we employ a Mixture of Experts (MoE) design to achieve this. Figure 1 illustrates the overall framework of the proposed model architecture. We follow the Mixture-of-Experts Transformer models proposed by [14]. A MoE layer for Transformer consists of E feed-forward networks (FFN) $\mathrm{FFN}_1 ... \mathrm{FFN}_E$. $\mathrm{FFN}_e(x_s) = Wo_e \cdot \mathrm{GeLU}(Wi_e \cdot x_s)$, $y_s = \sum_{e=1}^{E} G_{s,e} \cdot \mathrm{FFN_e}(x_s)$, where x_s is the input token at position s to the MoE layer and each $\mathrm{FFN_e}$ is a two layer neural network using a GeLU activation function. Wi_e and Wo_e are the input and output projection weights of the e-th expert. Vector $G_{s,E}$ is computed by a gating network. LoRA(Low-Rank Adapter) has been demonstrated to be an effective and efficient way to adapt pre-trained models to specific tasks[8]. Formally, for a pre-trained weight matrix $W_0 \in \mathbb{R}^{d \times k}$, LoRA updates the W with a low-rank decomposition: $W_0 + \mathit{\Delta} W = W_0 + BA$, $B \in \mathbb{R}^{d \times r}$, $A \in \mathbb{R}^{r \times k}$, and the rank $r \ll min(d, k)$. During training, W_0 is frozen and does not receive gradient updates, while A and B contain trainable parameters. We use a random Gaussian initialization for A and zero for B, so $\mathit{\Delta} W = BA$ is zero at the beginning of training and doesn't affect the generalization ability of the pre-trained weights W_0. We use Low-Rank Adapters as the experts for different tasks and adapt them for the FFN layers and the attention layers. This strategy facilitates adaptive model updates for new tasks without losing valuable information learned in previous tasks. Specifically, the forward process of the LoRA FFN MoE layer can be formulated as:

$$\mathrm{FFN_e}(x_s) = (Wo + \mathit{\Delta} W_e^o) \cdot \mathrm{GeLU}((W^i + \mathit{\Delta} W_e^i) \cdot x_s) \tag{2}$$

where $\mathit{\Delta} W_e^i$ and $\mathit{\Delta} W_e^o$ are the input and output low-rank projection weights of the e-th expert.

Low-Rank MoE Attention For the modified low-rank attention module, we replace four regular linear layers with low-rank linear layers. Formally, the matrix operation of the LoRA multi-head attention can be expressed as:

$$MultiHead(Q, K, V) = \mathrm{Concat(head_1, ..., head_h)(W^O + B^O A^O)} \tag{3}$$

where $W^O \in \mathbb{R}^{d_{model} \times d_{model}}$ and

$$\mathrm{head_i = Attention[Q(W_i^Q + B_i^Q A_i^Q), K(W_i^K + B_i^K A_i^K), V(W_i^V + B_i^V A_i^V)]} \tag{4}$$

where the projections are parameter matrices $W_i^Q \in \mathbb{R}^{d_{model} \times d_k}$, $B_i^Q \in (R)^{d_{model} \times r}$, $A_i^Q \in (R)^{r \times d_k}$; $W_i^K \in \mathbb{R}^{d_{model} \times d_k}$, $B_i^K \in (R)^{d_{model} \times r}$,$A_i^K \in (R)^{r \times d_k}$; $W_i^V \in \mathbb{R}^{d_{model} \times d_k}$, $B_i^V \in (R)^{d_{model} \times r}$, $A_i^V \in (R)^{r \times d_k}$. In this work, we employ $h = 8$ parallel attention heads[26], $d_k = d_v = d_{model}/h = 64$, r=8[8].

