Representation Compensation Networks for Continual Semantic Segmentation

Chang-Bin Zhang [1*] Jia-Wen Xiao [1*] Xialei Liu [1†] Ying-Cong Chen [2,3] Ming-Ming Cheng [1]
[1] TMCC, CS, Nankai University [2] The Hong Kong University of Science and Technology (Guangzhou)
[3] The Hong Kong University of Science and Technology

Abstract

In this work, we study the continual semantic segmentation problem, where the deep neural networks are required to incorporate new classes continually without catastrophic forgetting. We propose to use a structural re-parameterization mechanism, named representation compensation (RC) module, to decouple the representation learning of both old and new knowledge. The RC module consists of two dynamically evolved branches with one frozen and one trainable. Besides, we design a pooled cube knowledge distillation strategy on both spatial and channel dimensions to further enhance the plasticity and stability of the model. We conduct experiments on two challenging continual semantic segmentation scenarios, continual class segmentation and continual domain segmentation. Without any extra computational overhead and parameters during inference, our method outperforms state-of-the-art performance. The code is available at `https://github.com/zhangchbin/RCIL`.

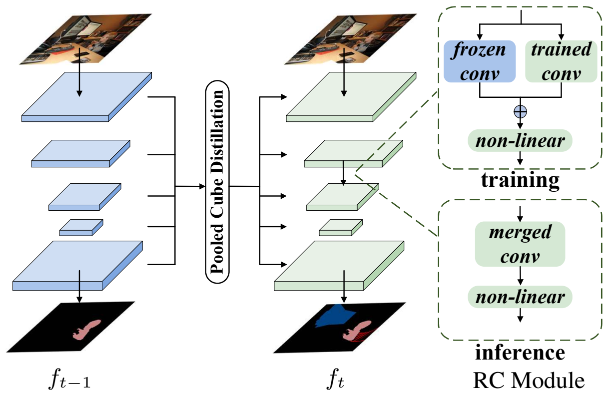


Figure 1. Illustration of our proposed training framework for continual semantic segmentation to avoid catastrophic forgetting. We design two mechanisms in our method, representation compensation (RC) module and pooled cube distillation (PCD).

1. Introduction

Data-driven deep neural networks [64, 72, 96, 108] have made many milestones in semantic segmentation. However, these fully-supervised models [16, 23, 93] can only handle a fixed number of classes. In real-world applications, it is preferable that a model can be dynamically extended to identify new classes. A straightforward solution is to rebuild the training set and retrain the model with all data available, known as *Joint Training*. However, considering the cost of retraining models, sustainable development of algorithms and privacy issues, it is particularly crucial to update the model with only current data to achieve the goal of recognizing both new and old classes. Nevertheless, naively fine-tuning a trained model with new data can result in catastrophic forgetting [48]. Therefore, in this paper, we seek continual learning, which can potentially enable a model to recognize new categories without catastrophic forgetting.

In the scenario of continual semantic segmentation [8, 27, 62, 63], given the previously trained model and the training data of the new classes, the model is supposed to distinguish all seen classes, including previous classes (old classes) and new classes. However, to save the labeling cost, the new training data often only has labels for the new classes, treating old classes as background. Learning with the new data directly without any additional designs is very challenging, which can easily lead to catastrophic forgetting [48].

As indicated in [28, 48, 51], fine-tuning the model on new data may lead to catastrophic forgetting, *i.e.,* the model quickly fits the data distribution of the new classes, while losing the discrimination for the old classes. Some methods [43,48,56,66,67,80,95] play regularization on model parameters to improve its stability. However, all parameters are updated on the training data of the new classes. This is however challenging, as new and old knowledge are entangled together in model parameters, making it extremely difficult to keep the fragile balance of learning new knowledge and keeping old ones. Some other methods [45,57,75,76,82,91] increase the capacity of the model to have a better trade-off of stability and plasticity, but with the cost of growing memory of the network.

In this work, we propose an easy-to-use representation compensation module, aiming at remembering the old knowl-

*The first two authors contribute equally.
†Corresponding author (xialei@nankai.edu.cn)

edge while allowing extra capacity for new knowledge. Inspired by structural re-parameterization [24, 25], we replace the convolution layers in the network with two parallel branches during training, which is named as representation compensation module. As shown in Fig. 1, during training, the output of two parallel convolutions is fused before the non-linear activation layer. At the beginning of each continual learning step, we equivalently merge the parameters of the two parallel convolutions into one convolution, which will be frozen to retain the old knowledge. Another branch is trainable and it inherits the parameters from the corresponding branch in the previous step. The representation compensation strategy is supposed to remember the old knowledge using the frozen branch while allowing extra capacity for new knowledge using the trainable branch. Importantly, this module brings no extra parameters and computation cost during inference.

To further alleviate catastrophic forgetting [48], we introduce a knowledge distillation mechanism [70] between intermediate layers (shown in Fig. 1), named Pooled Cube Distillation. It can suppress the negative impact of errors and noises in local feature maps. The main contributions of this paper are:

- We propose a representation compensation module with two branches during training, one for retaining the old knowledge and one for adapting to new data. It always keeps the same computation and memory cost during inference as the number of tasks grows.
- We conduct experiments on *continual class segmentation* and *continual domain segmentation*, respectively. Experimental results demonstrate that our method outperforms the state-of-the-art performance on three different datasets.

2. Related Work

Semantic Segmentation. Early methods focused on modeling contextual relationships [2,49,103]. Currently methods pay more attention to multi-scale feature aggregation [3, 34, 52,53,59,65,68,81]. Some methods [14,22,32,37,38,50,55] is inspired by Non-local [85], utilizing attention mechanisms to establish connections between image contexts. Another line of research [15, 61, 94] aimed at fusing features from different receptive fields. Recently, transformer architectures [7, 26, 86, 97, 104, 109] shine in semantic segmentation, focusing on multi-scale feature fusion [12, 84, 89, 101] and contextual feature aggregation [58, 79].

Continual Learning. Continual learning focuses on alleviating catastrophic forgetting while being discriminative for newly learned classes. To solve this problem, many work [4, 5,11,47,77] propose to review knowledge by rehearsal-based mechanism. The knowledge can be stored by multiple types, like examples [4, 6, 9, 11, 73, 83], prototypes [35, 106, 107], generative networks [60], *etc*. Although these rehearsal-based methods usually achieve high performance, they need storage and authority for storing. In the more challenging scenario without any replay, many methods explore regularization to maintain old knowledge, including knowledge distillation [10, 18, 21, 28, 51, 69, 74, 100], adversarial training [29,88], vanilla regularization [43,48,56,66,67,80,95,98] and so on. Others focus on the capacity of the neural network. One of the research line [45, 57, 75, 76, 82, 91] is to expand the network architecture while learning new knowledge. Another research line [1, 44] explores the sparsity regularization for network parameters, which aims at activating as few neurons as possible for each task. This sparsity regularization reduces the redundancy in the network, while limiting the learning capacity for each task. Some work propose to learn better representations by combining self-supervised learning for feature extractor [9, 87] and solving class imbalance [39, 46, 54, 99, 102].

Continual Semantic Segmentation. Continual semantic segmentation is still an urgent problem to solve, mainly focusing on catastrophic forgetting [48] in semantic segmentation. In this field, continual class segmentation is a classic setting, with great progress made by several previous work: [41,92] explore rehearsal-based methods to review old knowledge; MiB [8] models the potential classes to solve the ambiguous of *background* class; PLOP [27] applies knowledge distillation strategy to intermediate layers; SDR [63] takes advantage of prototype matching to perform consistency constraints in the latent space representation. While others [31, 78, 95] utilize high-dimensional information, self-training and model adaptation to overcome this problem. Moreover, continual domain segmentation is a novel setting proposed by PLOP [27], aiming at integrating new domain rather than new classes. Different from previous methods, we focus on expanding the network dynamically, decoupling the representation learning of old classes and new classes.

