Class-incremental Continual Learning for Instance Segmentation with Image-level Weak Supervision

Yu-Hsing Hsieh[1] Guan-Sheng Chen[1] Shun-Xian Cai[1] Ting-Yun Wei[1]
Huei-Fang Yang[2] Chu-Song Chen[1*]
[1]Dept. Computer Science and Information Engineering, National Taiwan University, Taiwan
[2]Dept. Information Management, National Sun Yat-sen University, Taiwan
{r10922024,r10922052,r11922081,r10922010,chusong}@csie.ntu.edu.tw hfyang@mis.nsysu.edu.tw

Abstract

Instance segmentation requires labor-intensive manual labeling of the contours of complex objects in images for training. The labels can also be provided incrementally in practice to balance the human labor in different time steps. However, research on incremental learning for instance segmentation with only weak labels is still lacking. In this paper, we propose a continual-learning method to segment object instances from image-level labels. Unlike most weakly-supervised instance segmentation (WSIS) which relies on traditional object proposals, we transfer the semantic knowledge from weakly-supervised semantic segmentation (WSSS) to WSIS to generate instance cues. To address the background shift problem in continual learning, we employ the old class segmentation results generated by the previous model to provide more reliable semantic and peak hypotheses. To our knowledge, this is the first work on weakly-supervised continual learning for instance segmentation of images. Experimental results show that our method can achieve better performance on Pascal VOC and COCO datasets under various incremental settings[1].

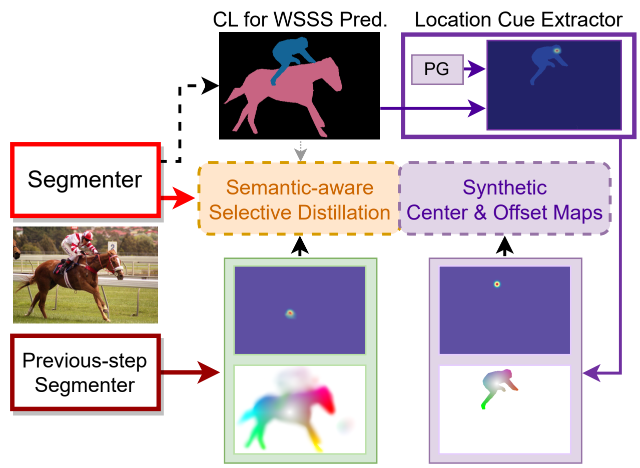

Figure 1. Our weakly-supervised incremental learning model. Assuming “horse” is the old class and “person” is the current during CL, our model leverages semantic knowledge yielded by CL for WSSS to produce synthetic center and offset labels for the current person class. Semantic-aware selective distillation is employed to preserve knowledge of the old horse class to achieve CL for WSIS.

1. Introduction

Continual learning (CL) aims to continually learn from data provided in sequential sessions while avoiding catastrophic forgetting [37, 19]. It has gained significant attention since incrementally learning a model is useful in many applications. CL has two main scenarios. The first, task-incremental CL [33, 54], assumes that we know the task indices of the input data during inference. The second, class-incremental CL [5, 45, 44, 59], assumes that the task index is inaccessible for inference and we aim to classify the data labels of all seen tasks, which is more generally applicable.

Based on the image labels, previous CL works mainly devote to image classification of sequential tasks. In this paper, we take a step forward in class-incremental CL and learn *instance segmentation* (IS) models from the image labels only. To learn an IS model, previous works often need pixel-level boundary annotations of training objects. Recently, methods that can learn instance segmenters incrementally are developed in CL [23, 10]. However, they need expensive pixel-wise supervisions at each incremental learning step. Our approach, on the other hand, requires only cheaper image-level labels that are easily available. To our knowledge, this is the first CL study using weakly supervised image labels for subsequent IS model learning.

On the other hand, IS has been studied for a long time and has made significant progress [18, 8, 26, 29, 38]. Many

*corresponding author.
[1]https://github.com/AI-Application-and-Integration-Lab/CL4WSIS

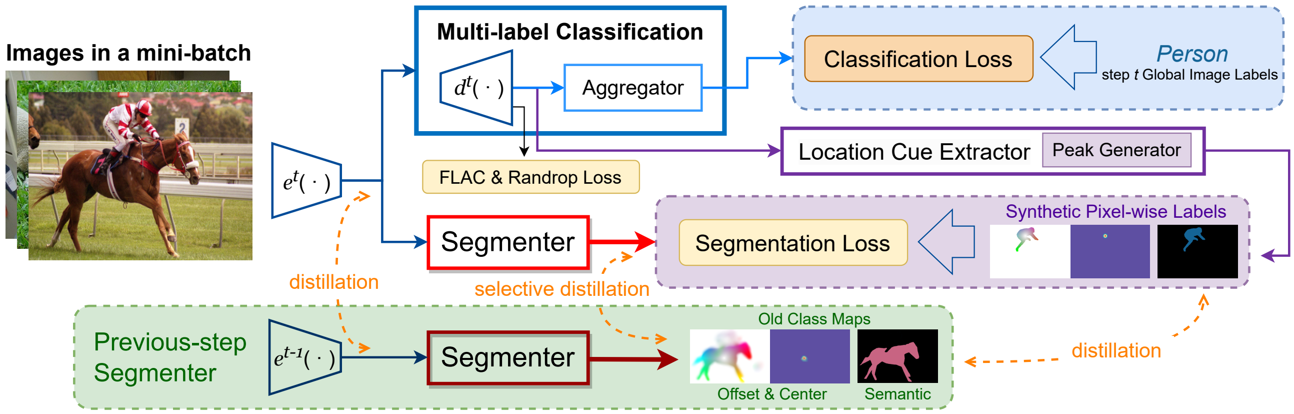

Figure 2. An overview of our CL4WSIS framework. Our model employs an encoder-decoder structure. At the CL step t, the Decoder d^t is responsible for generating Synthetic Pixel-wise Labels, *i.e.* semantic, center and offset maps, for the current classes to guide the Segmenter training. To learn with the Global Image Labels, our Decoder is combined with an Aggregator. Feature-level augmentation consistency (FLAC) and random dropout (Randrop) are further employed to enhance the reliability of WSSS generated from the Decoder. We leverage the instance cues from a Peak Generator (PG) in the Location Cue Extractor for synthetic center and offset maps for the current classes. The knowledge is maintained by distilling the previous knowledge provided by the Previous-step Segmenter through a selective distillation and also through feature distillation. The learned Segmenter is used for the current-step inference and preserved for CL in the next step.

