CoinSeg: Contrast Inter- and Intra- Class Representations for Incremental Segmentation

Zekang Zhang[1]*, Guangyu Gao[1]†, Jianbo Jiao[2], Chi Harold Liu[1], Yunchao Wei[3,4]

[1] School of Computer Science, Beijing Institute of Technology
[2] School of Computer Science, University of Birmingham
[3] WEI Lab, Institute of Information Science, Beijing Jiaotong University
[4] Beijing Key Laboratory of Advanced Information Science and Network
zkzhang1998@outlook.com

Abstract

*Class incremental semantic segmentation aims to strike a balance between the model's stability and plasticity by maintaining old knowledge while adapting to new concepts. However, most state-of-the-art methods use the freeze strategy for stability, which compromises the model's plasticity. In contrast, releasing parameter training for plasticity could lead to the best performance for all categories, but this requires discriminative feature representation. Therefore, we prioritize the model's plasticity and propose the **Co**ntrast **in**ter- and **in**tra-class representations for Incremental **Seg**mentation (**CoinSeg**), which pursues discriminative representations for flexible parameter tuning. Inspired by the Gaussian mixture model that samples from a mixture of Gaussian distributions, CoinSeg emphasizes intra-class diversity with multiple contrastive representation centroids. Specifically, we use mask proposals to identify regions with strong objectness that are likely to be diverse instances/centroids of a category. These mask proposals are then used for contrastive representations to reinforce intra-class diversity. Meanwhile, to avoid bias from intra-class diversity, we also apply category-level pseudo-labels to enhance category-level consistency and inter-category diversity. Additionally, CoinSeg ensures the model's stability and alleviates forgetting through a specific flexible tuning strategy. We validate CoinSeg on Pascal VOC 2012 and ADE20K datasets with multiple incremental scenarios and achieve superior results compared to previous state-of-the-art methods, especially in more challenging and realistic long-term scenarios. Code is available at https://github.com/zkzhang98/CoinSeg.*

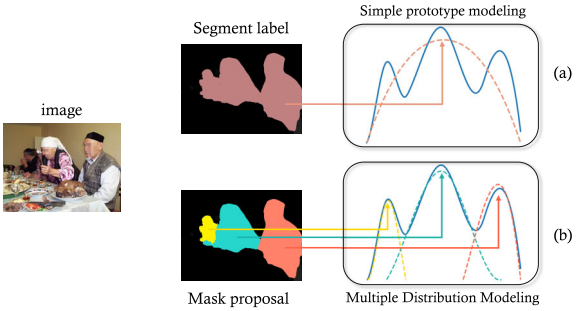


Figure 1. Comparison of Simple Prototype Modeling and Multiple Distribution Modeling. (a): Simple prototype modeling to represent a category with global average pooling; (b): Multiple Gussian distribution modeling to represent a category with its regional objectness, which is discovered with the guidance of mask proposals.

1. Introduction

In recent years, deep learning based methods have achieved satisfactory performance on various recognition tasks, with the assumption of fixed or stable data distribution [17]. However, real-world data is typically a continuous stream with an unstable distribution, making it difficult for models to retain old knowledge while acquiring new concepts, known as *catastrophic forgetting* [3, 24, 30]. To tackle this problem, *incremental learning* is proposed to adapt to changing data streams for new concepts, but also avoid forgetting old knowledge, especially for the classification task, *i.e.*, Class-Incremental Learning (CIL) [16, 18, 46].

Class Incremental Semantic Segmentation(CISS) aims to assign an image with the pixel-wise label of the CIL setting. In semantic segmentation tasks involving dense predictions, the problem of catastrophic forgetting typically becomes more challenging. Most recent works [4, 5, 12] have strug-

*Work done during an intern at WEI Lab of Beijing Jiaotong University.
†Corresponding author. *guangyugao@bit.edu.cn.*

gled to alleviate this problem, and the freeze strategy [5] (*i.e.*, freezing most of the parameters during all the incremental steps after learning base classes) was shown to be the most efficient way in the state-of-the-art CISS methods [5, 49].

However, while the freeze strategy effectively alleviates catastrophic forgetting, it is a compromise for the model's plasticity, meaning the model becomes hard to adapt to new classes. Furthermore, when considering the *lifelong learning* setting with an infinite number of novel classes, the plasticity of the incremental learner will be crucial. Therefore, ideally, the optimal solution should be fine-tuning all the parameters for new classes, while catastrophic forgetting needs to be handled properly, rather than a freeze strategy.

Thus, to address the above-mentioned issues in CISS, especially the limitations of the freeze strategy, we prioritize the model's plasticity by a flexible parameters tuning strategy, and pursue the more discriminative feature representation for the balance to stability. An intuitive idea for discriminative representation is to compute prototypes (category centroids) for each category and apply contrastive learning to improve the diversity among categories [26, 44]. However, as mentioned in the Gaussian Mixture Model (GMM) [33], the natural samples from the same category should come from the mixture of multiple Gaussian distributions. Furthermore, several prior works [28, 29, 33, 43] have also claimed that the representation of categories in the feature space should be multiple activations, as shown in Fig. 1.

To this end, we propose the **Co**ntrast **in**ter- and **in**tra-class representations for Incremental **Seg**mentation (**CoinSeg**), by pursuing discriminative representations. Although the idea of adapting contrastive learning to CISS is intuitive, it is worth studying and critical to choose the appropriate areas, to contrast inter- and intra-class representations for incremental segmentation. Firstly, the CoinSeg emphasizes intra-class diversity with multiple contrastive representation centroids. Specifically, we identify regions with strong objectness using mask proposals, which are more likely to be instances/centroids of a particular category, as shown in Fig. 1 (b). We contrast these regional objectness to reinforce intra-class diversity and robust representation of the model. In order to mitigate the potential bias from intra-class diversity, we incorporated category-level pseudo-labels to augment category-level consistency and inter-category diversity. Meanwhile, we apply Swin Transformer [19] to better extract the local representation of the samples. Additionally, CoinSeg ensures the model's stability and alleviates forgetting through a specific Flexible Tuning strategy. In this strategy, a better balance between plasticity and stability is achieved by designing an initial learning rate schedule and regularization terms.

Finally, the CoinSeg outperforms prior methods in multiple benchmarks, especially in realistic and hard long-term scenarios VOC 10-1 and 2-2, where our approaches show significant performance gains of 6.7% and 17.8%, comparing with previous state-of-the-art, respectively.

2. Related work

2.1. Class Incremental Learning

Class incremental learning (CIL) is incremental learning focusing on defying catastrophic forgetting in classification tasks. Replay-based approaches were proposed to store a sampler of historical data [10, 20, 32, 34, 38], which can be used for future training to prevent forgetting. The historical data can also be obtained with web search engine [23] or generation models [27, 39, 41]. Another intuitive thought to tackle CIL task is based on parameter isolation [2, 22, 35, 36, 37, 46]. Parameter isolation methods assign dedicated model parameters for each specific task, while bringing the continually increasing number of parameters with task increases. Regularization-based methods, such as knowledge distillation [1, 15, 16, 51, 14] and restricting model training with networks trained at previous tasks [18, 31, 47], are also effective in tackling the catastrophic forgetting problem in incremental learning. These methods allow the model to transfer knowledge from previous tasks to new tasks, which can help prevent forgetting.