3. Method

3.1. Preliminaries

Let $\mathcal{D} = \{x_i, y_i\}$ denotes the training set, where x_i denotes the input image and y_i is the corresponding segmentation ground-truth. In the challenging continual learning scenario, we call each training on the newly added dataset $\mathcal{D}_t$ as a *step*. At step t, given a model f_{t-1} with parameter θ_{t-1} trained on $\{\mathcal{D}_0, \mathcal{D}_1 ... \mathcal{D}_{t-1}\}$ with $\{C_0, C_1 ... C_{t-1}\}$ classes continually, the model is supposed to learn the discrimination for $\sum_{n=0}^{t} C_n$ classes when it encounters a newly added dataset $\mathcal{D}_t$ with extra C_t new classes. When training on $\mathcal{D}_t$, the training data of old classes are not accessible. Besides, to save the training cost, the ground-truth in $\mathcal{D}_t$ only contains the C_t new classes, while the old classes are labeled as

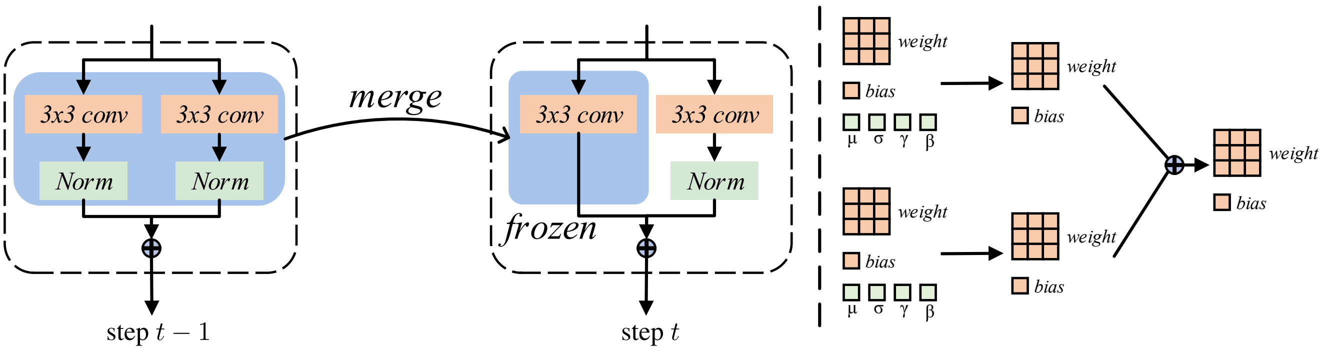


Figure 2. Illustration of our representation compensation mechanism. We modify the 3×3 convolution as two parallel convolutions. The features from the two branches are aggregated before the activation layer. At the beginning of step t, thus, the two parallel branches trained at step $t-1$ can be merged into an equivalent convolution layer, which will be frozen and is regarded as one branch of *step* t. Another branch in *step* t is initialized from the corresponding branch from *step* $t-1$. We demonstrate the merge operation in the right part of the figure.

background. Thus, there is an urgent problem, catastrophic forgetting. To verify the effectiveness of different methods, it is often necessary to perform the continual learning multiple times *e.g.,* N steps.

3.2. Representation Compensation Networks

To decouple the retaining of old knowledge and learning of new knowledge, as shown in Fig. 2, we introduce our representation compensation mechanism. In most of the deep neural networks, a 3×3 convolution followed by normalization and non-linear activation layer is a common component. We modify this architecture by adding a parallel 3×3 convolution followed by a normalization layer for each component. The output of two parallel convolution-normalization layers is fused, then is rectified by a non-linear activation layer. Formally, this architecture contains two parallel convolution layers with weight $\{W^0, W^1\}$ and bias $\{b^0, b^1\}$, followed by two independent normalization layers, respectively. Let $Norm^0 = \{\mu^0, \sigma^0, \gamma^0, \beta^0\}$ and $Norm^1 = \{\mu^1, \sigma^1, \gamma^1, \beta^1\}$ denote the mean, variance, weight and bias of two normalization layers $Norm^0$ and $Norm^1$. Thus, the calculation of input x before non-linear activation function can be denoted as

$$\begin{aligned}
\hat{x} &= \sum_{i=0}^{1} Norm_i(W_i x + b_i) \\
&= \sum_{i=0}^{1} (\gamma_i \frac{W_i x + b_i - \mu_i}{\sigma_i} + \beta_i) \\
&= (\sum_{i=0}^{1} \frac{\gamma_i W_i}{\sigma_i}) x + \sum_{i=0}^{1} (\frac{\gamma_i b_i - \gamma_i \mu_i}{\sigma_i} + \beta_i) \\
&= \hat{W} x + \hat{b}.
\end{aligned} \tag{1}$$

This equation demonstrates that two parallel branches can be equivalently represented as one with weight $\hat{W}$ and bias $\hat{b}$. We also display the transformation in the right part of Fig. 2. Therefore, for this modified architecture, we can equivalently merge the parameters of two branches into one convolution.

More precisely, in step 0, all parameters are trainable to train a model that can discriminate $\mathcal{C}_0$ classes. For the subsequent learning steps, the model is supposed to segment newly added classes. In these continual learning steps, the network will be initialized with the parameters trained in the previous step, which is beneficial to transfer knowledge [8]. At the beginning of step t, since the model is supposed to avoid forgetting old knowledge, we merge the parallel branches trained in step $t-1$ to one convolution layer. The parameters in this merged branch are frozen to memorize the old knowledge, as shown in Fig. 2. Another branch is trainable to learn new knowledge, which is initialized with the corresponding branch in the previous step. Besides, we design a drop-path strategy, which is applied on aggregating the output, x_1 and x_2 from two branches. During training, the output before the non-linear activation is denoted as

$$\hat{x} = \eta \cdot x_1 + (1-\eta) \cdot x_2, \tag{2}$$

where η is the random channel-wise weighted vector and sampling from the set $\{0, 0.5, 1\}$ uniformly. During inference, the element of vector η is set as 0.5. Experimental results demonstrate that this strategy brings slight improvement.

Analysis on RC-Module's Effectiveness. As shown in Fig. 3, the parallel convolution structure can be regarded as an implicit ensemble [36, 40] of numerous sub-networks. The parameters of some layers in these sub-networks are inherited from the the merged teacher model (trained at previous step) and are frozen. During training, similar to [33, 90], these frozen teacher layers will impose regularization to trainable parameters, encouraging trainable layers to behave like the teacher model. In a special case where only one layer in the sub-network is trainable, as shown in Fig. 3(a), during training, this layer will take into account both adapting for the representation of frozen layers and learning for new knowledge. Therefore, this mechanism will alleviate catastrophic forgetting of the trainable layer. We further promote this effect to general sub-networks like Fig. 3(b),

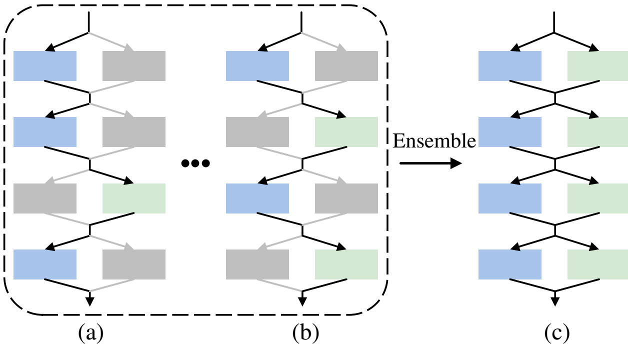


Figure 3. Illustration of our proposed Representation Compensation Network. Our architecture (c) can be regarded as an implicit ensemble of numerous sub-networks (a), (b), *etc*. The blue denotes the frozen layers inherited from the merged teacher model. The green denotes the trainable layers. The gray denotes the layers that are ignored in the sub-network.

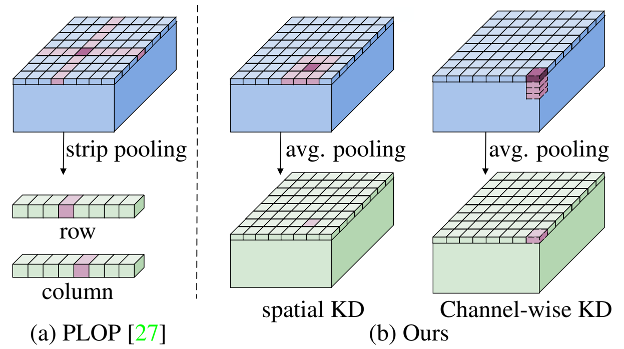


Figure 4. Comparison between PLOP [27] and our proposed Pooled Cube Knowledge Distillation mechanism.

which will also encourage the trainable layers to adapt to the representation of the frozen layers. Furthermore, all sub-networks are ensembled, integrating knowledge from different sub-networks to one network, like Fig. 3(c).

3.3. Pooled Cube Knowledge Distillation

In order to further alleviate the forgetting of old knowledge, following PLOP [27], we also explore feature distillation. As shown in Fig. 4(a), PLOP [27] introduces strip pooling [38] to integrate features. Pooling operation plays a key role in transferring knowledge. In our method, we design the average pooling-based knowledge distillation along the spatial dimension. Additionally, we use the average pooling in the channel dimension at each position as well to maintain their individual activation intensity. Overall, as shown in Fig. 4(b), we use the average pooling on both spatial and channel dimensions.