2.2 Continual Learning Gating Strategy

Task-level Gating: For task-level model, We utilize the popular SETR [29] as the backbone. As illustrated in 2, given a new task T_k associated with its

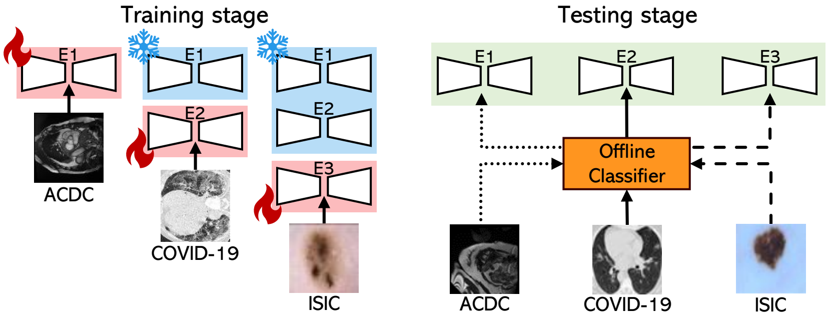

Fig. 2. Illustration of the proposed task-level gating pipeline.

data D_k, we directly train an expert E_k on D_k. For the segmentation task on each medical dataset, we learn a specialized expert for it. During step 1, dataset D_1 is processed by the first FFN, FFN_{E_1}. The matrix operation of the linear layer in the FFN_{E_1} can be expressed as $h = W_0x + B_{E_1}A_{E_1}x$. In step 2, dataset D_2 is processed by FFN_{E_2}, which is superimposed on FFN_{E_1}. The matrix operation of the linear layer in the FFN_{E_2} can be expressed as $h = W_0x + B_{E_1}A_{E_1}x + B_{E_2}A_{E_2}x$, the parameters in the FFN_{E_1} are fixed, i.e. $B_{E_1}A_{E_1}$ are frozen. We can represent the matrix operation of the linear layer in the task t in the FFN_{E_t} as:

$$h = W_0x + \sum_{t=1}^{T} B_{E_t}A_{E_t}x = \underbrace{(W_0 + \sum_{t=1}^{T-1} B_{E_t}A_{E_t})}_{frozen} x + B_{E_T}A_{E_T}x \tag{5}$$

Thus, according to the equation 2 and 5, the forward process of the LoRA MoE layer in the T_{th} task FFN layer can be represented as:

$$FFN_{E_T} = \underbrace{(W_0 + \sum_{t=1}^{T-1} B^o_{E_t}A^o_{E_t}}_{frozen} + B^o_{E_T}A^o_{E_T}) \cdot GeLU(\underbrace{(W^i + \sum_{t=1}^{T-1} B^i_{E_t}A^i_{E_t}}_{frozen} + B^i_{E_T}A^i_{E_T}) \cdot x) \tag{6}$$

where i and o represent the input and output linear layer since there are two linear layers in one FFN layer. We freeze the weights corresponding to the old tasks and only update the weights in the new task. When testing T_k, we first input the test data D_k into a scalable matching-based offline classifier (with high classification accuracy 99.7%) to determine the task number, and then use the corresponding expert model E_K to test D_k according to the task number.

Class-level Gating: Specifically, we first expand the standard image-based Swin Transformer [20] into 3D CT scans. We then incorporated our MoE structure onto this enhanced backbone. In the training phase, as illustrated in 3, we first describe the dataset T_1 trained in the first step (e.g.,"BTCV Segmentation dataset contains medical imaging data for abdominal organs with the following label definitions: 0.background; 1.spleen...13.left adrenal gland."). We use the text encoder from the CLIP model [23] to generate text embedding related to

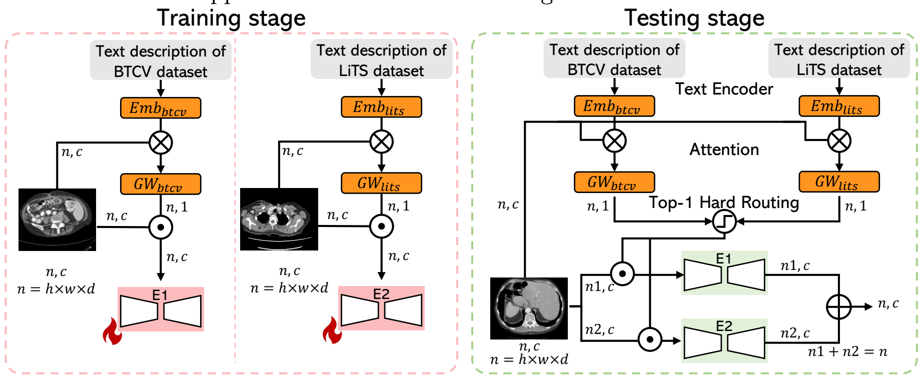

Fig. 3. Illustration of the proposed class-level gating pipeline. *Emb* means Embeddings and *GW* means Gating Weights. E_1 and E_2 indicate expert 1 and expert 2.