IS models are trained on existing benchmark datasets with pre-specified class labels. However, the learned model can only be used in limited cases of segmenting objects of pre-defined classes, but cannot handle new classes of objects. The lack of class extensibility limits the use of models. A promising approach to this problem is to enable the model to continually learn from newly labeled images. In the class-incremental setting, images collected in a new step can be used to train the model incrementally, where training data from previous steps cannot be used for learning in the new step. This setup has several advantages. For example, the training data in the previous steps may have to be protected and cannot be used in the next step. Joint training data in all steps could also scale up learning and make computational resources unaffordable. However, fine-tuning from the previous-step model to the new-step model easily leads to catastrophic forgetting. Due to the newly added classes, there is also a background shift problem where the background defined in the previous step is not consistent with the background in the new step images.

This paper aims to attain CL in a more effort-saving scenario for IS, where our model learns to predict instance-level segmentation using weakly supervised image labels. We introduce an end-to-end incremental learning model. As semantic segmentation can be generally seen as the union of IS for each class, we upgrade semantic segmentation to IS, as shown in Fig. 1. The model leverages a panoptic segmentation architecture whose decoder can generate semantic, instance center, and offset maps; the three maps can then yield our IS outcome.

To estimate the pixel-level semantic labels from only the image-level labels, it is observed that a single-round solution derived directly from the attention map is often insufficient for sophisticated boundary inference, and thus multi-round training is suggested [2, 34]. Our approach uses an attention mechanism leveraging the global image classification labels to extract the per-pixel location cues, which then helps synthesize the local labels for simultaneously training our Segmenter, as shown in Fig. 2. When training is finished, only the Segmenter is used in the inference stage.

To continually update the model, the Segmenter learned in previous step serves as a teacher for model distillation. Given images of the current step, in addition to training the Segmenter with the synthetic local labels of current classes, the Segmenter also distills from the teacher which provides the probability maps of old classes (Fig. 2). Hence, the model simultaneously learns from both fully supervised pixel-wise probability maps of the old classes and weakly supervised image labels of the new classes. Our method performs CL for WSSS at first and obtains a semantic map. We then perform CL for WSIS (CL4WSIS) leveraging the semantic map later. To handle the CL for WSSS, our learning mechanism addresses the background shift by an early occupation of the highly confident old-class objects found by the teacher model, and guides the seeking of new-class objects in the remaining regions. We also introduce the **augmentation consistency** and **random dropout** strategies to enhance the WSSS learning performance. We develop a **peak generator** to find more reliable location cues of the instances. To further tackle the background shift, we introduce a **selective distillation strategy** that learns the center and offset maps of old classes depending on the intermediate semantic map. Characteristics of our method include:

- As far as we know, we have conducted the first study of the CL4WSIS problem.

• Our method integrates CL and semantic knowledge transfer to IS. Not only can it outperform the previous incremental WSSS method, but it can further achieve IS effectively.

2. Related Work

We briefly review the related works including CL for IS, WSIS, WSSS, WSOD, and weak shot learning.

Continual Learning for Instance Segmentation. Many CL solutions are provided for image classification [41, 44, 52]. Despite recent progress in incremental semantic segmentation [11, 12, 20, 49, 61, 62], CL methods in instance segmentation (CLIS) are still underexplored. Besides the well-known catastrophic forgetting, CLIS is faced with another challenge, the background shift, which is caused by the missing annotations of objects in the old and future classes in the incremental learning step. To tackle the challenges, MTN [23] employs both the former and current teachers to guide the current student via knowledge distillation (KD) [30]. The Mask R-CNN [26] based MMA [10] is the most recent CLIS approach. It introduces unbiased KD to explicitly handle background shift while incrementally adapting the experiences. Both MTN and MMA require the expensive pixel-wise supervision at each incremental learning step. Our approach, on the other hand, is supervised by the image-level labels that are readily available.

Weakly-supervised Instance Segmentation. As collecting pixel-wise mask annotations for IS is labor-intensive, weakly-supervised IS (WSIS) based on image-level supervision [3, 1, 36, 46, 65] can greatly alleviate the human efforts and has attracted more interest recently. One way to obtain instance cues from image-level labels is via the Class Activation Maps (CAMs) [48, 63] that provide rough object regions per class. PRM [65] utilizes peaks detected from CAMs to localize informative object regions and combines with the object proposals provided by MCG [50] to extract instance masks. To eliminate the need for proposal approaches, IRN [1] generates a displacement vector field and a class boundary map for deriving pseudo instance labels. Recently, BESTIE [36] proposes a peak attention module (PAM) to enhance representative regions of objects for obtaining more accurate instance cues.

Weakly-supervised Semantic Segmentation. Many WSSS approache have been proposed based on the weak annotations such as scribble [42], bounding box [35, 15], points [4] and image labels [40, 2, 39]. Most recent WSSS with image-level supervision approaches have also utilized the pseudo masks derived from CAMs. To further improve the pseudo mask quality, PMM [40] expands the activation regions based on the CAM's coefficient of variation; the normalised Global Weighted Pooling (nGWP) [2] is proposed to compute better pixel-wise classification scores. Leveraging [2], the WSSS is extended to the incremental learning (CL for WSSS) scenario in WILSON [9] recently.

Weakly-supervised Object Detection. Recent methods [7, 58, 53, 32] usually regard Weakly-supervised Object Detection (WSOD) as a multiple instance learning problem, where a bag of instances is given by the off-the-shelf proposal methods. WSDDN [7] is a weakly-supervised detector re-purposed from a pre-trained image classifier. Based on WSDDN, an online multi-stage refinement method is proposed in [58] to better discover the entire object. [53] considers the proposal association between proposals and applies a learnable dropout augmentation that removes the object discriminative parts during training. Self-distillation is employed in CASD [32] to enhance the attention consistency between different transformations of the same image.

Weak-shot Learning. Weak-shot learning leverages full annotations of base classes to learn novel classes with only weak labels. With data from both base and novel classes, weak-shot learning methods commonly adopt mechanisms to transfer knowledge [31, 64, 13, 6, 14] from the base to novel to facilitate the learning of novel classes and do not consider sequential sessions of learning. By contrast, in CL4WSIS, new data are incrementally provided while previous data become inaccessible. CL4WSIS faces additional challenges of background shift and catastrophic forgetting.

Our work is the first study on CL4WSIS. The proposed method includes CL for WSSS as a special case and can further perform IS in class-incremental CL. By using instance cues together in training, experimental results show that our method can also provide better WSSS results in CL as well.

3. Methodology

Without loss of generality, we adopt the representation in Panoptic-Deeplab [18] and use semantic, center, and offset maps to describe instances. Semantic maps represent foreground regions. The center heatmap provides cues to extract the location of the center of the instance. Specifically, if a point on the heatmap has the same value before and after max pooling, it is considered the center. Finally, the 2D offset vector for each location points to the center. Hence, we can get instances by assigning an ID to each foreground pixel, a process called *instance grouping*. Instance ID k is assigned to pixel (i, j) if the k-th center is closest when we move the pixel by its offset. One advantage of this representation is that it allows for any semantic segmentation method to be upgraded to instance segmentation, as long as center and offset information can be generated.

