Continual Segment: Towards a Single, Unified and Non-forgetting Continual Segmentation Model of 143 Whole-body Organs in CT Scans

Zhanghexuan Ji[1,2†] Dazhou Guo[1†] Puyang Wang[1] Ke Yan[1,3] Le Lu[1] Minfeng Xu[1,3]
Qifeng Wang[4] Jia Ge[5] Mingchen Gao[2] Xianghua Ye[5*] Dakai Jin[1*]

[1]DAMO Academy, Alibaba Group [2] University at Buffalo [3]Hupan Lab, 310023, Hangzhou, China
[4]Sichuan Cancer Hospital [5]The First Affiliated Hospital of Zhejiang University

Abstract

Deep learning empowers the mainstream medical image segmentation methods. Nevertheless, current deep segmentation approaches are not capable of efficiently and effectively adapting and updating the trained models when new segmentation classes are incrementally added. In the real clinical environment, it can be preferred that segmentation models could be dynamically extended to segment new organs/tumors without the (re-)access to previous training datasets due to obstacles of patient privacy and data storage. This process can be viewed as a continual semantic segmentation (CSS) problem, being understudied for multi-organ segmentation. In this work, we propose a new architectural CSS learning framework to learn a single deep segmentation model for segmenting a total of 143 whole-body organs. Using the encoder/decoder network structure, we demonstrate that a continually trained then frozen encoder coupled with incrementally-added decoders can extract sufficiently representative image features for new classes to be subsequently and validly segmented, while avoiding the catastrophic forgetting in CSS. To maintain a single network model complexity, each decoder is progressively pruned using neural architecture search and teacher-student based knowledge distillation. Finally, we propose a body-part and anomaly-aware output merging module to combine organ predictions originating from different decoders and incorporate both healthy and pathological organs appearing in different datasets. Trained and validated on 3D CT scans of 2500+ patients from four datasets, our single network can segment a total of 143 whole-body organs with very high accuracy, closely reaching the upper bound performance level by training four separate segmentation models (i.e., one model per dataset/task).

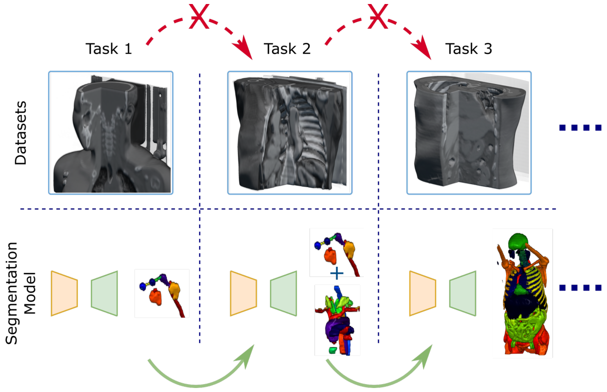

Figure 1. Illustration of the continual multi-organ segmentation. At each continual learning step, only the previously trained model is available (green arrow). Previous datasets are not accessible. We allow organs from different datasets to have overlaps, and these datasets may also contain diseased organs (with tumors).

1. Introduction

Multi-organ segmentation has been extensively studied in medical imaging because of its core importance for many downstream tasks, such as quantitative disease analysis [27, 17], computer-aided diagnosis [51, 7], and cancer radiotherapy planning [31, 67, 29]. With the emergence of many dedicated labeled organ datasets [2] and the fast developments in deep learning segmentation techniques [26], deep segmentation networks trained on specific datasets achieve comparable performance with human observers [59, 67, 56]. However, this setup can have serious limitations in practical deployment for clinical applications. These trained models are pre-trained to segment a fixed number of organs, while in real clinical practice, it is desirable that segmentation models can be dynamically extended to enable segmenting new organs without the (re-)access to previous training datasets or without training from scratch. In this way, patient privacy and data storage issues can be solved, and model development and deployment can be much more efficient. This clinically preferred

† ZJ and DG contribute equally. ∗ For correspondence, please contact XY (hye1982@zju.edu.cn) and DJ (dakai.jin@alibaba-inc.com).

process can be viewed as continual semantic segmentation (CSS), which is emerging very recently in the natural image domain [42, 6, 13, 71, 43, 65, 5] but has been only scarcely studied for medical imaging [45, 37]. Notably, if all labeled datasets are simultaneously accessible, it simplifies to a federated learning [50, 55] or partial label learning [16, 57] problem. However, labeled datasets are always sequentially built over time by annotating different organs of interest according to various clinical tasks.

Multi-organ CSS faces several major challenges. First, since old datasets are not accessible when training on the new dataset, deep networks may easily forget the previously learned knowledge if no additional constraints are added, which is the most prominent issue (known as catastrophic forgetting [60, 32]) in continual learning. Second, in contrast to natural image datasets that are often completely labeled [15, 73], fully annotated medical image datasets are rare, especially for comprehensive multi-organ datasets. For example, concerning both necessity and cost, labeling 143 organs for all datasets is simply infeasible or impossible. These partially labeled datasets bring up the label conflict issue (semantic shift of the background class [6]), meaning a labeled organ in dataset-1 may become unlabeled background in dataset-2. Third, domain incremental learning is common in multi-organ CSS, since different datasets may contain overlapped yet "style-different" organs. Appropriately tackling these domain gaps is non-trivial. E.g., dataset-1 is made up of healthy subjects with normal esophagus annotated, while dataset-2 is a dedicated esophageal cancer dataset where esophagus with tumor is labeled.

There are several recent CSS work in computer vision [42, 6, 13, 71, 43]. MiB loss is often applied to handle the background-label conflicting issue [6, 13]. Regularization-based methods are mostly adopted to reduce the forgetting of old knowledge while learning new classes. However, since network parameters are updated on the training of new classes, it is extremely difficult to achieve high performance on both old and new classes. There are few previous works of CSS in medical imaging [45, 37]. Ozdemir et al. employed only 9 patients with 2 labels to develop a regularization-based CSS preliminary model [45]. The most recent work [37] used MiB loss and prototype matching to continually segment a small number of 5 abdominal organs focusing only on the abdomen CT. When involving a large number of organs (e.g., ≥ 100 classes) affiliated with a variety of body parts, such as in whole-body CT scans for practical considerations, this strategy becomes non-scalable and suffers severe performance degradation (as demonstrated in our experiments later).

A most recent continual classification work [64] has empirically shown that a base classification model trained with a sufficiently large number of classes (e.g., 800) is capable of extracting representative features even for new classes. Hence, freezing most part of its parameters and incrementally fine-tuning the newly added last convolutional block for each new task leads to an almost non-forgetting continual classification model, whose performance is close to the joint learning upper bound for both old and new classes.

Motivated by the observation in continual classification, in this work, we propose a novel architecture-based continual multi-organ segmentation framework. On the basis of the common encoder + decoder architecture of segmentation networks, we demonstrate that its encoder is capable of extracting representative deep features (non-specific to organ or body part) for the new data. Hence, we can freeze the encoder and incrementally add a separate decoder for each new learning task. Under this scheme, when adding a new task, organs learned in previous tasks will never be forgotten because the encoder is frozen, and previous decoders are independent of the new task. In addition, the new decoder is trained separately to segment a fixed number of foreground organs using only the new dataset. Hence, it avoids the background-label conflict with previous datasets during training. Yet, this scheme can lead to a swelling model as tasks expand. To make it scalable, a progressive trimming method using neural architectural search (NAS) and teacher-student-based knowledge distillation (KD) is exploited to maintain the *overall model complexity* and *inference time* comparable to the original single network. Finally, to merge organ predictions originating from different decoders and incorporate both healthy and pathological organs appearing in different datasets, we propose a body-part and anomaly-aware output merging scheme using automated body part and tumor predictions.

In summary, the main contributions are as follows:

- We are the first to comprehensively study the multi-organ continual semantic segmentation (CSS) problem with a clinically desirable number of organs (143 organs) across different body parts (head & neck, chest, abdomen) to more sufficiently and efficiently support medical diagnosis and treatment planning purposes.

- We propose the first (pure) architecture-based multi-organ continual segmentation framework. Consisting of a general encoder, continually expanded and pruned decoders, and a body-part and anomaly-aware output merging module, the proposed network avoids the notorious catastrophic forgetting in CSS while being scalable (maintaining the model complexity similar to other types of CSS approaches).

