Cs^{2}K: Class-specific and Class-shared Knowledge Guidance for Incremental Semantic Segmentation

Wei Cong[1,2,3], Yang Cong[4,*], Yuyang Liu[5], and Gan Sun[4]

[1] State Key Laboratory of Robotics, Shenyang Institute of Automation, Chinese Academy of Sciences, Shenyang 110016, China
[2] Institutes for Robotics and Intelligent Manufacturing, Chinese Academy of Sciences, Shenyang 110169, China
[3] University of Chinese Academy of Sciences, Beijing 100049, China
[4] College of Automation Science and Engineering, South China University of Technology, Guangzhou 510640, China
[5] Peking University
{congwei45, congyang81, sungan1412}@gmail.com, liuyuyang13@pku.edu.cn

Abstract. Incremental semantic segmentation endeavors to segment newly encountered classes while maintaining knowledge of old classes. However, existing methods either 1) lack guidance from class-specific knowledge (*i.e.,* old class prototypes), leading to a bias towards new classes, or 2) constrain class-shared knowledge (*i.e.,* old model weights) excessively without discrimination, resulting in a preference for old classes. In this paper, to trade off model performance, we propose the **C**lass-**s**pecific and **C**lass-**s**hared **K**nowledge (**Cs^{2}K**) guidance for incremental semantic segmentation. Specifically, from the class-specific knowledge aspect, we design a prototype-guided pseudo labeling that exploits feature proximity from prototypes to correct pseudo labels, thereby overcoming catastrophic forgetting. Meanwhile, we develop a prototype-guided class adaptation that aligns class distribution across datasets via learning old augmented prototypes. Moreover, from the class-shared knowledge aspect, we propose a weight-guided selective consolidation to strengthen old memory while maintaining new memory by integrating old and new model weights based on weight importance relative to old classes. Experiments on public datasets demonstrate that our proposed Cs^{2}K significantly improves segmentation performance and is plug-and-play.

Keywords: Incremental learning · Semantic segmentation · Class-specific knowledge · Class-shared knowledge

1 Introduction

Semantic segmentation [19, 29], a fundamental task within the realm of computer vision [32], involves categorizing each pixel in an image to its class. Recent advancements [8, 37] have significantly enhanced the performance of semantic

* The corresponding author is Prof. Yang Cong.

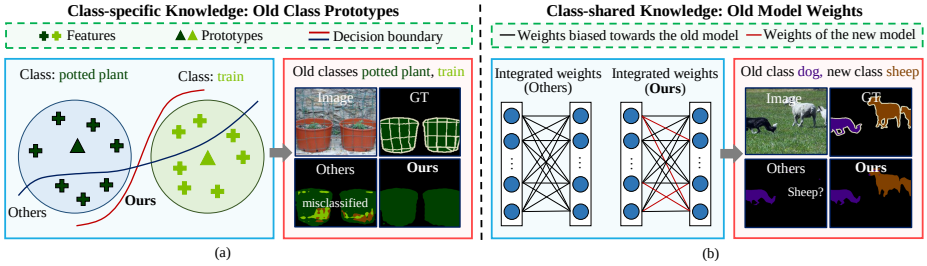

Fig. 1: Illustration of challenges for ISS. (a) The decision boundary between old classes `potted plant` and `train` undergoes a dramatic change without the guidance from old class prototypes. GT means ground truth. (b) Other methods integrate old and new model weights without discrimination, leading to the integrated model weights biased towards old model weights (remember `dog` but not recognize `sheep`).

segmentation models, contributing to their widespread use in various applications [2, 17]. However, adapting these models to new data streams or handling evolving classes poses challenges, as they tend to overfit new classes quickly and forget old classes during finetuning. This phenomenon is commonly known as catastrophic forgetting [26]. Incremental semantic segmentation (ISS) [4, 15, 27] emerges as a crucial solution to address catastrophic forgetting, focusing on maintaining knowledge about previous classes while efficiently incorporating knowledge from novel classes. We divide the stored knowledge for old classes in ISS into three forms: old class exemplars, old class features, and old model weights. Amidst the growing concerns over data privacy [12], we focus on the class-specific knowledge (*i.e.,* old class prototypes which are the average of old class features) and the class-shared knowledge (*i.e.,* old model weights) in exemplar-free methods [4, 15, 27, 28, 36, 41].

In ISS, the training dataset in one step only comprises images containing pixels belonging to the corresponding foreground classes, leading to a significantly higher proportion of these classes compared to training datasets in other steps. The discrepancies in the class distribution of different training datasets cause the overrepresentation of new classes. This phenomenon results in a dramatic change in the decision boundary, exacerbating catastrophic forgetting. As depicted in Fig. 1 (a), the segmentation model at step t misclassifies the old class `potted plant` as `train` and biases towards the new class `sheep`. However, most current methods [4,15,27,28,36,41] solely rely on class-shared knowledge (*i.e.,* old model weights), which only provides limited prevention against the average forgetting of previous classes without adapting to class distribution discrepancies. In contrast, we focus on crucial class-specific knowledge (*i.e.,* old class prototypes) as shown in Fig. 1 (a), which is a compact representation of the corresponding class distribution. On the one hand, current methods [10, 15, 39] are unable to adapt to disturbances in class distribution and generate noisy pseudo labels for background (the background pixels in ISS contain the future classes, the previous classes, and the true background), thus failing to adjust the decision boundary. In

this paper, we leverage the old class prototypes to correct the noisy pseudo labels. Specifically, we develop a prototype-guided pseudo labeling, which reweights the pseudo-label likelihoods assigned by the previous model, taking into account the proximity of features to prototypes. Then it corrects misclassified pixels and generates high-quality pseudo labels **from the class-specific knowledge aspect**. On the other hand, current methods [5, 25, 30, 44] preserving old samples to correct the decision boundary lack representative samples and leakage data privacy [12]. To address these limitations, we design a prototype-guided class adaptation to augment old class prototypes via self-prototype augmentation and inter-prototype augmentation. Subsequently, the augmented old class prototypes are jointly trained with new classes to maintain discriminability between old and new classes **from the class-specific knowledge aspect**. The proposed prototype-guided pseudo labeling and prototype-guided class adaptation eliminate the limitations of solely relying on class-shared knowledge by fusing class-specific knowledge, thus mitigating overrepresentation of new classes.

Some methods [15, 27, 28, 30] exploit regularization of class-shared knowledge (*i.e.,* old model weights) to overcome catastrophic forgetting but yield limited gain since only the representations are constrained to be consistent. Other methods [36, 41] integrate the weights of the old and new model without discrimination. As illustrated in Fig. 1 (b), these approaches cause the integrated model weights biased towards the old model weights, leading to remembering the old class `dog` but not recognizing the new class `sheep`. To address this issue, we introduce a weight-guided selective consolidation to simultaneously learn new classes and memorize old classes. As depicted in Fig. 1 (b), it calculates the importance of model weights for old classes based on Fisher information, then selects to integrate these important weights of the old and new models. Our weight-guided selective consolidation overcomes catastrophic forgetting while preserving new knowledge **from the class-shared knowledge aspect**.