2.2. Class Incremental Semantic Segmentation

Recently, there is a growing interest in the field of incremental learning for semantic segmentation(*i.e.*, Class incremental semantic segmentation, CISS), and researchers are proposing various approaches to tackle CISS. Modeling-the-Background (MiB) [4] first remodeling the background (dummy label) in the ground truth, and designs a distillation-based framework in CISS. Douillard et al. [12] proposed the approach of PLOP to define a pseudo label for CISS and proposes a local distillation method as an extended constraint based on MiB. The SSUL [5] first introduces replay-based approaches to the CISS task and maintains a memory bank of historical samples for future model training. Besides, the SSUL prevents the model from forgetting by freezing model parameters. RCIL [45] proposes a dual-branch architecture, in which one is freeze and the other is trainable, and introduces a channel-wise feature distillation loss. MicroSeg [49] introduces mask proposals to CISS and further clarifies image labels to tackle background shifts.

2.3. Vision Transformer

Transformers for computer vision (*i.e.*, Vision Transformer) have attracted more and more attention. ViT [11] is the first widely known vision transformer, which transposes transformer directly to image classification tasks, achieving comparable performances with CNN-based methods. Since then, researchers have explored modifications and optimizations to ViT, proposing numerous designs [9, 40] for Vision

Transformers that differ from those used in NLP. Swin Transformer [19] proposes shifted window-based self-attention, bringing greater efficiency for vision transformer. Other works have explored the use of transformers for image segmentation. Segformer [42] consists of hierarchically structured transformer layers to deal with semantic segmentation. Mask2Former [7] and MaskFormer [8] propose universal transformer-based architectures for semantic, instance and panoptic segmentation.

3. Method

3.1. Task Definition

We define the task of Class Incremental Semantic Segmentation (CISS) according to the common definition in previous works [4, 5, 12, 26, 45]. CISS is composed of a series of incremental *learning steps*, as $t = 1, \dots, T$. Each learning step has a sub-dataset $\boldsymbol{\mathcal{D}}^t$ and a corresponding class set of samples $\boldsymbol{\mathcal{C}}^t$. For any pair $(\boldsymbol{x}^t, \boldsymbol{y}^t)$ within $\boldsymbol{\mathcal{D}}^t$, $\boldsymbol{x}^t$ and $\boldsymbol{y}^t$ denote the input image and its ground-truth mask, respectively. We follow the definition of classes set learned in CISS as in MicroSeg [49]: In each learning step of class incremental semantic segmentation, the current classes set can be represented as the union of the classes to be learned, denoted by $\boldsymbol{\mathcal{C}}^t$, and a special class c_u denotes "areas does not belong to current foreground classes". From the perspective of each current learning step, the class c_u can be interpreted as the background class, and its composition varies across different learning steps.

The goal of the CISS model $f_{t,\boldsymbol{\theta}}$ with parameters $\boldsymbol{\theta}$ at the t_{th} learning step is to assign a probability to each class for every pixel in $\boldsymbol{x}^t$. The CISS model $f_{t,\boldsymbol{\theta}}$ is a composite of a feature extractor $g_{t,\boldsymbol{\theta}_1}$ and a classifier $h_{t,\boldsymbol{\theta}_2}$, which classifies each category in the union of $\boldsymbol{\mathcal{C}}^{1:t} = \bigcup_{i=1}^{t} \boldsymbol{\mathcal{C}}^i$ and the unseen class c_u (i.e., $\mathbb{C}^t = \boldsymbol{\mathcal{C}}^{1:t} \cup c_u$). After the t-th learning step, the model also needs to provide a prediction for *all seen classes* $\mathbb{C}^t$. The prediction of CISS with model $f_{t,\boldsymbol{\theta}}$ can be expressed as $\hat{\boldsymbol{y}}_t = \arg\max_{c \in \mathbb{C}^t} f^c_{t,\boldsymbol{\theta}}(\boldsymbol{x})$.

3.2. Contrast inter- and intra-class Representations

Previous studies [5, 49] have highlighted the advantages of the freeze strategy in CISS. However, while parameter freezing can be beneficial, it may also restrict the model's plasticity and block further exploration of CISS. Unlike previous methods, our approach prioritizes model plasticity and allows fine-tuning to address this concern. With the application of fine-tuning, it is possible to train the model to obtain more robust and discriminative representations. Thus, we designed two contrastive losses for enhancing the ability of model representation learning to improve the model plasticity, *i.e.*, pseudo label-guided representation learning for inter-class diversity and mask proposal-guided representation learning for intra-class diversity, as depicted in Fig. 2.

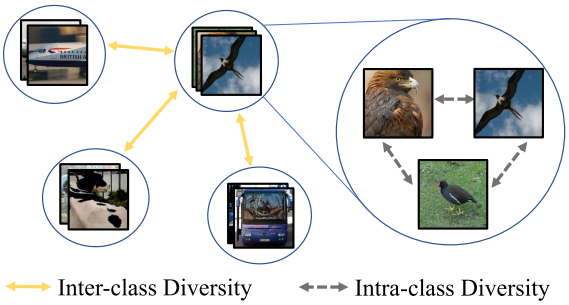


Figure 2. Illustration of the inter- & intra-class diversity. Only the contrastive learning with class "bird" is shown for example.

3.2.1 Contrast inter-class representations

We apply pseudo-label-guided representation learning to enhance inter-class diversity. The approach also highlights category-level consistency to avoid bias from intra-class diversity, which introduces in Sec. 3.2.2.

Pseudo labels. Due to the limitation of data acquisition in CISS, the current ground truth $\boldsymbol{y}^t$ is only annotated with classes at the current learning step, *i.e.*, $\boldsymbol{\mathcal{C}}^t$. So we need to extend the available ground truth $\boldsymbol{y}^t$ to include predictions from f_{t-1}, creating a more informative label for contrastive learning. Specifically, we get the label prediction $\hat{\boldsymbol{y}}^{t-1} \in \mathbb{R}^{|\boldsymbol{\mathcal{C}}^{1:t-1} \cup c_u| \times H \times W}$ with the inference of $f_{t-1}(\boldsymbol{x})$ and the confidence $\boldsymbol{s}^{t-1} \in \mathbb{R}^{H \times W}$ of prediction. Formally:

$$\begin{aligned} \hat{\boldsymbol{y}}^{t-1} &= \arg \max_{\boldsymbol{\mathcal{C}}^{1:t-1} \cup c_u} f_{t-1}(\boldsymbol{x}), \\ \boldsymbol{s}^{t-1} &= \max_{\boldsymbol{\mathcal{C}}^{1:t-1} \cup c_u} \sigma(f_{t-1}(\boldsymbol{x})), \end{aligned} \tag{1}$$

where $\sigma(\cdot)$ denotes Sigmoid function. With the ground truth mask $\boldsymbol{y}^t = \{\boldsymbol{y}^t_i\}$ in current step, we mix the supervision label $\tilde{\boldsymbol{y}}^t_i$ of pixel i according to the following rules:

$$\tilde{\boldsymbol{y}}^t_i = \begin{cases} \boldsymbol{y}^t_i & \text{where } \boldsymbol{y}^t_i \in \boldsymbol{\mathcal{C}}^t \text{ or } \boldsymbol{y}^t_i = c_u \wedge \boldsymbol{s}^{t-1}_i < \tau \\ \hat{\boldsymbol{y}}^{t-1}_i & \text{where } \boldsymbol{y}^t_i = c_u \wedge \boldsymbol{s}^{t-1}_i \geq \tau \end{cases}, \tag{2}$$

where threshold $\tau = 0.7$, '$\wedge$' represents the co-taking of conditions.

Inter-class contrastive loss. With the guidance of pseudo label, CoinSeg gets the prototypes of classes (*i.e.*, class centroids), and sets up contrastive loss to better represent inter-class diversity.