- Continually trained and validated on 3D CT scans of 2500+ patients compiled from four different datasets, our scalable unified model can segment total of 143 whole-body organs with very high accuracy, closely reaching the upper bound performance level of four well-trained individual models (i.e., nnUNet [26]).

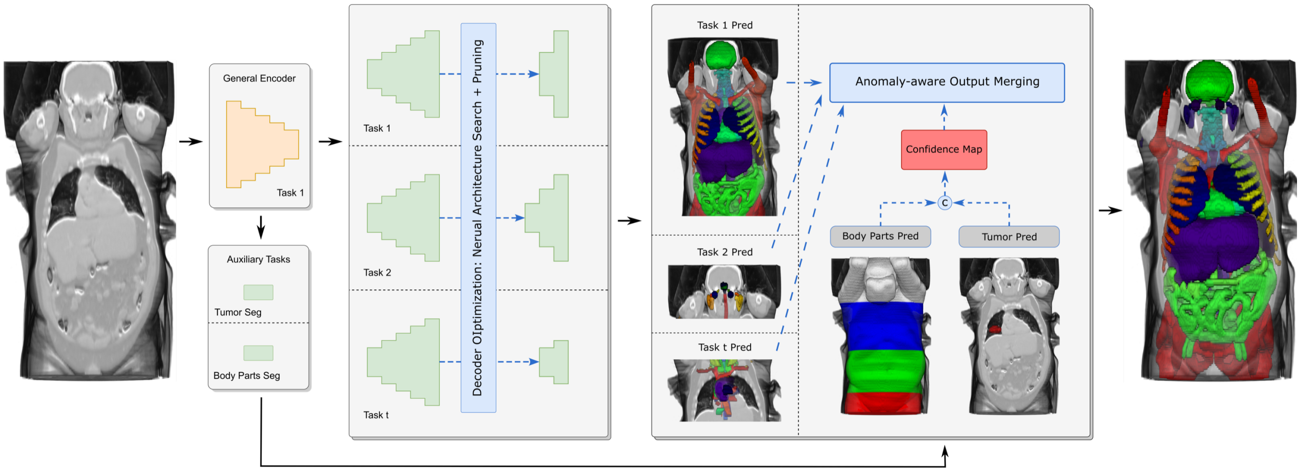

Figure 2. Overall framework of the proposed continual multi-organ segmentation, which is composed of a General Encoder, multiple optimized and pruned decoders (one for each learning step), and a body-part and anomaly-ware output merging module. After training the base encoding/decoding segmentation network using D_1, the General Encoder is frozen afterward, and separate trainable decoders are incrementally added to continually learn new datasets, which leads to a non-forgetting architecture. Decoder optimization and pruning are applied at each learning step to maintain a reasonable model complexity. Finally, the merging module is designed to combine organs from all decoders.

2. Related Work

Multi-Organ Segmentation. Automated multi-organ segmentation (MOS) is a challenging task in medical imaging with a long study history. The early registration-based atlas approach faces difficulty when large organ variation, tumor growth, or image acquisition differences exist. Recently, deep learning-based methods [26, 69, 31, 20, 21, 28, 48] have achieved great success when working on specific datasets with the same set of labeled organs. However, in practice, there are often partially labeled datasets, each with only one or a few labeled organs. Several recent works explore training a joint single model leveraging on multiple partially labeled datasets [74, 16, 57, 46, 72]. To address the major issue of background label conflicts, the marginal loss is often adopted to merge all unlabeled organs with the background [16, 57]. Different from these previous works that require all training datasets to be available/accessible at once, we train a single multi-organ segmentation model incrementally on multi-center partially-labeled datasets, with no access to previous datasets during the sequential process.

Continual Learning. Continual Learning aims to update a model from a sequence of new tasks and datasets without catastrophic forgetting [19, 32]. There are three main categories [11]. *Rehearsal-based* methods store a limited amount of training exemplars from old classes as raw images [49, 23, 38, 10, 3], embedded features [22, 25] or generators [44, 58]. However, it may be impracticable in real-world practice when data privacy is concerned, e.g., medical scans across multiple hospital sites are inaccessible. *Regularization-based* methods constrain the model plasticity either through regularization on weights [1, 8, 33, 70, 34] and gradients [39, 9], or knowledge distillation on output logits [36, 53, 49, 4] and intermediate features [12, 14, 75, 76]. Although without storing exemplars, they cannot guarantee desirable performance on challenging tasks. *Architecture-based* methods aim at either dynamically dividing task-specific partial network [18, 24, 41, 54], which suffers from running out of trainable parameters or expanding the network by freezing the old model and adding new parameters for new tasks [52, 35, 61, 68, 64, 62, 40], which guarantee no-forgetting performance but result in gradually growing/swelling model sizes. Our work falls into the expanding category, and we perform network pruning for each new task to control the overall model complexity.

Continual Semantic Segmentation. Continual semantic segmentation (CSS) is an emerging research topic with limited previous studies. Besides catastrophic forgetting, CSS faces the same challenge as partially labeled segmentation known as *background shift* [6]. ILT [42] proposes a CIS setting with a simple knowledge distillation solution. MiB [6] adapts marginal loss for both classification and distillation to solve background shift. A local-pooling-based distillation is applied to intermediate features in PLOP [13]. CSWKD [47] weights the distillation loss based on the old and new class similarity. SDR [43] propose to regularize the latent feature space using prototype matching and contrastive learning. Other than knowledge distillation, RCIL [71] designs a two-branch module for decoupling the representation learning of old and new classes. In multi-organ segmentation, only one study LISMO [37] applies CSS, based on MiB and prototype matching adapted from SDR [43], to segment five abdominal organs, which is an easy setting merely focusing on a single body part (abdomen). Our work is generalized for significantly more organ classes that are located in a large range of body parts (head & neck, chest, abdomen, hip & thigh).

3. Method

Problem Formulation. We aim to sequentially and continuously learn a single multi-organ segmentation model from several partially-labeled datasets one by one. Let $D = \{D_1, \ldots, D_T\}$ denote a sequence of data. When training on D_t, all previous training data $\{D_p, p < t\}$ are not accessible. For the t^{th} dataset $D_t = \{X_i^t, Y_i^t\}_{i=1}^{n_t}$ with C_t organ classes, let X^t and Y^t denote the input image and the corresponding organ label in the t^{th} dataset, the prediction map for voxel location, j, and output class c^t:

$$\hat{Y}^t(j) = f_d\left(Y^t(j) = c^t | f_e\left(X^t; W_e\right); W_d\right), \quad (1)$$

$$\hat{\mathbf{Y}} = \bigcup_{t=1}^{T} \hat{Y}^t, \quad (2)$$

where f_e, f_d, W_e, and W_d denote the CNN functions and the corresponding parameters for the encoding and decoding paths, respectively. The final prediction $\hat{\mathbf{Y}}$ is the union (with possible class overlapping) of all previous predictions.

Overall Training Process. Figure 2 illustrates the proposed multi-organ continual segmentation framework, which is composed of an encoder, multiple optimized and pruned decoders (one for each D_t), a body-part, and anomaly-ware output merging module. It starts from training a base encoding/decoding segmentation network using a comprehensive dataset D_1. We hypothesize that the well-trained encoder on D_1, represented as a General Encoder, is capable of extracting representative features (universal to all organs and datasets) to facilitate the subsequent learning tasks. Hence, this General Encoder is fixed afterward, and separate trainable decoders are incrementally added at the future learning steps, which leads to a non-forgetting architecture. Decoder optimization and pruning are also conducted at each learning step to maintain the model complexity comparable to a single network. Finally, by merging predictions from all decoders, we obtain a single unified model that can segment all organs of interest.

3.1. General Encoder Training

Ideally, for whole-body multi-organ segmentation, we expect to construct a sufficiently representative and universal General Encoder that extracts deep image features to capture and encode all visual information inside the full human body. Compared to the image statistics of broad natural image databases, medical images exist in a much more confined semantic domain, i.e., the human body is anatomically structured and composed of distinct body parts, no matter with or without diseases. This makes it feasible to learn a strong universal General Encoder competently capturing the holistic human body CT imaging statistics using large or not-so-limited multi-organ datasets. Sharing a similar idea, a very recent continual classification work [64] has empirically shown that a base classification model trained with a sufficiently large number of classes (e.g., 800) in ImageNet is capable of extracting representative features even for new classes. Here, our goal is to build a single unified segmentation model to accurately and continually segment up to 143 whole-body organs in CT scans (appeared in multiple datasets of both healthy subjects and diseased patients).