In summary, the key contributions are:

- We propose the **C**lass-**s**pecific and **C**lass-**s**hared **K**nowledge (**Cs^{2}K**) guidance model, which is an early exploration of considering both class-specific and class-shared knowledge to surmount ISS.
- To alleviate forgetting of old classes from the class-specific knowledge aspect, we introduce a prototype-guided pseudo labeling and a prototype-guided class adaptation to adapt to class distribution discrepancies.
- To prevent underfitting of new classes from the class-shared knowledge aspect, we design a weight-guided selective consolidation, which selectively integrates only the crucial weights from the old model, pertaining to the old classes, into the new model to obtain obvious segmentation performance gain.

2 Related Works

2.1 Incremental Learning

Incremental learning [33], a pivotal area in machine learning, endeavors to enable models to adapt to new classes while avoiding catastrophic forgetting [26]

of previously acquired knowledge. Various strategies have been proposed in this domain, encompassing structural-based methods [24, 38] that dynamically expand the model architecture to accommodate new classes, regularization-based methods [1, 11, 20, 21, 34, 40] employing constraints like knowledge distillation to maintain consistency of old classes, and rehearsal-based methods [14, 22, 33, 44] storing or generating old samples to participate in training alongside new samples. These diverse approaches collectively aim to empower the model to incrementally acquire new knowledge while preserving previous knowledge. In this paper, we focus on challenging ISS.

2.2 Knowledge-Guided Incremental Semantic Segmentation

ISS [3, 4, 15, 27] explores to gradually adapt the segmentation model to new classes. The stored knowledge for current ISS methods mainly comprises class-specific knowledge composed of old class exemplars and old class features, as well as class-shared knowledge represented by old model weights. However, storing previous exemplars is space-intensive and privacy-insecure. Therefore, we focus on exemplar-free ISS methods. The ISS methods of storing class-specific knowledge help the model better distinguish between classes. ALIFE [30] leverages distillation of old class features, while Incrementer [35] introduces tokens for new classes. In contrast, ISS methods that store class-shared knowledge help the model overcome the average forgetting of old classes. MiB [4] addresses semantic drift by modeling potential classes. PLOP [15] employs feature distillation with a multi-scale scheme. RCIL [41] introduces average-pooling-based distillation to overcome strip pooling drawbacks. EWF [36] integrates the old and new model containing the old and new knowledge, respectively. Additionally, a series of methods [5, 42] introduce additional auxiliaries, making a comparison with other methods unfair. However, thoroughly combining class-specific and class-shared knowledge remains to be explored. Our method stands out by synergizing these two aspects to enhance ISS performance.

3 Preliminaries

ISS sequentially learns a model $\mathcal{M}^t$ at $t \in \{0 \dots T\}$ steps, where $\mathcal{M}^t$ consisted of a feature extractor Ψ^t and a classifier Φ^t is over parameters Θ^t at step t. Each step t involves only one dataset $\mathcal{D}^t$ consisting of input images $\boldsymbol{x}^t$ and their corresponding ground truth (GT) $\boldsymbol{y}^t$. The training GT $\boldsymbol{y}^t$ of the dataset $\mathcal{D}^t$ contains the foreground classes C^t and the background c^{bg}. It is important to highlight that the foreground classes across steps are mutually exclusive, *i.e.,* $C^i \cap C^j = \emptyset$. The ISS model continuously encountering new classes without revisiting old ones brings about the issue of catastrophic forgetting. Additionally, pixels corresponding to future, previous, and true background classes are all labeled as the background c^{bg}, which exacerbates catastrophic forgetting. The objective of ISS is to achieve precise segmentation for all encountered classes throughout the incremental learning process.

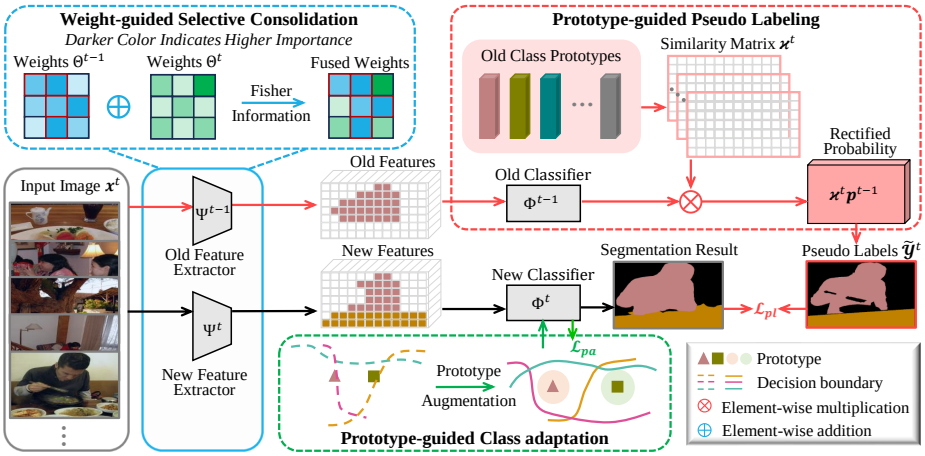

Fig. 2: Overview of our Cs^{2}K model. It updates model parameters with the proposed prototype-guided pseudo labeling and prototype-guided class adaptation from the class-specific knowledge aspect. Then, the old and new model weights are selectively integrated via the weight-guided selective consolidation to trade off performance between old and new classes from the class-shared knowledge aspect.

4 Method

The overview of our Cs^{2}K model is illustrated in Fig. 2. Our Cs^{2}K model updates parameters to overcome catastrophic forgetting via the prototype-guided pseudo labeling in Sec. 4.1 and the prototype-guided class adaptation in Sec. 4.2 from the class-specific knowledge aspect. Then the weight-guided selective consolidation in Sec. 4.3 is proposed to better distinguish between classes from the class-shared knowledge aspect. The overall framework is presented in Sec. 4.4.

