Continual Semantic Segmentation Leveraging Image-level Labels and Rehearsal

Mathieu Page Fortin[1] , **Brahim Chaib-draa**[2]
Laval University, Québec, Canada
[1]mathieu.page-fortin.1@ulaval.ca
[2]brahim.chaib-draa@ift.ulaval.ca

Abstract

Despite the remarkable progress of deep learning models for semantic segmentation, the success of these models is strongly limited by the following aspects: 1) large datasets with pixel-level annotations must be available and 2) training must be performed with all classes simultaneously. Indeed, in incremental learning scenarios, where new classes are added to an existing framework, these models are prone to catastrophic forgetting of previous classes. To address these two limitations, we propose a weakly-supervised mechanism for continual semantic segmentation that can leverage cheap image-level annotations and a novel rehearsal strategy that intertwines the learning of past and new classes. Specifically, we explore two rehearsal technique variants: 1) imprinting past objects on new images and 2) transferring past representations in intermediate features maps. We conduct extensive experiments on Pascal-VOC by varying the proportion of fully- and weakly-supervised data in various setups and show that our contributions consistently improve the mIoU on both past and novel classes. Interestingly, we also observe that models trained with less data in incremental steps sometimes outperform the same architectures trained with more data. We discuss the significance of these results and propose some hypotheses regarding the dynamics between forgetting and learning.

1 Introduction

Semantic segmentation is a fundamental task of computer vision, as it can play a key role for perception in many applications. While deep convolutional networks have significantly advanced this field in the last decade for static datasets, these models struggle in incremental scenarios. Indeed, when new classes have to be added to existing models, fine-tuning these models on new data tends to erase previous knowledge, a phenomenon termed catastrophic forgetting [Michieli and Zanuttigh, 2019]. This poses a key challenge for many real applications as a model deployed in a dynamic environment should be able to adapt without losing its acquired skills.

An increasing amount of work in continual learning has been done in the last few years to address catastrophic forgetting, especially from the side of classification tasks [Delange *et al.*, 2021]. On the other hand, continual semantic segmentation (CSS) only began to be investigated recently [Michieli and Zanuttigh, 2019]. In CSS, forgetting manifests itself differently due to the problem of background shift [Cermelli *et al.*, 2020], a challenge that is unique to this task. Specifically, it designates the fact that the set of object categories belonging to the background changes over time. Indeed, past and future classes are annotated as background when they appear in novel images since learning is focused on the new classes. Forgetting is thus amplified because this actively pushes the model to erase previous learning and can also interfere with future training steps [Cermelli *et al.*, 2020].

Previous work on background shift was mainly tackled with one or a combination of the following strategies: 1) adapting the cross-entropy and distillation losses to account for the uncertainty of the background [Cermelli *et al.*, 2020], 2) training the model with pseudo-labels of past classes [Douillard *et al.*, 2021] and 3) replaying past data stored in memory [Maracani *et al.*, 2021].

In this paper, we explore a complementary strategy to the ones cited above by proposing a weakly-supervised mechanism for CSS that can leverage image-level annotations. Notably, we explore a scenario in which past classes are only labelled at the image-level while current classes have their full segmentation masks. While [Cermelli *et al.*, 2020] justified the experimental setting of having past classes annotated as background because segmentation masks are costly to produce, image-level labels are much cheaper to collect than pixel-level annotations. Leveraging weak labels is thus a promising strategy to improve CSS at a small additional cost.

In a similar vein, self-training [Yu *et al.*, 2020] and webly-supervised learning [Maracani *et al.*, 2021] approaches have been proposed to learn from unlabelled examples in continual scenarios. While the results of these works supported the benefits of using additional unlabelled data in CSS, their frameworks required large auxiliary datasets. For instance, [Yu *et al.*, 2020] assumed access to $\sim$120K and $\sim$1.8M additional unlabelled images with classes that overlap those of the main dataset, which is a strong assumption that cannot hold in many real applications.

In this work, we propose a data-efficient weakly-supervised method that does not require additional images from auxiliary datasets. An overview of the weakly-supervised mechanism is shown in Fig. 1. Specifically, we develop a module that leverages the Grad-CAM technique [Selvaraju *et al.*, 2017] to localize objects from a multi-label classification head, and we use a self-attention mechanism [Vaswani *et al.*, 2017] to automatically learn to refine the mask proposals in an end-to-end manner. This module is coupled with a pseudo-labelling mechanism [Zou *et al.*, 2021] to efficiently train our model from weak labels in a self-supervised manner. By simply providing image-level annotations of past classes on a small number of images, we significantly improve the mIoU of both past and new classes in all settings. We also conduct experiments by reducing the amount of training images with pixel-level annotations on Pascal-VOC [Everingham *et al.*, 2015] while the remaining part is weakly-labelled. Our experiments show that our weakly-supervised approach successfully outperforms a corresponding baseline that is limited to pixel-level annotations. Interestingly, we observe that in some settings, training with less fully-supervised data sometimes provides better results.

Moreover, we propose two variants of a novel rehearsal strategy that takes the form of a mix between data augmentation and knowledge distillation. Instead of simply adding past images in current batches, the first variant crops objects from past images and randomly pastes them on new images. The second variant applies a similar idea at the feature-level: a memory bank module extracts intermediate representations of past classes from the old model and stochastically injects them in the feature maps of the new model while adjusting the groundtruth segmentation mask accordingly. Contrarily to common rehearsal strategies, this second approach does not need access to raw images after the extraction of intermediate representations. This reduces the privacy issues often raised against rehearsal approaches [Delange *et al.*, 2021], and equalizes or outperforms the first variant with raw images. Our experiments show that, with as little as 20 examples per class, we can significantly improve the mIoU in most settings.