T_1. The text embedding is then matrix-multiplied with the input passed through a linear layer and a sigmoid layer to obtain a layer of language-guided gating weights (GW). GW is then multiplied by the input to obtain the parameters of E_1. This is the gating function of one layer of an expert. In the second step of continual learning, the parameters of the first FFN E_1 are fixed and loaded, and the second expert FFN E_2 is trained in the same gating manner.

In the testing phase, we calculate the weights for the input concerning the text embeddings of T1 and T2, respectively. For each token in the input, we only select the expert with the higher weight to perform the calculation, which is a form of top-1 hard routing. This allows each token to choose the more appropriate expert for calculation without introducing additional computational cost, thereby enabling the model to handle different categories in T_1 and T_1 simultaneously.

3 Experimental Setup & Result

3.1 Dataset

Task-level Continual Learning: We train and then do continual learning on the following three datasets with different orders. ACDC [1] is a publicly-available dataset for the automated cardiac diagnosis challenge. Following [18], the dataset is split into 70 training samples, 10 validation samples and 20 testing samples. ISIC [3] is a publicly-availavle dataset for skin lesion segmentation, which contains 2594 images with lesions and corresponding skin lesion labels marked by experts. Among them, 162 pictures are used for testing, 162 pictures are used for verification, and 2270 pictures are used for training. COVID-19 CT Segmentation dataset [5] consists of 100 axial CT images.
Class-level Continual Learning: We first train on the BTCV [13] and then do CL on the LiTS [2]. BTCV contains 30 labeled abdominal CT scans, which we divided according to MONAI[5], with 24 for training and 6 for testing. LiTS

[5] MONAI: `https://monai.io/`

dataset consists of 131 contrast-enhanced abdominal CT images for liver and liver tumor segmentation, originating from 7 different medical centers, which we divided into train/val/test sets according to [28].

3.2 Implementation Details

Task-level Continual Learning : We adapt the SETR [29] segmentation framework for evaluating all methods, ensuring a fair comparison. We employ an AdamW optimizer [21] for 100 epochs using a cosine learning rate scheduler with 10-epoch linear warm-up. A batch size of 16, an initial learning rate of 0.001 and a weight decay of 1e-6 are used. We use a hidden dimension of 8 for all low-rank layers. For the offline task classifier, we randomly select 8 images from each dataset as its support set and utilize the image encoder from CLIP [23] to extract their features for the task matching process. We use the most commonly used Dice score (DSC) as our evaluation metric.

Class-level Continual Learning : We use an improved Swin-UNETR [7] implemented in MONAI as the segmentation framework. We employ an AdamW optimizer [21] with 500 epochs for BTCV and 200 epochs for LiTS. A cosine learning rate scheduler with 10-epoch linear warm-up, a batch size of 3, an initial learning rate of 0.001 and a weight decay of 1e-5 are used. We use a hidden dimension of 8 for all low-rank layers. We use the most commonly used Dice score (DSC) as our evaluation metric. Following [28], we report the average DSC across 13 classes from the BTCV dataset in step-1 learning phase and the DSC for the newly introduced liver tumor class from the LiTS dataset in the step-2 learning phase. Comprehensive implementation details are available in the Appendix.