Formally, we select feature maps $\{X^1, X^2, ..., X^L\}$ before the last non-linear activation layer for all L stages, including decoder and all stages in the backbone. For the features from the teacher model and the student model, we firstly calculate the square of value at each pixel to retain the negative information. Then, we perform multi-scale average pooling on spatial and channel dimensions, respectively. The features $\hat{X}^l_T, \hat{X}^l_S$ of the teacher model and the student model can be calculated by the average pooling operation $\odot$:

$$\begin{aligned} \hat{X}_T^{l,m} &= M \odot [(X^l_{T,ij})^2] \\ \hat{X}_S^{l,m} &= M \odot [(X^l_{S,ij})^2], \end{aligned} \tag{3}$$

where M denotes the m_{th} average pooling kernel, and l denotes the l_{th} stage. For the average pooling on the spatial dimension, we use the multi-scale windows to model the relationships between pixels in the local region. The size of kernel M belongs to $\mathcal{M} = \{4, 8, 12, 16, 20, 24\}$ and the step size is set to 1. And we simply set the window size as 3 for the average pooling on channel dimension. Then, the spatial knowledge distillation loss function L_{skd} for the intermediate layers can be denoted as

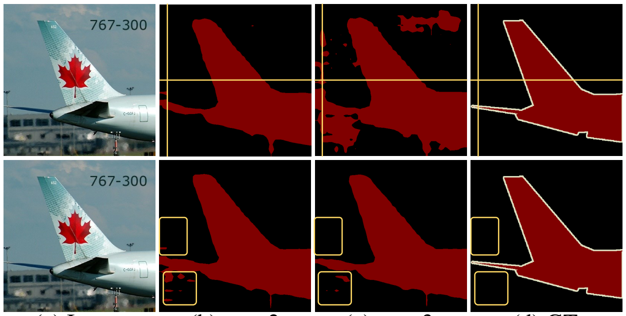
$$L_{skd} = \frac{1}{L}\frac{1}{|\mathcal{M}|}\sum_{l=1}^{L}\sum_{m=1}^{|\mathcal{M}|}\sqrt{\sum_{i=1}^{H}\sum_{j=1}^{W}\sum_{d=1}^{D}[(\hat{X}^{l,m}_{T,ijd} - \hat{X}^{l,m}_{S,ijd})^2]}, \tag{4}$$

where H, W, D denote the height, width and the number of channels. The same equation can be applied on channel dimension with $\mathcal{M} = \{3\}$ to form L_{ckd}. Overall, the distillation objective can be denoted as:

$$L = L_{skd} + L_{ckd}. \tag{5}$$

Average pooling ***vs.*** **Strip pooling**. Benefiting from its strong ability to aggregate features and model long-range dependency, strip pooling shines in many fully-supervised semantic segmentation models [38, 42]. The performance of continual segmentation is still much worse than that of fully-supervised segmentation. In the scenario of continual segmentation, there are often more noise or errors in the prediction results than fully-supervised segmentation. Thus, in the distillation process, when using strip pooling to aggregate features, this long-range dependency will introduce some uncorrelated noise to the cross point, causing noise diffusion. This will lead to further deterioration of the prediction results of the student model. In our method, we use average pooling in the local region to suppress the negative impact of noise. Specifically, because the semantics of local regions are often similar, the current key point can find more neighbors to support its decision by aggregating features in the local region. Thus, the current key point is less negatively affected by the noise in the local region.

As an example shown in Fig. 5(b) top, the strip pooling introduces noise or errors to the cross point for the teacher model. During the distillation process, the noise is further propagated to the student model, making the noise diffusion. For the average pooling in Fig. 5 bottom, the key point will



(a) Image (b) step 2 (c) step 3 (d) GT

Figure 5. The impact of the strip pooling (top row) used in PLOP [27] and the average pooling (bottom row) in our method.

consider many nearby neighbors, resulting in an aggregated feature that is more robust to noise.

4. Experiments

In this section, we first demonstrate the details of our experimental setups, *e.g.,* datasets, protocols and training details. Then we illustrate the effectiveness of our method from quantitative and qualitative experiments.

4.1. Experimental setups

4.1.1 Datasets

PASCAL VOC 2012 [30] is a commonly used dataset, which contains 10,582 training images and 1449 validation images with 20 object classes and the background class. **ADE20K** [105] is a dataset for semantic segmentation covering daily life scenes. It contains 20,210 training images and 2,000 validation images with 150 classes. **Cityscapes** [19] contains 2,975 training images, 500 validation images and 1,525 test images. There are 19 classes from 21 cities.

4.1.2 Protocols

Continual Class Segmentation. In continual class segmentation, the model is trained to recognize different classes sequentially in multiple steps. Each step the model learns one or several classes. Following [8, 27, 63], we assume training data of previous steps are not available, *i.e.,* the model can only access data of the current step. Besides, only classes to be learned in the current step are labeled. All other classes are treated as *background*. There are two commonly used settings proposed by [8] for continual class segmentation, *disjoint* and *overlapped*. In the *disjoint* setting, assuming we know all classes in the future, the images in the current training step do not contain any classes in the future. The *overlapped* setting is more realistic. It allows potential classes in the future to appear in the current training images.

We conduct continual class segmentation experiments on the PASCAL VOC 2012 [30] and ADE20K [105]. Following [8,27,63], as defined in Sec. 3.1, we call each training on the newly added dataset as a *step*. Formally, X-Y denotes the continual setting in our experiments, where X denotes the number of classes that we need to train in the first step. In each subsequent learning step, the newly added dataset contains Y classes. On PASCAL VOC 2012 [30], we conduct experiments on three settings, 15-5 (2 steps), 15-1 (6 steps) and 10-1 (11 steps). For example, 15-1 denotes that we train the model on the initial 15 object classes in the first step. In the subsequent five steps, the model is expected to be trained on new datasets, where each dataset contains one new added class. Thus, the model can discriminate 20 object classes in the last step. On ADE20K [105], we apply four settings, *100-50 (2 steps)*, *50-50 (3 steps)*, *100-10 (6 steps)*, and *100-5 (11 steps)* .

Continual Domain Segmentation. It is proposed by [27]. Different from continual class segmentation, this setting is to deal with the domain shift phenomenon rather than integrating new classes. In the real-world scene, domain shift can also occur frequently. We assume the classes in different domains are the same. The training data of the old domain is not accessible when training on new domain data. We conduct continual domain segmentation experiments on Cityscapes [19]. Following PLOP [27], we regard the training data in each city as a domain. We also apply three settings, *11-5 (3 steps)*, *11-1 (11 steps)* and *1-1 (21 steps)*. In these experimental settings, we use the same recording as the continual class segmentation, but each step adds new domains (cities) instead of classes.

4.1.3 Implementation Details

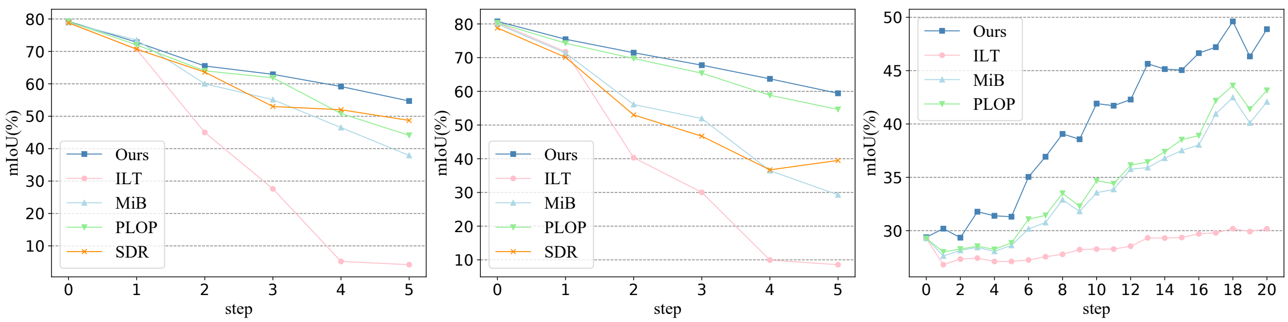
Following [8, 27, 63], we use the Deeplab-v3 [13] architecture with ResNet-101 [36] as backbone. The output stride of Deeplab-v3 is set to 16. We also apply the in-place activated batch normalization [71] in the backbone pre-trained on the ImageNet [20], as the above methods. We utilized the loss function proposed by MiB [8] to assist our training process. And we apply the same training strategy as [8,27,63]. Specifically, we apply the same data augmentation, *e.g.,* horizontal flip and random crop. The batch size is set to 24 for all experiments. We set the initial learning rate as 0.02 for the first training step and 0.001 for the next continual learning steps. The learning rate is adjusted by the *poly* schedule. We train the model using SGD optimizer for each step with 30 (PASCAL VOC 2012 [30]), 50 (Cityscapes [19]), and 60 epochs (ADE20K [105]), respectively. We also use 20% of the training set as validation following [8, 27, 63]. We report the mean Intersect over Union (mIoU) on the original validation set.