In summary, our contributions are as follows:

- We propose a data-efficient weakly-supervised framework for CSS to leverage weakly-labelled data of **current classes** in addition to fully-annotated examples. Our approach localizes the presence of objects with Grad-CAM and learns to improve the pseudo-labels proposals in an end-to-end fashion.

- We explore a new realistic scenario of CSS in which objects of **past classes** appearing on new images are labelled at the image-level. Our experiments show that these weak labels can significantly reduce the effects of background shift and thus improve CSS at a small cost, which is of strong practical importance.

- We propose a novel rehearsal strategy at the intersection of data augmentation and knowledge distillation. Our experiments show that our feature-level rehearsal often outperforms the image-level one, as well as reducing the privacy issues raised by keeping raw images.

2 Related work

2.1 Semantic segmentation

Semantic segmentation aims to produce pixel-level class annotations, a fundamental issue for computer vision. This task has significantly improved since the development of fully-convolutional networks [Long *et al.*, 2015]. Modelling contextual information has then been a successful avenue of improvement since extracting a more holistic view helps modelling long-range inter-relations between pixels and to produce more consistent semantic masks. Notably, context has been modelled with dilated convolutions [Chen *et al.*, 2017] which increase the receptive field of convolutional filters. In a similar vein, PSPNet [Zhao *et al.*, 2017] introduced the Pyramid Pooling Module to aggregate features at different scales. Atrous Spatial Pyramid Pooling has then been popularized by Deeplab-v3 [Chen *et al.*, 2018] which combines Pyramid Pooling and dilated convolutions for better performance. These approaches, however, have been developed for static datasets and they still struggle in continual learning scenarios.

2.2 Continual learning

Continual learning aims to provide solutions to the practical need to increment existing models with new classes without retraining them from scratch. The main challenge is catastrophic forgetting, i.e. performance on previous classes quickly deteriorates while learning novel ones [Delange *et al.*, 2021]. Strategies against catastrophic forgetting are often categorized as regularization-based, rehearsal-based, and architectural [Serra *et al.*, 2018].

Regularization methods [Zenke *et al.*, 2017; Wu *et al.*, 2019; Li and Hoiem, 2017] apply additional losses to prevent the existing model from erasing previous knowledge. Among these, knowledge distillation is a popular technique [Li and Hoiem, 2017]. In the context of CSS, [Michieli and Zanuttigh, 2019] studied the application of distillation losses on the encoder, the decoder and the output layer. The authors of MiB [Cermelli *et al.*, 2020] then highlighted the challenge of background shift that naturally arises in CSS, which has been neglected in previous work. The authors also introduced a more realistic experimental scenario in which past classes are not annotated if they appear on images of novel classes. Recently, a few approaches have built on top of MiB. For instance PLOP [Douillard *et al.*, 2021] applies multi-scale and multi-layer distillation losses to preserve the behavior of the old network, which also produces pseudo-labels of background objects. SDR [Michieli and Zanuttigh, 2021] shapes the latent space with metric learning principles to accommodate new classes without interfering with previous ones. RECALL uses synthetic replays produced by a Generative Adversarial Network and large unlabelled datasets crawled from the web. Our contributions are orthogonal to the approaches cited above, therefore for simplicity we also build on top of MiB which constitutes our baseline.

Rehearsal-based methods [Prabhu *et al.*, 2020] are often the most successful to prevent forgetting as they store past examples to replay them with new training data. A typical approach is the one used by GDumb [Prabhu *et al.*, 2020] that simply selects representative examplars of past classes and adds them to the training set of new classes. In our work, we propose two variants of a new rehearsal strategy for CSS; one that dynamically imprints past examples in novel images, and one that replays intermediate representations. The latter is more related to REMIND [Hayes *et al.*, 2020]. Our strategy nonetheless differs in several aspects. Notably, REMIND applied compressed representations of past classes with a variant of manifold mixup for Visual Question Answering problems, while we apply past representations by adapting Copy-Paste [Ghiasi *et al.*, 2021] for semantic segmentation.

3 Our approach

3.1 Preliminaries

Semantic segmentation aims to produce pixel-level labels $\hat{\mathbf{M}} \in \mathcal{C}^{H \times W}$ given an image $x \in \mathbb{R}^{H \times W \times 3}$, where H and W are the height and width, respectively, and $\mathcal{C}$ is the set of classes. In class-incremental scenarios, classes are seen in n successive steps $t = \{1, ..., n\}$ such that $\mathcal{C}^t \subset \mathcal{C}$. We follow the experimental setup introduced by [Cermelli *et al.*, 2020] in which, for each step t, groundtruth semantic masks $\mathbf{M}^t$ are only available for current classes $\mathcal{C}^t$. This implies that past classes $\mathcal{C}^{1:t-1}$ are labelled as background as well as future classes $\mathcal{C}^{t+1:n}$ considered as unknown. The goal of CSS is thus to successfully increment a model $f_\theta^{t-1}(x) = \hat{\mathbf{M}}^{1:t-1}$, parameterized by θ, to a model $f_{\theta'}^{t}(x) = \hat{\mathbf{M}}^{1:t}$ that performs well on both past and new classes. Contrarily to most previous work, which are limited to pixel-level supervision, we additionally consider a weakly-supervised scenario in which image-level labels of past or new classes $y \in \mathcal{C}^{1:t}$ are available for some images.

3.2 Model overview

Our first intuition is to design a dedicated branch that is specialized in the production of pseudo-labels from weakly-labelled data and to learn from a consistency loss [Zou *et al.*, 2021].