3.3 Results

Task-level Continual Learning Results. We first analyze the quantitative results of models on the task-level CL MIS task, as shown in Tables 1. For more comprehensive experimental outcomes, please refer to the Appendix. The Single-task model in Tables 1 indicates results achieved by training and testing solely on the ACDC, ISIC, and COVID-19 CT datasets. The other eight rows in Tables 1 represent eight distinct scenarios, for example, ACDC → ISIC means training initially on ACDC, followed by ISIC. The following observations can be made: firstly, the Low-Rank MoE model's performance on both previous and current tasks is on par with or even exceeds the baseline (Single-task model), signifying that the MoE structure has effectively addressed the issue of catastrophic forgetting in CL. Secondly, compared to the baseline and models w/o low-rank architectures, those using the Low-Rank MoE structure are markedly more lightweight. Thirdly, when compared to the baseline, there is a notable improvement in the results on the current task. This improvement is attributed to the utilization of LoRA weights from previous tasks as initial values for the current LoRA weight, allowing the transfer of beneficial information from previous tasks to the current one.

Class-level Continual Learning Results. We now analyze the quantitative results of models on the class-level CL multi-organ segmentation task, as shown in Table2. Three single-task models refer to the results trained only on BTCV or LiTS, indicating that the results based on Swin-Tiny serve as an upper bound for all corresponding results. Compared to the other four popular CL methods, our results consistently achieve optimal performance in step 2. Qualitative result can be seen in the Appendix.

Table 1. Benchmark task-level continual learning methods. ∇, $\triangle$ and $\square$ represents ACDC, ISIC and COVID-19 CT dataset respectively. Red indicates the performance of the data trained in the final step. #param indicates the number of trainable parameters.

Model	#param	low-rank	MoE	∇	$\triangle$	$\square$
SOTAs	-	-	-	91.46 [27]	89.03 [27]	68.20 [5]
Single-task model	88.1M	×	×	92.02	90.63	72.71
$\nabla \rightarrow \triangle$	88.1M	×	×	50.19	90.69	-
$\nabla \rightarrow \square$	88.1M	×	×	32.41	-	72.44
$\nabla \rightarrow \triangle \rightarrow \square$	88.1M	×	×	3.03	63.60	72.30
$\nabla \rightarrow \square \rightarrow \triangle$	88.1M	×	×	39.08	90.46	50.20
Single-task model	3.4M	✓	×	92.08	90.61	73.27
$\nabla \rightarrow \triangle$	3.4M	✓	✓	92.08	90.77	-
$\nabla \rightarrow \square$	3.4M	✓	✓	92.08	-	73.68
$\nabla \rightarrow \triangle \rightarrow \square$	3.4M	✓	✓	92.08	90.77	74.17
$\nabla \rightarrow \square \rightarrow \triangle$	3.4M	✓	✓	92.08	90.75	73.68

Table 2. Benchmark class-level continual learning methods. † indicates using our improved Swin-UNETR framework. Performance for both the validation and testing sets of the LITS dataset are reported (val/test).

		Step1		Step2	
Method	#param	BTCV	LiTS	BTCV	LiTS
Single-task model	62.1M	81.9	-	-	-
Single-task model†	30.7M	82.7	-	-	-
Single-task model†	30.7M	-	56.9/49.5	-	-
LwF [17]†	37.5M	82.7	-	76.2	49.9/43.1
ILT [22]†	37.5M	82.7	-	77.8	39.0/32.8
PLOP [4]†	37.5M	82.7	-	78.0	41.2/36.6
CLAMTS [28]†	37.5M	81.9	-	78.5	50.9/45.4
Low-Rank MoE†	2.3M	82.6	-	80.6	53.5/46.7

4 Conclusion

In this article, we propose a Low-Rank Mixture of Experts (MoE) network to address continual learning (CL) in medical image segmentation. Whenever new data arrives, the MoE structure fixes most of the parameters, only updating the data-specific expert FFN. As a result, the old data parameters are frozen within the data-specific expert, while the parameters in the new expert are activated by new data, and the parameters of the two parts of the network do not affect each other, thus resolving the catastrophic forgetting problem. To address the increased computational cost and parameter overhead in the MoE structure, we propose a low-rank decoupling parameter strategy. The experimental results on five public datasets demonstrate the high performance of the proposed method.