To train the General Encoder for multi-organ continual segmentation, we recommend starting with the publicly available TotalSegmentator [63] dataset as D_1, which consists of 1204 CT scans with a total of 103 labeled whole-body organs. These are routine diagnostic CT scans of different body parts with various scanning protocols. Besides this comprehensive dataset, we also supplement the General Encoder with auxiliary body-part segmentation and abnormal/tumor segmentation tasks. The body part labels can be obtained based on axial CT slice scores predicted by an automated body part regression algorithm [66]. As the slice score is monotonously correlated with the patient's anatomic height, slices with key landmarks can be determined to divide the whole body into four major regions, i.e., head & neck, chest, abdomen, and hip & thigh. The abnormal/tumor segmentation head is trained using dedicated tumor datasets. By involving these additional tasks, the General Encoder explicitly recognizes each pixel's anatomy region (body part) and potential abnormal tissues, which may be beneficial for learning better pixel representations. Moreover, the body part and tumor segmentation results can be further utilized in the output merging step to combine outputs from all decoders and reduce potential distal false positives from different decoders. For implementation, light-weighted body parts and tumor segmentation heads are added to the General Encoder using only the FCN8-like projection layers (0.04× size of a regular decoder) [30].

3.2. Decoder Optimization & Pruning

As the continual segmentation step extends, the proposed model complexity may escalate. Therefore, after initially training the decoder at each continual step, we further apply

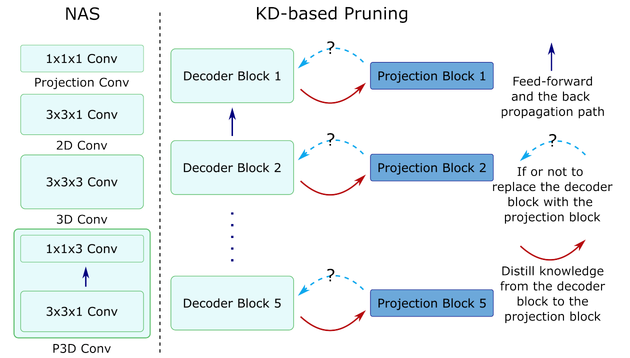

Figure 3. Illustration of the decoder optimization and pruning via neural architectural search and knowledge distillation.

a progressive optimization and pruning procedure to scale down the decoder complexity with the least possible performance drop.

Decoder Optimization via Neural Architectural Search (NAS). We first conduct NAS to optimize the decoder's segmentation performance and possibly reduce the decoder's parameters. Let $\phi\left(\cdot;\omega_{x\times y\times z}\right)$ denote a composition function of consecutive operations: batch normalization, a rectified linear unit, and a convolution layer with an $x \times y \times z$ dimension kernel. Inspired by previous work [77, 20], different convolutional layers may require various 2D/3D kernel types to segment 3D organs. Hence, we search for a set of possible convolutional kernels tailored to our problem: projection convolution $\phi\left(\cdot;\omega_{1\times1\times1}\right)$, 2D convolution $\phi\left(\cdot;\omega_{3\times3\times1}\right)$, pseudo-3D (P3D) convolution $\phi\left(\phi\left(\cdot;\omega_{3\times3\times1}\right);\omega_{1\times1\times3}\right)$, and 3D convolution $\phi\left(\cdot;\omega_{3\times3\times3}\right)$. To simplify the searching process, we use only one type of convolutional kernel to build each decoding block. At the end of the search, we determine the architecture of each block by choosing the ϕ corresponding to the largest weight value. Besides the optimized decoder performance, the searched 2D and P3D kernel parameters are only 1/3 and 4/9 of the 3D one, which also trims down the network parameter numbers.

Decoder Compression via Knowledge Distillation (KD). After NAS, we further prune the decoder by designing a convolution block-wise teacher-student-based Knowledge Distillation (KD) method. Each convolutional block is fixed and used as the teacher block. Next, we pair each teacher block with a projection block (i.e., a convolutional block with projection layers with kernel size 1), aiming to replace the teacher block with this projection block. The mean-square error loss is adopted to match the feature maps of the teacher block to the student block. Note that the student blocks have no path connection (hence no gradient back-propagation). To reduce the optimization difficulty, the deeper level of the decoding blocks is optimized first. Once the KD training of the deeper blocks is saturated, we freeze them and progressively move to the shallower ones.

Figure 3 illustrates the pruning method. After this process, there are 2^5 decoding paths when choosing between the original and the projection convolutional block, where all possible combinations are enumerated, and the corresponding segmentation performance and decoding parameter numbers are recorded. We use the decreased segmentation Dice score (%) to select the most possibly pruned decoding path. This decreased Dice score is defined by a performance drop tolerance parameter τ. In ablation experiments, we use $\tau \in \{1\%, 3\%, 5\%\}$ to inspect the model compression results. The final results are reported using $\tau = 1\%$. For the detailed distillation training process, please refer to the supplementary materials.

3.3. Body-part & Anomaly-aware Output Merging

We exploit the body part and anomaly predictions from two auxiliary tasks and propose a simple yet effective rule-based approach to combine the predictions from all decoders. Specifically, for each dataset/task, we pre-compute the merged bounding boxes of all labeled organs. Next, we calculate the average body part distribution map P^t for each dataset t by overlapping the averaged bounding box to the body part labels. Let $\hat{Y}^\epsilon$ denote the distinct tumor prediction, $\odot$ denote the element-wise multiplication, and J denote the matrix of ones, the weighting map M^t is calculated using Eq. (3), i.e., only when $\hat{Y}^\epsilon \to 0$ and $P^t \to 1$ s.t. the $M^t \to 1$, whereas $M^t \to 0.5$ for the rest states. We use the entropy function Eq. (4) to compute the confidence map.

$$M^t = J - \frac{1}{2}\left(J - P^t + \hat{Y}^\epsilon \odot P^t\right) \tag{3}$$

$$H^t = -\left(M^t \odot \hat{Y}^t\right) \log\left(M^t \odot \hat{Y}^t\right), \tag{4}$$

$$\mathbf{H}(j) = \bigcup_{\forall \hat{Y}(j)^t \neq 0} H^t(j), t \in \{1,\dots,T\}, \tag{5}$$

$$\hat{\mathbf{Y}}(j) = \hat{Y}^{\arg\min(\mathbf{H}(j))}(j) \tag{6}$$

For each voxel, we collect a set $\mathbf{H}(j)$, for all $\hat{Y}(j)^t \neq 0$. Depicted in Eq. (6), the final output class $\hat{\mathbf{Y}}(j)$ is determined using the prediction $\hat{Y}^t(j)$, of which with the smallest $H^t(j)$. For the detailed merging setups, please refer to the supplementary Sec. B.

4. Experiments

Datasets: We evaluated our method using 2500+ patients from one public and three private partially labeled multi-organ datasets. TotalSegmentator [63] consists of 1204 CT scans of different body parts with a total of 103 labeled anatomical structures (26 major organs, 59 bone instances, 10 muscles, and 8 vessels). Note that the face label is removed as it is an artificially created label for patient de-identification purposes after blurring the facial area. In the

Table 1. Continual multi-organ segmentation final results on two orders of our datasets. Dataset names are followed by their class numbers. Mean DSC (%, ↑), HD95 (mm, ↓) and ASD (mm, ↓) are evaluated on each dataset as well as all classes (All). 'Params #': decoder(s) parameter number of the final model (# (MB)) and the relative number (Rel #) compared to the original nnUNet decoder. †: ILT is reimplemented using a frozen encoder setting and the unbiased loss from MiB for better performance.