CL4WSIS aims to build a model through incremental learning in $t = 1, ..., T$ steps that rely only on image class labels. We assume that the model is provided with fully pixel-wise annotations at the initial step ($t = 0$). In the incremental learning step $t > 0$, the model learns to segment new instance classes from the training data $\mathcal{D}^t = \{x_n^t, y_n^t\}_{n=1}^{N_t}$, where x_n^t is an image of size $H \times W$, y_n^t

is the image class label, and N^t is the number of images. We denote the label set of the new classes by $\mathcal{Y}^t$. Like in conventional fully supervised CL for IS [10], previous data are not available when the model is incrementally updated.

Our method consists of two phases, CL for WSSS and CL4WSIS. Fig. 2 shows an overview of these two phases, and their details are illustrated in the supplementary materials. In Phase 1, we train the semantic branch of the t-th step Segmenter. The learned CL for WSSS network then serves for predicting the semantic segmentation results and producing further the synthetic center & offset maps in Phase 2 to train the instance branch of the Segmenter for CL4WSIS. We will elaborate on each phase in the following sections.

3.1. CL for WSSS

For a current-step input image $\boldsymbol{x}^t$, we first feed it to an encoder network $e^t(\cdot)$ and obtain a feature map. The encoder distills from the previous-step encoder $e^{t-1}(\cdot)$ by using L_2^2 loss to preserve the basic capabilities of the old task.

Our model contains a Multi-label Classification module trained with the classification loss by the global image labels, as shown in Fig. 2. In our implementation, binary cross entropy (BCE) is used as the loss for each class. Despite being trained with a classification loss, the module is mainly responsible for estimating each pixel's "contribution score" to each class by decomposing its intermediate output, which is then used by the Location Cue Extractor that provides the synthetic pixel-wise labels to train our Segmenter (Fig. 2). For the purpose, this module is often designed as a Decoder $d(\cdot)$ followed by an Aggregator ($\mathbb{A}$). The Decoder output $Z^t = d(e(\boldsymbol{x}^t)) \in \mathbb{R}^{|\mathcal{Y}^t| \times H \times W}$ is a feature map containing the per-pixel semantic score of each class in $\mathcal{Y}^t$, and $\mathbb{A}(Z^t)$ aggregates the scores of all pixels to produce the logits for classification.

Since our purpose is primarily to estimate pixel-level contributions to each class, global image classification performance is not necessarily the most important concern. Therefore, instead of directly using global average pooling (GAP) like CAM [63], many strategies have been developed which can help generate appropriate semantic scores for finer segmentation [21, 2, 51, 55]. Without loss of generality, we adopt the Normalized Global Weighted Pooling (nGWP) combined with the focal penalty [2] as our aggregator. The approach aggregates pixels based on their contributions to the relevant class instead of treating each pixel equally, which can generate finer semantic maps.

The Decoder output Z^t is sent to the Location Cue Extractor, which simply performs label smoothing [47] on Z^t and produces synthetic semantic maps as pixel-level labels to train our Segmenter on the semantic part. We use pixel-wise BCE as the Segmentation Loss in our implementation.

Besides the current (t-th step) weak-label supervision, our model distills additionally from the $(t-1)$-th Segmenter output in CL. For input x^t, let $S^{0:t-1}$ be the pixel-wise probability maps generated by the previous Segmenter for all the old classes in $\mathcal{Y}^{0:t-1}$, we also adopt the pixel-wise BCE (denoted as $BCE(Z_{i,j,c}^{0:t-1}, S_{i,j,c}^{0:t-1})$) as the distillation loss for pixel (i,j), $c \in \mathcal{Y}^{0:t-1}$. In addition to the decoder, the current Segmenter distills from the previous Segmenter by using the same loss for the previous classes in $\mathcal{Y}^{0:t-1}$.

As Decoder is the main component responsible for producing the synthetic labels for per-pixel supervision, it highly influences the training performance of the Segmenter. In our experience, the architecture of Decoder may not be the main concern to affect the performance. We use the DeeplabV3 [16] decoder as our Segmenter, but only a simple few-layer CNN model as our Decoder for efficient training. On the other hand, how to train a better Decoder for WSSS is an important issue. To this end, we introduce two further strategies, *feature-level augmentation consistency* and *random dropout*, which can improve the Decoder's training as depicted below.

Augmentation Consistency. The Decoder output often gives rough estimates of object regions only. We employ an augmentation consistency strategy to strengthen the maps. The idea is that when we apply a transformation $\mathcal{T}$ to an image x and perform semantic segmentation on $T(x)$, the segmentation result $S(x)$ should be equal to $\mathcal{T}^{-1}(S(\mathcal{T}(x)))$ for some transformations $\mathcal{T}$. The transformations used in our work include horizontal flip and a random rotation in $\{90°, 180°, 270°\}$. Unlike other methods (*e.g.*, [17, 25, 32]) that perform augmentations directly on images, our method, inspired by [56], performs the transformations on the lower-dimensional feature map of the encoder output, resulting in a more efficient data augmentation training. We denote the two transformed outputs of the Decoder as Z_{flp} and Z_{rot}. Then, their inverse transformations Z_{flp}^{-1}, Z_{rot}^{-1} should be consistent with the original Decoder output Z. To enforce this constraint, we design an augmentation consistency loss. For each pixel, we average the classification probabilities of all classes at first and obtain $\bar{Z}$, $\bar{Z}_{flp}^{-1}$, and $\bar{Z}_{rot}^{-1}$. Then, inspired by the method [32] developed for WSOD, we apply a pixel-wise max to the three maps and obtain $\tilde{Z}$ which serves as the target map to encourage the mutual consistency between the augmentations. As the max operation acts as the union of the segmentation maps, it helps resolve the part domination problems in segmentation. However, unlike [32], our approach performs the data augmentation in the feature space with higher efficiency. Our feature-level augmentation consistency (**FLAC**) loss is defined as

$$\frac{1}{K}\frac{1}{HW}(||\bar{Z} - \tilde{Z}||^2 + ||\bar{Z}_{flp}^{-1} - \tilde{Z}||^2 + ||\bar{Z}_{rot}^{-1} - \tilde{Z}||^2), \quad (1)$$

with K the number of augmentations.

Random Dropout. Mainly guided by global image labels, the Decoder tends to produce pixel-wise scores on which only discriminative regions of objects are highlighted for the current-step classes. Studies on object detection of weak supervision [57, 53] have shown that randomly removing some discriminating regions is an effective solution to force the network to exploit other regions when performing pixel aggregation for classification. Directly masking out image content in the input is a common approach, but it may not be straightforward to adapt into our method. Since we already use nGWP [2] in the aggregation of Decoder outputs, we propose a trainable soft pixel masking strategy to force the Decoder to recognize whole objects.