Given the feature maps $\boldsymbol{M}^t = g_t(\boldsymbol{x})$ and binary masks $\tilde{\boldsymbol{y}}^t \in \{0,1\}^{|\mathbb{C}^t| \times h \times w}$ from the pseudo labels, CoinSeg applies masked average pooling (MAP) [48] to obtain prototypes of each foreground class as $\boldsymbol{P}^t_{int}$ for contrast inter-class representation:

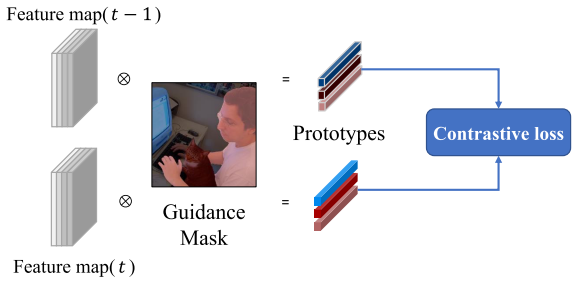


Figure 3. The pipeline of contrastive learning.

$$\boldsymbol{P}_{int}^{t} = MAP(\boldsymbol{M}^{t}, \tilde{\boldsymbol{y}}^{t}) = \frac{\sum_{i=1,j=1}^{h,w}(\tilde{\boldsymbol{y}}^{t,i,j} * M^{t,i,j})}{\sum_{i=1,j=1}^{h,w}\tilde{\boldsymbol{y}}^{t,i,j}}. \quad (3)$$

We abbreviate the operation as $\boldsymbol{P}_{int}^{t} = MAP(\boldsymbol{M}^{t}, \tilde{\boldsymbol{y}}^{t})$. $\boldsymbol{P}_{int}^{t}$ is a series of vectors as prototypes, *i.e.*, discriminative representation of each classes. At learning step $t > 1$, CoinSeg adapts model f_{t-1} from learning step $t-1$ as guidance to train the current model f_t, and we note $\boldsymbol{M}^{t-1} = g_{t-1}(\boldsymbol{x})$. $\boldsymbol{P}_{int}^{t-1}$ can also be obtained with feature maps of previous step $\boldsymbol{M}^{t-1}$ through a similar operation. CoinSeg assigns prototypes clustered from the same pseudo label as positive pairs in contrastive learning, and prototypes from different labels as negative pairs.

The distance of prototypes is measured with the inner product $Dot(\cdot,\cdot)$, and the contrastive loss with the guidance of the pseudo label can be expressed as:

$$\mathcal{L}_{int} = -\frac{1}{K}\sum_{i=1}^{K}\log\frac{\exp(Dot(\boldsymbol{P}_{int}^{t,i}, \boldsymbol{P}_{int}^{t-1,i}))}{\sum_{j=1,j\neq i}^{2K}\exp(Dot(\boldsymbol{P}_{int}^{t,i}, \boldsymbol{P}_{int}^{*,j}))}, \quad (4)$$

where $\boldsymbol{P}_{int}^{*} = \boldsymbol{P}_{int}^{t} \cup \boldsymbol{P}_{int}^{t-1}$, $K = |\boldsymbol{P}_{int}^{t}|$ is the number of inter-class prototypes, thus $2K = |\boldsymbol{P}_{int}^{t}| + |\boldsymbol{P}_{int}^{t-1}|$. Generally, as shown in Fig. 3, we abbreviate the contrastive loss by previous operations with the guidance of mask $\tilde{\boldsymbol{y}}^{t}$ as:

$$\mathcal{L}_{int} = CON(\tilde{\boldsymbol{y}}^{t}, M^{t}, M^{t-1}). \quad (5)$$

3.2.2 Contrast intra-class representations

Inspired by the Gaussian mixture models, we focus on intra-class diversity benefits to more robust representation learning. Thus we mine potential regional objectness within categories, and emphasize intra-class diversity through mask proposals-guided representation learning.

Mask proposals. CoinSeg adapts a set of *class-agnostic* binary mask $\boldsymbol{B} \in \{0,1\}^{N\times H\times W}$ as proposals (*i.e.*, mask proposals), where N denotes the number of mask proposals. Following the practice of [49], mask proposals are generated with Mask2Former [7]. Note each pixel in an image belongs and only belongs to one of the mask proposals.

Intra-class contrastive loss. Mask proposals discover regional objectness in images, which are likely to be diverse instances or centroids of a category, and benefit the construction of intra-class contrastive learning. For $\boldsymbol{M}^{t} = g_t(\boldsymbol{x})$, CoinSeg obtains prototypes of each mask proposal by $\boldsymbol{P}_{itr}^{t} = MAP(\boldsymbol{M}^{t}, \boldsymbol{B})$. Namely, $\boldsymbol{P}_{itr}^{t}$ is a series of mask proposal-based prototypes with size of $N \times C$. $\boldsymbol{P}_{itr}^{t-1}$ can also be obtained with the similar operation. To better characterize the intra-class diversity by contrastive learning, CoinSeg assigns prototypes from the same mask proposal as positive pairs, and prototypes from different mask proposals as negative pairs. The contrastive loss with the guidance of mask proposals can be expressed as:

$$\begin{aligned}\mathcal{L}_{itr} &= CON(\boldsymbol{B}, \boldsymbol{M}^{t}, \boldsymbol{M}^{t-1}) \\ &= -\frac{1}{N}\sum_{i=1}^{N}\log\frac{\exp(Dot(\boldsymbol{P}_{itr}^{t,i}, \boldsymbol{P}_{itr}^{t-1,i}))}{\sum_{j=1,j\neq i}^{2N}\exp(Dot(\boldsymbol{P}_{itr}^{t,i}, \boldsymbol{P}_{itr}^{*,j}))}.\end{aligned} \quad (6)$$

3.2.3 Summary

As explained in Sec. 3.1, due to the incremental learning task, there is a limitation in the acquisition of category labels in the ground truth, which leads to the fact that it is difficult for the CISS model to distinguish between categories. To this end, we enhance the model's category discrimination ability by emphasizing inter- and intra-class diversity. Moreover, contrastive learning overcomes catastrophic forgetting by the design of positive contrast pairs, *i.e.*, constraining the corresponding prototypes from feature map $\boldsymbol{M}^{t}$ and $\boldsymbol{M}^{t-1}$ to be consistent. This knowledge distillation-like mechanism helps to alleviate forgetting and improve performance in CISS.

Finally, the total loss for the learning of the Contrast inter- and intra-class Representations is:

$$\mathcal{L}_{ct} = \mathcal{L}_{int} + \mathcal{L}_{itr}. \quad (7)$$

3.3. Flexible Tuning Strategy

As mentioned before, releasing parameter training for plasticity could lead to the best performance, but more discriminative feature representation is needed. Although we have designed the Contrast inter- and intra-class Representations for the discriminative representation, some more specific parameter tuning strategy is necessary to ensure stability (*i.e.*, handling catastrophic forgetting) as well. Therefore, we introduce the Flexible Tuning (FT) strategy as that, which allows for training the model while mitigating the effects of forgetting, achieving a balance between stability and plasticity. Fig. 4 shows the comparison of the freeze strategy and flexible tuning strategy. The freeze strategy involves keeping the parameters of the feature extractor and classifier for historical classes fixed for learning step $t > 1$. In contrast, the flexible tuning strategy uses a lower learning

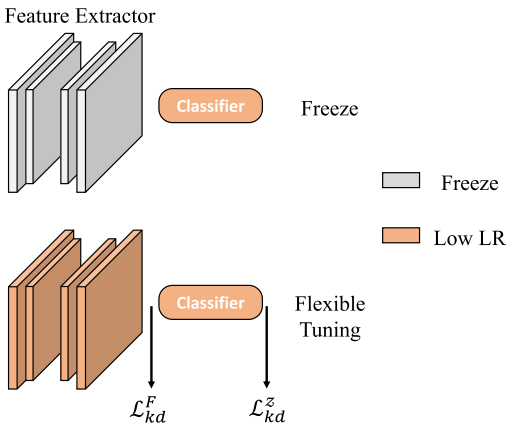


Figure 4. Comparison of Freeze and Flexible Tuning strategy for learning step $t > 1$. Best viewed in color.

rate and regularization constraints to allow for more flexible adjustments of these parameters.