Methods	TotalSeg (103)			ChestOrgan (31)			HNOrgan (13)			EsoOrgan (1)			All (143)			Params #	
	DSC	HD95	ASD	DSC	HD95	ASD	DSC	HD95	ASD	DSC	HD95	ASD	DSC	HD95	ASD	# (MB)	Rel #
Order A: TotalSeg → ChestOrgan → HNOrgan → EsoOrgan																	
MiB [6]	7.65	119.66	67.41	19.24	37.14	8.34	6.37	7.40	2.38	86.92	4.33	1.09	8.51	98.98	51.98	15.068	1.00
ILT† [42]	10.87	192.23	116.20	27.87	36.93	7.41	6.39	4.04	0.81	85.75	4.57	1.17	11.99	148.96	86.34		
PLOP [13]	37.30	53.71	23.33	51.74	35.36	8.71	25.38	16.12	9.24	82.90	6.21	1.62	39.01	46.63	18.48		
LISMO [37]	10.82	129.82	76.92	28.24	36.33	9.08	6.30	12.93	4.14	**87.12**	**4.24**	**1.05**	12.11	96.89	54.71		
RCIL [71]	42.58	48.28	23.24	57.76	33.95	9.12	27.96	16.88	8.59	84.72	5.95	1.16	42.43	44.89	18.67		
Ours	**92.98**	**4.09**	**0.98**	**78.26**	**9.17**	**1.82**	**83.97**	**2.22**	**0.59**	86.94	5.04	1.11	**88.74**	**5.28**	**1.14**	14.669	0.98
Order B: TotalSeg → HNOrgan → ChestOrgan → EsoOrgan																	
MiB [6]	10.35	136.77	63.51	65.63	14.37	1.94	6.29	24.83	7.22	86.79	4.31	1.08	20.00	68.82	29.87	15.068	1.00
ILT† [42]	13.12	201.66	106.51	67.28	14.21	1.88	6.18	3.12	0.95	85.52	4.80	1.25	22.31	115.23	59.34		
PLOP [13]	30.82	62.07	23.14	70.18	13.05	2.36	15.77	11.09	3.84	83.41	6.11	1.54	36.49	44.78	16.01		
LISMO [37]	14.04	90.17	47.81	67.19	14.88	1.93	6.15	9.13	1.44	86.87	**4.18**	**1.03**	22.92	57.71	28.22		
RCIL [71]	35.24	59.81	24.20	70.74	12.98	2.22	18.43	11.81	3.65	84.17	6.14	1.09	39.85	45.52	15.07		
Ours	**92.98**	**4.09**	**0.98**	**78.26**	**9.17**	**1.82**	**83.97**	**2.22**	**0.59**	**86.94**	5.04	1.11	**88.74**	**5.28**	**1.14**	14.669	0.98
Upper bound	93.24	3.29	0.83	78.45	8.16	1.83	84.35	2.38	0.60	87.15	4.44	0.98	89.02	4.41	1.06	15.07×4	1.0×4

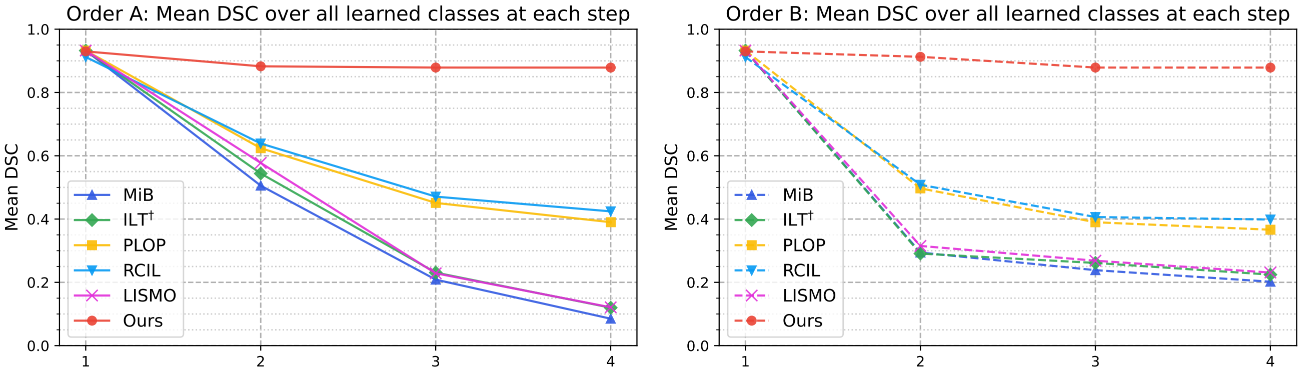

Figure 4. The mean DSCs over all learned classes at each step of **Order A** (left, solid line) and **Order B** (right, dashed line).

in-house collection, the ChestOrgan dataset contains 292 chest CT scans, most of which come from early esophageal or lung cancer patients. For the ChestOrgan dataset, 31 chest anatomical structures are labeled, among which 4 overlapped with organs in TotalSegmentator (esophagus, trachea, SVC, pulmonary artery). Another dataset includes 447 head & neck CT scans (denoted as HNOrgan dataset), where 13 organs are annotated as organs at risk (OARs) in radiation therapy and do not have class overlap with all other datasets. The fourth dataset is a dedicated cancer dataset validating the domain change of CSS, containing 640 diagnostic CT scans of advanced esophageal cancer patients where only the esophagus is labeled (denoted as the EsoOrgan dataset). For the detailed organ list, please refer to the supplementary Sec. A. By combining all datasets, we have a total of $103+27+13 = 143$ organ classes from 2583 unique patients. For each of these four datasets, 20% is randomly chosen as an independent testing set, while the rest is used as training + validation in each continual learning step.

In addition, for the purpose of training and validating our abnormality segmentation module, we further collect CT scans from 304 esophageal (private) and 625 lung cancer (public with labels) patients where the 3D tumor masks are segmented.

Overall CSS Training Process: In our CSS experiment, the model is trained to segment organs sequentially at multiple steps. At each step t, the model is trained on the specific dataset D_t without access to any other datasets. Specifically, at step-1, D_1 is first used to train both the General Encoder and the associated decoder, where the decoder is further optimized and pruned using D_1. After that, D_1 cannot be accessed in any future steps. This process is repeated for step 2, ..., T, except that at each step-t, D_t is only used to train, optimize and prune D_t dedicated decoder keeping the General Encoder always frozen.

CSS Protocols: We examine two CSS orders with four learning steps. **Order A** goes as: *TotalSegmentator* → *ChestOrgan* → *HNOrgan* → *EsoOrgan*. **Order B** goes as: *TotalSegmentator* → *HNOrgan* → *ChestOrgan* → *EsoOrgan*, which exchanges the *ChestOrgan* with *HNOrgan* to demonstrate the effect of different body parts in CSS. All methods (including ours) are trained and evaluated in both orders. To report the final results in CSS setting, we compute segmentation metrics after the last learning step for all the previous datasets. For reporting the results in any intermediate step t, these metrics are calculated after the learning step t for all the datasets $i \leq t$.

Metrics: We report the Dice similarity coefficient (DSC), 95% Hausdorff distance (HD95) and average surface distance (ASD) to quantify the organ segmentation results.

4.1. Implementation Details

A $[-1024, 1024]$ HU CT windowing is applied to every CT image. We resample all CT scans to the same resolution: $0.75 \times 0.75 \times 3.0$mm. The ratio between the training and validation set is 4:1. "3d-full res" version (+ "moreDA" data augmentation) of nnUNet [26] with DSC+CE losses is adopted for all model training with a batch size of 2. The training patch size is $128 \times 128 \times 64$. We set 8000 epochs for training General Encoder and the associated decoder using the TotalSegmentator dataset in step-1, and 1000 epochs for training the dedicated decoder at each future step-t.

NAS Setting: At a learning step $t, 1 \leq t \leq T$, after initially training the decoder, we further exploit NAS to search for the optimal network architecture for the associate decoder. For NAS training, the dataset D_t (training+validation) is initially divided into 1) 60% for network training, 2) 30% for NAS training, and 3) 10% for validation evaluation. The initial learning rate is set to 0.01. We first fix the NAS parameters and train the network for 400 epochs. Then we alternatively update the NAS and network parameters for additional 600 epochs. The batch size is set to 4 in NAS training. Only the validation set is used for updating NAS parameters. After NAS training, we follow the same 'moreDA' data augmentation scheme and retrain the searched decoding architecture from scratch using D_t (training+validation) with a re-divided 'training-validation' ratio of 4:1.

Pruning Setting: After NAS, we perform a block-wise teacher-student KD to compress the decoder by replacing the searched convolutional kernels with the projection kernels. The initial learning rate is 0.01. We fix the teacher networks and train the paired student network for another 500 epochs. MSE loss is used for teacher-student feature map matching. After the pruning is completed, we replace the selected teacher blocks with the student blocks and finetune the trimmed network for 500 epochs with a learning rate of 0.001. All models are developed using PyTorch and trained on one NVIDIA A100 GPU. Please refer to the supplementary Sec. B for more implementation details.