4.2. Continual Class Segmentation

PASCAL VOC 2012. Applying the same experimental settings as [8,27,63], we performed experiments on different

Method	15-5 (2 steps)						15-1 (6 steps)						10-1 (11 steps)					
	Disjoint			Overlapped			Disjoint			Overlapped			Disjoint			Overlapped		
	0-15	*16-20*	*all*	*0-15*	*16-20*	*all*	*0-15*	*16-20*	*all*	*0-15*	*16-20*	*all*	*0-10*	*11-20*	*all*	*0-10*	*11-20*	*all*
Fine-tuning	5.7	33.6	12.3	6.6	33.1	12.9	4.6	1.8	3.8	4.6	1.8	3.9	6.3	1.1	3.8	6.4	1.2	3.9
Joint	79.8	72.6	78.2	79.8	72.6	78.2	79.8	72.6	78.2	79.8	72.6	78.2	78.2	78.0	78.2	78.2	78.0	78.2
LwF [51]	60.4	37.4	54.9	60.8	36.6	55.0	5.8	3.6	5.3	6.0	3.9	5.5	7.2	1.2	4.3	8.0	2.0	4.8
ILT [62]	64.9	39.5	58.9	67.8	40.6	61.3	8.6	5.7	7.9	9.6	7.8	9.2	7.3	3.2	5.4	7.2	3.7	5.5
MiB [8]	73.0	43.3	65.9	76.4	49.4	70.0	48.4	12.9	39.9	38.0	13.5	32.2	9.5	4.1	6.9	20.0	20.1	20.1
SDR [63]	74.6	44.1	**67.3**	76.3	50.2	70.1	59.4	14.3	**48.7**	47.3	14.7	39.5	17.3	11.0	**14.3**	32.4	17.1	25.1
PLOP [27]	71.0	42.8	64.3	75.7	51.7	**70.1**	57.9	13.7	46.5	65.1	21.1	**54.6**	9.7	7.0	8.4	44.0	15.5	**30.5**
Ours	75.0	42.8	**67.3**	78.8	52.0	**72.4**	66.1	18.2	**54.7**	70.6	23.7	**59.4**	30.6	4.7	**18.2**	55.4	15.1	**34.3**

Table 1. The mIoU(%) of the last step on the Pascal VOC 2012 dataset for different continual class segmentation scenarios. The **red** denotes the highest results and the **blue** denotes the second highest results.



(a) 15-1 disjoint on PASCAL VOC 2012 (b) 15-1 overlapped on PASCAL VOC 2012 (c) 1-1 on Cityscapes

Figure 6. The mIoU (%) at each step in three experimental settings. (a)(b) are settings of continual class segmentation. (c) is the setting of continual domain segmentation.

continual learning settings, *15-5*, *15-1* and *10-1*. As shown in Tab. 1, we report the experimental results of the last step. The vanilla fine-tuning method suffers from the catastrophic forgetting phenomena. The model quickly forgets the old knowledge and is unable to learn the new knowledge well. Experimental results demonstrate that our method significantly improves the segmentation performance both on the overlapped and disjoint settings. Especially in the challenging 15-1 settings, our method outperforms the state-of-the-art by 6.0% (*disjoint*) and 4.8% (*overlapped*) in terms of mIoU, respectively. We also display the performance of each step for different methods as shown in Fig. 6a and Fig. 6b. This demonstrates that our method can reduce the forgetting of old knowledge in the continual learning process. In Tab. 1, we also report the performance over the old classes and new classes, respectively. For all settings, the performance of the old classes is greatly improved. This is benefited from the representation compensation module and distillation mechanism, which can effectively retain the old knowledge. On the other hand, our proposed representation module and distillation mechanism allow room for learning new knowledge. In Sec. 4.4, we will further analyze the effectiveness of these two mechanisms. We further show the qualitative results of different methods in the 15-1 *overlapped* setting in Fig. 7.

ADE20K. To verify the effectiveness of our method, we conduct experiments on a challenging semantic segmentation dataset, ADE20K [105]. Experimental results are shown in Tab. 2 and Tab. 3. On different continual learning tasks, 100-50, 100-10 and 50-50, our method achieves an average improvement of 1.4% over the state-of-the-art. To further verify our method, we also perform experiments on a more challenging scenario, 100-5, which contains 11 steps. In this scenario, our method also achieves the state-of-the-art, outperforming the previous method by about 0.9% in terms of mIoU, as shown in Tab. 3. The improvement is due to our proposed representation compensation module and pooled cube distillation mechanism.

4.3. Continual Domain Segmentation

In the context of continual semantic segmentation, in addition to the need to segment new classes, it is also of great significance to increase the processing capabilities of new domains. Following [27], we conducted experiments of continual domain semantic segmentation on Cityscapes [19]. Each city in Cityscapes [19] can be regarded as a domain,

	100-50 (2 steps)			100-10 (6 steps)							50-50 (3 steps)			
Method	*1-100*	*101-150*	*all*	*1-100*	*101-110*	*111-120*	*121-130*	*131-140*	*141-150*	*all*	*1-50*	*51-100*	*101-150*	*all*
ILT [62]	18.3	14.8	17.0	0.1	0.0	0.1	0.9	4.1	9.3	1.1	13.6	12.3	0.0	9.7
MiB [8]	40.7	17.7	32.8	38.3	12.6	10.6	8.7	9.5	15.1	29.2	45.3	26.1	17.1	29.3
PLOP [27]	41.9	14.9	**32.9**	40.6	15.2	16.9	18.7	11.9	7.9	**31.6**	48.6	30.0	13.1	**30.4**
Ours	42.3	18.8	**34.5**	39.3	14.6	26.3	23.2	12.1	11.8	**32.1**	48.3	31.3	18.7	**32.5**
Joint	44.3	28.2	38.9	44.3	26.1	42.8	26.7	28.1	17.3	38.9	51.1	38.3	28.2	38.9

Table 2. The mIoU(%) of the last step on the ADE20K dataset for different overlapped continual learning scenarios. The **red** denotes the highest results and the **blue** denotes the second highest results.

Method	*1-100*	*101-150*	*all*
ILT [62]	0.1	1.3	0.5
MiB [8]	36.0	5.6	25.9
PLOP [27]	39.1	7.8	**28.7**
Ours	38.5	11.5	**29.6**

Table 3. The final mIoU(%) of 100-5 overlapped on ADE20K.

Method	**11-5** (3 steps)	**11-1** (11 steps)	**1-1** (21 steps)
Fine-tuning	61.7	60.4	42.9
LwF [51]	59.7	57.3	33.0
LwF-MC [69]	58.7	57.0	31.4
ILT [62]	59.1	57.8	30.1
MiB [8]	61.5	60.0	42.2
PLOP [27]	**63.5**	**62.1**	**45.2**
Ours	**64.3**	**63.0**	**48.9**

Table 4. The final mIoU(%) for continual domain semantic segmentation on Cityscapes [19].

which is widely used by domain adaptive semantic segmentation tasks [17]. In this scenario, we do not consider the difference in classes between domains. As shown in Tab. 4, experimental results demonstrate that our method achieves higher mIoU than previous methods [8, 27, 62] in all three settings. Our method outperforms the state-of-the-art by 3.7% on the challenging 1-1 setting with 21 learning steps. For this setting, we display the performance of each step in Fig. 6c. Since MiB [8] aims at solving the problem of semantic shift which is not existing in continual domain segmentation, MiB [8] performs slightly worse than Fine-tuning. These experiments indicate that our method is also effective for continual domain semantic segmentation, benefiting from the ability to retain old knowledge while allowing to learn new knowledge.

4.4. Ablation Study

In this section, we firstly analyze the effectiveness of our proposed representation compensation and pooled cube distillation mechanism. Then we discuss the robustness to class orders in the continual learning scenario.

Representation Compensation. We conduct ablation

MiB‡ [8]	RC	Strip [38]	S-KD	C-KD	15-1
✓					36.1
✓	✓				43.0
✓	✓		✓		58.3
✓	✓			✓	58.4
✓			✓	✓	57.8
✓	✓	✓			57.9
✓	✓		✓	✓	**59.4**

Table 5. The final mIoU(%) of ablation study about representation compensation module (RC) and pooled cube distillation mechanism on spatial (S-KD) and channel dimension (C-KD). Experiments are conducted on 15-1 overlapped setting on PASCAL VOC 2012. † denotes that the baseline is improved by an adaptive factor [27].