Flexible initial learning rate. The segmentation model $f_{\boldsymbol{\theta}}$ is represented with a feature extractor $g_{\boldsymbol{\theta}_1}$ and a classifier $h_{\boldsymbol{\theta}_2}$, with parameters set $\boldsymbol{\theta}_1$ and $\boldsymbol{\theta}_2$ accordingly. Unlike the freeze strategy [5], which sets the learning rate to zero for both $\boldsymbol{\theta}_1$ and $\boldsymbol{\theta}_2$ of historical classes, the FT strategy applies a **initial** learning rate schedule of each learning step to these parameters. Specifically, during the first learning step ($t = 1$), the initial learning rate lr will be kept as original lr_0. For subsequent learning steps ($t > 1$), the initial learning rate is gradually reduced to better preserve the knowledge of the historical categories. So the initial lr of step t is expressed as:

$$lr^t = \begin{cases} lr_0 & \text{if } t = 1 \\ e^{-t}\lambda_{lr} \cdot lr_0 & \text{if } t > 1 \end{cases}, \tag{8}$$

where λ_{lr} is a hyper-parameter, and e^{-t} refers to the exponential decay of the learning rate.

Regularization constraints. When the flexible learning schedule allows the model to adapt more to new concepts, regularization constraints for mitigating forgetting become even more crucial than before. To better alleviate forgetting and ensure stability when the model adapts to new categories, CoinSeg applies some regularization constraints. To be more specific, for a sample $\boldsymbol{x}$, CoinSeg extract feature map $\boldsymbol{M}^{t-1} \in \mathbb{R}^{C\times H'\times W'}$ by the feature extractor g_{t-1}. *i.e.*, $\boldsymbol{M}^{t-1} = g_{t-1}(\boldsymbol{x})$, and C denotes the number of channels of $\boldsymbol{M}_{t-1}$. Similarly, $\boldsymbol{M}^t = g_t(\boldsymbol{x})$. Then, CoinSeg constrains the consistency of $\boldsymbol{M}^t$ and $\boldsymbol{M}^{t-1}$ with the Mean Square Error (MSE) as:

$$\begin{aligned} \mathcal{L}_{kd}^{F} &= MSE(\boldsymbol{M}^t, \boldsymbol{M}^{t-1}) \\ &= \frac{1}{C}\frac{1}{H'W'}\sum_{j=1}^{C}\sum_{i=1}^{H'W'}(\boldsymbol{M}_{i,j}^t - \boldsymbol{M}_{i,j}^{t-1})^2. \end{aligned} \tag{9}$$

Furthermore, CoinSeg also adapts knowledge distillation of logits z^t and z^{t-1}. Logits is the output of the model, *i.e.*, $z^t = f_t(x)$. Note $z^t \in \mathbb{R}^{|\boldsymbol{\mathcal{C}}^{1:t}\cup c_u|\times H\times W}$, where $\boldsymbol{\mathcal{C}}^{1:t} = \bigcup_{i=1}^{t}\boldsymbol{\mathcal{C}}^i$ is all seen categories of step t, including historical and present classes. As for logits $z^{t-1} \in \mathbb{R}^{|\boldsymbol{\mathcal{C}}^{1:t-1}\cup c_u|\times H\times W}$, while $\boldsymbol{\mathcal{C}}^t$ is not exist for previous model f_{t-1}. Following the common practice [4, 12, 26, 45], CoinSeg adopts the following approaches to remodeling the logits $\hat{z}^t$ for pixel i with class c :

$$\hat{z}_{i,c}^t = \begin{cases} z_{i,c}^t & \text{if } c \neq c_u \\ \sum_{j\in\boldsymbol{\mathcal{C}}^t} z_{i,j}^t & \text{if } c = c_u \end{cases}. \tag{10}$$

Then cross entropy loss L_{kd}^z is adapted to distill logits:

$$\begin{aligned} \mathcal{L}_{kd}^{z} &= CE(\hat{z}^t, z^{t-1}) \\ &= -\frac{1}{|\mathbb{C}^{t-1}|}\frac{1}{H'W'}\sum_{j=1}^{|\mathbb{C}^{t-1}|}\sum_{i=1}^{H'W'}(\hat{z}_{i,j}^t \log z_{i,j}^{t-1}), \end{aligned} \tag{11}$$

where we assign $|\mathbb{C}^{t-1}| = |\boldsymbol{\mathcal{C}}^{1:t-1} \cup c_u|$.

In summary, the regularization constraints in the flexible tuning strategy are:

$$\mathcal{L}_{reg} = \mathcal{L}_{kd}^{F} + \mathcal{L}_{kd}^{z}. \tag{12}$$

3.4. Objective Function

Following the practice of previous methods[49], CoinSeg adapt common Binary Cross-Entropy (BCE) loss as supervised segmentation loss $\mathcal{L}_{BCE}$ with the augmented label $\tilde{\boldsymbol{y}}$, and the final objective function of CoinSeg is:

$$\mathcal{L} = \mathcal{L}_{BCE} + \lambda_c \cdot \mathcal{L}_{ct} + \lambda_r \cdot \mathcal{L}_{reg}. \tag{13}$$

4. Experiment

4.1. Experimental Setups

Dataset. We have evaluated our approach on datasets of Pascal VOC 2012 [13] and ADE20K [50]. Pascal VOC 2012 contains 10,582 training images and 1449 validation images, with a total of 20 foreground classes and one background class. ADE20K contains 20,210 training images and 2,000 validation images with 100 thing classes and 50 stuff classes.

Protocols. Following the conventions of previous work [5, 12, 49], we mainly evaluate our approach for CISS with *overlapped* experimental setup, which is more realistic than

the *disjoint* setting that was studied by [4, 26, 45]. For each benchmark dataset, we examine our approach under multiple *incremental scenarios*. We abbreviate each incremental scenario in the form of $X - Y$, where X means the initial number of base classes, and Y refers to the incremental number of classes in each step. For instance, the 10-1 scenario of the VOC dataset (VOC 10-1) signifies 10 classes learned at the first learning step, and 1 incremental class learned in each subsequent step, *i.e.*, VOC 10-1 scenario takes a total of 11 steps to learn the entire dataset.

Implementation details. In accordance with the established practice [4, 5, 26], we use DeepLabv3 [6] as the segmentation network. CoinSeg chooses Swin Transformer-base (Swin-B) [19] pretrained on ImageNet-1K as the backbone. Swin Transformer provides a better feature representation of the local patches, which is concerned in our CoinSeg. For the segmentation head, we apply the dual-head architecture from MicroSeg, including dense prediction branch and proposal classification branch [49]. We optimize the model by ADAMW [21] with an learning rate of $lr_0 = 10^{-4}$. The batch size is 16 for Pascal VOC 2012, and 12 for ADE20K. The window size for Swin Transformer is 12. Data augmentation [5] are applied for all samples. To ensure a fair comparison with previous methods, we apply the same class-agnostic mask proposals with MicroSeg [49]. Specifically, mask proposals are generated with parameters-fixed Mask2Former [7] pre-trained on MS-COCO, with $N = 100$ for all experiments. To prevent information leakage, the Mask2Former is **not** fine-tuned on any benchmark dataset [49]. All experiments are implemented with PyTorch on two NVIDIA GeForce RTX 3090 GPUs. Hyper-parameters $\lambda_{lr} = 10^{-3}$, $\lambda_c = 0.01$ and $\lambda_r = 0.1$ are set for all experiments.