Comparing Methods: We compare our method with five latest leading CSS works, including four regularization-based methods (ILT [42], MiB [6], PLOP [13], RCIL [71]) and a hybrid of regularization and rehearsal-based method (LISMO [37]). To ensure comparisons' fairness, we reimplement ILT, MiB, LISMO, and PLOP in the nnUNet framework to guarantee consistent data pre-processing and data augmentation (Re-implementation details are fully disclosed in the supplementary Sec. E). Noted that all four datasets in our experiment are partially labeled, hence, it is not straightforward to compute the upper bound performance using a single model. In this work, we train a separate nnUNet [26] model for each dataset, the results of which can serve as the CSS performance upper bound for each dataset.

Table 2. Segmentation performance under two 1-step continual learning scenarios with and without freezing the General Encoder. Mean DSC (%, ↑), HD95 (mm, ↓) and ASD (mm, ↓) are evaluated.

Settings	TotalSeg → ChestOrgan				TotalSeg → HNOrgan			
	TotalSeg		ChestOrgan		TotalSeg		HNOrgan	
	DSC	HD95	DSC	HD95	DSC	HD95	DSC	HD95
unfreezing	51.42	26.52	78.45	8.16	2.90	162.09	84.35	2.38
freezing	92.98	4.09	77.91	8.37	92.98	4.09	84.14	2.39

4.2. Comparison to Leading CSS Methods

Overall Performance: Table 1 and Figure 4 show final segmentation results after continually learning on two orders (each with four steps) of our datasets. Our proposed method significantly outperforms other leading methods on the previously learned three datasets as well as the total 143 organs in both CSS orders. The second best performing method RCIL [71] still experiences catastrophic forgetting and has a mean DSC of 41.43% and 39.85% in CSS order A and B, far less than our mean DSC of 88.74%. Similar performance gaps are noticed on HD95 and ASD metrics (e.g., ∼45mm vs. ∼5mm in terms of HD95). Our proposed method achieves very similar performance to the upper bound with a 0.28% marginal decrease in DSC and a 0.08mm increase in ASD (see Figure 5 for qualitative results). In model complexity, the overall parameter number of our four pruned decoders (14.7 MB) is 98% of an original nnUNet decoder (15.1 MB), which is only 24% size of the decoders required by achieving the upper bound performance. The running time of the proposed framework (segmenting 143 organs) is slightly longer (+12%) than the running time of a single nnUNet to segment 103 organs. For our detailed results of individual organs or organ groups, please refer to the supplementary Sec. D.

Two CSS Orders: Table 1 also demonstrates the segmentation results under two CSS orders (order A and B). Because the proposed framework consists of a frozen General Encoder, independent decoders (each for one continual learning step), and a unified output merging module, our method is order invariant if the base dataset for training General Encoder is the same. On the other hand, the continual learning order may significantly affects the comparison methods. E.g., LISMO has a mean DSC of 28.24% v.s. 67.19% on

Table 3. Multi-organ segmentation results using decoder optimization & pruning. We report the number of decoder parameters and the relative size percentage compared to the original nnUNet decoder when the DSC (%) is dropped by $\tau \in \{1\%, 3\%, 5\%\}$.

		DSC Drop		
		1%	3%	5%
TotalSeg	**DSC**	92.98	90.72	88.83
	#(MB)	6.53	4.50	3.28
	Rel #	0.43	0.30	0.22
ChestOrgan	**DSC**	78.26	77.16	74.88
	#(MB)	3.39	2.85	1.23
	Rel #	0.23	0.19	0.08
HNOrgan	**DSC**	83.97	82.24	80.27
	#(MB)	4.18	4.04	1.88
	Rel #	0.28	0.27	0.12
EsoOrgan	**DSC**	86.94	85.97	–
	#(MB)	0.67	0.57	–
	Rel #	0.04	0.04	–

ChestOrgan dataset in order A and B, respectively.

The significant performance drop of the comparing methods could be caused by the catastrophic forgetting induced from the bodypart-related domain gap. In our experiments, we observe that the comparing methods generally work well if new and old datasets share similar domains/body-parts, e.g. ChestOrgan → EsoOrgan (second from the left plot of bottom row in the supplementary Figure C.1). However, in whole-body organ continual segmentation, different datasets may cover various body parts with limited overlaps, which causes a large gap in the image domain and significantly deteriorates the performance, e.g. ChestOrgan → HNOrgan (third from the left plot of bottom row in the supplementary Figure C.1). In contrast, when learning new tasks, our framework keeps previously learned parameters unchanged and avoids knowledge forgetting. Please refer to the supplementary Sec. C for more detailed results and discussion on the step-wise results of our method and comparing methods.

4.3. Ablation Study Results

Effectiveness of General Encoder: To demonstrate the importance of freezing the General Encoder when learning subsequent tasks, we compare the segmentation performance with and without freezing the General Encoder when continually learning on new datasets (using two CSS orders with two learning steps). Results are summarized in Table 2. First, it is observed that without freezing the General Encoder, the model has catastrophic forgetting, e.g., segmentation DSC of the old dataset in TotalSegmentator → ChestOrgan decreases from 93.24% to 51.42% as compared to that with the frozen encoder. Second, the performance for segmenting the new dataset is similar regardless of the encoder status (freezing or trainable). For instance, 84.14% vs. 84.35% DSC of HNOrgan dataset is achieved in TotalSegmentator → HNOrgan. The experimental results demonstrate that a well-trained and subsequently frozen General Encoder could generalize well to support specialized tasks.

Table 4. Quantitative results of using different output merging methods. Mean DSC (%), HD95 (mm) and ASD (mm) are evaluated. Better performance is indicated in bold.

	Ensemble			**Anomaly-aware merging**		
	DSC	**HD95**	**ASD**	**DSC**	**HD95**	**ASD**
TotalSeg	88.59	4.41	1.09	**92.98**	**4.09**	**0.98**
ChestOrgan	76.78	9.44	1.89	**78.26**	**9.17**	**1.82**
HNOrgan	77.84	2.65	0.67	**83.97**	**2.22**	**0.59**
EsoOrgan	80.22	7.62	1.92	**86.94**	**5.04**	**1.11**

Effectiveness of Decoder Pruning: Table 3 shows the detailed decoder pruning results. Several conclusions can be drawn. First, the proposed decoder pruning method achieves a good trade-off between model complexity and accuracy reduction. For example, for the TotalSegmentator decoder, with 1% DSC decrease, the number of parameters is reduced from 15.07 MB to 6.53 MB with a relative 43% of the original decoder size. As the larger performance drop is allowed, e.g., 3% and 5% DSC decrease, the size of pruned decoder decreases to 30% and 22% of the original decoder, respectively. Second, as the number of segmented organs becomes smaller, a higher compressed ratio can be achieved. With 1% DSC performance decrease, the pruned ChestOrgan decoder (segmenting 31 organs) has 3.39 MB parameters as compared to 6.53 MB of pruned TotalSegmentator decoder. Third, the EsoOrgan decoder has the highest model compression ratio with only 0.67 MB parameters (4% of original decoder size). This indicates that domain-incremental segmentation may be an easier task as compared to class-incremental continual segmentation.

Effectiveness of Merging Module: Table 4 presents the segmentation results using two merging methods. It is observed that a simple ensemble-based merging method exhibits decreased performance in all metrics on all datasets. The proposed anomaly-aware output merging significantly boosts the performance on the EsoOrgan dataset (DSC: 80.22% to 86.94%, HD95: 7.62 to 5.04mm, ASD: 1.92 to 1.11mm). This demonstrates the effectiveness and importance of the abnormal detection module. The proposed merging module can identify the esophageal tumor and subsequently generate a high confidence score for the EsoOrgan decoder suitable for segmenting advanced esophageal cancer patients. In contrast, the ensemble method could not differentiate if there exists abnormality in an image. Hence, averaging the esophagus predictions from three decoders that predict the esophagus leads to significantly decreased performance.