Acknowledgments. This work was supported in part by the Natural Science Foundation of China under Grant 82371112, 623B2001, 62394311, in part by Beijing Natural Science Foundation under Grant Z210008, in part by Shenzhen Science and Technology Program, China under Grants KQTD20180412181221912, and in part by High-grade, Precision and Advanced University Discipline Construction Project of Beijing (BMU2024GJJXK004).

Disclosure of Interests. The authors have no competing interests to declare that are relevant to the content of this article.

References

1. Bernard, O., Lalande, A., Zotti, C., Cervenansky, F., Yang, X., Heng, P.A., Cetin, I., Lekadir, K., Camara, O., Ballester, M.A.G., et al.: Deep learning techniques for automatic mri cardiac multi-structures segmentation and diagnosis: is the problem solved? IEEE transactions on medical imaging **37**(11), 2514–2525 (2018)
2. Bilic, P., Christ, P., Li, H.B., Vorontsov, E., Ben-Cohen, A., Kaissis, G., Szeskin, A., Jacobs, C., Mamani, G.E.H., Chartrand, G., et al.: The liver tumor segmentation benchmark (lits). Medical Image Analysis **84**, 102680 (2023)
3. Codella, N., Rotemberg, V., Tschandl, P., Celebi, M.E., Dusza, S., Gutman, D., Helba, B., Kalloo, A., Liopyris, K., Marchetti, M., et al.: Skin lesion analysis toward melanoma detection 2018: A challenge hosted by the international skin imaging collaboration (isic). arXiv preprint arXiv:1902.03368 (2019)
4. Douillard, A., Chen, Y., Dapogny, A., Cord, M.: Plop: Learning without forgetting for continual semantic segmentation. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 4040–4050 (2021)
5. Fan, D.P., Zhou, T., Ji, G.P., Zhou, Y., Chen, G., Fu, H., Shen, J., Shao, L.: Inf-net: Automatic covid-19 lung infection segmentation from ct images. IEEE transactions on medical imaging **39**(8), 2626–2637 (2020)
6. Golkar, S., Kagan, M., Cho, K.: Continual learning via neural pruning. arXiv preprint arXiv:1903.04476 (2019)
7. Hatamizadeh, A., Nath, V., Tang, Y., Yang, D., Roth, H.R., Xu, D.: Swin unetr: Swin transformers for semantic segmentation of brain tumors in mri images. In: International MICCAI Brainlesion Workshop. pp. 272–284. Springer (2021)
8. Hu, E.J., Shen, Y., Wallis, P., Allen-Zhu, Z., Li, Y., Wang, S., Wang, L., Chen, W.: Lora: Low-rank adaptation of large language models. arXiv preprint arXiv:2106.09685 (2021)
9. Hung, C.Y., Tu, C.H., Wu, C.E., Chen, C.H., Chan, Y.M., Chen, C.S.: Compacting, picking and growing for unforgetting continual learning. Advances in Neural Information Processing Systems **32** (2019)
10. Iyer, K.S., Newell Jr, J.D., Jin, D., Fuld, M.K., Saha, P.K., Hansdottir, S., Hoffman, E.A.: Quantitative dual-energy computed tomography supports a vascular etiology of smoking-induced inflammatory lung disease. American journal of respiratory and critical care medicine **193**(6), 652–661 (2016)
11. Ji, Z., Guo, D., Wang, P., Yan, K., Lu, L., Xu, M., Wang, Q., Ge, J., Gao, M., Ye, X., et al.: Continual segment: Towards a single, unified and non-forgetting continual segmentation model of 143 whole-body organs in ct scans. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 21140–21151 (2023)