The principle of nGWP [2] is to aggregate pixels based on their relevance to the class. Thus, for pixels likely to belong to the current class, randomly increasing their probability of being in the old class has the similar effect of getting them out of the current class during aggregation. To achieve this, for a pixel (i, j), we consider its highest potential of being some class c in the current (t-th) step in the nGWP aggregation process,

$$\hat{Z}_{i,j} = max\{Z_{i,j,c} | c \in \mathcal{Y}^t\}. \tag{2}$$

Let P denote the set of pixels whose highest probability of belonging to some current class is higher than 0.5,

$$P = \{(i, j) | \sigma(\hat{Z}_{i,j}) > 0.5\}, \tag{3}$$

with $\sigma(\cdot)$ the sigmoid function. Then, for a pixel (i, j) in P, we randomly choose an old class $C_{i,j} \in \mathcal{Y}^{0:t-1}$ and apply further the following cross-entropy loss during the training process with nGWP aggregation, so as to randomly raise the pixel's probability of belonging to some old class C,

$$-\frac{1}{|P|} \sum_{(i,j) \in P} \log(\sigma(Z_{i,j,C_{i,j}})). \tag{4}$$

By doing so, our method can successfully integrate the random dropout effect into the nGWP aggregation process in CL for WSSS. It not only retains the advantage of using nGWP in CL [9], that is, the regions of old classes highly confirmed by the previous Segmenter will not be occupied by the current class, but also provides the effect of random dropout in a smooth training process and forces the learner to explore wider regions than just the discriminating ones.

3.2. CL for WSIS

After CL for WSSS, we proceed towards CL4WSIS by acquiring center and offset maps. One way to generate these maps from WSSS is to determine whether a mask derived from semantic segmentation through connected-component labeling (CCL) [28] contains only a single instance. This is because for such masks, generating the center and offset maps can be achieved simply by calculating the centroid of the mask and directing pixels belonging to the mask towards the centroid.

Table 1. Comparison of different peak generation approaches. Our PG generates peaks of higher quality than PAM does.

SBD 15-5 overlap	**AP@.5**			**AP@.5:.95**		
Peaks from	1-15	16-20	All	1-15	16-20	All
PAM [36]	**47.2**	9.1	37.7	28.1	3.6	22.0
Peak Generator (Ours)	47.1	**17.3**	**39.7**	**28.2**	**8.5**	**23.3**

Peak Generator (PG). To this end, we propose a Peak Generator (PG) (included in Location Cue Extractor in Fig. 2 and is appended after the Decoder), inspired by recent WSIS methods [36, 65]. PG aims to yield one appropriate, accurate cue (*i.e.*, peak) per instance. Therefore, peaks, which represent instances, can be helpful for identifying whether a CCL-obtained mask contains only one instance by counting the peaks it includes. Specifically, PG takes as input the Decoder's output Z^t and produces $Z^{\text{pg}} \in \mathbf{R}^{|\mathcal{Y}^t| \times H \times W}$. We then highlight the core regions and suppress the noisy regions in Z^{pg} as follows. Pixels on channel c are treated as core pixels if their values are greater than the channel-specific threshold τ_c. τ, the threshold vector for all channels, is computed by pixel-wise multiplying a hyper-paramenter γ with $G \in \mathbb{R}^{|\mathcal{Y}^t| \times 1 \times 1}$, where G is the global max pooling of Z^{pg} and γ is set to 0.7 in our implementation. The peak map Z^{pg} is then processed by a convolution layer, followed by our pixel aggregator to enable the training with the Global Image Labels.

Though in the same vein as the peak attention module (PAM) [36], our PG is enforced to place the peaks in the semantic foreground, whereas PAM trained irrespective of the WSSS output, may generate peaks not in the foreground. Furthermore, since the Decoder's output already considers the occupation of old classes, PG is guided to generate peaks in proper regions during CL. Based on this reason, PG also encourages Decoder to strengthen the activation of the current class semantic map. As shown in Table 1, PG helps achieve more favorable performance than PAM.

Synthetic Center & Offset Maps Generation. With the WSSS results and PG's instance cues, our Location Cue Extractor then generates the synthetic center and offset labels for the current classes for supervision (see Fig. 2). We adopt the method introduced in [36] for the center and offset maps generation. First, CCL is applied to the WSSS to obtain multiple instance mask candidates. Then, a mask candidate is regarded as an isolated object if only one peak is included. Once the isolated instances are identified, their corresponding center and offset labels can be generated. We denote these synthetic center and offset maps as C^{syn} and O^{syn}, respectively. We also use the self-refinement strategy [36] to yield the instance-level supervision for the overlapped instances. The idea behind self-refinement is that by learning with reliable synthetic labels generated from isolated instances, the Segmenter should progressively capture missing or overlapped instances as the training proceeds. One thus can utilize the Segmenter's outputs as guidance to generate synthetic center and offset labels for these instances. To achieve this, a magnitude map is first created using the Segmenter's output offset map. Each pixel in the map represents the magnitude of its corresponding 2D offset vector. Since the offset magnitudes around a center are usually small, the CCL algorithm is performed on the pixels with magnitudes smaller than a threshold to obtain candidate masks. The centroids of the candidate masks are regarded as new centers. Finally, *instance grouping* aforementioned is performed on the semantic and offset maps outputted by Segmenter using the new centers to generate synthetic labels for the newly identified instances, *i.e.*, a pixel's offset will be redirected to a new center if the pixel is closest the new center after moving by its offset predicted by Segmenter. To learn from the synthesized labels, a weight mask, $\mathcal{W}^{syn}$ with N foreground pixels is generated. $\mathcal{W}^{syn}_{i,j}$ is set to 1 if the pixel (i, j) belongs to isolated objects; otherwise it is set to the confidence of the synthetic instance it belongs to. Following Panoptic-DeepLab, we use L2 loss for the center and L1 loss for the offset. The Segmentation Loss for the center and offset maps is then formulated as

$$\ell^{syn}_{center} = \frac{1}{N}||\mathcal{W}^{syn} \odot (C^t - C^{syn})||^2, \tag{5}$$

$$\ell^{syn}_{offset} = \frac{1}{N}||\mathcal{W}^{syn} \odot (O^t - O^{syn})||_1, \tag{6}$$

with $\odot$ the pixel-wise multiplication.

Semantic-aware Selective Distillation. To retain the old-class knowledge in $\mathcal{Y}^{0:(t-1)}$, an intuitive method would be distilling from the entire center and offset maps of the previous Segmenter. However, as the previous center and offset maps contain no information about the current classes, this would hinder the Segmenter learning of current classes from the synthetic labels provided by Location Cue Extractor.