Baselines. We evaluate our CoinSeg on multiple incremental scenarios. The performance of CoinSeg is compared with some representative approaches in CIL, including Lwf-MC [18] and ILT [25], which are applied to the experimental setup of CISS. Besides, we provide the results of the prior state-of-the-art CISS methods, including MiB [4], SDR [26], PLOP [12], SSUL [5], RCIL [45], and MicroSeg [49]. To ensure a fair comparison with recent state-of-the-art CISS methods [5, 49] that employ different backbones, we produced results by replacing their backbones with Swin-B. These methods are re-implemented with their official codes. Methods with suffix 'M' (for example, *SSUL-M*) are with memory sampling strategy [5], which maintains a memory bank of historical samples for rehearsal at future learning steps. Besides, we also provide the experimental results of *joint* training (training all classes together), on both Resnet101 and Swin-B backbones. The result of joint training is usually regarded as the upper bound of incremental learning [4, 18], *i.e.*, offline training. The mean Intersection-over-Union (mIoU) is applied as the evaluation metric for all experiments and analyses. For each method in CISS, three perspectives are presented: the performance of base classes, novel classes, and total performance, respectively. Please refer to the supplementary material for more details.

4.2. Experimental Results

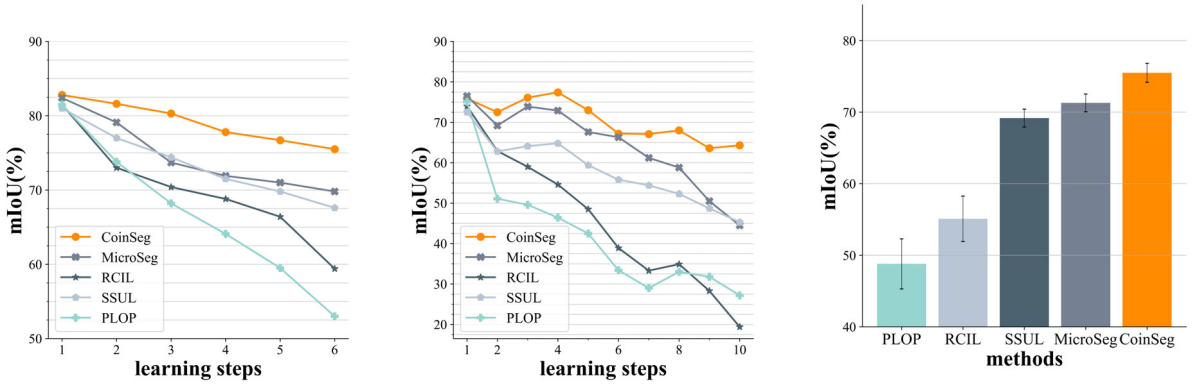
Comparison on VOC. On the Pascal VOC 2012 dataset, we evaluate CoinSeg on various incremental scenarios, including long-term scenarios with lots of learning steps (10-1, 15-1), with a large number of base classes (19-1, 15-5) and an equally-divided long-term scenario (2-2). The performance comparison of our approach with classical CIL methods and prior CISS methods is presented in Tab. 1. Our CoinSeg exhibits a significant performance advantage in all incremental scenarios, even when compared to the state-of-the-art methods using Swin Transformer as the backbone.

In particular, in very long-term incremental scenarios with few base classes, like 10-1 and 2-2, our CoinSeg brings huge performance gaps of 6.7% and 17.8%, respectively, in comparison to MicroSeg. The freeze strategy in SSUL and MicroSeg prevented these methods from performing well in incremental scenarios with few base classes. In such scenarios, the base classes contain fewer concepts, which can easily introduce bias for the representation learning of novel classes. Our approach addresses this challenge by allowing the model to adapt to novel classes with the design of 'contrast inter- and intra-class representations'. In such incremental scenarios, the base classes contain fewer concepts and can easily introduce bias for the representation learning of novel classes. Besides, we concern the scenarios with a large number of base classes (*i.e.*, 19-1 and 15-5), in which, CoinSeg achieves state-of-the-art with a performance gain of 3.4% and 2.4%, compared with MicroSeg.

Besides, Fig. 5 (a,b) depicts the variations of mIoU of **all seen classes** by learning steps during incremental learning, in two long-term incremental scenarios, *i.e.*, VOC 15-1 and VOC 2-2, respectively. The performance of each step depends on two factors: 1) the forgetting of historical classes, and 2) the ability to learn new classes. Fig. 5 (a) shows the results of VOC 15-1, which involves a large number of base classes. The performance of our CoinSeg exhibits the least decrease with increasing training steps, suggesting that CoinSeg causes less forgetting of previously learned knowledge than prior methods. In contrast, while VOC 2-2 consists of 18 novel classes, results of Fig. 5 (b) clearly demonstrate that our CoinSeg is more adaptable to learning new classes, compared to previous methods. Additionally, our CoinSeg even shows a significant performance improvement while learning new classes in some steps (steps 3, 4, and 8). Besides, to demonstrate the robustness of CoinSeg, we provide experimental results of VOC 15-1 with 20 different class incremental orders, including average and standard variance

Table 1. Comparison with state-of-the-art methods on Pascal VOC 2012. †: Re-implemented with Swin-B backbone; Joint is the upperbound.

Method	Backbone	VOC 10-1 (11 steps)			VOC 15-1 (6 steps)			VOC 19-1 (2 steps)			VOC 15-5 (2 steps)			VOC 2-2 (10 steps)		
		0-10	11-20	all	0-15	16-20	all	0-19	20	all	0-15	16-20	all	0-2	3-20	all
Joint	Resnet101	82.1	79.6	80.9	82.7	75.0	80.9	81.0	79.1	80.9	82.7	75.0	80.9	76.5	81.6	80.9
LwF-MC [18]	Resnet101	4.7	5.9	4.9	6.4	8.4	6.9	64.4	13.3	61.9	58.1	35.0	52.3	3.5	4.7	4.5
ILT [25]	Resnet101	7.2	3.7	5.5	8.8	8.0	8.6	67.8	10.9	65.1	67.1	39.2	60.5	5.8	5.0	5.1
MiB [4]	Resnet101	12.3	13.1	12.7	34.2	13.5	29.3	71.4	23.6	69.2	76.4	50.0	70.1	41.1	23.4	25.9
SDR [26]	Resnet101	32.1	17.0	24.9	44.7	21.8	39.2	69.1	32.6	67.4	57.4	52.6	69.9	13.0	5.1	6.2
PLOP [12]	Resnet101	44.0	15.5	30.5	65.1	21.1	54.6	75.4	37.4	73.5	75.7	51.7	70.1	24.1	11.9	13.7
RCIL [45]	Resnet101	55.4	15.1	34.3	70.6	23.7	59.4	68.5	12.1	65.8	78.8	52.0	72.4	28.3	19.0	19.4
SSUL [5]	Resnet101	71.3	46.0	59.3	77.3	36.6	67.6	77.7	29.7	75.4	77.8	50.1	71.2	62.4	42.5	45.3
MicroSeg [49]	Resnet101	72.6	48.7	61.2	80.1	36.8	69.8	78.8	14.0	75.7	80.4	52.8	73.8	61.4	40.6	43.5
Joint†	Swin-B	82.4	83.0	82.7	83.8	79.3	82.7	82.6	84.4	82.7	83.8	79.3	82.7	75.8	83.9	82.7
SSUL†[5]	Swin-B	74.3	51.0	63.2	78.1	33.4	67.5	80.8	31.5	78.4	79.7	55.3	73.9	60.3	40.6	44.0
MicroSeg† [49]	Swin-B	73.5	53.0	63.8	80.5	40.8	71.0	79.0	25.3	76.4	81.9	54.0	75.2	64.8	43.4	46.5
CoinSeg (Ours)	Swin-B	**80.1**	**60.0**	**70.5**	**82.7**	**52.5**	**75.5**	**81.5**	**44.8**	**79.8**	**82.1**	**63.2**	**77.6**	**70.1**	**63.3**	**64.3**



(a) (b) (c)

Figure 5. Performance comparisons for VOC. Illustration of the change of miou with learning step in (a) VOC 15-1, (b) VOC 2-2. Note we calculate mean IoU for all seen classes until current learning step. *e.g.*, the mIoU of classes 0-10 for VOC 2-2 step 5. And (c) Average performance comparisons with 20 different incremental orders of VOC 15-1.

of mIoU, illustrate as Fig. 5 (c). The results indicate that our approach performs better and is more robust, with a lower standard variance.