Parallel-Conv	Merge	Frozen	Drop-path	15-1
✓				40.1
✓	✓			42.0
✓	✓	✓		42.8
✓	✓	✓	✓	**43.0**

Table 6. Ablation study of representation compensation module. All experiments are conducted on PASCAL VOC 2012 without pooled cube distillation.

experiments on PASCAL VOC 2012 [30].

As shown in Tab. 5, our proposed representation compensation module achieves about 7% improvement than the MiB [8] baseline. With this module, our method reaches state-of-the-art performance. We argue this performance benefits from the scheme of remembering old knowledge in our method while allowing the learning for new knowledge. In our method, the operations of merging and freezing parameters aim at alleviating the forgetting of old knowledge. Thus, in Tab. 6, we further study the effectiveness of these two operations. Specifically, based on the plain parallel convolution branches (Parallel-Conv), the operations of merging (Merge) and freezing (Frozen) can bring 2.7% improvement. Experimental results demonstrate that the model can benefit from the frozen knowledge in previous steps.

Distillation Mechanism. In Tab. 5, we study the importance of knowledge distillation mechanism on spatial and channel dimensions, respectively. The knowledge distillation

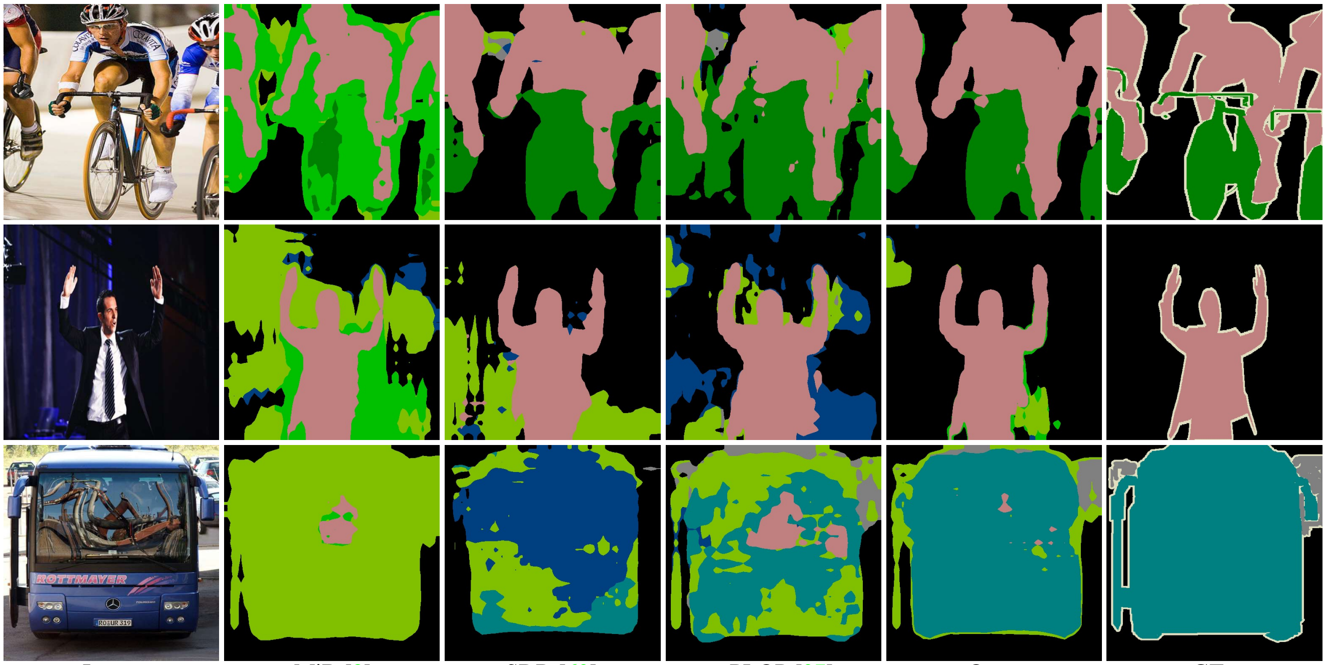


Image MiB [8] SDR [63] PLOP [27] Ours GT

Figure 7. The qualitative comparison between different methods. All the prediction results are from the last step of 15-1 overlapped setting.

W/o Pooling	GAP	Max Pooling	Strip Pooling	Avg. Pooling
52.0	36.1	48.0	54.6	**56.1**

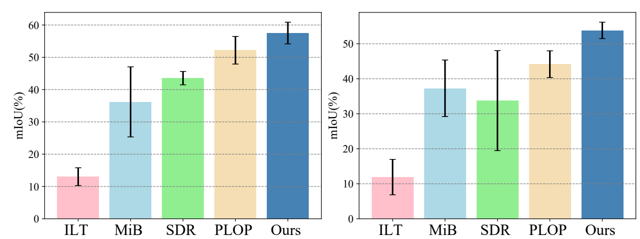
Table 7. Comparison between different pooling methods in distillation mechanism. All experiments are conducted on 15-1 overlapped on PASCAL VOC 2012 using PLOP framework. GAP denotes the global average pooling.

on spatial and channel dimensions achieves similar performance, outperforming baseline by about 15.3% in terms of mIoU. With the representation compensation module, the combination of these two distillation schemes can reach state-of-the-art performance. We further compare the effectiveness of different pooling methods used in the knowledge distillation mechanism, as shown in Tab. 7. Experimental results demonstrate that average pooling outperforms strip pooling by 1.5%.

Robustness to Class Orders. In the scenario of continual semantic segmentation, the class orders in the pipeline is particularly important. To verify the robustness to class orders, we perform experiments on five different class orders, including four random orders and the original ascending order. In Fig. 8, we display the average performance and standard variance for different methods [8, 27, 62, 63]. Experimental results demonstrate that our method is more robust against different class orders than previous methods.

5. Conclusion and Limitation

In this work, aiming at remembering the knowledge for old classes while allowing capacity for learning new classes,



(a) 15-1 overlapped (b) 15-1 disjoint

Figure 8. The average performance and standard variance under different continual learning class orders.

we propose the representation compensation module, which dynamically expands the network without any extra inference cost. Besides, to further alleviate the forgetting for old knowledge, we propose Pooled Cube Distillation mechanism on spatial and channel dimensions. We conduct experiments on two commonly used benchmarks, continual class segmentation and continual domain segmentation. Our method outperforms state-of-the-art performance.

Although we have proposed two components, which outperform the state-of-the-art performance, we have a poor performance in the continual learning process with many steps, like 10-1 setting shown in Tab. 1. In these challenging scenarios, how to improve the performance of the model still has a long way to go. Besides, our method requires more computation costs during training.

Acknowledgment This work is funded by the National Key Research and Development Program of China (NO. 2018AAA0100400) ans NSFC (NO. 61922046), and S&T innovation project from Chinese Ministry of Education.

References

[1] Davide Abati, Jakub Tomczak, Tijmen Blankevoort, Simone Calderara, Rita Cucchiara, and Babak Ehteshami Bejnordi. Conditional channel gated networks for task-aware continual learning. In *IEEE Conf. Comput. Vis. Pattern Recog.*, pages 3931–3940, 2020. 2

[2] Anurag Arnab, Sadeep Jayasumana, Shuai Zheng, and Philip HS Torr. Higher order conditional random fields in deep neural networks. In *Eur. Conf. Comput. Vis.*, 2016. 2

[3] Vijay Badrinarayanan, Alex Kendall, and Roberto Cipolla. Segnet: A deep convolutional encoder-decoder architecture for image segmentation. *IEEE Trans. Pattern Anal. Mach. Intell.*, 39(12):2481–2495, 2017. 2

[4] Jihwan Bang, Heesu Kim, YoungJoon Yoo, Jung-Woo Ha, and Jonghyun Choi. Rainbow memory: Continual learning with a memory of diverse samples. In *IEEE Conf. Comput. Vis. Pattern Recog.*, 2021. 2

[5] Eden Belouadah and Adrian Popescu. Il2m: Class incremental learning with dual memory. In *Int. Conf. Comput. Vis.*, pages 583–592, 2019. 2

[6] Pietro Buzzega, Matteo Boschini, Angelo Porrello, Davide Abati, and Simone Calderara. Dark experience for general continual learning: a strong, simple baseline. In *Adv. Neural Inform. Process. Syst.*, 2020. 2

[7] Nicolas Carion, Francisco Massa, Gabriel Synnaeve, Nicolas Usunier, Alexander Kirillov, and Sergey Zagoruyko. End-to-end object detection with transformers. In *Eur. Conf. Comput. Vis.*, pages 213–229, 2020. 2