4.1 Prototype-guided Pseudo Labeling

The discrepancies in class distribution within the training datasets of different steps cause the overrepresentation of new classes, resulting in significant changes of the decision boundary. This change leads to noisy pseudo labels [15] of the background, posing a challenge since precise pseudo labels are vital for refining the decision boundary. They effectively consolidate pixels for previous classes while accommodating current class pixels. In this paper, we develop the prototype-guided pseudo labeling, as illustrated in Fig. 2, which attempts to adapt old class prototypes to correct misclassifications in pseudo labels from the class-specific knowledge aspect. We choose old class prototypes to produce high-quality pseudo labels due to the following two reasons: 1) The prototypes are not sensitive to outliers that are minority; 2) The prototypes treat classes with different occurrence frequencies equally in semantic segmentation. At step

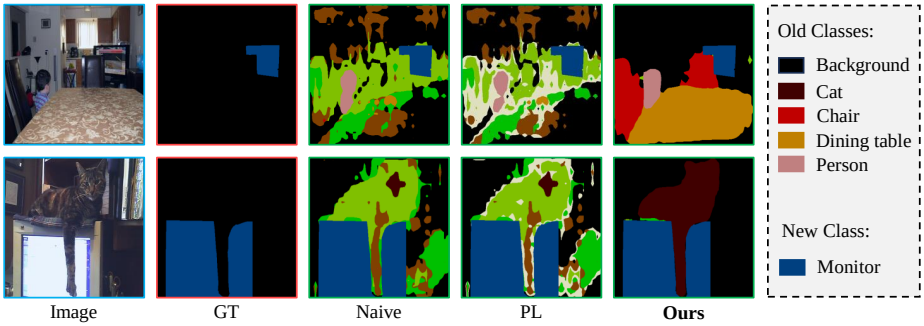

Fig. 3: The visualization comparison between pseudo labelss on Pascal VOC 2012 [16].

$t-1$, the calculation for the prototype $\boldsymbol{\eta}_c$ of the old class c is formulated as:

$$\boldsymbol{\eta}_c = \frac{\sum_{\boldsymbol{x}^{t-1}\in\mathcal{D}^{t-1}} \sum_i \Psi^{t-1}(x_i^{t-1}) * \mathbb{1}(y_{i,c}^{t-1} == 1)}{\sum_{\boldsymbol{x}^{t-1}\in\mathcal{D}^{t-1}} \sum_i \mathbb{1}(y_{i,c}^{t-1} == 1)}, \tag{1}$$

where x_i^{t-1} is the i-th pixel in the image $\boldsymbol{x}^{t-1}$ belonging to the dataset $\mathcal{D}^{t-1}$ at step $t-1$. $\Psi^{t-1}(x_i^{t-1})$ denotes the features of x_i^{t-1} via the feature extractor Ψ^{t-1}. $y_{i,c}^{t-1}$ represents that the GT of the pixel x_i^{t-1} is the c-th class. $\mathbb{1}(\cdot)$ denotes the indicator function, which gets 1 when the condition is true and 0 otherwise. The prototype $\boldsymbol{\eta}_c$ is the average of pixel features for the old class c, serving as compact representations of the corresponding class distribution. It is worth noting that we recalculate the prototype of the background at each step t, as its features are continuously changing. Instead of directly using old class prototypes for classification [33], we try to correct pseudo labels for the previous class c in background adopting the similarity weight $\kappa_{i,c}^t$ according to the old class prototype $\boldsymbol{\eta}_c$. Specifically, the similarity weight $\kappa_{i,c}^t$ at step t that exploits feature proximity between the pixel x_i^t and the old class prototype $\boldsymbol{\eta}_c$ is obtained as:

$$\kappa_{i,c}^t = \frac{\exp(-||\Psi^{t-1}(x_i^t) - \boldsymbol{\eta}_c||/\tau)}{\sum_{c'\in(C^{0:t-1}\cup c^{bg})} \exp\big(-||\Psi^{t-1}(x_i^t) - \boldsymbol{\eta}_{c'}||/\tau\big)}, \tag{2}$$

where $\tau = 1$ is the temperature. c' represents any previously seen old class. The similarity weight $\kappa_{i,c}^t$ reflects the confidence that the pixel x_i^t belongs to the c-th class. It adapts to the disturbances of class distribution when generating pseudo labels. The formulation of the pseudo label $\tilde{y}_i^t$ for the pixel x_i^t obtained by our proposed prototype-guided pseudo labeling is as follows:

$$\tilde{y}_i^t = \begin{cases} y_i^t & \text{if } y_i^t > 0 \\ \arg\max \boldsymbol{\kappa}_i^t \boldsymbol{p}_i^{t-1} & \text{if } y_i^t = 0 \text{ and } \arg\max \boldsymbol{\kappa}_i^t \boldsymbol{p}_i^{t-1} > 0 \\ 0 & \text{otherwise,} \end{cases} \tag{3}$$

where y_i^t is the GT of the pixel x_i^t in the image $\boldsymbol{x}^t$. $\boldsymbol{\kappa}_i^t$ represents the similarity matrix between the pixel x_i^t and old class prototypes. $\boldsymbol{p}_i^{t-1}$ denotes the softmax

probability for the pixel x_i^t via the model $\mathcal{M}^{t-1}$. When the GT $y_i^t > 0$, it indicates that the pixel x_i^t belongs to the new classes. Thus, we directly assign y_i^t to the pseudo label $\tilde{y}_i^t$. If the GT y_i^t is 0 (*i.e.,* background) and the rectified probability $\boldsymbol{\kappa}_i^t \boldsymbol{p}_i^{t-1}$ considers the pixel as an old class, the pseudo label $\tilde{y}_i^t$ is equal to the old class (*i.e.,* $\arg\max \boldsymbol{\kappa}_i^t \boldsymbol{p}_i^{t-1}$). Otherwise, when the GT y_i^t and the rectified probability $\boldsymbol{\kappa}_i^t \boldsymbol{p}_i^{t-1}$ both consider the pixel as 0 (*i.e.,* background), we assign 0 to the pseudo label $\tilde{y}_i^t$. The prototypes have the capability to rectify misclassified pseudo labels near the decision boundary, given that the distance from pixels near the decision boundary to the corresponding prototype is much closer than to other prototypes. As shown in Fig. 3, the proposed prototype-guided pseudo labeling guides ISS model learning by generating high-quality pseudo labels compared with Naive and pseudo-labeling strategy (PL) [15]. Naive means choosing the channel with the highest old probability as the pseudo label. PL [15] generates pseudo labels by adopting median entropy as the measurement.

Finally, we update the ISS model using the loss $\mathcal{L}_{\text{pl}}$ based on the pseudo label $\tilde{\boldsymbol{y}}^t$ of $\boldsymbol{x}^t$, which is formulated as follows:

$$\mathcal{L}_{\text{pl}} = -\frac{1}{|\mathcal{D}^t|} \sum_{\boldsymbol{x}^t \in \mathcal{D}^t} \mathcal{L}_{ce}(\boldsymbol{p}^t, \tilde{\boldsymbol{y}}^t), \tag{4}$$

where $\mathcal{L}_{ce}(\cdot)$ is the function of cross entropy. $\boldsymbol{p}^t$ is the softmax probability of $\boldsymbol{x}^t$ by the model $\mathcal{M}^t$. $|\mathcal{D}^t|$ denotes the number of samples in the dataset $\mathcal{D}^t$.