To resolve this issue, we introduce a semantic-aware selective distillation strategy to help the Segmenter learn effectively for the current classes while preserving the old-class information. Assume $\mathcal{S}(\cdot, \cdot)$ to be the semantic map already generated in our CL for WSSS. Leveraging $\mathcal{S}$, we construct a weight mask $\mathcal{W}^{old}$ for the old classes, where $\mathcal{W}^{old}_{i,j} = 1$ if $\mathcal{S}(i, j)$ belongs to $\mathcal{Y}^{0:(t-1)}$ (w/o background class) and $\mathcal{W}^{old}_{i,j} = 0$ otherwise. The distillation losses for learning the center and offset maps C^t and O^t from the previous Segmenter are then respectively defined as

$$\ell^{dist}_{center} = \frac{1}{M}||\mathcal{W}^{old} \odot (C^t - C^{t-1})||^2 \tag{7}$$

$$\ell^{dist}_{offset} = \frac{1}{M}||\mathcal{W}^{old} \odot (O^t - O^{t-1})||_1 \tag{8}$$

with M the number of foreground pixels of old classes.

In sum, established on the CL for WSSS results, our approach achieves CL4WSIS via PG's instance cues for synthesizing local labels for current classes and selective distillation for maintaining old knowledge. Further details are given in the supplementary material.

4. Experiments

In this section, we present experimental results to verify the performance of our CL4WSIS method.

4.1. Datasets and Settings

We conduct experiments on Pascal SBD 2012 [24], Pascal VOC 2012 [22] and COCO [43]. Pascal SBD 2012 consists of 8,498 training and 2,857 validation images annotated on 20 objects categories. Following [36, 1, 46], we augment Pascal VOC with PASCAL SBD and obtain 10,582 images for training and 1,499 for validation, with objects in 20 categories. COCO comprises 118K training and 5K validation images with 80 object categories.

We follow [9] and adopt two incremental learning settings on Pascal SBD: **15-5** and **10-10**. The M-N refers to that M classes are learned in the first step and N classes in the second. While comparing with pixel-level methods, we report the results in two scenarios: 1) the *disjoint*, in which an image is included in the current-step data if all instances in this image are of previous or current-step classes, and 2) the *overlap*, in which an image is included in the current-step data if the image contains at least one instance belonging to the current-step classes. When compared with CL4WSIS adapted methods, we focus on the overlap scenario and increase a setting of **10-5-5** with more incremental steps. Besides, another challenging **COCO-to-VOC** cross-dataset scenario is adopted. There are two incremental learning steps. The first learns 60 COCO classes not present in Pascal VOC. Note that all the images containing the VOC classes are excluded. The second learns the 20 Pascal VOC classes. The performance is evaluated using the mean average precision (mAP) with intersection-over-union (IoU) threshold from 0.5 to 0.95 and 0.5. We report the performance of the final model that finishes all the incremental steps on all the learned classes of the validation sets of PASCAL and COCO.

4.2. Baselines

Since CL for IS with image-level supervision is a new setting, our approach is compared to the recent incremental IS approach MMA [10]. We note that MMA is trained under pixel-level annotations (using Mask R-CNN [26]), which can be regarded as an upper bound of our method. The reported AP@0.5:0.95 of MMA in the Pascal SBD 15-5 overlap is directly taken from their paper, while the rest are obtained by running their official implementation. Another two state-of-the-art WSIS methods, IRN [1] and

BESTIE [36], are also adapted into incremental scenario for further comparison. Same as ours, the instance segmentation model is provided with fully pixel-wise annotations on the initial step. As for the incremental steps, we follow their implementation to generate the pixel-level pseudo labels for the current classes using the image-level labels. We also generate the pixel-level pseudo labels for the old classes by using the previous model. Then, we train the instance segmentation model using these pseudo-labels.

4.3. Implementation Details

Our architecture is adapted from Panoptic-DeepLab [18] by appending an additional decoder. The Panoptic-DeepLab decoder (Segmenter in our paper) consists of a semantic branch and an instance branch (for center and offset), and the semantic branch is replaced with DeepLabv3 in our implementation. We use a ResNet101 [27] as the encoder for Pascal SBD experiments and use a Wide-ResNet-38 [60] for COCO-to-VOC. Both models are initialized with ImageNet pretrained weights. The Decoder is composed of 3 convolution layers followed by batch normalization and Leaky ReLU, where the kernel size is 3×3 for the first two while 1x1 for the last, with channel numbers $\{256, 256, |\mathcal{Y}^{0:t}|\}$, and stride 1. PG consists of 1 layer for keeping the high activation values followed by a convolution layer with kernel size 1x1, $|\mathcal{Y}^t|$ channels, and stride 1. For base step, our model is trained for 100 epochs on Pascal SBD and 200 epochs on COCO-to-VOC using Adam with an initial learning rate (lr) of 5e-5 (5e-4 for the Segmenter). As for other incremental steps, we first train the model (w/o instance branch) using SGD with an initial lr of 0.001 (0.01 for the Segmenter and Decoder) for 40 epochs (30 epochs on COCO-to-VOC), and then only train the instance branch using Adam with an initial lr of 5e-4 for 50 epochs on both settings. In all experiments, we use the batch size 16 and a polynomial scheduler with a power of 0.9.

4.4. Results

Comparison with Pixel-Level Methods. Table 2 shows the results on the Pascal SBD 15-5 setting. The joint training learns from pixel-wise annotations using all the data and is the upper-bound. The lower-bound fine-tuning (FT) does not employ any component to preserve old knowledge and simply learns the current classes in a fully supervised manner. Hence, it performs well on the current classes but forgets the old completely, resulting in unsatisfactory overall performance. Our method maintains satisfactory performance on the old classes thanks to the employed incremental components for knowledge preservation. Furthermore, it effectively learns current classes, even with only image-level labels. As such, overall, our approach outperforms the FT by a large margin. Although MMA is superior to ours, it requires fully pixel-level annotations. A similar trend is

Table 2. Results on the Pascal SBD 15-5 setting. $\mathcal{P}$ denotes pixel-wise supervision, and $\mathcal{I}$ denotes image-level supervision.

AP@0.5		Disjoint			Overlap		
Method	**Sup**	1-15	16-20	All	1-15	16-20	All
Joint	$\mathcal{P}$	58.8	56.1	58.2	58.8	56.1	58.2
FT	$\mathcal{P}$	0.0	26.7	6.7	0.0	21.9	5.5
MMA [10]	$\mathcal{P}$	65.3	50.9	61.7	64.0	50.7	60.7
Ours	$\mathcal{I}$	47.6	19.8	40.7	50.7	23.3	43.9
AP@0.5:0.95		**Disjoint**			**Overlap**		
Method	**Sup**	1-15	16-20	All	1-15	16-20	All
Joint	$\mathcal{P}$	38.3	38.3	38.3	38.3	38.3	38.3
FT	$\mathcal{P}$	0.0	13.6	3.4	0.0	10.8	2.7
MMA [10]	$\mathcal{P}$	39.5	30.9	37.3	40.2	32.2	38.2
Ours	$\mathcal{I}$	28.8	9.4	24.0	30.9	11.6	26.1

Table 3. Results on the Pascal SBD 10-10 setting. $\mathcal{P}$ denotes pixel-wise supervision, and $\mathcal{I}$ denotes image-level supervision.