[8] Fabio Cermelli, Massimiliano Mancini, Samuel Rota Bulo, Elisa Ricci, and Barbara Caputo. Modeling the background for incremental learning in semantic segmentation. In *IEEE Conf. Comput. Vis. Pattern Recog.*, pages 9233–9242, 2020. 1, 2, 3, 5, 6, 7, 8

[9] Hyuntak Cha, Jaeho Lee, and Jinwoo Shin. Co2l: Contrastive continual learning. In *Int. Conf. Comput. Vis.*, pages 9516–9525, 2021. 2

[10] Arslan Chaudhry, Puneet K Dokania, Thalaiyasingam Ajanthan, and Philip HS Torr. Riemannian walk for incremental learning: Understanding forgetting and intransigence. In *Eur. Conf. Comput. Vis.*, 2018. 2

[11] Arslan Chaudhry, Albert Gordo, Puneet K Dokania, Philip Torr, and David Lopez-Paz. Using hindsight to anchor past knowledge in continual learning. In *The National Conference on Artificial Intelligence (AAAI)*, 2021. 2

[12] Chun-Fu Chen, Quanfu Fan, and Rameswar Panda. Crossvit: Cross-attention multi-scale vision transformer for image classification. In *Int. Conf. Comput. Vis.*, 2021. 2

[13] Liang-Chieh Chen, George Papandreou, Iasonas Kokkinos, Kevin Murphy, and Alan L Yuille. Deeplab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected crfs. *IEEE Trans. Pattern Anal. Mach. Intell.*, 40(4):834–848, 2017. 5

[14] Liang-Chieh Chen, Yi Yang, Jiang Wang, Wei Xu, and Alan L Yuille. Attention to scale: Scale-aware semantic image segmentation. In *IEEE Conf. Comput. Vis. Pattern Recog.*, 2016. 2

[15] Liang-Chieh Chen, Yukun Zhu, George Papandreou, Florian Schroff, and Hartwig Adam. Encoder-decoder with atrous separable convolution for semantic image segmentation. In *Eur. Conf. Comput. Vis.*, 2018. 2

[16] Lin-Zhuo Chen, Zheng Lin, Ziqin Wang, Yong-Liang Yang, and Ming-Ming Cheng. Spatial information guided convolution for real-time rgbd semantic segmentation. *IEEE Trans. Image Process.*, 30:2313–2324, 2021. 1

[17] Yi-Hsin Chen, Wei-Yu Chen, Yu-Ting Chen, Bo-Cheng Tsai, Yu-Chiang Frank Wang, and Min Sun. No more discrimination: Cross city adaptation of road scene segmenters. In *Int. Conf. Comput. Vis.*, pages 1992–2001, 2017. 7

[18] Ali Cheraghian, Shafin Rahman, Pengfei Fang, Soumava Kumar Roy, Lars Petersson, and Mehrtash Harandi. Semantic-aware knowledge distillation for few-shot class-incremental learning. In *IEEE Conf. Comput. Vis. Pattern Recog.*, 2021. 2

[19] Marius Cordts, Mohamed Omran, Sebastian Ramos, Timo Rehfeld, Markus Enzweiler, Rodrigo Benenson, Uwe Franke, Stefan Roth, and Bernt Schiele. The cityscapes dataset for semantic urban scene understanding. In *IEEE Conf. Comput. Vis. Pattern Recog.*, pages 3213–3223, 2016. 5, 6, 7

[20] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *IEEE Conf. Comput. Vis. Pattern Recog.*, pages 248–255, 2009. 5

[21] Prithviraj Dhar, Rajat Vikram Singh, Kuan-Chuan Peng, Ziyan Wu, and Rama Chellappa. Learning without memorizing. In *IEEE Conf. Comput. Vis. Pattern Recog.*, 2019. 2

[22] Henghui Ding, Xudong Jiang, Bing Shuai, Ai Qun Liu, and Gang Wang. Context contrasted feature and gated multi-scale aggregation for scene segmentation. In *IEEE Conf. Comput. Vis. Pattern Recog.*, pages 2393–2402, 2018. 2

[23] Henghui Ding, Xudong Jiang, Bing Shuai, Ai Qun Liu, and Gang Wang. Semantic segmentation with context encoding and multi-path decoding. *IEEE Trans. Image Process.*, 29:3520–3533, 2020. 1

[24] Xiaohan Ding, Yuchen Guo, Guiguang Ding, and Jungong Han. Acnet: Strengthening the kernel skeletons for powerful cnn via asymmetric convolution blocks. In *Int. Conf. Comput. Vis.*, October 2019. 2

[25] Xiaohan Ding, Xiangyu Zhang, Ningning Ma, Jungong Han, Guiguang Ding, and Jian Sun. Repvgg: Making vgg-style convnets great again. In *IEEE Conf. Comput. Vis. Pattern Recog.*, 2021. 2

[26] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, et al. An image is worth 16x16 words: Transformers for image recognition at scale. In *Int. Conf. Learn. Represent.*, 2021. 2

[27] Arthur Douillard, Yifu Chen, Arnaud Dapogny, and Matthieu Cord. Plop: Learning without forgetting for continual semantic segmentation. In *IEEE Conf. Comput. Vis. Pattern Recog.*, 2021. 1, 2, 4, 5, 6, 7, 8

[28] Arthur Douillard, Matthieu Cord, Charles Ollion, Thomas Robert, and Eduardo Valle. Podnet: Pooled outputs distillation for small-tasks incremental learning. In *Eur. Conf. Comput. Vis.*, volume 12365, pages 86–102, 2020. 1, 2

[29] Sayna Ebrahimi, Franziska Meier, Roberto Calandra, Trevor Darrell, and Marcus Rohrbach. Adversarial continual learning. In *Adv. Neural Inform. Process. Syst.*, 2020. 2

[30] M. Everingham, L. Van Gool, C. K. I. Williams, J. Winn, and A. Zisserman. The PASCAL Visual Object Classes Challenge 2012 (VOC2012) Results. http://www.pascal-network.org/challenges/VOC/voc2012/workshop/index.html. 5, 7

[31] Jonas Frey, Hermann Blum, Francesco Milano, Roland Siegwart, and Cesar Cadena. Continual learning of semantic segmentation using complementary 2d-3d data representations. *arXiv preprint arXiv:2111.02156*, 2021. 2

[32] Jun Fu, Jing Liu, Haijie Tian, Yong Li, Yongjun Bao, Zhiwei Fang, and Hanqing Lu. Dual attention network for scene segmentation. In *IEEE Conf. Comput. Vis. Pattern Recog.*, pages 3146–3154, 2019. 2

[33] Shipeng Fu, Zhen Li, Jun Xu, Ming-Ming Cheng, Zitao Liu, and Xiaomin Yang. Interactive knowledge distillation. *arXiv preprint arXiv:2007.01476*, 2020. 3

[34] Bharath Hariharan, Pablo Arbeláez, Ross Girshick, and Jitendra Malik. Hypercolumns for object segmentation and fine-grained localization. In *IEEE Conf. Comput. Vis. Pattern Recog.*, pages 447–456, 2015. 2

[35] Tyler L Hayes, Kushal Kafle, Robik Shrestha, Manoj Acharya, and Christopher Kanan. Remind your neural network to prevent catastrophic forgetting. In *Eur. Conf. Comput. Vis.*, pages 466–483, 2020. 2

[36] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *IEEE Conf. Comput. Vis. Pattern Recog.*, 2016. 3, 5

[37] Seunghoon Hong, Junhyuk Oh, Honglak Lee, and Bohyung Han. Learning transferrable knowledge for semantic segmentation with deep convolutional neural network. In *IEEE Conf. Comput. Vis. Pattern Recog.*, pages 3204–3212, 2016. 2

[38] Qibin Hou, Li Zhang, Ming-Ming Cheng, and Jiashi Feng. Strip pooling: Rethinking spatial pooling for scene parsing. In *IEEE Conf. Comput. Vis. Pattern Recog.*, pages 4003–4012, 2020. 2, 4, 7

[39] Saihui Hou, Xinyu Pan, Chen Change Loy, Zilei Wang, and Dahua Lin. Learning a unified classifier incrementally via rebalancing. In *IEEE Conf. Comput. Vis. Pattern Recog.*, pages 831–839, 2019. 2

[40] Gao Huang, Yu Sun, Zhuang Liu, Daniel Sedra, and Kilian Q Weinberger. Deep networks with stochastic depth. In *Eur. Conf. Comput. Vis.*, pages 646–661, 2016. 3

[41] Zilong Huang, Wentian Hao, Xinggang Wang, Mingyuan Tao, Jianqiang Huang, Wenyu Liu, and Xian-Sheng Hua. Half-real half-fake distillation for class-incremental semantic segmentation. *arXiv preprint arXiv:2104.00875*, 2021. 2