4.2 Prototype-guided Class Adaptation

Apart from accurate pseudo labels, replay techniques [5, 33] have been proven effective to reduce changes to the decision boundary resulting from class distribution discrepancies. Current replay strategies [5,30] in ISS select old samples that lack representativeness and leak data privacy to participate in the training. In contrast, we in this paper replay the representative old class prototypes to maintain a well-separated decision boundary from the class-specific knowledge aspect. As shown in Fig. 2, we design the prototype-guided class adaptation to optimize the shape of the decision boundary via augmented prototypes, thereby making it more adaptable to the complex distribution between different classes. Specifically, we perform prototype augmentation of the old prototype $\boldsymbol{\eta}_c$ in Eq. (1) through self-prototype augmentation and inter-prototype augmentation motivated by data augmentation [9, 23]. Given the considerable shift presented in background pixels, we abstain from prototype augmentation on the background. The augmented prototypes Γ_c via self-prototype augmentation is calculated by:

$$\Gamma_c = \boldsymbol{\eta}_c + \boldsymbol{\mu} * s^t, \tag{5}$$

where $\boldsymbol{\mu} \sim \mathcal{N}(0, 1)$ is the Gaussian distribution with the same dimension as the prototype $\boldsymbol{\eta}_c$. s^t represents the scaling factor at step t, which is as follows:

$$s^t = \begin{cases} \sigma^{t-1} & \text{if } t = 1 \\ \frac{|C^{t-1}| * \sigma^{t-1} + \sum_{m=0}^{t-2} |C^m| * \sigma^{t-2}}{\sum_{m=0}^{t-1} |C^m|} & \text{if } t > 1, \end{cases} \tag{6}$$

where σ^t is the standard deviation for the features of classes C^t at step t. $|C^{t-1}|$ denotes the number of classes at step $t-1$. The scaling factor s^t is a dynamic parameter that changes with step t, which can adaptively fit the class distribution to augment prototypes. self-prototype augmentation enhances the capability of the ISS model to thoroughly explore the feature space, mitigating the risk of being trapped in local optima.

Then the augmented prototypes Π_c of $\boldsymbol{\eta}_c$ via inter-prototype augmentation is formulated as the following:

$$\Pi_c = \lambda * \boldsymbol{\eta}_c + (1-\lambda) * \boldsymbol{\eta}_{c'}, s.t.\ \ c', c \in C^{0:t-1}, c' \neq c, \tag{7}$$

where $\lambda \sim U(0,1)$ is a random value from a uniform distribution. Performing inter-prototype augmentation can adapt to class distribution discrepancies, fostering a more balanced acquisition of distinctive features across diverse classes.

Subsequently, we update the ISS model using the loss $\mathcal{L}_{pa}$, which incorporates the augmented prototypes Γ_c and Π_c into the classifier Φ^t:

$$\mathcal{L}_{pa} = \frac{\sum_c \Big(\mathcal{L}_{ce}\big(\Phi^t(\Gamma_c), y_c\big) + \lambda * \mathcal{L}_{ce}\big(\Phi^t(\Pi_c), y_c\big) + (1-\lambda) * \mathcal{L}_{ce}\big(\Phi^t(\Pi_c), y_{c'}\big) \Big)}{\sum_{m=0}^{t-1} |C^m| \\ s.t.\ \ c', c \in C^{0:t-1}, c' \neq c,} \tag{8}$$

where $\Phi^t(\Gamma_c)$ and $\Phi^t(\Pi_c)$ represent the probabilities of the augmented prototypes Γ_c and Π_c via the classifier $\Phi^t(\cdot)$ at step t, respectively. y_c denotes the GT of the corresponding augmented prototypes Γ_c and Π_c.

4.3 Weight-guided Selective Consolidation

To address the challenge of class imbalance, which often favors new classes, existing methods [36, 41] tend to overly restrict the old model weights, resulting in the preference for old classes. In this paper, we propose the weight-guided selective consolidation (as depicted in Fig. 2) to selectively merge the old and new model weights based on weight importance for old classes, which effectively learns new classes while preserving the memory of previously learned classes from the class-shared knowledge aspect. In specific, the weight importance $\boldsymbol{F}^{t-1}$ is quantified by the Fisher information [31] of corresponding gradients at step $t-1$. After learning the step t, the formulation of fusing the old and new model weights based on the weight importance $\boldsymbol{F}^{t-1}$ for old classes is as follows:

$$\Theta_i^t = \begin{cases} \omega * \Theta_i^{t-1} + (1-\omega) * \Theta_i^t & \text{if } F_i^{t-1} > \text{TopK}(\boldsymbol{F}^{t-1}, \beta |\boldsymbol{F}^{t-1}|) \\ \Theta_i^t & \text{otherwise,} \end{cases} \tag{9}$$

where F_i^{t-1} represents the importance of the i-th old model weight for old classes. $|\boldsymbol{F}^{t-1}|$ denotes the number of weights in the model $\mathcal{M}^{t-1}$. TopK$(\boldsymbol{F}^{t-1}, \beta|\boldsymbol{F}^{t-1}|)$ represents the $\beta|\boldsymbol{F}^{t-1}|$-th largest value in $\boldsymbol{F}^{t-1}$. Θ_i^{t-1} denotes the weights of the model $\mathcal{M}^{t-1}$, containing the discriminative information for old classes. Θ_i^t denotes the weights of the model $\mathcal{M}^t$, which is regarded as the best container for new classes. The selection of the number of important old model weights (*i.e.,* β) and the strength of constraints applied to these weights (*i.e.,* ω) are crucial factors affecting the final model performance. Specifically, β serves as the threshold to distinguish weight importance, and it is closely linked to the disparity in the quantity of classes acquired at step t compared to those learned previously. Hence, the calculation of β is designed as:

$$\beta = \left(1 + \exp\left(\frac{|C^t| - \sum_{m=0}^{t-1} |C^m| - 1}{\sum_{m=0}^{t} |C^m| + 1}\right)\right)^{-1}. \tag{10}$$

ω denotes the balance factor that governs the trade-off between the performance of the new and old classes, which is more associated with the ratio of the number of classes learned at step t to the total number of classes encountered over time. The formulation of ω is as follows:

$$\omega = 1 - \left(\frac{|C^t|}{\sum_{m=0}^{t} |C^m| + 1}\right)^{\frac{1}{2}}. \tag{11}$$

β and ω are dynamic factors, which can be automatically adjusted in various steps and scenarios. Our weight-guided selective consolidation constrains the essential model weights for old classes to overcome catastrophic forgetting, while recognizing new classes by retaining the remaining new model weights.