12. Jin, D., Guo, D., Ho, T.Y., Harrison, A.P., Xiao, J., Tseng, C.K., Lu, L.: Deeptarget: Gross tumor and clinical target volume segmentation in esophageal cancer radiotherapy. Medical Image Analysis **68**, 101909 (2021)
13. Landman, B., Xu, Z., Igelsias, J., Styner, M., Langerak, T., Klein, A.: Miccai multi-atlas labeling beyond the cranial vault–workshop and challenge. In: Proc. MICCAI Multi-Atlas Labeling Beyond Cranial Vault—Workshop Challenge. vol. 5, p. 12 (2015)
14. Lepikhin, D., Lee, H., Xu, Y., Chen, D., Firat, O., Huang, Y., Krikun, M., Shazeer, N., Chen, Z.: Gshard: Scaling giant models with conditional computation and automatic sharding. arXiv preprint arXiv:2006.16668 (2020)
15. Lewandowsky, S., Li, S.C.: Catastrophic interference in neural networks: Causes, solutions, and data. In: Interference and inhibition in cognition, pp. 329–361. Elsevier (1995)
16. Li, X., Zhou, Y., Wu, T., Socher, R., Xiong, C.: Learn to grow: A continual structure learning framework for overcoming catastrophic forgetting. In: International Conference on Machine Learning. pp. 3925–3934. PMLR (2019)
17. Li, Z., Hoiem, D.: Learning without forgetting. IEEE transactions on pattern analysis and machine intelligence **40**(12), 2935–2947 (2017)
18. Lin, X., Yan, Z., Deng, X., Zheng, C., Yu, L.: Convformer: Plug-and-play cnn-style transformers for improving medical image segmentation. In: International Conference on Medical Image Computing and Computer-Assisted Intervention. pp. 642–651. Springer (2023)
19. Liu, P., Wang, X., Fan, M., Pan, H., Yin, M., Zhu, X., Du, D., Zhao, X., Xiao, L., Ding, L., et al.: Learning incrementally to segment multiple organs in a ct image. In: International Conference on Medical Image Computing and Computer-Assisted Intervention. pp. 714–724. Springer (2022)
20. Liu, Z., Lin, Y., Cao, Y., Hu, H., Wei, Y., Zhang, Z., Lin, S., Guo, B.: Swin transformer: Hierarchical vision transformer using shifted windows. In: Proceedings of the IEEE/CVF international conference on computer vision. pp. 10012–10022 (2021)
21. Loshchilov, I., Hutter, F.: Decoupled weight decay regularization. In: International Conference on Learning Representations (2019), `https://openreview.net/forum?id=Bkg6RiCqY7`
22. Michieli, U., Zanuttigh, P.: Incremental learning techniques for semantic segmentation. In: Proceedings of the IEEE/CVF international conference on computer vision workshops. pp. 0–0 (2019)
23. Radford, A., Kim, J.W., Hallacy, C., Ramesh, A., Goh, G., Agarwal, S., Sastry, G., Askell, A., Mishkin, P., Clark, J., et al.: Learning transferable visual models from natural language supervision. In: International conference on machine learning. pp. 8748–8763. PMLR (2021)
24. Roth, H.R., Lu, L., Liu, J., Yao, J., Seff, A., Cherry, K., Kim, L., Summers, R.M.: Improving computer-aided detection using convolutional neural networks and random view aggregation. IEEE transactions on medical imaging **35**(5), 1170–1181 (2015)
25. Rusu, A.A., Rabinowitz, N.C., Desjardins, G., Soyer, H., Kirkpatrick, J., Kavukcuoglu, K., Pascanu, R., Hadsell, R.: Progressive neural networks. arXiv preprint arXiv:1606.04671 (2016)
26. Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A.N., Kaiser, Ł., Polosukhin, I.: Attention is all you need. Advances in neural information processing systems **30** (2017)

27. Wu, H., Chen, S., Chen, G., Wang, W., Lei, B., Wen, Z.: Fat-net: Feature adaptive transformers for automated skin lesion segmentation. Medical image analysis **76**, 102327 (2022)
28. Zhang, Y., Li, X., Chen, H., Yuille, A.L., Liu, Y., Zhou, Z.: Continual learning for abdominal multi-organ and tumor segmentation. In: International conference on medical image computing and computer-assisted intervention. pp. 35–45. Springer (2023)
29. Zheng, S., Lu, J., Zhao, H., Zhu, X., Luo, Z., Wang, Y., Fu, Y., Feng, J., Xiang, T., Torr, P.H., et al.: Rethinking semantic segmentation from a sequence-to-sequence perspective with transformers. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 6881–6890 (2021)