[42] Zilong Huang, Xinggang Wang, Lichao Huang, Chang Huang, Yunchao Wei, and Wenyu Liu. Ccnet: Criss-cross attention for semantic segmentation. In *Int. Conf. Comput. Vis.*, pages 603–612, 2019. 4

[43] Ahmet Iscen, Jeffrey Zhang, Svetlana Lazebnik, and Cordelia Schmid. Memory-efficient incremental learning through feature adaptation. In *Eur. Conf. Comput. Vis.*, pages 699–715, 2020. 1, 2

[44] Sangwon Jung, Hongjoon Ahn, Sungmin Cha, and Taesup Moon. Continual learning with node-importance based adaptive group sparse regularization. In *Adv. Neural Inform. Process. Syst.*, 2020. 2

[45] Menelaos Kanakis, David Bruggemann, Suman Saha, Stamatios Georgoulis, Anton Obukhov, and Luc Van Gool. Reparameterizing convolutions for incremental multi-task learning without task interference. In *Eur. Conf. Comput. Vis.*, pages 689–707, 2020. 1, 2

[46] Chris Dongjoo Kim, Jinseo Jeong, and Gunhee Kim. Imbalanced continual learning with partitioning reservoir sampling. In *Eur. Conf. Comput. Vis.*, pages 411–428, 2020. 2

[47] Chris Dongjoo Kim, Jinseo Jeong, Sangwoo Moon, and Gunhee Kim. Continual learning on noisy data streams via self-purified replay. In *Int. Conf. Comput. Vis.*, pages 537–547, 2021. 2

[48] 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. 1, 2

[49] Vladlen Koltun et al. Efficient inference in fully connected crfs with gaussian edge potentials. In *Adv. Neural Inform. Process. Syst.*, 2011. 2

[50] Xia Li, Zhisheng Zhong, Jianlong Wu, Yibo Yang, Zhouchen Lin, and Hong Liu. Expectation-maximization attention networks for semantic segmentation. In *Int. Conf. Comput. Vis.*, pages 9167–9176, 2019. 2

[51] Zhizhong Li and Derek Hoiem. Learning without forgetting. *IEEE Trans. Pattern Anal. Mach. Intell.*, 40(12):2935–2947, 2017. 1, 2, 6, 7

[52] Di Lin, Yuanfeng Ji, Dani Lischinski, Daniel Cohen-Or, and Hui Huang. Multi-scale context intertwining for semantic segmentation. In *Eur. Conf. Comput. Vis.*, pages 603–619, 2018. 2

[53] Guosheng Lin, Anton Milan, Chunhua Shen, and Ian Reid. Refinenet: Multi-path refinement networks for high-resolution semantic segmentation. In *IEEE Conf. Comput. Vis. Pattern Recog.*, 2017. 2

[54] Qing Liu, Orchid Majumder, Alessandro Achille, Avinash Ravichandran, Rahul Bhotika, and Stefano Soatto. Incremental meta-learning via indirect discriminant alignment. In *Eur. Conf. Comput. Vis.*, 2020. 2

[55] Sifei Liu, Shalini De Mello, Jinwei Gu, Guangyu Zhong, Ming-Hsuan Yang, and Jan Kautz. Learning affinity via spatial propagation networks. In *Adv. Neural Inform. Process. Syst.*, 2017. 2

[56] Yu Liu, Sarah Parisot, Gregory Slabaugh, Xu Jia, Ales Leonardis, and Tinne Tuytelaars. More classifiers, less forgetting: A generic multi-classifier paradigm for incremental

learning. In *Eur. Conf. Comput. Vis.*, pages 699–716, 2020. 1, 2

[57] Yaoyao Liu, Bernt Schiele, and Qianru Sun. Adaptive aggregation networks for class-incremental learning. In *IEEE Conf. Comput. Vis. Pattern Recog.*, 2021. 1, 2

[58] Ze Liu, Yutong Lin, Yue Cao, Han Hu, Yixuan Wei, Zheng Zhang, Stephen Lin, and Baining Guo. Swin transformer: Hierarchical vision transformer using shifted windows. In *Int. Conf. Comput. Vis.*, 2021. 2

[59] Jonathan Long, Evan Shelhamer, and Trevor Darrell. Fully convolutional networks for semantic segmentation. In *IEEE Conf. Comput. Vis. Pattern Recog.*, 2015. 2

[60] Andrea Maracani, Umberto Michieli, Marco Toldo, and Pietro Zanuttigh. Recall: Replay-based continual learning in semantic segmentation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 7026–7035, 2021. 2

[61] Sachin Mehta, Mohammad Rastegari, Anat Caspi, Linda Shapiro, and Hannaneh Hajishirzi. Espnet: Efficient spatial pyramid of dilated convolutions for semantic segmentation. In *Eur. Conf. Comput. Vis.*, pages 552–568, 2018. 2

[62] Umberto Michieli and Pietro Zanuttigh. Incremental learning techniques for semantic segmentation. In *Int. Conf. Comput. Vis. Worksh.*, 2019. 1, 6, 7, 8

[63] Umberto Michieli and Pietro Zanuttigh. Continual semantic segmentation via repulsion-attraction of sparse and disentangled latent representations. In *IEEE Conf. Comput. Vis. Pattern Recog.*, 2021. 1, 2, 5, 6, 8

[64] Yuval Nirkin, Lior Wolf, and Tal Hassner. Hyperseg: Patch-wise hypernetwork for real-time semantic segmentation. In *IEEE Conf. Comput. Vis. Pattern Recog.*, pages 4061–4070, 2021. 1

[65] Hyeonwoo Noh, Seunghoon Hong, and Bohyung Han. Learning deconvolution network for semantic segmentation. In *Int. Conf. Comput. Vis.*, pages 1520–1528, 2015. 2

[66] Pingbo Pan, Siddharth Swaroop, Alexander Immer, Runa Eschenhagen, Richard E Turner, and Mohammad Emtiyaz Khan. Continual deep learning by functional regularisation of memorable past. In *Adv. Neural Inform. Process. Syst.*, 2020. 1, 2

[67] Dongmin Park, Seokil Hong, Bohyung Han, and Kyoung Mu Lee. Continual learning by asymmetric loss approximation with single-side overestimation. In *Int. Conf. Comput. Vis.*, pages 3335–3344, 2019. 1, 2

[68] Chao Peng, Xiangyu Zhang, Gang Yu, Guiming Luo, and Jian Sun. Large kernel matters–improve semantic segmentation by global convolutional network. In *IEEE Conf. Comput. Vis. Pattern Recog.*, pages 4353–4361, 2017. 2

[69] Sylvestre-Alvise Rebuffi, Alexander Kolesnikov, Georg Sperl, and Christoph H Lampert. icarl: Incremental classifier and representation learning. In *IEEE Conf. Comput. Vis. Pattern Recog.*, 2017. 2, 7

[70] Adriana Romero, Nicolas Ballas, Samira Ebrahimi Kahou, Antoine Chassang, Carlo Gatta, and Yoshua Bengio. Fitnets: Hints for thin deep nets. In *Int. Conf. Learn. Represent.*, 2015. 2

[71] Samuel Rota Bulò, Lorenzo Porzi, and Peter Kontschieder. In-place activated batchnorm for memory-optimized training of dnns. In *IEEE Conf. Comput. Vis. Pattern Recog.*, 2018. 5

[72] Mojtaba Seyedhosseini and Tolga Tasdizen. Semantic image segmentation with contextual hierarchical models. *IEEE Trans. Pattern Anal. Mach. Intell.*, 38(5):951–964, 2016. 1

[73] Dongsub Shim, Zheda Mai, Jihwan Jeong, Scott Sanner, Hyunwoo Kim, and Jongseong Jang. Online class-incremental continual learning with adversarial shapley value. In *The National Conference on Artificial Intelligence (AAAI)*, 2021. 2

[74] Christian Simon, Piotr Koniusz, and Mehrtash Harandi. On learning the geodesic path for incremental learning. In *IEEE Conf. Comput. Vis. Pattern Recog.*, 2021. 2

[75] Pravendra Singh, Pratik Mazumder, Piyush Rai, and Vinay P Namboodiri. Rectification-based knowledge retention for continual learning. In *IEEE Conf. Comput. Vis. Pattern Recog.*, pages 15282–15291, 2021. 1, 2

[76] Pravendra Singh, Vinay Kumar Verma, Pratik Mazumder, Lawrence Carin, and Piyush Rai. Calibrating cnns for lifelong learning. In *Adv. Neural Inform. Process. Syst.*, volume 33, 2020. 1, 2