4.4 Overall Framework

The ISS model is updated continually with the proposed prototype-guided pseudo labeling and the prototype-guided class adaptation from the class-specific knowledge aspect when learning new classes. The overall loss $\mathcal{L}$ is formulated as follows:

$$\mathcal{L} = \mathcal{L}_{pl} + \mathcal{L}_{pa}. \tag{12}$$

After learning new classes, the old and new model weights are selectively integrated via the weight-guided selective consolidation from the class-shared knowledge aspect. Combining the above techniques, our Cs^{2}K model effectively learns new classes without forgetting previously learned ones.

Table 1: Comparison on Pascal VOC 2012 [16]. **Red** highlights the highest results.

Method	15-1 (6 steps)			10-1 (11 steps)			5-3 (6 steps)		
	0-15	16-20	all	0-10	11-20	all	0-5	6-20	all
FT [7]	0.2	1.8	0.6	6.3	1.1	3.8	11.8	5.2	7.1
Joint [7]	79.5	74.0	78.2	79.0	77.3	78.2	78.0	78.3	78.2
LWF [21]	6.0	3.9	5.5	8.0	2.0	4.8	20.9	36.7	24.7
ILT [27]	9.6	7.8	9.2	7.2	3.7	5.5	22.5	31.7	29.0
SDR [28]	47.3	14.7	39.5	32.4	17.1	25.1	-	-	-
RCIL [41]	70.6	23.7	59.4	55.4	15.1	34.3	63.1	34.6	42.8
GSC [10]	72.1	24.4	60.8	50.6	17.3	34.7	32.7	30.1	30.9
MiB [4]	38.0	13.5	32.2	12.2	13.1	12.6	57.1	42.5	46.7
MiB+EWF [36]	78.0	25.5	65.5	56.0	16.7	37.3	69.0	45.0	51.8
MiB+Cs^{2}K (**Ours**)	76.2	41.8	68.0	43.0	35.2	39.3	70.6	50.4	**56.2**
PLOP [15]	65.1	21.1	54.6	44.0	15.5	30.5	25.7	30.0	28.7
PLOP+EWF [36]	77.7	32.7	67.0	71.5	30.3	51.9	61.7	42.2	47.7
PLOP+Cs^{2}K (**Ours**)	77.9	46.4	**70.4**	74.4	47.2	**61.5**	58.4	53.4	54.8

5 Experiments

5.1 Experimental Setups

Evaluation Protocols. The ISS training process is typically divided into T steps, with each step representing an individual task, and the labeled classes within each step are disjoint. We adhere to the widely-used *overlapped* setting, as adopted in previous works [6,36]. This choice stems from the acknowledgment that within the current training, the background class contains both old and future classes. Following previous methods [4, 15, 36], we perform experiments on two public datasets: PASCAL VOC 2012 [16] and ADE20K [43]. The former comprises 20 distinct classes and the background class, while the latter consists of 150 classes. We evaluate the effectiveness under 15-1, 10-1, and 5-3 scenarios on Pascal VOC 2012 [16]. Additionally, we conduct experiments on ADE20K [43] under 100-10 and 100-5 scenarios. The X-Y scenario indicates learning X classes in the first step, followed by learning Y classes in the subsequent steps. At each step, we only access the current data. Moreover, we adopt mIoU as the metric.
Implementation Details. Following popular works [4,15,36], our architecture utilizes Deeplab-v3 [7] with a ResNet-101 [18] pre-trained on ImageNet [13]. We ensure that details such as learning rate, batch size, optimizer, and dataset processing remain consistent with previous methods [15,36]. Due to the commonality in the first step across all methods, we reuse the weights acquired during this phase. We conduct experiments on four NVIDIA RTX 3090 GPUs.

5.2 Comparisons

In our comparative analysis, we benchmark our Cs^{2}K model with the classic continual learning method LWF [21] and several ISS algorithms ILT [27], MiB [4], PLOP [15], SDR [28], RCIL [41], GSC [10], and EWF [36]. Notably, EWF [36] is applied to MiB [4] and PLOP [15] following the original paper. FT [7] serves

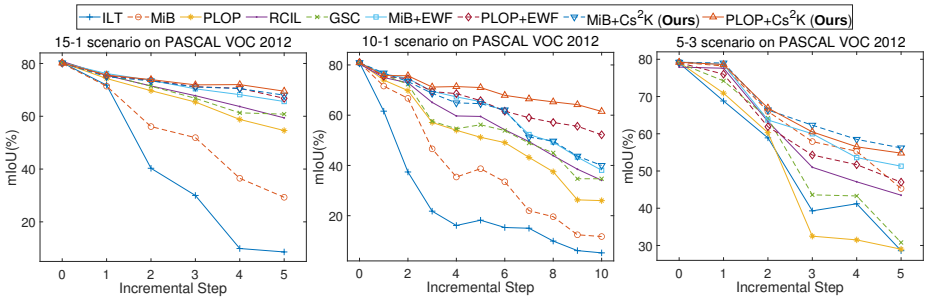

Fig. 4: Quantitative comparison at each step with different methods for 15-1, 10-1, and 5-3 class incremental segmentation scenarios on Pascal VOC 2012 [16].

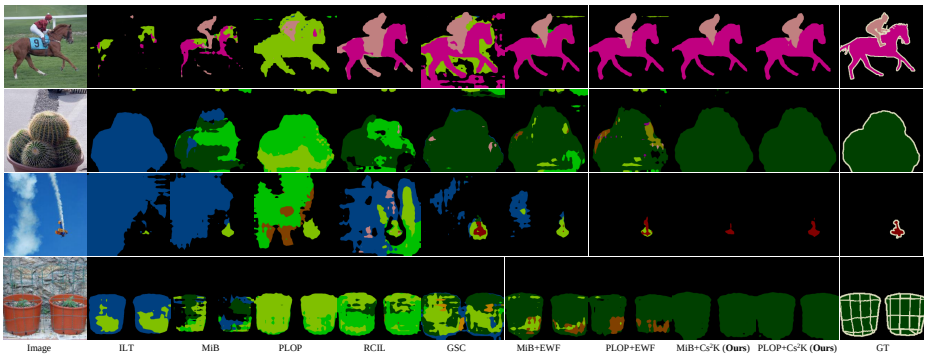

Fig. 5: The visualization comparison from the last step on Pascal VOC 2012 [16].

as a lower bound, training only on the newly encountered data. Joint [7] trains on all seen classes, which is an upper bound. Additionally, our proposed Cs^{2}K is plug-and-play and we apply it to MiB [4] and PLOP [15] for evaluation.