This branch leverages class-activation maps (CAM) [Zhou *et al.*, 2016] to localize classes on input images and learns to refine these localization masks in an end-to-end manner with a self-attention mechanism [Vaswani *et al.*, 2017]. Then, we propose two variants of rehearsal that take the form of a mix between data augmentation and knowledge distillation to complement the pseudo-labels of weakly-supervised learning. Finally, we apply knowledge distillation losses to prevent forgetting. The total loss is then defined as:

$$\mathcal{L} = \mathcal{L}_s + \mathcal{L}_w + \mathcal{L}_{cls} + \mathcal{L}_{kd}, \tag{1}$$

where $\mathcal{L}_s$ is a supervised loss, $\mathcal{L}_w$ is a weakly-supervised loss, $\mathcal{L}_{cls}$ is a multi-label classification loss used to produce CAMs, and $\mathcal{L}_{kd}$ is a knowledge distillation loss.

3.3 Weakly-supervised learning for CSS

We now describe the mechanism by which our model can learn from image-level labels. We design a dedicated module that produces pseudo-labels to train the main network as shown in Fig. 1. Specifically, our strategy combines Grad-CAM [Selvaraju *et al.*, 2017] to obtain coarse localization masks, a self-attention mechanism to automatically learn to propagate and improve the quality of these masks, and strong data augmentation with consistency training to perform weakly-supervised learning from pseudo-labels.

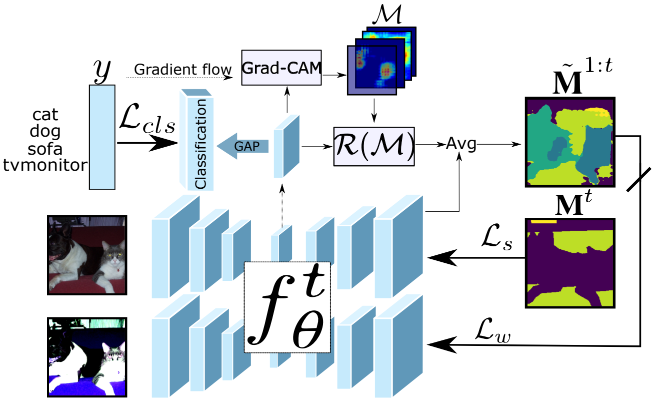

Figure 1: Overview of the weakly-supervised learning. The feature map given by the encoder is used to extract CAMs with Grad-CAM. These CAMs are then refined by the self-attention module $\mathcal{R}$ and combined with the decoder prediction. The resulting pseudo-labels $\hat{\mathbf{M}}^{1:t}$ then serves in the weakly-supervised loss $\mathcal{L}_w$ for self-training by comparing the prediction of the same model given a strongly augmented image. We stop the gradient after the production of pseudo-labels which is represented by the broken line.

Grad-CAM. To produce the pseudo-labels $\tilde{\mathbf{M}}^{1:t}$, we first attach a localization branch to the output of the encoder network. This localization branch contains a multi-label classification head to enable the extraction of coarse masks by using CAMs [Zhou *et al.*, 2016]. This classification head is trained with the image-level labels y according to $\mathcal{L}_{cls}$, which is a standard multi-label cross-entropy loss. We use Grad-CAM [Selvaraju *et al.*, 2017], a variant of CAM that consists of using the class-specific gradients of a classification layer. These gradients are then backpropagated up to the last spatial feature maps before pooling to compute a localization mask for each class, i.e. $\mathcal{M} \in \mathbb{R}^{H' \times W' \times |C^{1:t}|}$, where H' and W' are the height and width of the last feature maps, respectively. CAMs can be obtained without groundtruth annotations but they can be significantly improved when image-level labels are provided by filtering out the CAMs of classes that are not present on the image.

Masks refinement. As these localization masks are too coarse for segmentation, they are then refined by a given function $\mathcal{R}(\mathcal{M})$ (see Fig. 1), which in our experiments is modelled by a self-attention layer to automatically learn the pairwise similarities of image regions and propagates CAMs accordingly. The refined CAMs are then resized to the output dimensions, normalized with softmax, and averaged with the softmax predictions of the decoder to produce the pseudo-labels $\tilde{\mathbf{M}}^{1:t}$ for the weakly-supervised loss $\mathcal{L}_w$.

Consistency loss between pseudo-labels and the predictions of strongly-augmented data. To train our model from self-produced pseudo-labels, we exploit the semantic invariance of images to strong data augmentation [1]. Indeed, the semantic segmentation masks predicted by the model given an image x and its distorted version x_{aug} should be consistent with each other since the object classes are not affected by the alteration of pixel values (for example see the inputs in Fig 1). We exploit this invariance by training our model such that the output of $f_\theta^t(x_{aug})$ is consistent with the pseudo-labels $\tilde{\mathbf{M}}^{1:t}$ based on the following cross-entropy consistency loss:

$$\mathcal{L}_w = -\frac{1}{WH}\sum_{w,h}^{W,H}\sum_{c\in\mathcal{C}^{1:t}}\tilde{\mathbf{M}}^{1:t}[w,h,c]\log f_\theta^t(x_{aug})[w,h,c]. \tag{2}$$

3.4 Supervised and distillation losses

The supervised and distillation losses each comprise multiple terms. Indeed, when pixel-level supervision is available, we train the main network and the refinement module $\mathcal{R}$ with groundtruth segmentation masks according to the unbiased cross-entropy loss ($\mathcal{L}_{unce}$) proposed in [Cermelli *et al.*, 2020]. Therefore, the supervised loss is the following:

$$\mathcal{L}_s = \mathcal{L}_{unce}(f^t(x), \mathbf{M}^t) + \mathcal{L}_{unce}(\mathcal{R}^t(x), \mathbf{M}^t). \tag{3}$$