AP@0.5		Disjoint			Overlap		
Method	**Sup**	1-10	11-20	All	1-10	11-20	All
Joint	$\mathcal{P}$	57.0	59.3	58.2	57.0	59.3	58.2
FT	$\mathcal{P}$	0.0	49.7	24.9	0.0	57.3	28.6
MMA [10]	$\mathcal{P}$	53.3	40.8	47.1	53.3	40.8	47.1
Ours	$\mathcal{I}$	41.8	21.6	31.7	48.6	29.2	38.9
AP@0.5:0.95		**Disjoint**			**Overlap**		
Method	**Sup**	1-10	11-20	1-20	1-10	11-20	All
Joint	$\mathcal{P}$	37.8	38.8	38.3	37.8	38.8	38.3
FT	$\mathcal{P}$	0.0	30.0	15.0	0.0	35.6	17.8
MMA [10]	$\mathcal{P}$	34.0	21.4	27.7	39.1	24.1	31.6
Ours	$\mathcal{I}$	26.8	8.9	17.9	30.3	13.8	22.0

also observed in the 10-10 setting in Table 3.

Comparison with Adapted WSIS Methods. As revealed in Table 4, our approach performs more favorably against the ones adapted from the state-of-the-art WSIS on Pascal SBD for both old and current classes on all settings. It is worth noting that on the 15-5 setting, ours achieves a 10.8% AP@.5 higher than IRN, and 19.2% AP@.5 higher than BESTIE. Although IRN performs well on the 10-5-5, our method still well maintains the balance between learning the current classes and preserving the old knowledge. Table 5 presents the per-step performance, which shows IRN faces challenges in retaining prior knowledge and acquiring new knowledge especially in step 2. These results demonstrate the effectiveness of our approach.

COCO-to-VOC results. In this setting, the model performs incremental updates using data from different domains. The catastrophic forgetting problem could become harder to tackle because some of the first 60 classes in COCO in the 1st step are unlikely to appear in VOC in the

Table 4. Comparison of our approach to other WSIS approaches adapted into the **CL4WSIS** scenario on Pascal SBD. $\mathcal{P}$ denotes pixel-wise supervision, and $\mathcal{I}$ denotes image-level supervision.

15-5 Overlap		**AP@.5**			**AP@.5:.95**		
Method	**Sup**	1-15	16-20	All	1-15	16-20	All
IRN [1]	$\mathcal{I}$	39.2	14.8	33.1	21.6	6.2	17.8
BESTIE [36]	$\mathcal{I}$	30.1	8.5	24.7	15.0	2.8	12.0
Ours	$\mathcal{I}$	**50.7**	**23.3**	**43.9**	**30.9**	**11.6**	**26.1**
10-10 Overlap		**AP@.5**			**AP@.5:.95**		
Method	**Sup**	1-10	11-20	All	1-10	11-20	All
IRN [1]	$\mathcal{I}$	44.6	23.0	33.8	26.0	9.9	18.0
BESTIE [36]	$\mathcal{I}$	40.5	17.0	28.7	23.0	6.4	14.7
Ours	$\mathcal{I}$	**48.6**	**29.2**	**38.9**	**30.3**	**13.8**	**22.0**

10-5-5 Overlap		**AP@.5**				**AP@.5:.95**			
Method	**Sup**	1-10	11-15	16-20	All	1-10	11-15	16-20	All
FT	$\mathcal{P}$	0.0	0.0	34.9	8.7	0.0	0.0	19.6	4.9
IRN [1]	$\mathcal{I}$	36.6	**28.1**	16.4	29.4	19.9	11.2	6.6	14.4
BESTIE [36]	$\mathcal{I}$	32.3	17.5	10.8	23.2	17.1	6.9	3.4	11.1
Ours	$\mathcal{I}$	**37.4**	27.1	**20.3**	**30.5**	**21.1**	**12.6**	**8.6**	**15.8**

Table 5. The performance after training each CL step on the 10-5-5 overlap setting.

10-5-5 AP@.5:.95	**Step 0**	**Step 1**		**Step 2**		
Method	1-10	1-10	11-15	1-10	11-15	16-20
IRN [1]	34.1	25.8	**14.0**	19.9	11.2	6.6
Ours	34.1	**26.6**	13.9	**21.1**	**12.6**	**8.6**

Table 6. **CL4WSIS** results on the COCO-to-VOC setting. Our approach produces more favorable results than the other approaches.

AP@0.5		**COCO**	**VOC**	**COCO$_{\text{(only testing)}}$**
Method	**Sup**	1-60	61-80	61-80
FT	$\mathcal{P}$	0.0	52.7	30.0
IRN [1]	$\mathcal{I}$	10.9	13.7	6.7
BESTIE [36]	$\mathcal{I}$	10.7	10.3	3.9
Ours	$\mathcal{I}$	**14.4**	**14.7**	**6.8**
AP@0.5:0.95		**COCO**	**VOC**	**COCO$_{\text{(only testing)}}$**
Method	**Sup**	1-60	61-80	61-80
FT	$\mathcal{P}$	0.0	32.3	18.3
IRN [1]	$\mathcal{I}$	6.2	5.4	2.6
BESTIE [36]	$\mathcal{I}$	5.8	3.7	1.4
Ours	$\mathcal{I}$	**8.2**	**5.7**	**2.9**

2nd step. Table 6 shows the result. Note that we also report the performance on the VOC classes (61-80) included in COCO, which are not used for training, and denote it as COCO$_{\text{(only testing)}}$ in the table. Under such a challenging learning scenario, the comparison in Table 6 shows that our method still yields the best results. This is attributed to our model's ability to better attain the previous experience and effectively learn the current classes.

Comparison on CL for WSSS. Although designed for CL4WSIS, our approach can perform CL for WSSS. Table 7 shows the results of different approaches on the SBD 15-5 setting. As can be observed, our approach surpasses WILSON on the both scenarios. This is mainly because of the introduced modules, and we study their effect on performance in Table 8. Our approach includes [9] as a special case when no introduced modules are employed, as indicated in the first row with no checks. While PG was initially developed to supply instance cues for CL4WSIS, appending it after the Decoder has also shown to be highly beneficial in improving the performance of current classes of WSSS. FLAC helps by encouraging the Decoder to produce same semantic segmentations for different views generated from one sample. Random Dropout forces the Decoder to explore more regions, bringing additional gain.

Table 7. **CL for WSSS** results on the Pascal SBD 15-5 setting. $\mathcal{P}$ denotes pixel-wise supervision, and $\mathcal{I}$ denotes image-level supervision.

mIoU		**Disjoint**			**Overlap**		
Method	**Sup**	1-15	16-20	All	1-15	16-20	All
Joint	$\mathcal{P}$	73.6	65.7	72.6	73.6	65.7	72.6
FT	$\mathcal{P}$	0.0	29.6	10.5	0.1	29.9	10.6
WILSON [9]	$\mathcal{I}$	69.4	35.7	62.2	69.7	36.4	62.6
Ours	$\mathcal{I}$	**69.8**	**39.4**	**63.4**	**70.5**	**41.3**	**64.4**

Table 8. **CL for WSSS** ablation study on Pascal SBD 15-5 overlap.