Comparison on ADE. For the more challenging ADE dataset, we choose incremental scenarios that have been widely compared in past methods, including 100-5, 100-10, 100-50, and 50-50, shown in Tab. 2. Our method, CoinSeg, outperforms previous state-of-the-art once again, which shows that our method is not specific to a particular dataset and is effective under multiple benchmarks.

4.3. Ablation Studies

Overall ablation of CoinSeg. In this section, we present an evaluation of each specific design of the proposed CoinSeg, including the Swin Transformer, Flexible Tuning (FT) strategy, and Contrast inter- and intra-class Representations (*Coin*). As shown in Tab. 4, we can find that all these proposed designs benefit the performance of CISS. By replacing the backbone with Swin-Transformer (row 1 *vs* row 2), it improves the ability of feature representation of the feature extractor, which is especially beneficial for the representation of local features, and necessary for CoinSeg. The comparison between row 2 and row 3 reflects the experiment of replacing the freeze strategy with the FT strategy, showing that the FT enhances model plasticity, allowing adaptation to novel classes, and slightly improving the performance. The application of *Coin* (row3 *vs* row4) significantly improves the performance by imposing explicit constraints on intra-class and inter-class diversity, thereby enhancing the model's ability to represent data. Overall, these components enable CoinSeg to achieve state-of-the-art performance.

Number of proposal in $\mathcal{L}_{itr}$. We also investigated the influence of hyperparameter N on the method's performance concerning *contrast intra-class diversity*. Specifically, we explored how altering the number of proposals would affect the proposed CoinSeg. The results are presented in Tab. 5. It can be observed that with N increases, there is a slight improvement in performance, albeit with diminishing returns

Table 2. Comparison with state-of-the-art methods on ADE20K. †: Re-implemented with Swin-B backbones; Joint is the upperbound.

Method	Backbone	ADE 100-5 (11 steps)			ADE 100-10 (6 steps)			ADE 100-50 (2 steps)			ADE 50-50 (3 steps)		
		0-100	101-150	all	0-100	101-150	all	0-100	101-150	all	0-50	51-150	all
Joint	Resnet101	43.8	28.9	38.9	43.8	28.9	38.9	43.8	28.9	38.9	50.7	32.8	38.9
ILT [25]	Resnet101	0.1	1.3	0.5	0.1	3.1	1.1	18.3	14.4	17.0	3.5	12.9	9.7
MiB [4]	Resnet101	36.0	5.7	26.0	38.2	11.1	29.2	40.5	17.2	32.8	45.6	21.0	29.3
PLOP [12]	Resnet101	39.1	7.8	28.8	40.5	13.6	31.6	41.9	14.9	32.9	48.8	21.0	30.4
SSUL [5]	Resnet101	39.9	17.4	32.5	40.2	18.8	33.1	41.3	18.0	33.6	48.4	20.2	29.6
RCIL [45]	Resnet101	38.5	11.5	29.6	39.3	17.7	32.1	**42.3**	18.8	34.5	48.3	24.6	32.5
MicroSeg [49]	Resnet101	40.4	20.5	33.8	41.5	21.6	34.9	40.2	18.8	33.1	48.6	24.8	32.9
Joint †	Swin-B	43.5	30.6	39.2	43.5	30.6	39.2	43.5	30.6	39.2	50.2	33.7	39.2
SSUL † [5]	Swin-B	41.3	16.0	32.9	40.7	19.0	33.5	41.9	20.1	34.6	49.5	21.3	30.7
MicroSeg † [49]	Swin-B	41.2	21.0	34.5	41.0	22.6	34.8	41.1	24.1	35.4	**49.8**	23.9	32.5
CoinSeg	Swin-B	**43.1**	**24.1**	**36.8**	**42.1**	**24.5**	**36.2**	41.6	**26.7**	**36.6**	49.0	**28.9**	**35.6**

Table 3. Comparisons of CoinSeg using memory sampling strategy. †: Re-implemented with Swin-B backbone.

Method	Backbone	VOC 10-1 (11 steps)			VOC 15-1 (6 steps)			VOC 19-1 (2 steps)			VOC 15-5 (2 steps)			VOC 2-2 (10 steps)		
		0-10	11-20	all	0-15	16-20	all	0-19	20	all	0-15	16-20	all	0-2	3-20	all
SSUL-M [5]	Resnet101	74.0	53.2	64.1	78.4	49.0	71.4	77.8	49.8	76.5	78.4	55.8	73.0	58.8	45.8	47.6
MicroSeg-M [49]	Resnet101	77.2	57.2	67.7	81.3	52.5	74.4	79.3	62.9	78.5	82.0	59.2	76.6	60.0	50.9	52.2
SSUL-M† [5]	Swin-B	75.3	54.1	65.2	78.8	49.7	71.9	78.5	50.0	77.1	79.3	55.1	73.5	61.1	47.5	49.4
MicroSeg-M† [49]	Swin-B	78.9	59.2	70.1	82.0	47.3	73.7	81.0	**62.4**	80.0	82.9	60.1	77.5	62.7	51.4	53.0
CoinSeg (Ours)	Swin-B	80.0	63.4	72.5	82.7	52.5	75.5	81.5	44.8	79.8	82.1	63.2	77.6	**70.1**	63.3	64.3
CoinSeg-M (Ours)	Swin-B	**81.3**	**64.4**	**73.7**	**84.1**	**65.6**	**79.6**	**82.7**	52.6	**81.3**	**84.1**	**69.9**	**80.8**	68.4	**65.6**	**66.0**

Table 4. Ablation Studies for our proposed methods. *Coin*: contrast inter- and intra-class representations, Fz: Freeze strategy, FLR: flexible initial learning rate. Numbers in the brackets (): gains w.r.t. the preceding row.

Backbone	parameter strategy			*Coin*		VOC 15-1 (6 steps)		
	LR	$\mathcal{L}_{kd}^{F}$	$\mathcal{L}_{kd}^{z}$	$\mathcal{L}_{int}$	$\mathcal{L}_{itr}$	0-15	16-20	all
ResNet101	Fz	✗	✗	✗	✗	74.9	26.4	63.3
Swin-B	Fz	✗	✗	✗	✗	79.5	42.4	70.5
Swin-B	FLR	✗	✗	✗	✗	73.6	44.0	66.7
Swin-B	FLR	✓	✗	✗	✗	78.9	42.7	70.3 (+3.6)
Swin-B	FLR	✓	✓	✗	✗	80.4	43.7	71.6 (+1.3)
Swin-B	FLR	✓	✓	✓	✗	80.8	45.9	72.4 (+0.8)
Swin-B	FLR	✓	✓	✓	✓	82.7	52.5	**75.5** (+3.1)

Table 5. Ablations to # of proposals (N) in $\mathcal{L}_{itr}$ (Left), and pseudo-labeling in $\mathcal{L}_{int}$ (Right). GT: ground truth, PL: pseudo label.