We also regularize, with unbiased distillation losses ($\mathcal{L}_{unkd}$) [Cermelli *et al.*, 2020], the outputs of the segmentation network given the original input x and the strongly-augmented input x_{aug}, and the outputs of the refinement module $\mathcal{R}$. The total distillation loss is thus defined as the following:

$$\begin{aligned}\mathcal{L}_{kd} =& \mathcal{L}_{unkd}(f^t(x), f^{t-1}(x)) + \\ & \mathcal{L}_{unkd}(f^t(x_{aug}), f^{t-1}(x_{aug})) + \\ & \mathcal{L}_{unkd}(\mathcal{R}^t(x), \mathcal{R}^{t-1}(x)).\end{aligned} \tag{4}$$

Note that, contrarily to the main network, the refinement module is not regularized with strongly-augmented data since this branch never uses this input.

3.5 Rehearsal as data augmentation

We also propose two rehearsal strategies in the form of a data augmentation technique that is especially suited for continual learning (see Figure 2). We begin by introducing an additional set of notations to define our two strategies of rehearsal in a unified framework.

Notations. We define the input of layer l as the tensor $x^l \in \mathbb{R}^{m\times n\times d}$ such that $x^{l+1} = g^l(x^l)$, where m and n are the spatial dimensions, d is the number of channels, and g^l is the l-th layer of the model. Note that the initial input is $x^0 = x \in \mathbb{R}^{H\times W\times 3}$, i.e. the input RGB image. This notation enables us to conveniently define replay of raw images (i.e. x^0) and replay of intermediate features (i.e. x^l for $l > 0$) in a unified framework. We also define the binary segmentation mask of class c, i.e. $\mathbf{M}_c = \mathbf{M}[:,:,c] \in \mathbb{R}^{H\times W}$, and $\mathbf{M}_c^l \in \mathbb{R}^{m\times n}$ which is the binary segmentation mask of class c that has been downsized to the same spatial dimensions than x^l. For conciseness, in the text we call $x^l \odot \mathbf{M}_c^l$ the masked features, even though the particular case x^0 corresponds to raw pixels.

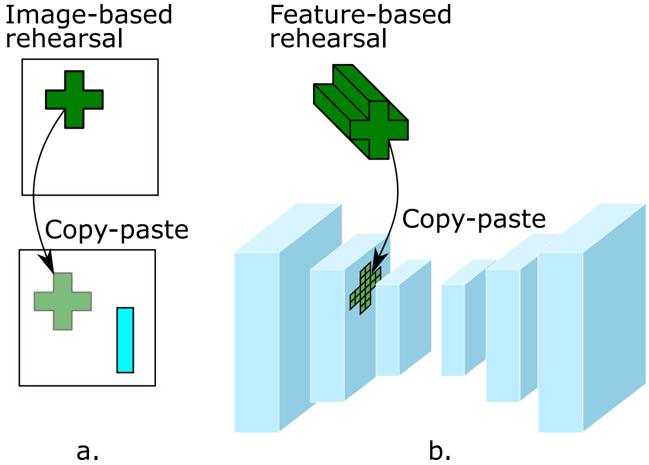

Figure 2: Overview of our two rehearsal strategies. In this simplified example, the green cross and cyan rectangle are respectively a past and a novel class. a. The image-based rehearsal consists of copying examples of previous classes stored in memory on new training images. b. The feature-based rehearsal keeps representations of previous classes and transfers them in the current network during the forward pass while learning novel classes.

Copy-Paste augmentation for rehearsal. We model our rehearsal in the form of a data augmentation technique that intertwines the learning of past and current classes. Specifically, we explore the Copy-Paste technique [Ghiasi *et al.*, 2021] that consists of copying objects from an image A into an image B while adapting the groundtruth segmentation mask accordingly.

Different from its original role of standard data augmentation, here we adapt the Copy-Paste algorithm as a way of revisiting past classes while learning novel ones. Our first adaptation consists of copying the pixels of past objects from a memory bank into images of the current step (see Figure 2a). We then make a step further by suggesting another adaptation that uses features rather than pixels. Our second adaptation stores intermediate representations of feature maps and replays them in the new model (see Figure 2b). The advantages of this variant are the following: 1) it contributes to the protection of privacy since it stores abstract features rather than RGB pixels, which is important in some applications [Delange *et al.*, 2021]; 2) it increases the diversity of replay data as the past representations can be extracted from many different layers; and 3) it is more biologically plausible than replaying raw images, by mimicking a more realistic activity of the brain similar to how neural pathways are reactivated during memory consolidation [Walker and Stickgold, 2004].

Writing in memory. In our two variants of rehearsal, we keep a memory bank of size $N = \sum_{c\in\mathcal{C}^{1:t-1}} n_c$, where n_c is the number of stored examples for class c. This memory bank only keeps masked features, extracted from n_c examples per class, by using the corresponding groundtruth mask. Formally, a new memory of class c is formed by storing the tuple $(x^l \odot \mathbf{M}_c^l, \mathbf{M}_c)$. In this work, we simply set $n_c = \frac{N}{|\mathcal{C}^t|}$ where N is a hyperparameter and the examples are sampled uniformly.

[1] In this work, we use color jittering as the strong augmentation function.

Reading from memory. For each training data encountered during step t, there is a probability p that we randomly sample a tuple from the memory bank and inject the masked features on the new tensor x^l_{new} at a random location. Namely, if a training example at step t is augmented with our strategy, then the modified tensor becomes:

$$\tilde{x}^l_{new} = \mathbf{M}^l_c \odot x^l_{old} + (1 - \mathbf{M}^l_c) \odot x^l_{new}. \qquad (5)$$

The next layer g^l thus receives $\tilde{x}^l_{new}$ as input instead of x^l_{new}.