SBD 15-5 overlap			**mIoU**		
PG	FLAC	Randrop	1-15	16-20	All
			69.7	36.4	62.6
✓			69.9	40.7	63.8
✓	✓		70.1	41.0	64.1
✓	✓	✓	**70.5**	**41.3**	**64.4**

Table 9. **CL4WSIS** ablation study on Pascal SBD 15-5 overlap. SD stands for the selective distillation strategy.

SBD 15-5 overlap				**AP@.5**			**AP@.5:.95**		
PG	SD	FLAC	Randrop	1-15	16-20	All	1-15	16-20	All
✓				47.1	17.3	39.7	28.2	8.5	23.3
✓	✓			50.1	22.7	43.3	30.6	11.5	25.9
✓	✓	✓		50.1	23.2	43.4	30.3	**11.9**	25.7
✓	✓	✓	✓	**50.7**	**23.3**	**43.9**	**30.9**	11.6	**26.1**

Influences of Different Modules. We ablate on the introduced modules for their relative contribution to CL4WSIS and report the results on SBD in Table 9. PG enables transferring semantic knowledge to instance segmentation and hence is the first module to be included. Our selective distillation strategy has proven effective for retaining previous experiences and learning new knowledge, as evidenced by 3.0% AP@.5 improvement for old classes and 4.5% AP@.5 boost for current classes. Since the knowledge is transferred from CL for WSSS to CL4WSIS, FLAC also helps learning object instances of current classes while Random Dropout

Table 10. Comparison of our approach to other WSIS approaches adapted into the CL4WSIS scenario on the Pascal SBD overlap setting. **old & current classes**: the results where all the seen labels are weakly annotated in the incremental step. **only current classes**: the results where only the current-task class labels are weakly provided in the incremental step.

15-5 setting

old & current classes	**AP@.5**			**AP@.5:.95**		
Method	1-15	16-20	All	1-15	16-20	All
IRN [1]	28.8	12.9	24.8	14.7	5.5	12.4
BESTIE [36]	41.3	16.5	35.1	23.7	6.4	19.3
Ours	**51.5**	**25.7**	**45.1**	**31.2**	**12.6**	**26.5**

only current classes	**AP@.5**			**AP@.5:.95**		
Method	1-15	16-20	All	1-15	16-20	All
IRN [1]	39.2	14.8	33.1	21.6	6.2	17.8
BESTIE [36]	30.1	8.5	24.7	15.0	2.8	12.0
Ours	**50.7**	**23.3**	**43.9**	**30.9**	**11.6**	**26.1**

10-10 setting

old & current classes	**AP@.5**			**AP@.5:.95**		
Method	1-10	11-20	All	1-10	11-20	All
IRN [1]	39.9	25.0	32.5	22.8	10.4	16.6
BESTIE [36]	40.6	18.5	29.5	23.0	7.3	15.2
Ours	**54.3**	**29.9**	**42.1**	**32.3**	**13.7**	**23.0**

only current classes	**AP@.5**			**AP@.5:.95**		
Method	1-10	11-20	All	1-10	11-20	All
IRN [1]	44.6	23.0	33.8	26.0	9.9	18.0
BESTIE [36]	40.5	17.0	28.7	23.0	6.4	14.7
Ours	**48.6**	**29.2**	**38.9**	**30.3**	**13.8**	**22.0**

has a positive effect on both old and current classes. We found that when all the modules are adopted, our approach yields the best performance for CL4WSIS.

Qualitative Analysis. We visualize the qualitative results in Fig. 3 for the images from the Pascal SBD 15-5 overlap setting. IRN partly maintains the knowledge about old classes (*e.g.*, person and diningtable) and also partly learns the current (*e.g.*, sheep and train). BESTIE fails to learn current classes and produces many false-positive predictions (*e.g.*, sheep). On the other hand, our approach produces higher-quality predictions for both old and current classes.

Incremental Steps with All Weak Labels Provided. Settings above follow the class-incremental continual learning, where newly collected data are provided with only new labels (*i.e.*, labels in $\mathcal{Y}^t$) at the current step t and data from previous steps are unavailable. Because image-level labels are cheaper to obtain, we could consider a setting where image class labels up to the current step, *i.e.*, $\mathcal{Y}^{0:t}$, are provided, and call it All-Seen-Label-Annotation (ASLA).

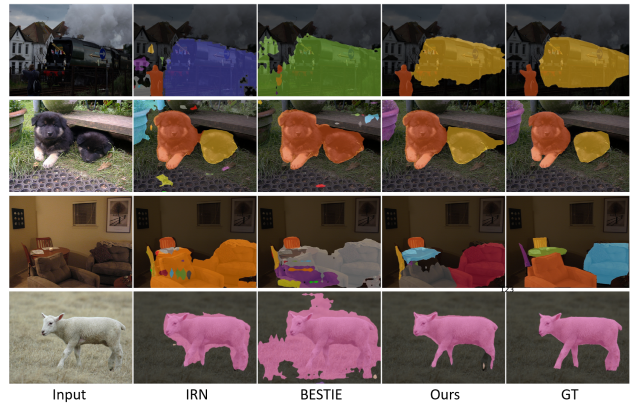

Figure 3. Qualitative results on SBD 15-5 overlap setting. Our method produces higher-quality segmentations on both current classes (*e.g.*, sheep, sofa, train) and old classes (*e.g.*, person, dog). From left to right: image, IRN, BESTIE, OURS and ground truth.

Table 10 shows the results, where the upper half of each setting is for the ASLA and the lower half is for the original one. Compared to the original one, our approach obtains additional performance gain in ASLA. It is because when Global Image Labels of old classes are also provided, incorrect predictions from previous Segmenter can be removed, thereby providing more proper information for both the Decoder and Segmenter. Again, our approach performs more favorably than IRN and BESTIE in the ASLA setting, too.

5. Conclusion

We have presented a novel framework and conducted the first study for the new CL4WSIS problem. To incrementally extend knowledge through cheap image-level supervision, our framework generates pseudo instance-level supervision by leveraging the semantic knowledge from a Decoder and instance cues from a peak generator. We further introduce feature-level augmentation consistency (FLAC) and employ random dropout for obtaining more reliable pseudo supervision. Besides, by leveraging the knowledge from the previous model using proposed selective distillation, our model maintains the learned experiences while learning new skills. Experiments in various incremental settings have verified the effectiveness of our approach. We hope our study could provide insights for future research on CL4WSIS.

6. Acknowledgement

This work was supported in part by the National Science and Technology Council, Taiwan under Grant NSTC 111-2634-F-006-012, 110-2221-E-002-185-MY2 and 112-2221-E-110-047-MY3. We thank to National Center for High-performance Computing (NCHC) of National Applied Research Laboratories (NARLabs) in Taiwan for providing computational and storage resources.