Pascal VOC 2012. Tab. 1 presents the comparison results of challenging 15-1, 10-1 and 5-3 scenarios on PASCAL VOC 2012 [16]. We observe that our method surpasses MiB [4] and PLOP [15] by a substantial margin in all scenarios, achieving notable mIoU gains of 35.8% and 31.0% respectively in the 15-1 and 10-1 scenarios. Besides, comparing with advanced methods still significantly highlights our advantages. Specifically, our method surpasses the latest advancements PLOP+EWF [36] and MiB+EWF [36] by 9.6% and 4.4% mIoU in the 10-1 and 5-3 scenarios, respectively. This underscores the effectiveness of correcting decision boundaries for better distinguishing between classes from the class-specific knowledge aspect. Furthermore, our method maintains comparable performance on previous classes and achieves excellent performance on current classes. Compared to PLOP+EWF [36], we observe substantial mIoU improvements of 13.7%, 16.9%, and 11.2% on new classes in the 15-1, 10-1, and 5-3 scenarios, respectively. It is attributed to the discriminative integration of weights between the old and new model, effectively preserving new knowledge

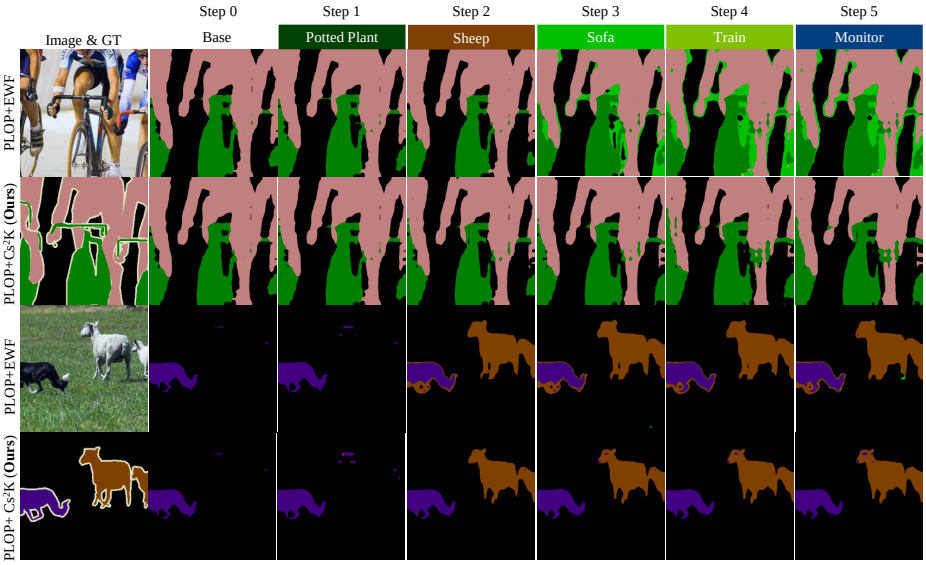

Fig. 6: The visualization comparison across steps on Pascal VOC 2012 [16].

Table 2: Comparison on ADE20K [43]. **Red** highlights the highest results.

Method	100-10 (6 steps)							100-5 (11 steps)		
	1-100	101-110	111-120	121-130	131-140	141-150	all	1-100	101-150	all
Joint [7]	44.3	26.1	42.8	26.7	28.1	17.3	38.9	44.3	28.2	38.9
ILT [27]	0.1	0.0	0.1	0.9	4.1	9.3	1.1	0.1	1.3	0.5
PLOP [15]	40.6	15.2	16.9	18.7	11.9	7.9	31.6	39.1	7.8	28.7
RCIL [41]	39.3	14.6	26.3	23.2	12.1	11.8	32.1	38.5	11.5	29.6
GSC [10]	40.8	14.3	24.6	22.2	15.2	11.7	32.6	39.5	11.2	30.2
MiB [4]	38.3	12.6	10.6	8.7	9.5	15.1	29.2	36.0	5.6	25.9
MiB+EWF [36]	41.5	12.8	22.5	23.2	14.4	8.8	33.2	41.4	13.4	32.1
MiB+Cs²K (**Ours**)	42.4	11.6	29.4	22.8	14.5	7.7	**34.1**	41.9	18.4	**34.2**

from the class-shared knowledge aspect. The comparison at each step for 15-1, 10-1, and 5-3 scenarios is shown in Fig. 4. Our method consistently maintains a leading performance, especially in the challenging 10-1 scenario. This illustrates our robustness in combining the class-specific and class-shared knowledge.

Visualization. Fig. 5 visually showcases the results of the final step in the 15-1 scenario. Unlike previous methods that display varying degrees of misclassification for old classes, our method excels in producing accurate segmentation. This intuitively indicates that our proposed method consistently exhibits outstanding performance. Additionally, Fig. 6 presents the visualization results across steps in the 15-1 scenario. Both methods generate the same visualization results as there is no distinction at the first step. However, MiB+EWF [36] rapidly forgets previous classes, showing a bias towards new classes. In contrast, our Cs²K ex-

Table 3: Ablation study of different pseudo label strategies on PASCAL VOC 2012 [16]. **Red** highlights the highest results.

Pseudo label Strategy	Step										
	0	1	2	3	4	5	6	7	8	9	10
Naive	80.9	75.8	75.5	71.2	71.4	69.9	67.8	65.7	63.9	61.5	58.3
PL [15]	80.9	75.5	75.5	71.1	71.3	71.0	66.3	60.9	57.9	57.2	54.0
Cs^{2}K (**Ours**)	80.9	75.8	75.7	71.2	71.4	71.0	67.9	66.5	65.3	64.2	**61.5**

Table 4: Ablation study of weight integration strategies on PASCAL VOC 2012 [16]. **Red** highlights the highest results.

Fusion Strategy	Step										
	0	1	2	3	4	5	6	7	8	9	10
WF [36]	80.9	75.7	75.6	71.1	71.0	69.7	67.6	66.1	64.7	63.0	60.3
Cs^{2}K (**Ours**)	80.9	75.8	75.7	71.2	71.4	71.0	67.9	66.5	65.3	64.2	**61.5**

hibits greater stability. It is attributed to our prototype-guided pseudo labeling, prototype-guided class adaptation, and weight-guided selective consolidation.

ADE20K. We further validate our method on ADE20K [43]. Experiments are conducted on the most challenging scenarios, 100-10 and 100-5, while discarding the less meaningful 100-50 scenario. Our method, as depicted in Tab. 2, consistently surpasses all other competing methods across all scenarios on ADE20K [43]. For instance, our method attains a 2.1% mIoU increase compared to MiB+EWF [36] in the 100-5 scenario. This underscores the robustness and generalizability of our proposed method on the more realistic dataset.