Similarly, the corresponding groundtruth values are modified accordingly by pasting the sampled class on the segmentation mask:

$$\bar{\mathbf{M}}_{new}[:, :, c] = \mathbf{M}_c + (1 - \mathbf{M}_c) \odot \mathbf{M}_{new}[:, :, c] \qquad (6)$$

4 Experiments

4.1 Implementation details

During our experiments we use Deeplab-v3 [Chen *et al.*, 2017] as the segmentation decoder with a ResNet-101 backbone [He *et al.*, 2016] pretrained on ImageNet [Russakovsky *et al.*, 2015]. We append a global average pooling layer and a linear classifier after the last ResNet module to extract localization masks with Grad-CAM (see Section 3.3). The mask refinement module consists of a self-attention layer with hidden dimensions of 128. The last feature map of ResNet-101 is projected by two distinct sets of 1×1 convolution filters to form the "keys" and "queries", and the class-wise localization masks extracted by Grad-CAM act as the "values". Lastly, stop-gradient is applied to the pseudo-labels that are produced since we only want the training signals to flow in the main network.

We train our models with SGD with momentum on 4 Nvidia A100 GPUs with a total batch size of 24 for 30 epochs for each step. An initial learning rate of 0.01 and 0.001 are used for the first and subsequent steps, respectively, with a polynomial decay of power 0.9. Similar to [Cermelli *et al.*, 2020], the hyper-parameters are searched by keeping 20% of the training set for validation. Among the most influential hyper-parameters, we found that it was important to start learning from weakly-supervised data only after a few epochs, i.e. 5-15 epochs, especially in our experiments with scarcer pixel-level annotations. Additionally, with weakly-supervised data a threshold of 0.75 is used to only keep the most confident predictions as pseudo-labels. The weights for distillation losses are 100 in the 19-1 and 15-1 scenarios, and 10 for the 15-5 scenarios.

4.2 Experimental settings

Dataset. We train and evaluate our model in various continual learning scenarios built from the commonly used PASCAL-VOC 2012 dataset [Everingham *et al.*, 2015]. This dataset contains a *train* split of 10,582 images and a *val* split of 1,449 images used for testing. A total of 20 semantic classes, plus the background, are learned in separated sets to simulate incremental steps. In each experiment, the specific scenario is designated by the amount of classes in the first step, followed by the number of classes in each of the next steps. For instance, 15-5 means that the first step includes 15 classes and the second step includes the remaining 5 classes. Similarly, 15-1 indicates that after the first step of 15 classes, each remaining class is learned in a distinct step (for a total of 6 steps).

Disjoint *vs* overlapped setups. To compare our approach in similar settings to previous work, we follow the common experimental setups proposed by [Cermelli *et al.*, 2020] which further divides each scenario into a *disjoint* setup or an *overlapped* setup. In the *disjoint* setup, the training set at a given step is formed by images that contain either old or new classes, but the images that contain future classes are removed. On the opposite, the *overlapped* setup also includes images that contain future classes if they contain at least one pixel of a novel class. Note that old and future classes are always labelled as background in the groundtruth semantic masks of training images.

Metrics and baselines. We evaluate the models based on the mean Intersection-over-union (mIoU). In our tables we differentiate the mIoU for the old and novel classes to have a better picture of incremental steps. We compare our models to standard approaches such as EWC [Kirkpatrick *et al.*, 2017], LwF-MC [Rebuffi *et al.*, 2017], ILT [Michieli and Zanuttigh, 2019] and MiB [Cermelli *et al.*, 2020]. Most of their results are from [Cermelli *et al.*, 2020]. We also compare our approach to recent state-of-the-art models such as SDR [Michieli and Zanuttigh, 2021], RECALL [Maracani *et al.*, 2021] and PLOP [Douillard *et al.*, 2021]. Note that MiB is our most interesting baseline as our contributions are built on top of this approach.

4.3 Results

Our experiments aim to evaluate several aspects of our proposed approach. First, we aim to evaluate the contribution of leveraging image-level labels of old classes while learning novel categories. Second, we want to evaluate and compare our image-level and feature-level rehearsal techniques. Third, we aim to compare our contributions in several scenarios of weakly-supervised continual learning by varying the proportion of fully-annotated data.

4.4 Learning with image-level labels of previous classes.

We begin by benchmarking our model on the same fully-supervised experimental setup as previous work with image-level labels of previous classes. Table 1 shows the mIoU on Pascal-VOC for three scenarios. First, it appears that our approach stands out more in the 15-5 settings while scenarios with one class per step are more challenging, which reveal the benefits and limits of leveraging weak labels and rehearsal. For instance, we successfully verify the ability of our model to leverage image-level labels of old classes. Indeed, `Ours` consistently outperforms MiB by a large margin, both on old and new classes. For instance, in the 19-1 overlapped setting, the mIoU of our approach that leverages weak supervision of old classes obtains 72.3% on old classes and 31.0% on novel ones, whereas MiB obtains 70.2% and 22.1%, respectively.