Alternative Training Dataset for the General Encoder: We recommend starting with TotalSegentator as D_1 to train the General Encoder as it covers most body parts with a large set of labeled organs for comprehensive feature extraction. However, this is not a hard requirement. Alterna-

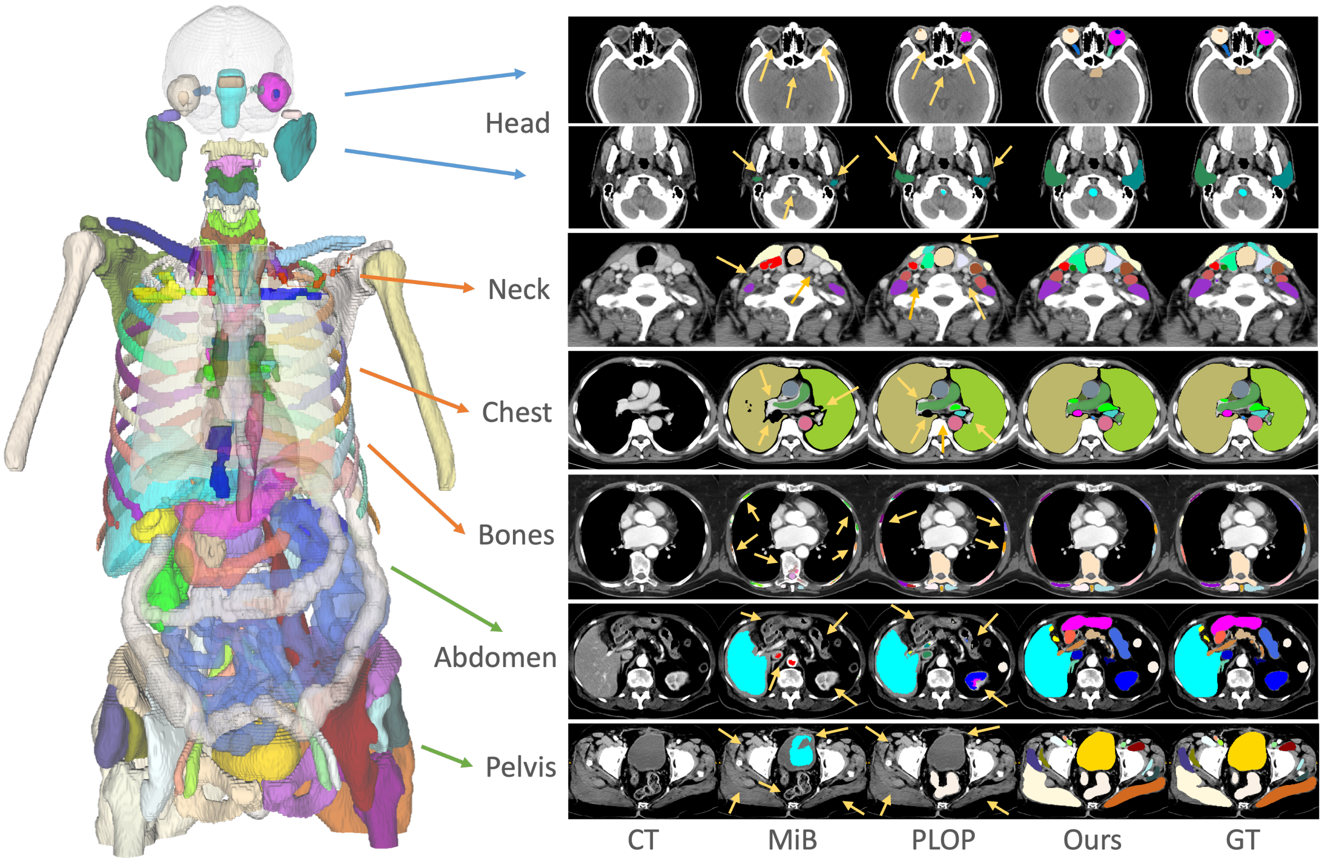

Figure 5. The qualitative comparison of our method with MiB [6] and PLOP [13]. All the segmentation results are from the last step of CSS order A. Seven quality examples are shown covering different body parts and organ groups. Yellow arrows indicate missing/wrong predictions. MiB and PLOP experience severe forgetting in the head, abdomen and pelvis regions, since these body parts only appear in one or two tasks/datasets; while less forgetting is observed in neck and upper chest regions, which appear in all four tasks/datasets (more suitable for CSS by the MiB and PLOP methods). (For visualization purpose, not all the organs are shown in each example.)

tively, other datasets can also be used as the starting dataset to train the General Encoder. When training the General Encoder using the ChestOrgan dataset with much less training scans (292 vs. 1204 CT scans) and organ classes (31 vs. 103 anatomical structures), a tolerable performance drop (<1% Dice) of our method is observed in the final results of CSS Order A. The assumed reason is that CT scans in the ChestOrgan dataset covers most of the torso region with diverse anatomies, which allows the General Encoder to learn sufficient representative features. Hence, General Encoder trained with ChestOrgan exhibits similar performance as the one trained using TotalSegmentator.

5. Conclusion

In this work, we propose a new CSS framework to continually segment a total of 143 whole-body organs from four partially labeled datasets. With the trained and frozen General Encoder and continually-added and architecturally optimized decoders, our model avoids catastrophic forgetting while effectively segmenting new organs with high accuracy. We further propose a body-part and anomaly-aware output merging module to combine organ predictions originating from different decoders and incorporate both healthy and pathological organs appearing in different datasets. Continually trained and validated on 3D CT scans of 2500+ patients of four datasets, our single network can segment 143 whole-body organs with very high accuracy, closely reaching the upper bound performance level by training four separate segmentation models.

References

[1] Rahaf Aljundi, Francesca Babiloni, Mohamed Elhoseiny, Marcus Rohrbach, and Tinne Tuytelaars. Memory aware synapses: Learning what (not) to forget. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 139–154, 2018. 3

[2] Michela Antonelli, Annika Reinke, Spyridon Bakas, Keyvan Farahani, Annette Kopp-Schneider, Bennett A Landman, Geert Litjens, Bjoern Menze, Olaf Ronneberger, Ronald M

Summers, et al. The medical segmentation decathlon. *Nature communications*, 13(1):1–13, 2022. 1

[3] 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*, 2020. 3

[4] Francisco M Castro, Manuel J Marín-Jiménez, Nicolás Guil, Cordelia Schmid, and Karteek Alahari. End-to-end incremental learning. In *Proceedings of the European conference on computer vision (ECCV)*, pages 233–248, 2018. 3

[5] Fabio Cermelli, Matthieu Cord, and Arthur Douillard. Comformer: Continual learning in semantic and panoptic segmentation. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3010–3020, 2023. 2

[6] Fabio Cermelli, Massimiliano Mancini, Samuel Rota Bulo, Elisa Ricci, and Barbara Caputo. Modeling the background for incremental learning in semantic segmentation. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9233–9242, 2020. 2, 3, 6, 7, 9

[7] Chun-Hung Chao, Zhuotun Zhu, Dazhou Guo, Ke Yan, Tsung-Ying Ho, Jinzheng Cai, Adam P Harrison, Xianghua Ye, Jing Xiao, Alan Yuille, et al. Lymph node gross tumor volume detection in oncology imaging via relationship learning using graph neural network. In *International Conference on Medical Image Computing and Computer-Assisted Intervention*, pages 772–782. Springer, 2020. 1

[8] Arslan Chaudhry, Puneet K Dokania, Thalaiyasingam Ajanthan, and Philip HS Torr. Riemannian walk for incremental learning: Understanding forgetting and intransigence. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 532–547, 2018. 3

[9] Arslan Chaudhry, Marc'Aurelio Ranzato, Marcus Rohrbach, and Mohamed Elhoseiny. Efficient lifelong learning with a-gem. *arXiv preprint arXiv:1812.00420*, 2018. 3

[10] Arslan Chaudhry, Marcus Rohrbach, Mohamed Elhoseiny, Thalaiyasingam Ajanthan, Puneet K Dokania, Philip HS Torr, and Marc'Aurelio Ranzato. On tiny episodic memories in continual learning. *International Conference on Machine Learning (ICML) Workshop*, 2019. 3

[11] Matthias Delange, Rahaf Aljundi, Marc Masana, Sarah Parisot, Xu Jia, Ales Leonardis, Greg Slabaugh, and Tinne Tuytelaars. A continual learning survey: Defying forgetting in classification tasks. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, pages 1–1, 2021. 3

[12] Prithviraj Dhar, Rajat Vikram Singh, Kuan-Chuan Peng, Ziyan Wu, and Rama Chellappa. Learning without memorizing. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5138–5146, 2019. 3