Method	N	VOC 15-1 (6 steps)		
		0-15	16-20	all
CoinSeg	50	81.4	51.1	74.2
	100	82.7	52.5	75.5
	200	83.1	53.8	**76.1**

as the cardinality grows. Consequently, considering a trade-off between performance and computational complexity, we choose $N = 100$ in our method.

4.4. Qualitative Analysis

We have conducted a qualitative analysis using two examples in Fig. 6. In the first example (rows 1,3 & 5), during incremental learning, CoinSeg is able to retain knowledge about past classes and accurate predictions for them, whereas prior methods exhibit forgetting and misclassification, which demonstrated the stability of CoinSeg. The second example (rows 2, 4 & 6), demonstrates the plasticity of our method, which refers to the ability to learn new classes. For example, while prior methods predicted wrong bounds for the class 'train' in incremental learning, our method is better adapted to these new classes making appropriate predictions.

Table 6. The performance of Flexible tuning for prior freeze-strategy-based method MicroSeg, 'FT' denotes flexible tuning .

Method	Backbone	parameter strategy	VOC 15-1 (6 steps)		
			0-15	16-20	all
MicroSeg	Resnet101	Freeze	80.5	40.8	71.0
	Resnet101	FT	80.9	41.5	**72.3** (+1.3)
	Swin-B	Freeze	80.1	36.8	69.8
	Swin-B	FT	79.8	40.2	**70.4** (+0.6)

4.5. Expansibility of CoinSeg

Flexible tuning on prior methods. As claimed, the FT strategy flexibly releases parameter training for plasticity and provides some more specific parameter tuning ways to ensure stability as well. To better validate the effectiveness of the FT strategy, we applied the FT strategy to the state-of-the-art method MicroSeg, and compared the results to its original freeze strategy, as shown in Tab. 6. The results demonstrate that on both backbones, applying the FT strategy leads to better performance, particularly, for the new classes, which proves that model plasticity is significantly enhanced by

Figure 6. Qualitative analysis for VOC, comparing with prior CISS methods. Text points out the **uncorrectly predicted** areas.

the FT strategy. It means that the FT strategy can be also extended to other CISS methods for better performance, especially performance relying on model plasticity.

CoinSeg with memory sampling. The memory sampling strategy alleviates forgetting by rehearsing samples from past learning steps, and significantly improves performance [5, 49]. Thus, we produce the results of CoinSeg with this strategy, denoted as CoinSeg-M, for comparison with prior methods with the same strategy. As shown in Tab. 3, our CoinSeg (the 5_{th} row) achieves better performance in almost all incremental scenarios, even when compared with previous work using the sampling strategy. When equipped with the memory sampling method, *i.e.*, CoinSeg-M, it undoubtedly achieves state-of-the-art performance.

5. Conclusion

In this work, we studied class incremental semantic segmentation and proposed an effective method CoinSeg. Inspired by the Gaussian mixture model, we proposed *Coin* to better characterize samples with explicitly constrain inter- and intra- class diversity. Furthermore, we proposed a flexible tuning strategy, to keep the stability of the model and alleviate forgetting by the flexible initial learning rate and regularization constraints. Extensive experimental evaluations show the effectiveness of our method. CoinSeg outperforms prior state-of-the-art CISS methods, especially on more challenging long-term incremental scenarios.

Acknowledgment. This work was supported in part by the National Key R&D Program of China (No.2021ZD0112100), the National Natural Science Foundation of China (No. 61972036), the Fundamental Research Funds for the Central Universities (No. K22RC00010).

References

[1] Rahaf Aljundi, Francesca Babiloni, Mohamed Elhoseiny, Marcus Rohrbach, and Tinne Tuytelaars. Memory aware synapses: Learning what (not) to forget. In *Proc. European Conference on Computer Vision*, pages 139–154, 2018. 2

[2] Rahaf Aljundi, Punarjay Chakravarty, and Tinne Tuytelaars. Expert gate: Lifelong learning with a network of experts. In *Proc. IEEE Conference on Computer Vision and Pattern Recognition*, pages 3366–3375, 2017. 2

[3] Gert Cauwenberghs and Tomaso Poggio. Incremental and decremental support vector machine learning. *Advances in neural information processing systems*, 13, 2000. 1

[4] Fabio Cermelli, Massimiliano Mancini, Samuel Rota Bulo, Elisa Ricci, and Barbara Caputo. Modeling the background for incremental learning in semantic segmentation. In *Proc. IEEE Conference on Computer Vision and Pattern Recognition*, pages 9233–9242, 2020. 1, 2, 3, 5, 6, 7, 8

[5] Sungmin Cha, YoungJoon Yoo, Taesup Moon, et al. Ssul: Semantic segmentation with unknown label for exemplar-based class-incremental learning. *Advances in neural information processing systems*, 34, 2021. 1, 2, 3, 5, 6, 7, 8, 9

[6] 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 Transactions on Pattern Recognition and Machine Intelligence*, 40(4):834–848, 2017. 6

[7] Bowen Cheng, Ishan Misra, Alexander G Schwing, Alexander Kirillov, and Rohit Girdhar. Masked-attention mask transformer for universal image segmentation. *arXiv preprint arXiv:2112.01527*, 2021. 3, 4, 6

[8] Bowen Cheng, Alexander G. Schwing, and Alexander Kirillov. Per-pixel classification is not all you need for semantic segmentation. *Advances in neural information processing systems*, 34, 2021. 3

[9] Xiangxiang Chu, Zhi Tian, Bo Zhang, Xinlong Wang, Xiaolin Wei, Huaxia Xia, and Chunhua Shen. Conditional positional encodings for vision transformers. *arXiv preprint arXiv:2102.10882*, 2021. 2

[10] Matthias De Lange and Tinne Tuytelaars. Continual prototype evolution: Learning online from non-stationary data streams. In *Proc. IEEE International Conference on Computer Vision*, pages 8250–8259, 2021. 2

[11] 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. *arXiv preprint arXiv:2010.11929*, 2020. 2

[12] Arthur Douillard, Yifu Chen, Arnaud Dapogny, and Matthieu Cord. Plop: Learning without forgetting for continual semantic segmentation. In *Proc. IEEE Conference on Computer Vision and Pattern Recognition*, pages 4040–4050, 2021. 1, 2, 3, 5, 6, 7, 8

[13] Mark Everingham, Luc Van Gool, Christopher KI Williams, John Winn, and Andrew Zisserman. The pascal visual object classes (voc) challenge. *International Journal on Computer Vision*, 88(2):303–338, 2010. 5

[14] Kunyang Han, Yong Liu, Jun Hao Liew, Henghui Ding, Jiajun Liu, Yitong Wang, Yansong Tang, Yujiu Yang, Jiashi Feng, Yao Zhao, and Yunchao Wei. Global knowledge calibration for fast open-vocabulary segmentation, 2023. 2

[15] Geoffrey Hinton, Oriol Vinyals, Jeff Dean, et al. Distilling the knowledge in a neural network. *arXiv preprint arXiv:1503.02531*, 2(7), 2015. 2

[16] Heechul Jung, Jeongwoo Ju, Minju Jung, and Junmo Kim. Less-forgetting learning in deep neural networks. *arXiv preprint arXiv:1607.00122*, 2016. 1, 2

[17] Yann LeCun, Yoshua Bengio, and Geoffrey Hinton. Deep learning. *nature*, 521(7553):436–444, 2015. 1