	19-1						**15-5**						**15-1**					
	Disjoint			**Overlapped**			**Disjoint**			**Overlapped**			**Disjoint**			**Overlapped**		
Method	*1-19*	*20*	*all*	*1-19*	*20*	*all*	*1-15*	*16-20*	*all*	*1-15*	*16-20*	*all*	*1-15*	*16-20*	*all*	*1-15*	*16-20*	*all*
FT	5.8	12.3	6.2	6.8	12.9	7.1	1.1	33.6	9.2	2.10	33.1	9.8	0.2	1.8	0.6	0.2	1.8	0.6
EWC	23.2	16.0	22.9	26.9	14.0	26.3	26.7	37.7	29.4	24.3	35.5	27.1	0.3	4.3	1.3	0.3	4.3	1.3
LwF-MC	63.0	13.2	60.5	64.4	13.3	61.9	67.2	41.2	60.7	58.1	35.0	52.3	4.5	7.0	5.2	6.4	8.4	6.9
ILT	69.1	16.4	66.4	67.1	12.3	64.4	63.2	39.5	57.3	66.3	40.6	59.9	3.7	5.7	4.2	4.9	7.8	5.7
SDR	69.9	37.3	68.4	69.1	32.6	67.4	73.5	47.3	67.2	75.4	52.6	69.9	59.2	12.9	48.1	44.7	21.8	39.2
RECALL–GAN	65.2	**50.1**	65.8	67.9	53.5	68.4	66.3	49.8	63.5	66.6	50.9	64.0	66.0	44.9	62.1	65.7	47.8	62.7
RECALL–WEB	65.0	47.1	65.4	68.1	**55.3**	68.6	69.2	**52.9**	66.3	67.7	54.3	65.6	**67.6**	**49.2**	**64.3**	**67.8**	**50.9**	64.8
PLOP	**75.4**	38.9	**73.6**	**75.4**	37.4	**73.5**	71.0	42.8	64.3	75.7	51.7	70.1	57.9	13.7	46.5	65.1	21.1	54.6
MiB	69.6	25.6	67.4	70.2	22.1	67.8	71.8	43.3	64.7	75.5	49.4	69.0	46.2	12.9	37.9	35.1	13.5	29.7
Ours	71.6	28.2	69.4	72.3	31.0	70.2	73.9	48.0	67.4	75.9	53.4	70.3	48.2	14.2	39.7	41.7	14.8	35.0
Ours–x^0	73.2	33.1	71.2	75.3	32.6	73.1	**76.5**	51.6	**70.0**	76.2	54.8	70.8	49.3	14.1	40.5	44.0	15.3	36.8
Ours–x^3	73.2	36.5	71.4	73.2	36.0	71.3	75.7	52.2	69.8	**76.9**	**55.8**	**71.6**	51.3	14.5	42.1	44.7	15.1	37.3

Table 1: Mean-IoU on VOC2012 val set. We denote by Ours–x^0 our model that uses image-based rehearsal and by Ours–x^3 our model that uses feature-based rehearsal with features from the 3^{th} ResNet module. Both of these models only use 20 examples per old class. The best and second best results are shown in bold and underlined, respectively.

Although learning one novel class in the 19-1 scenarios seems more challenging to our method, we obtain competitive results on old classes. Similarly, we still improve over MiB in the 15-1 scenarios. Together, these results show that our model can effectively leverage weak annotations to reduce catastrophic forgetting.

Image- and feature-based rehearsal. We also show the results of our rehearsal techniques with a small budget of 20 examples per old classes. For our feature-level rehearsal, we extract the masked features from the output of the third module of ResNet-101 (denoted by Ours–x^3) since it gave slightly better results than other modules in most scenarios on the validation set. In Table 1, we can see that both of our rehearsal techniques strongly improve the mIoU of our model in all settings. Note that using a memory bank of old classes also improves the results even on new classes. For instance in the 15-5 disjoint setting, our usage of feature-based memory improves the mIoU to 52.2% on novel classes, as compared to 48.0% for our model without memory. This is explained by the fact that false positives of new classes are reduced since the training process is balanced with examples of past classes. Finally, when comparing our two variants of rehearsal, we can see that both obtain generally similar results. The feature-based rehearsal seems to allow better learning of novel classes but sometimes at the expense of a performance drop on previous ones. For instance, in the 15-5 disjoint setup, Ours–x^3 obtains 75.7%/52.2% on past and new classes, respectively, while Ours–x^0 obtains 76.5%/51.6%, respectively.

Summary. In scenarios with the addition of one class at a time, our approach generally obtains overall performances that are competitive with state-of-the-art methods. For instance, RECALL outperforms our approach on the new class in the 19-1 setups but at the expense of mIoU on previous ones, even if RECALL used 500 examples per old class generated from a GAN. The main limitation of our approach appears in the 15-1 scenario, where stronger knowledge distillation (i.e. PLOP) and metric learning principles (i.e. SDR) seem to exceed the benefits of weak labels and rehearsal. Since our contributions are orthogonal to these methods, future work should investigate the combination of weakly-supervised mechanisms and rehearsal with other strategies such as theirs.

It appears from our experiments that the more suitable scenario to leverage our weakly-supervised learning and rehearsal comes when there are many classes at each step. Our approach especially shines in 15-5 settings where we outperform state-of-the-art methods by strongly reducing forgetting of previous classes, which was the main focus of using image-level labels and rehearsal, as well as performing better on novel classes.

4.5 Learning with image-level labels of *new* classes.

We then explored the ability of our model to leverage additional data with only image-level labels of new classes. To do so, we reduce the fraction of fully-supervised data while the remaining portion becomes weakly-annotated with image-level labels, both for old and new classes. We thus compare our approach with its corresponding baseline, i.e. MiB, which cannot use the weakly-annotated portion. These results are shown in Table 2 for the two setups of 19-1 and 15-5.