[13] Arthur Douillard, Yifu Chen, Arnaud Dapogny, and Matthieu Cord. Plop: Learning without forgetting for continual semantic segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4040–4050, 2021. 2, 3, 6, 7, 9

[14] Arthur Douillard, Matthieu Cord, Charles Ollion, Thomas Robert, and Eduardo Valle. Podnet: Pooled outputs distillation for small-tasks incremental learning. In *European Conference on Computer Vision*, pages 86–102, 2020. 3

[15] Mark Everingham, SM Eslami, Luc Van Gool, Christopher KI Williams, John Winn, and Andrew Zisserman. The pascal visual object classes challenge: A retrospective. *International journal of computer vision*, 111(1):98–136, 2015. 2

[16] Xi Fang and Pingkun Yan. Multi-organ segmentation over partially labeled datasets with multi-scale feature abstraction. *IEEE Transactions on Medical Imaging*, 39(11):3619–3629, 2020. 2, 3

[17] Elise MN Ferré, Timothy J Break, Peter D Burbelo, Michael Allgäuer, David E Kleiner, Dakai Jin, Ziyue Xu, Les R Folio, Daniel J Mollura, Muthulekha Swamydas, et al. Lymphocyte-driven regional immunopathology in pneumonitis caused by impaired central immune tolerance. *Science translational medicine*, 11(495):eaav5597, 2019. 1

[18] Siavash Golkar, Michael Kagan, and Kyunghyun Cho. Continual learning via neural pruning. *Advances in Neural Information Processing Systems (NeurIPS) Workshop*, 2019. 3

[19] Ian J Goodfellow, Mehdi Mirza, Da Xiao, Aaron Courville, and Yoshua Bengio. An empirical investigation of catastrophic forgetting in gradient-based neural networks. *Proceedings of the International Conference on Learning Representations (ICLR)*, 2014. 3

[20] Dazhou Guo, Dakai Jin, Zhuotun Zhu, Tsung-Ying Ho, Adam P Harrison, Chun-Hung Chao, Jing Xiao, and Le Lu. Organ at risk segmentation for head and neck cancer using stratified learning and neural architecture search. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4223–4232, 2020. 3, 5

[21] Dazhou Guo, Xianghua Ye, Jia Ge, Xing Di, Le Lu, Lingyun Huang, Guotong Xie, Jing Xiao, Zhongjie Lu, Ling Peng, et al. Deepstationing: thoracic lymph node station parsing in ct scans using anatomical context encoding and key organ auto-search. In *Medical Image Computing and Computer Assisted Intervention*, pages 3–12. Springer, 2021. 3

[22] Tyler L Hayes, Kushal Kafle, Robik Shrestha, Manoj Acharya, and Christopher Kanan. Remind your neural network to prevent catastrophic forgetting. In *European Conference on Computer Vision*, pages 466–483, 2020. 3

[23] Saihui Hou, Xinyu Pan, Chen Change Loy, Zilei Wang, and Dahua Lin. Learning a unified classifier incrementally via rebalancing. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 831–839, 2019. 3

[24] Ching-Yi Hung, Cheng-Hao Tu, Cheng-En Wu, Chien-Hung Chen, Yi-Ming Chan, and Chu-Song Chen. Compacting, picking and growing for unforgetting continual learning. *Advances in Neural Information Processing Systems*, 2019. 3

[25] Ahmet Iscen, Jeffrey Zhang, Svetlana Lazebnik, and Cordelia Schmid. Memory-efficient incremental learning through feature adaptation. In *European Conference on Computer Vision*, pages 699–715. Springer, 2020. 3

[26] Fabian Isensee, Paul F Jaeger, Simon AA Kohl, Jens Petersen, and Klaus H Maier-Hein. nnu-net: a self-configuring method for deep learning-based biomedical image segmentation. *Nature methods*, 18(2):203–211, 2021. 1, 2, 3, 7

[27] Krishna S Iyer, John D Newell Jr, Dakai Jin, Matthew K Fuld, Punam K Saha, Sif Hansdottir, and Eric A Hoffman. 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. 1

[28] Zhanghexuan Ji, Yan Shen, Chunwei Ma, and Mingchen Gao. Scribble-based hierarchical weakly supervised learning for brain tumor segmentation. In *Medical Image Computing and Computer Assisted Intervention*, pages 175–183. Springer, 2019. 3

[29] Dakai Jin, Dazhou Guo, Jia Ge, Xianghua Ye, and Le Lu. Towards automated organs at risk and target volumes contouring: Defining precision radiation therapy in the modern era. *Journal of the National Cancer Center*, 2022. 1

[30] Dakai Jin, Dazhou Guo, Tsung-Ying Ho, Adam P Harrison, Jing Xiao, Chen-Kan Tseng, and Le Lu. Accurate esophageal gross tumor volume segmentation in pet/ct using two-stream chained 3d deep network fusion. In *International Conference on Medical Image Computing and Computer-Assisted Intervention*, pages 182–191. Springer, 2019. 4

[31] Dakai Jin, Dazhou Guo, Tsung-Ying Ho, Adam P Harrison, Jing Xiao, Chen-Kan Tseng, and Le Lu. Deeptarget: Gross tumor and clinical target volume segmentation in esophageal cancer radiotherapy. *Medical Image Analysis*, 68:101909, 2021. 1, 3

[32] Ronald Kemker, Marc McClure, Angelina Abitino, Tyler Hayes, and Christopher Kanan. Measuring catastrophic forgetting in neural networks. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 32, 2018. 2, 3

[33] James Kirkpatrick, Razvan Pascanu, Neil Rabinowitz, Joel Veness, Guillaume Desjardins, Andrei A Rusu, Kieran Milan, John Quan, Tiago Ramalho, Agnieszka Grabska-Barwinska, et al. Overcoming catastrophic forgetting in neural networks. *Proceedings of the national academy of sciences*, 114(13):3521–3526, 2017. 3

[34] Abhishek Kumar, Sunabha Chatterjee, and Piyush Rai. Bayesian structural adaptation for continual learning. In *International Conference on Machine Learning*, pages 5850–5860. PMLR, 2021. 3

[35] Xilai Li, Yingbo Zhou, Tianfu Wu, Richard Socher, and Caiming Xiong. Learn to grow: A continual structure learning framework for overcoming catastrophic forgetting. In *International Conference on Machine Learning*, pages 3925–3934. PMLR, 2019. 3

[36] Zhizhong Li and Derek Hoiem. Learning without forgetting. *IEEE transactions on pattern analysis and machine intelligence*, 40(12):2935–2947, 2017. 3

[37] Pengbo Liu, Xia Wang, Mengsi Fan, Hongli Pan, Minmin Yin, Xiaohong Zhu, et al. Learning incrementally to segment multiple organs in a CT image. In *International Conference on Medical Image Computing and Computer-Assisted Intervention*, pages 714–724. Springer, 2022. 2, 4, 6, 7

[38] Yaoyao Liu, Bernt Schiele, and Qianru Sun. Rmm: Reinforced memory management for class-incremental learning. *Advances in Neural Information Processing Systems*, 34, 2021. 3

[39] David Lopez-Paz and Marc'Aurelio Ranzato. Gradient episodic memory for continual learning. *Advances in neural information processing systems*, 30, 2017. 3

[40] Chunwei Ma, Zhanghexuan Ji, Ziyun Huang, Yan Shen, Mingchen Gao, and Jinhui Xu. Progressive voronoi diagram subdivision enables accurate data-free class-incremental learning. In *The Eleventh International Conference on Learning Representations*, 2022. 3

[41] Arun Mallya and Svetlana Lazebnik. Packnet: Adding multiple tasks to a single network by iterative pruning. In *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*, pages 7765–7773, 2018. 3

[42] Umberto Michieli and Pietro Zanuttigh. Incremental learning techniques for semantic segmentation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision Workshops*, pages 0–0, 2019. 2, 3, 6, 7

[43] Umberto Michieli and Pietro Zanuttigh. Continual semantic segmentation via repulsion-attraction of sparse and disentangled latent representations. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1114–1124, 2021. 2, 4

[44] Oleksiy Ostapenko, Mihai Puscas, Tassilo Klein, Patrick Jahnichen, and Moin Nabi. Learning to remember: A synaptic plasticity driven framework for continual learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 11321–11329, 2019. 3