[18] Zhizhong Li and Derek Hoiem. Learning without forgetting. *IEEE Transactions on Pattern Recognition and Machine Intelligence*, 40(12):2935–2947, 2017. 1, 2, 6, 7

[19] 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 *Proceedings of the IEEE/CVF international conference on computer vision*, pages 10012–10022, 2021. 2, 3, 6

[20] David Lopez-Paz and Marc'Aurelio Ranzato. Gradient episodic memory for continual learning. *Advances in neural information processing systems*, 30, 2017. 2

[21] Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. *arXiv preprint arXiv:1711.05101*, 2017. 6

[22] Arun Mallya and Svetlana Lazebnik. Packnet: Adding multiple tasks to a single network by iterative pruning. In *Proc. IEEE Conference on Computer Vision and Pattern Recognition*, pages 7765–7773, 2018. 2

[23] 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

[24] Michael McCloskey and Neal J Cohen. Catastrophic interference in connectionist networks: The sequential learning problem. In *Psychology of learning and motivation*, volume 24, pages 109–165. Elsevier, 1989. 1

[25] Umberto Michieli and Pietro Zanuttigh. Incremental learning techniques for semantic segmentation. In *Proc. IEEE International Conference on Computer Vision Workshops*, pages 0–0, 2019. 6, 7, 8

[26] Umberto Michieli and Pietro Zanuttigh. Continual semantic segmentation via repulsion-attraction of sparse and disentangled latent representations. In *Proc. IEEE Conference on Computer Vision and Pattern Recognition*, pages 1114–1124, 2021. 2, 3, 5, 6, 7

[27] Oleksiy Ostapenko, Mihai Puscas, Tassilo Klein, Patrick Jahnichen, and Moin Nabi. Learning to remember: A synaptic plasticity driven framework for continual learning. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 11321–11329, 2019. 2

[28] Haim Permuter, Joseph Francos, and Ian Jermyn. A study of gaussian mixture models of color and texture features for image classification and segmentation. *Pattern recognition*, 39(4):695–706, 2006. 2

[29] Franz Pernkopf and Djamel Bouchaffra. Genetic-based em algorithm for learning gaussian mixture models. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 27(8):1344–1348, 2005. 2

[30] Robi Polikar, Lalita Upda, Satish S Upda, and Vasant Honavar. Learn++: An incremental learning algorithm for supervised neural networks. *IEEE transactions on systems, man, and cybernetics, part C (applications and reviews)*, 31(4):497–508, 2001. 1

[31] Amal Rannen, Rahaf Aljundi, Matthew B Blaschko, and Tinne Tuytelaars. Encoder based lifelong learning. In *Proc. IEEE International Conference on Computer Vision*, pages 1320–1328, 2017. 2

[32] Sylvestre-Alvise Rebuffi, Alexander Kolesnikov, Georg Sperl, and Christoph H Lampert. icarl: Incremental classifier and representation learning. In *Proc. IEEE Conference on Computer Vision and Pattern Recognition*, pages 2001–2010, 2017. 2

[33] Douglas A Reynolds et al. Gaussian mixture models. *Encyclopedia of biometrics*, 741(659-663), 2009. 2

[34] David Rolnick, Arun Ahuja, Jonathan Schwarz, Timothy Lillicrap, and Gregory Wayne. Experience replay for continual learning. *Advances in neural information processing systems*, 32, 2019. 2

[35] Amir Rosenfeld and John K Tsotsos. Incremental learning through deep adaptation. *IEEE Transactions on Pattern Recognition and Machine Intelligence*, 42(3):651–663, 2018. 2

[36] Andrei A Rusu, Neil C Rabinowitz, Guillaume Desjardins, Hubert Soyer, James Kirkpatrick, Koray Kavukcuoglu, Razvan Pascanu, and Raia Hadsell. Progressive neural networks. *arXiv preprint arXiv:1606.04671*, 2016. 2

[37] Joan Serra, Didac Suris, Marius Miron, and Alexandros Karatzoglou. Overcoming catastrophic forgetting with hard attention to the task. In *Proc. International Conference on Machine Learning*, pages 4548–4557, 2018. 2

[38] Hanul Shin, Jung Kwon Lee, Jaehong Kim, and Jiwon Kim. Continual learning with deep generative replay. *Advances in neural information processing systems*, 30, 2017. 2

[39] Hanul Shin, Jung Kwon Lee, Jaehong Kim, and Jiwon Kim. Continual learning with deep generative replay. *Advances in neural information processing systems*, 30, 2017. 2

[40] Hugo Touvron, Matthieu Cord, Matthijs Douze, Francisco Massa, Alexandre Sablayrolles, and Hervé Jégou. Training data-efficient image transformers & distillation through attention. In *International conference on machine learning*, pages 10347–10357. PMLR, 2021. 2

[41] Chenshen Wu, Luis Herranz, Xialei Liu, Joost Van De Weijer, Bogdan Raducanu, et al. Memory replay gans: Learning to generate new categories without forgetting. *Advances in Neural Information Processing Systems*, 31, 2018. 2

[42] Enze Xie, Wenhai Wang, Zhiding Yu, Anima Anandkumar, Jose M Alvarez, and Ping Luo. Segformer: Simple and efficient design for semantic segmentation with transformers. *Advances in Neural Information Processing Systems*, 34:12077–12090, 2021. 3

[43] Boyu Yang, Chang Liu, Bohao Li, Jianbin Jiao, and Qixiang Ye. Prototype mixture models for few-shot semantic segmentation. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part VIII 16*, pages 763–778. Springer, 2020. 2

[44] Guanglei Yang, Enrico Fini, Dan Xu, Paolo Rota, Mingli Ding, Moin Nabi, Xavier Alameda-Pineda, and Elisa Ricci. Uncertainty-aware contrastive distillation for incremental semantic segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2022. 2

[45] Chang-Bin Zhang, Jia-Wen Xiao, Xialei Liu, Ying-Cong Chen, and Ming-Ming Cheng. Representation compensation networks for continual semantic segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7053–7064, 2022. 2, 3, 5, 6, 7, 8

[46] Gengwei Zhang, Liyuan Wang, Guoliang Kang, Ling Chen, and Yunchao Wei. Slca: Slow learner with classifier alignment for continual learning on a pre-trained model. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2023. 1, 2

[47] Junting Zhang, Jie Zhang, Shalini Ghosh, Dawei Li, Serafettin Tasci, Larry Heck, Heming Zhang, and C-C Jay Kuo. Class-incremental learning via deep model consolidation. In *Proc. IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 1131–1140, 2020. 2

[48] Xiaolin Zhang, Yunchao Wei, Yi Yang, and Thomas S Huang. Sg-one: Similarity guidance network for one-shot semantic segmentation. *IEEE Transactions on Cybernetics*, 50(9):3855–3865, 2020. 3

[49] Zekang Zhang, Guangyu Gao, Zhiyuan Fang, Jianbo Jiao, and Yunchao Wei. Mining unseen classes via regional objectness: A simple baseline for incremental segmentation. *Advances in neural information processing systems*, 35, 2022. 2, 3, 4, 5, 6, 7, 8, 9

[50] Bolei Zhou, Hang Zhao, Xavier Puig, Sanja Fidler, Adela Barriuso, and Antonio Torralba. Scene parsing through ade20k dataset. In *Proc. IEEE Conference on Computer Vision and Pattern Recognition*, pages 633–641, 2017. 5

[51] Hongguang Zhu, Yunchao Wei, Xiaodan Liang, Chunjie Zhang, and Yao Zhao. Ctp: Towards vision-language continual pretraining via compatible momentum contrast and topology preservation, 2023. 2