Less training data can be better. We gradually reduced the amount of training data and their quality by providing image-level labels instead of pixel-level annotations for a portion of the dataset, and we expected that the performance of all models would deteriorate accordingly. Surprisingly, while it is true that our models leveraging weakly-annotated data outperform a baseline that is limited to pixel-level annotations, we observe that in many cases the best results are obtained when the models are trained with less than 100% of fully-supervised data, especially for the novel classes. For instance, for the old/new classes respectively, MiB obtains 72.2%/28.9% MIoU in the 19-1 overlapped setup with 1/2 of data, whereas it obtained 70.2%/22.1% in the fully-supervised setting (Table 1). The same phenomenon is observed in the 15-5 disjoint and overlapped setups, where MiB respectively obtained 71.8%/43.3% and 75.5%/49.4% in the fully-supervised setting, while it obtains 71.8%/44.9% and 74.4%/51.0%, respectively, with half of the training data.

Similarly, our approach with and without a memory bank obtain better results with less training data in a few scenarios. Notably, our model without memory gets 75.0%/53.2% in

		19-1																	
		Disjoint									**Overlap**								
		3/4			1/2			1/4			3/4			1/2			1/4		
Supervision	**Method**	1-19	20	all	1-19	20	all	1-19	20	all	1-19	20	all	1-19	20	all	1-19	20	all
Full	MiB	69.8	24.7	67.5	69.6	25.6	67.4	69.8	12.3	66.9	71.5	24.5	69.2	72.2	28.9	70.0	71.6	13.2	68.7
+ Weak	Ours	71.1	28.3	69.0	71.6	24.2	69.2	71.5	21.6	69.0	72.3	34.6	70.4	72.7	28.7	70.5	73.0	21.4	70.4
	Ours–x^0	**74.6**	32.1	**72.5**	**74.3**	**32.9**	**72.3**	**73.2**	19.7	**70.5**	73.8	**41.9**	72.2	**74.1**	**35.0**	**72.1**	**73.1**	**31.3**	**71.0**
	Ours–x^3	72.3	**35.8**	70.4	72.5	28.4	70.3	70.5	**24.9**	68.2	**74.2**	40.8	**72.5**	73.5	31.9	71.4	72.8	29.7	70.6

		15-5																	
		Disjoint									**Overlap**								
		3/4			1/2			1/4			3/4			1/2			1/4		
		1-15	16-20	all	1-15	16-20	all	1-15	16-20	all	1-15	16-20	all	1-15	16-20	all	1-15	16-20	all
Full	MiB	70.3	43.4	63.6	71.8	44.9	65.1	71.6	41.2	64.0	74.2	49.6	68.5	74.4	51.0	68.6	73.9	48.2	67.5
+ Weak	Ours	73.8	48.8	67.6	**75.0**	**53.2**	**69.6**	72.5	49.2	66.7	74.9	51.2	69.0	75.1	51.6	69.2	**74.7**	51.0	68.8
	Ours–x^0	75.1	**51.7**	69.3	74.9	53.1	69.5	**73.4**	**51.1**	**67.8**	75.0	53.2	69.6	75.7	53.7	70.2	74.6	54.1	**69.5**
	Ours–x^3	**75.3**	51.5	**69.4**	74.8	52.8	69.3	73.3	50.7	67.7	**75.4**	**54.4**	**70.2**	**75.9**	**54.8**	**70.6**	74.5	**54.5**	**69.5**

Table 2: Comparison between our models and a corresponding baseline on VOC2012 val set by reducing the proportion of pixel-level annotations. The remaining portion is weakly-labelled.

the 15-5 disjoint setup when half of the data are annotated at the pixel-level and the other half is only labelled at the image level, while it obtained 73.9%/48.0% with 100% of fully-supervised data, as was shown in Table 1.

Dynamics between forgetting and learning. Generally, it is expected that training machine learning models with more and cleaner data leads to better generalization, thus obtaining better results. However, in continual settings, learning new classes is coupled with the deterioration of old ones since the new weights tend to move away from the previous solution. To address that, many approaches have considered regulating the updates of weights, for instance with a quadratic regularization penalty [Kirkpatrick *et al.*, 2017]. In our case, we hypothesize that a similar regularization effect might occur when training with less data. Optimizing the network on fewer examples of novel classes could tend to find a reasonable minimum in the parameter space that is closer to the previous weights. On the opposite, finding a set of weights that satisfy a larger training set might lead to weights that are farther from the previous optimization step, therefore increasing forgetting and biasing the network towards novel classes. A proposition resulting from this would be to investigate few-shot learning approaches, especially meta-learning strategies, to improve CSS.

5 Conclusion

We have presented a continual semantic segmentation model that, differently from previous work, is also able to leverage weakly-annotated data. We have exploited this ability by exploring two scenarios: one in which past classes are annotated at the image-level, and another in which pixel-level annotations of new classes are scarcer while additional data is available in the form of weak labels. We showed that weak annotations can greatly improve segmentation performance, both on previous and novel classes. Interestingly, we also found that using less training data can sometimes be beneficial, and we discussed some hypotheses explaining this phenomenon.

Finally, we have also proposed two variants of a rehearsal technique that shares principles that are common to both data augmentation and knowledge distillation. Our results showed that with as little as 20 examples per class, we can significantly improve the mean intersection-over-union in several continual semantic segmentation scenarios.

Acknowledgements

This work was granted access to the HPC resources of Compute Canada.

References

[Cermelli *et al.*, 2020] Fabio Cermelli, Massimiliano Mancini, Samuel Rota Bulo, Elisa Ricci, and Barbara Caputo. Modeling the background for incremental learning in semantic segmentation. In *CVPR*, pages 9233–9242, 2020.

[Chen *et al.*, 2017] 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. *TPAMI*, 40(4):834–848, 2017.

[Chen *et al.*, 2018] Liang-Chieh Chen, Yukun Zhu, George Papandreou, Florian Schroff, and Hartwig Adam. Encoder-decoder with atrous separable convolution for semantic image segmentation. In *ECCV*, pages 801–818, 2018.