[45] Firat Ozdemir, Philipp Fuernstahl, and Orcun Goksel. Learn the new, keep the old: Extending pretrained models with new anatomy and images. In *International Conference on Medical Image Computing and Computer-Assisted Intervention*, pages 361–369. Springer, 2018. 2

[46] Olivier Petit, Nicolas Thome, and Luc Soler. Iterative confidence relabeling with deep convnets for organ segmentation with partial labels. *Computerized Medical Imaging and Graphics*, 91:101938, 2021. 3

[47] Minh Hieu Phan, Son Lam Phung, Long Tran-Thanh, Abdesselam Bouzerdoum, et al. Class similarity weighted knowledge distillation for continual semantic segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 16866–16875, 2022. 3

[48] Ashwin Raju, Zhanghexuan Ji, Chi Tung Cheng, Jinzheng Cai, Junzhou Huang, Jing Xiao, et al. User-guided domain adaptation for rapid annotation from user interactions: a study on pathological liver segmentation. In *International Conference on Medical Image Computing and Computer-Assisted Intervention*, pages 457–467. Springer, 2020. 3

[49] Sylvestre-Alvise Rebuffi, Alexander Kolesnikov, Georg Sperl, and Christoph H Lampert. icarl: Incremental classifier and representation learning. In *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*, pages 2001–2010, 2017. 3

[50] Nicola Rieke, Jonny Hancox, Wenqi Li, Fausto Milletari, Holger R Roth, Shadi Albarqouni, Spyridon Bakas, Mathieu N Galtier, Bennett A Landman, Klaus Maier-Hein, et al. The future of digital health with federated learning. *NPJ digital medicine*, 3(1):1–7, 2020. 2

[51] Holger R Roth, Le Lu, Jiamin Liu, Jianhua Yao, Ari Seff, Kevin Cherry, Lauren Kim, and Ronald M Summers. Improving computer-aided detection using convolutional neural networks and random view aggregation. *IEEE transactions on medical imaging*, 35(5):1170–1181, 2015. 1

[52] Andrei A Rusu, Neil C Rabinowitz, Guillaume Desjardins, Hubert Soyer, James Kirkpatrick, Koray Kavukcuoglu, Raz-

van Pascanu, and Raia Hadsell. Progressive neural networks. *arXiv preprint arXiv:1606.04671*, 2016. 3

[53] Jonathan Schwarz, Wojciech Czarnecki, Jelena Luketina, Agnieszka Grabska-Barwinska, Yee Whye Teh, Razvan Pascanu, and Raia Hadsell. Progress & compress: A scalable framework for continual learning. In *International Conference on Machine Learning*, pages 4528–4537, 2018. 3

[54] Joan Serra, Didac Suris, Marius Miron, and Alexandros Karatzoglou. Overcoming catastrophic forgetting with hard attention to the task. In *International Conference on Machine Learning*, pages 4548–4557. PMLR, 2018. 3

[55] Yan Shen, Jian Du, Han Zhao, Zhanghexuan Ji, Chunwei Ma, and Mingchen Gao. Fedmm: A communication efficient solver for federated adversarial domain adaptation. In *International Conference on Autonomous Agents and Multiagent Systems*, pages 1808–1816, 2023. 2

[56] Feng Shi, Weigang Hu, Jiaojiao Wu, Miaofei Han, Jiazhou Wang, Wei Zhang, Qing Zhou, Jingjie Zhou, Ying Wei, Ying Shao, et al. Deep learning empowered volume delineation of whole-body organs-at-risk for accelerated radiotherapy. *Nature Communications*, 13(1):1–13, 2022. 1

[57] Gonglei Shi, Li Xiao, Yang Chen, and S Kevin Zhou. Marginal loss and exclusion loss for partially supervised multi-organ segmentation. *Medical Image Analysis*, 70:101979, 2021. 2, 3

[58] Hanul Shin, Jung Kwon Lee, Jaehong Kim, and Jiwon Kim. Continual learning with deep generative replay. *Advances in neural information processing systems*, 30, 2017. 3

[59] Hao Tang, Xuming Chen, Yang Liu, Zhipeng Lu, Junhua You, Mingzhou Yang, Shengyu Yao, Guoqi Zhao, Yi Xu, Tingfeng Chen, et al. Clinically applicable deep learning framework for organs at risk delineation in ct images. *Nature Machine Intelligence*, 1(10):480–491, 2019. 1

[60] Sebastian Thrun. Lifelong learning algorithms. In *Learning to learn*, pages 181–209. Springer, 1998. 2

[61] Yu-Xiong Wang, Deva Ramanan, and Martial Hebert. Growing a brain: Fine-tuning by increasing model capacity. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2471–2480, 2017. 3

[62] 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 *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 139–149, 2022. 3

[63] Jakob Wasserthal, Manfred Meyer, Hanns-Christian Breit, Joshy Cyriac, Shan Yang, and Martin Segeroth. Totalsegmentator: robust segmentation of 104 anatomical structures in ct images. *arXiv preprint arXiv:2208.05868*, 2022. 4, 5

[64] Tz-Ying Wu, Gurumurthy Swaminathan, Zhizhong Li, Avinash Ravichandran, Nuno Vasconcelos, Rahul Bhotika, and Stefano Soatto. Class-incremental learning with strong pre-trained models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9601–9610, 2022. 2, 3, 4

[65] Jia-Wen Xiao, Chang-Bin Zhang, Jiekang Feng, Xialei Liu, Joost van de Weijer, and Ming-Ming Cheng. Endpoints weight fusion for class incremental semantic segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7204–7213, 2023. 2

[66] Ke Yan, Le Lu, and Ronald M. Summers. Unsupervised body part regression via spatially self-ordering convolutional neural networks. In *IEEE ISBI*, pages 1022–2025, 2018. 4

[67] Xianghua Ye, Dazhou Guo, Jia Ge, Senxiang Yan, Yi Xin, Yuchen Song, Yongheng Yan, Bing-shen Huang, et al. Comprehensive and clinically accurate head and neck cancer organs-at-risk delineation on a multi-institutional study. *Nature communications*, 13(1):1–15, 2022. 1

[68] Jaehong Yoon, Eunho Yang, Jeongtae Lee, and Sung Ju Hwang. Lifelong learning with dynamically expandable networks. *Proceedings of the International Conference on Learning Representations (ICLR)*, 2017. 3

[69] Qihang Yu, Dong Yang, Holger Roth, Yutong Bai, Yixiao Zhang, Alan L Yuille, and Daguang Xu. C2fnas: Coarse-to-fine neural architecture search for 3d medical image segmentation. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4126–4135, 2020. 3

[70] Friedemann Zenke, Ben Poole, and Surya Ganguli. Continual learning through synaptic intelligence. In *International Conference on Machine Learning*, pages 987–995, 2017. 3

[71] Chang-Bin Zhang, Jia-Wen Xiao, Xialei Liu, Ying-Cong Chen, and Ming-Ming Cheng. Representation compensation networks for continual semantic segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7053–7064, 2022. 2, 4, 6, 7

[72] Jianpeng Zhang, Yutong Xie, Yong Xia, and Chunhua Shen. Dodnet: Learning to segment multi-organ and tumors from multiple partially labeled datasets. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1195–1204, 2021. 3

[73] Bolei Zhou, Hang Zhao, Xavier Puig, Sanja Fidler, Adela Barriuso, and Antonio Torralba. Scene parsing through ade20k dataset. In *IEEE conference on computer vision and pattern recognition*, pages 633–641, 2017. 2

[74] Yuyin Zhou, Zhe Li, Song Bai, Chong Wang, Xinlei Chen, Mei Han, Elliot Fishman, and Alan L Yuille. Prior-aware neural network for partially-supervised multi-organ segmentation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 10672–10681, 2019. 3

[75] Fei Zhu, Xu-Yao Zhang, Chuang Wang, Fei Yin, and Cheng-Lin Liu. Prototype augmentation and self-supervision for incremental learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5871–5880, 2021. 3

[76] Kai Zhu, Wei Zhai, Yang Cao, Jiebo Luo, and Zheng-Jun Zha. Self-sustaining representation expansion for non-exemplar class-incremental learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9296–9305, 2022. 3

[77] Zhuotun Zhu, Chenxi Liu, Dong Yang, Alan Yuille, and Daguang Xu. V-nas: Neural architecture search for volumetric medical image segmentation. In *2019 International conference on 3d vision (3DV)*, pages 240–248, 2019. 5