[77] James Smith, Yen-Chang Hsu, Jonathan Balloch, Yilin Shen, Hongxia Jin, and Zsolt Kira. Always be dreaming: A new approach for data-free class-incremental learning. In *Int. Conf. Comput. Vis.*, 2021. 2

[78] Serban Stan and Mohammad Rostami. Unsupervised model adaptation for continual semantic segmentation. In *The National Conference on Artificial Intelligence (AAAI)*, 2022. 2

[79] Robin Strudel, Ricardo Garcia, Ivan Laptev, and Cordelia Schmid. Segmenter: Transformer for semantic segmentation. In *Int. Conf. Comput. Vis.*, 2021. 2

[80] Xiaoyu Tao, Xinyuan Chang, Xiaopeng Hong, Xing Wei, and Yihong Gong. Topology-preserving class-incremental learning. In *Eur. Conf. Comput. Vis.*, pages 254–270, 2020. 1, 2

[81] Zhi Tian, Tong He, Chunhua Shen, and Youliang Yan. Decoders matter for semantic segmentation: Data-dependent decoding enables flexible feature aggregation. In *IEEE Conf. Comput. Vis. Pattern Recog.*, pages 3126–3135, 2019. 2

[82] Vinay Kumar Verma, Kevin J Liang, Nikhil Mehta, Piyush Rai, and Lawrence Carin. Efficient feature transformations for discriminative and generative continual learning. In *IEEE Conf. Comput. Vis. Pattern Recog.*, 2021. 1, 2

[83] Eli Verwimp, Matthias De Lange, and Tinne Tuytelaars. Rehearsal revealed: The limits and merits of revisiting samples in continual learning. In *Int. Conf. Comput. Vis.*, pages 9385–9394, 2021. 2

[84] Wenhai Wang, Enze Xie, Xiang Li, Deng-Ping Fan, Kaitao Song, Ding Liang, Tong Lu, Ping Luo, and Ling Shao. Pyramid vision transformer: A versatile backbone for dense prediction without convolutions. In *Int. Conf. Comput. Vis.*, 2021. 2

[85] Xiaolong Wang, Ross Girshick, Abhinav Gupta, and Kaiming He. Non-local neural networks. In *IEEE Conf. Comput. Vis. Pattern Recog.*, pages 7794–7803, 2018. 2

[86] Yuqing Wang, Zhaoliang Xu, Xinlong Wang, Chunhua Shen, Baoshan Cheng, Hao Shen, and Huaxia Xia. End-to-end video instance segmentation with transformers. In *IEEE Conf. Comput. Vis. Pattern Recog.*, 2021. 2

[87] Guile Wu, Shaogang Gong, and Pan Li. Striking a balance between stability and plasticity for class-incremental learning. In *Int. Conf. Comput. Vis.*, pages 1124–1133, 2021. 2

[88] Ye Xiang, Ying Fu, Pan Ji, and Hua Huang. Incremental learning using conditional adversarial networks. In *Int. Conf. Comput. Vis.*, pages 6619–6628, 2019. 2

[89] Enze Xie, Wenhai Wang, Zhiding Yu, Anima Anandkumar3, Jose Alvarez, and Ping Luo. Segformer: Simple and efficient design for semantic segmentation with transformers. In *Int. Conf. Comput. Vis.*, 2021. 2

[90] Canwen Xu, Wangchunshu Zhou, Tao Ge, Furu Wei, and Ming Zhou. Bert-of-theseus: Compressing bert by progressive module replacing. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing*, pages 7859–7869, 2020. 3

[91] Shipeng Yan, Jiangwei Xie, and Xuming He. Der: Dynamically expandable representation for class incremental learning. In *IEEE Conf. Comput. Vis. Pattern Recog.*, 2021. 1, 2

[92] Shipeng Yan, Jiale Zhou, Jiangwei Xie, Songyang Zhang, and Xuming He. An em framework for online incremental learning of semantic segmentation. In *ACM Int. Conf. Multimedia*, 2021. 2

[93] Kailun Yang, Xinxin Hu, and Rainer Stiefelhagen. Is context-aware cnn ready for the surroundings? panoramic semantic segmentation in the wild. *IEEE Trans. Image Process.*, 30:1866–1881, 2021. 1

[94] Maoke Yang, Kun Yu, Chi Zhang, Zhiwei Li, and Kuiyuan Yang. Denseaspp for semantic segmentation in street scenes. In *IEEE Conf. Comput. Vis. Pattern Recog.*, pages 3684–3692, 2018. 2

[95] Lu Yu, Bartlomiej Twardowski, Xialei Liu, Luis Herranz, Kai Wang, Yongmei Cheng, Shangling Jui, and Joost van de Weijer. Semantic drift compensation for class-incremental learning. In *IEEE Conf. Comput. Vis. Pattern Recog.*, pages 6982–6991, 2020. 1, 2

[96] Mohsen Zand, Shyamala Doraisamy, Alfian Abdul Halin, and Mas Rina Mustaffa. Ontology-based semantic image segmentation using mixture models and multiple crfs. *IEEE Trans. Image Process.*, 25(7):3233–3248, 2016. 1

[97] Yanhong Zeng, Jianlong Fu, and Hongyang Chao. Learning joint spatial-temporal transformations for video inpainting. In *Eur. Conf. Comput. Vis.*, pages 528–543, 2020. 2

[98] Friedemann Zenke, Ben Poole, and Surya Ganguli. Continual learning through synaptic intelligence. In *Int. Conf. Mach. Learn.*, 2017. 2

[99] Chi Zhang, Nan Song, Guosheng Lin, Yun Zheng, Pan Pan, and Yinghui Xu. Few-shot incremental learning with continually evolved classifiers. In *IEEE Conf. Comput. Vis. Pattern Recog.*, 2021. 2

[100] Chang-Bin Zhang, Peng-Tao Jiang, Qibin Hou, Yunchao Wei, Qi Han, Zhen Li, and Ming-Ming Cheng. Delving deep into label smoothing. *IEEE Trans. Image Process.*, 2021. 2

[101] Dong Zhang, Hanwang Zhang, Jinhui Tang, Meng Wang, Xiansheng Hua, and Qianru Sun. Feature pyramid transformer. In *Eur. Conf. Comput. Vis.*, pages 323–339, 2020. 2

[102] Bowen Zhao, Xi Xiao, Guojun Gan, Bin Zhang, and Shu-Tao Xia. Maintaining discrimination and fairness in class incremental learning. In *IEEE Conf. Comput. Vis. Pattern Recog.*, pages 13208–13217, 2020. 2

[103] Shuai Zheng, Sadeep Jayasumana, Bernardino Romera-Paredes, Vibhav Vineet, Zhizhong Su, Dalong Du, Chang Huang, and Philip HS Torr. Conditional random fields as recurrent neural networks. In *Int. Conf. Comput. Vis.*, 2015. 2

[104] Sixiao Zheng, Jiachen Lu, Hengshuang Zhao, Xiatian Zhu, Zekun Luo, Yabiao Wang, Yanwei Fu, Jianfeng Feng, Tao Xiang, Philip HS Torr, et al. Rethinking semantic segmentation from a sequence-to-sequence perspective with transformers. In *IEEE Conf. Comput. Vis. Pattern Recog.*, 2021. 2

[105] Bolei Zhou, Hang Zhao, Xavier Puig, Sanja Fidler, Adela Barriuso, and Antonio Torralba. Scene parsing through ade20k dataset. In *IEEE Conf. Comput. Vis. Pattern Recog.*, 2017. 5, 6

[106] Fei Zhu, Xu-Yao Zhang, Chuang Wang, Fei Yin, and Cheng-Lin Liu. Prototype augmentation and self-supervision for incremental learning. In *IEEE Conf. Comput. Vis. Pattern Recog.*, pages 5871–5880, 2021. 2

[107] Kai Zhu, Yang Cao, Wei Zhai, Jie Cheng, and Zheng-Jun Zha. Self-promoted prototype refinement for few-shot class-incremental learning. In *IEEE Conf. Comput. Vis. Pattern Recog.*, pages 6801–6810, 2021. 2

[108] Lanyun Zhu, Deyi Ji, Shiping Zhu, Weihao Gan, Wei Wu, and Junjie Yan. Learning statistical texture for semantic segmentation. In *IEEE Conf. Comput. Vis. Pattern Recog.*, pages 12537–12546, 2021. 1

[109] Xizhou Zhu, Weijie Su, Lewei Lu, Bin Li, Xiaogang Wang, and Jifeng Dai. Deformable detr: Deformable transformers for end-to-end object detection. In *Int. Conf. Learn. Represent.*, 2021. 2