[Delange *et al.*, 2021] Matthias Delange, Rahaf Aljundi, Marc Masana, Sarah Parisot, Xu Jia, Ales Leonardis, Greg Slabaugh, and Tinne Tuytelaars. A continual learning survey: Defying forgetting in classification tasks. *TPAMI*, 2021.

[Douillard *et al.*, 2021] Arthur Douillard, Yifu Chen, Arnaud Dapogny, and Matthieu Cord. Plop: Learning without forgetting for continual semantic segmentation. In *CVPR*, pages 4040–4050, 2021.

[Everingham *et al.*, 2015] Mark Everingham, SM Ali Eslami, Luc Van Gool, Christopher KI Williams, John Winn, and Andrew Zisserman. The pascal visual object classes challenge: A retrospective. *IJCV*, 111(1):98–136, 2015.

[Ghiasi *et al.*, 2021] Golnaz Ghiasi, Yin Cui, Aravind Srinivas, Rui Qian, Tsung-Yi Lin, Ekin D Cubuk, Quoc V Le, and Barret Zoph. Simple copy-paste is a strong data augmentation method for instance segmentation. In *CVPR*, pages 2918–2928, 2021.

[Hayes *et al.*, 2020] Tyler L Hayes, Kushal Kafle, Robik Shrestha, Manoj Acharya, and Christopher Kanan. Remind your neural network to prevent catastrophic forgetting. In *ECCV*, pages 466–483. Springer, 2020.

[He *et al.*, 2016] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *CVPR*, pages 770–778, 2016.

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

[Li and Hoiem, 2017] Zhizhong Li and Derek Hoiem. Learning without forgetting. *TPAMI*, 40(12):2935–2947, 2017.

[Long *et al.*, 2015] Jonathan Long, Evan Shelhamer, and Trevor Darrell. Fully convolutional networks for semantic segmentation. In *CVPR*, pages 3431–3440, 2015.

[Maracani *et al.*, 2021] Andrea Maracani, Umberto Michieli, Marco Toldo, and Pietro Zanuttigh. Recall: Replay-based continual learning in semantic segmentation. In *ICCV*, pages 7026–7035, 2021.

[Michieli and Zanuttigh, 2019] Umberto Michieli and Pietro Zanuttigh. Incremental learning techniques for semantic segmentation. In *ICCVW*, pages 3205–3212, 2019.

[Michieli and Zanuttigh, 2021] Umberto Michieli and Pietro Zanuttigh. Continual semantic segmentation via repulsion-attraction of sparse and disentangled latent representations. In *CVPR*, pages 1114–1124, 2021.

[Prabhu *et al.*, 2020] Ameya Prabhu, Philip HS Torr, and Puneet K Dokania. Gdumb: A simple approach that questions our progress in continual learning. In *ECCV*, pages 524–540. Springer, 2020.

[Rebuffi *et al.*, 2017] Sylvestre-Alvise Rebuffi, Alexander Kolesnikov, Georg Sperl, and Christoph H Lampert. icarl: Incremental classifier and representation learning. In *CVPR*, pages 2001–2010, 2017.

[Russakovsky *et al.*, 2015] Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, et al. Imagenet large scale visual recognition challenge. *IJCV*, 115(3):211–252, 2015.

[Selvaraju *et al.*, 2017] Ramprasaath R Selvaraju, Michael Cogswell, Abhishek Das, Ramakrishna Vedantam, Devi Parikh, and Dhruv Batra. Grad-cam: Visual explanations from deep networks via gradient-based localization. In *ICCV*, pages 618–626, 2017.

[Serra *et al.*, 2018] Joan Serra, Didac Suris, Marius Miron, and Alexandros Karatzoglou. Overcoming catastrophic forgetting with hard attention to the task. In *ICML*, pages 4548–4557. PMLR, 2018.

[Vaswani *et al.*, 2017] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *NeurIPS*, pages 5998–6008, 2017.

[Walker and Stickgold, 2004] Matthew P Walker and Robert Stickgold. Sleep-dependent learning and memory consolidation. *Neuron*, 44(1):121–133, 2004.

[Wu *et al.*, 2019] Yue Wu, Yinpeng Chen, Lijuan Wang, Yuancheng Ye, Zicheng Liu, Yandong Guo, and Yun Fu. Large scale incremental learning. In *CVPR*, pages 374–382, 2019.

[Yu *et al.*, 2020] Lu Yu, Xialei Liu, and Joost van de Weijer. Self-training for class-incremental semantic segmentation. *arXiv preprint arXiv:2012.03362*, 2020.

[Zenke *et al.*, 2017] Friedemann Zenke, Ben Poole, and Surya Ganguli. Continual learning through synaptic intelligence. In *ICML*, pages 3987–3995. PMLR, 2017.

[Zhao *et al.*, 2017] Hengshuang Zhao, Jianping Shi, Xiaojuan Qi, Xiaogang Wang, and Jiaya Jia. Pyramid scene parsing network. In *CVPR*, pages 2881–2890, 2017.

[Zhou *et al.*, 2016] Bolei Zhou, Aditya Khosla, Agata Lapedriza, Aude Oliva, and Antonio Torralba. Learning deep features for discriminative localization. In *CVPR*, pages 2921–2929, 2016.

[Zou *et al.*, 2021] Yuliang Zou, Zizhao Zhang, Han Zhang, Chun-Liang Li, Xiao Bian, Jia-Bin Huang, and Tomas Pfister. Pseudoseg: Designing pseudo labels for semantic segmentation. In *ICLR*, 2021.