5.3 Ablation Study

Pseudo Label Strategy. To validate the effectiveness of our prototype-guided pseudo labeling, we compare it with Naive and the pseudo-labeling strategy (PL) [15]. Naive determines the pseudo label by adopting the channel with the highest old probability. Fig. 3 presents the visualization results. Naive generates the noisiest pseudo labels, while PL removes some uncertain pixels based on the median entropy. When the model exhibits strong performance, removing uncertain pseudo labels leads to poor results. In contrast, our method, guided by the correction of representative prototypes, consistently produces accurate pseudo labels. Quantitative results in Tab. 3 align with the visualization.

Weight Integration Strategy. We compare our weight-guided selective consolidation with the weight fusion (WF) [36] which equally constrains all old model weights to overcome catastrophic forgetting. However, as shown in Tab. 4, WF [36] yields suboptimal results. In contrast, our method attains superior performance, underscoring the effectiveness of selectively integrating crucial previous model weights.

Ablation Study on Proposed Components of Cs^{2}K. To further assess the impact of the introduced prototype-guided pseudo labeling (PPL), prototype-

Table 5: Ablation study of the 15-1 scenario. **Red** highlights the highest results.

Settings	Variants				15-1 Scenario		
	PPL	PCA-SA	PCA-IA	WSC	0-15	16-20	all
Ours-w/o PPL	✗	✓	✓	✓	70.4	49.0	65.3
Ours-w/o PCA	✓	✗	✗	✓	78.5	37.5	68.7
Ours-w/o PCA-SA	✓	✗	✓	✓	77.3	42.2	69.0
Ours-w/o PCA-IA	✓	✓	✗	✓	77.8	43.7	69.7
Ours-w/o WSC	✓	✓	✓	✗	58.0	18.8	48.6
Cs^2K (**Ours**)	✓	✓	✓	✓	77.9	46.4	**70.4**

guided class adaptation (PCA), and weight-guided selective consolidation (WSC), we perform the ablation study in the 15-1 scenario on PASCAL VOC 2012 [16]. We refer to PCA with only self-prototype augmentation as PCA-SA, and PCA with only inter-prototype augmentation as PCA-IA. Tab. 5 reveals that our proposed PPL can improve performance by 5.1% mIoU, indicating that prototype correction can generate high-quality pseudo labels. Our PCA achieves 1.7% mIoU gain with effective prototype augmentation. This illustrates its ability to adapt to class distribution discrepancies and distinguish between old and new classes. Additionally, our proposed WSC improves the performance by 21.8% mIoU, highlighting the necessity of selectively consolidating important old knowledge for ISS. In summary, each component in our Cs^2K has been proven effective, and their simultaneous utilization contributes to the overall superior performance.

6 Conclusion

In this paper, we mitigate catastrophic forgetting of old classes and underfitting of new classes caused by neglecting the class-specific knowledge and equally treating the class-shared knowledge. We propose the **C**lass-**s**pecific and **C**lass-**s**hared **K**nowledge (**Cs^2K**) guidance to surmount ISS. The prototype-guided pseudo labeling and prototype-guided class adaptation are designed to adapt to class distribution discrepancies from the class-specific knowledge aspect. Then the weight-guided selective consolidation is proposed to distinguish between classes from the class-shared knowledge aspect. Our effectiveness is rigorously validated through extensive experiments on public benchmark datasets. While acknowledging a performance gap compared to joint training in long sequence tasks, we emphasize that this serves as a foundation for future endeavors, where we aim to further investigate and bridge this gap.

Acknowledgments

This work is supported by National Key R&D Program of China (2023YFB4704800), and National Nature Science Foundation of China under Grant (62225310, 62127807).

References

1. Aljundi, R., Babiloni, F., Elhoseiny, M., Rohrbach, M., Tuytelaars, T.: Memory aware synapses: Learning what (not) to forget. In: Eur. Conf. Comput. Vis. (2018)
2. Asgari Taghanaki, S., Abhishek, K., Cohen, J.P., Cohen-Adad, J.: Deep semantic segmentation of natural and medical images: a review. Artif. Intell. Rev. **54**(1), 137–178 (2021)
3. Baek, D., Oh, Y., Lee, S., Lee, J., Ham, B.: Decomposed knowledge distillation for class-incremental semantic segmentation. In: Adv. Neural Inform. Process. Syst. (2022)
4. Cermelli, F., Mancini, M., Bulo, S.R., Ricci, E., Caputo, B.: Modeling the background for incremental learning in semantic segmentation. In: IEEE Conf. Comput. Vis. Pattern Recog. (2020)
5. Cha, S., kim, b., Yoo, Y., Moon, T.: Ssul: Semantic segmentation with unknown label for exemplar-based class-incremental learning. In: Adv. Neural Inform. Process. Syst. vol. 34, pp. 10919–10930 (2021)
6. Chen, J., Cong, R., Yuxuan, L., Ip, H., Kwong, S.: Saving 100x storage: Prototype replay for reconstructing training sample distribution in class-incremental semantic segmentation. In: Adv. Neural Inform. Process. Syst. (2023)
7. Chen, L.C., Papandreou, G., Kokkinos, I., Murphy, K., Yuille, A.L.: Deeplab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected crfs. IEEE Trans. Pattern Anal. Mach. Intell. **40**(4), 834–848 (2018)
8. Cheng, B., Misra, I., Schwing, A.G., Kirillov, A., Girdhar, R.: Masked-attention mask transformer for universal image segmentation. In: IEEE Conf. Comput. Vis. Pattern Recog. pp. 1290–1299 (2022)
9. Chou, H.P., Chang, S.C., Pan, J.Y., Wei, W., Juan, D.C.: Remix: rebalanced mixup. In: Eur. Conf. Comput. Vis. pp. 95–110 (2020)
10. Cong, W., Cong, Y., Dong, J., Sun, G., Ding, H.: Gradient-semantic compensation for incremental semantic segmentation. IEEE Trans. Multimedia (2023)
11. Cong, W., Cong, Y., Sun, G., Liu, Y., Dong, J.: Self-paced weight consolidation for continual learning. IEEE Trans. Circuits Syst. Video Technol. (2023)
12. De Lange, M., Aljundi, R., Masana, M., Parisot, S., Jia, X., Leonardis, Leonardis, A., Slabaugh, G., Tuytelaars, T.: A continual learning survey: Defying forgetting in classification tasks. IEEE Trans. Pattern Anal. Mach. Intell. **44**(7), 3366–3385 (2022)
13. Deng, J., Dong, W., Socher, R., Li, L.J., Li, K., Fei-Fei, L.: Imagenet: A large-scale hierarchical image database. In: IEEE Conf. Comput. Vis. Pattern Recog. pp. 248–255 (2009)
14. Dong, J., Wang, L., Fang, Z., Sun, G., Xu, S., Wang, X., Zhu, Q.: Federated class-incremental learning. In: IEEE Conf. Comput. Vis. Pattern Recog. pp. 10164–10173 (2022)
15. Douillard, A., Chen, Y., Dapogny, A., Cord, M.: Plop: Learning without forgetting for continual semantic segmentation. In: IEEE Conf. Comput. Vis. Pattern Recog. pp. 4040–4050 (2021)
16. Everingham, M., Eslami, S., Van Gool, L., Williams, C.K., Winn, J., Zisserman, A.: The pascal visual object classes challenge: A retrospective. Int. J. Comput. Vis. **111**(1), 98–136 (2015)
17. Feng, D., Haase-Schütz, C., Rosenbaum, L., Hertlein, H., Gläser, C., Timm, F., Wiesbeck, W.: Deep multi-modal object detection and semantic segmentation for autonomous driving: Datasets, methods, and challenges. IEEE trans. Intell. Transp. Syst. **22**(3), 1341–1360 (2021)

18. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. In: IEEE Conf. Comput. Vis. Pattern Recog. (2016)
19. Huang, Z., Jiang, B., Liu, Y.: A few-shot semantic segmentation method based on adaptively mining correlation network. Robotica **41**(6), 1828–1836 (2023)
20. Kirkpatrick, J., Pascanu, R., Rabinowitz, N., Veness, J., Desjardins, G., Rusu, A.A., Milan, K., Quan, J., Ramalho, T., Grabska-Barwinska, A., et al.: Overcoming catastrophic forgetting in neural networks. Proc. Natl. Acad. Sci. U.S.A.. **114**(13), 3521–3526 (2017)
21. Li, Z., Hoiem, D.: Learning without forgetting. IEEE Trans. Pattern Anal. Mach. Intell. **40**(12), 2935–2947 (2018)
22. Lopez-Paz, D., Ranzato, M.A.: Gradient episodic memory for continual learning. In: Adv. Neural Inform. Process. Syst. vol. 30. Curran Associates, Inc. (2017)
23. Malepathirana, T., Senanayake, D., Halgamuge, S.: Napa-vq: Neighborhood-aware prototype augmentation with vector quantization for continual learning. In: IEEE Conf. Comput. Vis. Pattern Recog. pp. 11674–11684 (2023)
24. Mallya, A., Lazebnik, S.: Packnet: Adding multiple tasks to a single network by iterative pruning. In: IEEE Conf. Comput. Vis. Pattern Recog. (2018)
25. Maracani, A., Michieli, U., Toldo, M., Zanuttigh, P.: Recall: Replay-based continual learning in semantic segmentation. In: IEEE Conf. Comput. Vis. Pattern Recog. pp. 7026–7035 (2021)
26. McCloskey, M., Cohen, N.J.: Catastrophic interference in connectionist networks: The sequential learning problem. Psychol. Learn. Motiv. **24**, 109–165 (1989)
27. Michieli, U., Zanuttigh, P.: Incremental learning techniques for semantic segmentation. In: Int. Conf. Comput. Vis. (2019)
28. Michieli, U., Zanuttigh, P.: Continual semantic segmentation via repulsion-attraction of sparse and disentangled latent representations. In: IEEE Conf. Comput. Vis. Pattern Recog. pp. 1114–1124 (2021)
29. Minaee, S., Boykov, Y., Porikli, F., Plaza, A., Kehtarnavaz, N., Terzopoulos, D.: Image segmentation using deep learning: A survey. IEEE Trans. Pattern Anal. Mach. Intell. **44**(7), 3523–3542 (2022)
30. Oh, Y., Baek, D., Ham, B.: Alife: Adaptive logit regularizer and feature replay for incremental semantic segmentation. In: Adv. Neural Inform. Process. Syst. (2022)
31. Pascanu, R., Bengio, Y.: Revisiting natural gradient for deep networks. arXiv preprint arXiv:1301.3584 (2013)
32. Qu, M., Wu, Y., Wei, Y., Liu, W., Liang, X., Zhao, Y.: Learning to segment every referring object point by point. In: IEEE Conf. Comput. Vis. Pattern Recog. pp. 3021–3030 (2023)
33. Rebuffi, S.A., Kolesnikov, A., Sperl, G., Lampert, C.H.: icarl: Incremental classifier and representation learning. In: IEEE Conf. Comput. Vis. Pattern Recog. (2017)
34. Schwarz, J., Czarnecki, W., Luketina, J., Grabska-Barwinska, A., Teh, Y.W., Pascanu, R., Hadsell, R.: Progress & compress: A scalable framework for continual learning. In: Proc. Int. Conf. Mach. Learn. pp. 4528–4537. PMLR (2018)
35. Shang, C., Li, H., Meng, F., Wu, Q., Qiu, H., Wang, L.: Incrementer: Transformer for class-incremental semantic segmentation with knowledge distillation focusing on old class. In: IEEE Conf. Comput. Vis. Pattern Recog. pp. 7214–7224 (2023)
36. Xiao, J.W., Zhang, C.B., Feng, J., Liu, X., van de Weijer, J., Cheng, M.M.: Endpoints weight fusion for class incremental semantic segmentation. In: IEEE Conf. Comput. Vis. Pattern Recog. pp. 7204–7213 (2023)
37. Xie, E., Wang, W., Yu, Z., Anandkumar, A., Alvarez, J.M., Luo, P.: Segformer: Simple and efficient design for semantic segmentation with transformers. In: Adv. Neural Inform. Process. Syst. vol. 34, pp. 12077–12090 (2021)

38. Yoon, J., Yang, E., Lee, J., Hwang, S.J.: Lifelong learning with dynamically expandable networks. In: Int. Conf. Learn. Represent. (2018)
39. Yu, L., Liu, X., Van de Weijer, J.: Self-training for class-incremental semantic segmentation. IEEE Trans. Neural Networks Learn. Syst. (2022)
40. Zenke, F., Poole, B., Ganguli, S.: Continual learning through synaptic intelligence. In: Proc. Int. Conf. Mach. Learn. pp. 3987–3995 (2017)
41. Zhang, C.B., Xiao, J.W., Liu, X., Chen, Y.C., Cheng, M.M.: Representation compensation networks for continual semantic segmentation. In: IEEE Conf. Comput. Vis. Pattern Recog. pp. 7053–7064 (2022)
42. Zhang, Z., Gao, G., Fang, Z., Jiao, J., Wei, Y.: Mining unseen classes via regional objectness: A simple baseline for incremental segmentation. In: Adv. Neural Inform. Process. Syst. (2022)
43. Zhou, B., Zhao, H., Puig, X., Fidler, S., Barriuso, A., Torralba, A.: Scene parsing through ade20k dataset. In: IEEE Conf. Comput. Vis. Pattern Recog. pp. 633–641 (2017)
44. Zhu, L., Chen, T., Yin, J., See, S., Liu, J.: Continual semantic segmentation with automatic memory sample selection. In: IEEE Conf. Comput. Vis. Pattern Recog. pp. 3082–3092 (2023)