References

[1] Jiwoon Ahn, Sunghyun Cho, and Suha Kwak. Weakly supervised learning of instance segmentation with inter-pixel relations. In *CVPR*, pages 2204–2213, 2019. 3, 6, 8, 9

[2] Nikita Araslanov and Stefan Roth. Single-stage semantic segmentation from image labels. In *CVPR*, pages 4252–4261, 2020. 2, 3, 4, 5

[3] Aditya Arun, C. V. Jawahar, and M. Pawan Kumar. Weakly supervised instance segmentation by learning annotation consistent instances. In *ECCV*, pages 254–270, 2020. 3

[4] Amy Bearman, Olga Russakovsky, Vittorio Ferrari, and Li Fei-Fei. What's the point: Semantic segmentation with point supervision. In *ECCV*, pages 549–565, 2016. 3

[5] Eden Belouadah and Adrian Popescu. Il2m: Class incremental learning with dual memory. In *ICCV*, October 2019. 1

[6] David Biertimpel, Sindi Shkodrani, Anil S Baslamisli, and Nóra Baka. Prior to segment: Foreground cues for weakly annotated classes in partially supervised instance segmentation. In *ICCV*, pages 2824–2833, 2021. 3

[7] Hakan Bilen and Andrea Vedaldi. Weakly supervised deep detection networks. In *CVPR*, pages 2846–2854, 2016. 3

[8] Daniel Bolya, Chong Zhou, Fanyi Xiao, and Yong Jae Lee. YOLACT++ better real-time instance segmentation. *IEEE TPAMI*, 44(2):1108–1121, 2022. 1

[9] Fabio Cermelli, Dario Fontanel, Antonio Tavera, Marco Ciccone, and Barbara Caputo. Incremental learning in semantic segmentation from image labels. In *CVPR*, pages 4361–4371, 2022. 3, 5, 6, 8

[10] Fabio Cermelli, Antonino Geraci, Dario Fontanel, and Barbara Caputo. Modeling missing annotations for incremental learning in object detection. In *CVPRW*, pages 3699–3709, 2022. 1, 3, 4, 6, 7

[11] Fabio Cermelli, Massimiliano Mancini, Samuel Rota Bulò, Elisa Ricci, and Barbara Caputo. Modeling the background for incremental learning in semantic segmentation. In *CVPR*, pages 9230–9239, 2020. 3

[12] Sungmin Cha, Beomyoung Kim, Youngjoon Yoo, and Taesup Moon. SSUL: Semantic segmentation with unknown label for exemplar-based class-incremental learning. In *NeurIPS*, pages 10919–10930, 2021. 3

[13] Junjie Chen, Li Niu, Liu Liu, and Liqing Zhang. Weak-shot fine-grained classification via similarity transfer. In M. Ranzato, A. Beygelzimer, Y. Dauphin, P.S. Liang, and J. Wortman Vaughan, editors, *NeurIPS*, pages 7306–7318, 2021. 3

[14] Junjie Chen, Li Niu, Siyuan Zhou, Jianlou Si, Chen Qian, and Liqing Zhang. Weak-shot semantic segmentation via dual similarity transfer. In *Advances in Neural Information Processing Systems*, pages 32525–32536, 2022. 3

[15] Liang-Chieh Chen, George Papandreou, Iasonas Kokkinos, Kevin Murphy, and Alan L. Yuille. Semantic image segmentation with deep convolutional nets and fully connected crfs. In *ICLR*, 2015. 3

[16] Liang-Chieh Chen, George Papandreou, Florian Schroff, and Hartwig Adam. Rethinking atrous convolution for semantic image segmentation. *arXiv*, 2017. 4

[17] Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. A simple framework for contrastive learning of visual representations. In *ICML*, pages 1597–1607, 2020. 4

[18] Bowen Cheng, Maxwell D. Collins, Yukun Zhu, Ting Liu, Thomas S. Huang, Hartwig Adam, and Liang-Chieh Chen. Panoptic-DeepLab: A simple, strong, and fast baseline for bottom-up panoptic segmentation. In *CVPR*, pages 12472–12482, 2020. 1, 3, 7

[19] Matthias Delange, Rahaf Aljundi, Marc Masana, Sarah Parisot, Xu Jia, Ales Leonardis, Greg Slabaugh, and Tinne Tuytelaars. A continual learning survey: Defying forgetting in classification tasks. *IEEE Trans. Pattern Anal. Mach. Intell.*, 2021. 1

[20] Arthur Douillard, Yifu Chen, Arnaud Dapogny, and Matthieu Cord. PLOP: Learning without forgetting for continual semantic segmentation. In *CVPR*, pages 4039–4049, 2021. 3

[21] Thibaut Durand, Taylor Mordan, Nicolas Thome, and Matthieu Cord. Wildcat: Weakly supervised learning of deep convnets for image classification, pointwise localization and segmentation. In *CVPR*, July 2017. 4

[22] Mark Everingham, Luc Van Gool, Christopher K. I. Williams, John M. Winn, and Andrew Zisserman. The pascal visual object classes (voc) challenge. *IJCV*, pages 303–338, 2010. 6

[23] Yanan Gu, Cheng Deng, and Kun Wei. Class-incremental instance segmentation via multi-teacher networks. In *AAAI*, pages 1478–1486, 2021. 1, 3

[24] Bharath Hariharan, Pablo Arbeláez, Lubomir Bourdev, Subhransu Maji, and Jitendra Malik. Semantic contours from inverse detectors. In *ICCV*, pages 991–998, 2011. 6

[25] Kaiming He, Haoqi Fan, Yuxin Wu, Saining Xie, and Ross Girshick. Momentum contrast for unsupervised visual representation learning. In *CVPR*, pages 9729–9738, 2020. 4

[26] Kaiming He, Georgia Gkioxari, Piotr Dollar, and Ross Girshick. Mask R-CNN. In *ICCV*, pages 2961–2969, 2017. 1, 3, 6

[27] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *CVPR*, pages 770–778, 2016. 7

[28] Lifeng He, Yuyan Chao, Kenji Suzuki, and Kesheng Wu. Fast connected-component labeling. *PR*, pages 1977–1987, 2009. 5

[29] Yin-Yin He, Peizhen Zhang, Xiu-Shen Wei, Xiangyu Zhang, and Jian Sun. Relieving long-tailed instance segmentation via pairwise class balance. In *CVPR*, pages 6990–6999, 2022. 1

[30] Geoffrey E. Hinton, Oriol Vinyals, and Jeffrey Dean. Distilling the knowledge in a neural network. In *NeurIPS Workshop*, 2014. 3

[31] Ronghang Hu, Piotr Dollár, Kaiming He, Trevor Darrell, and Ross Girshick. Learning to segment every thing. In *CVPR*, pages 4233–4241, 2018. 3

[32] Zeyi Huang, Yang Zou, B. V. K. Vijaya Kumar, and Dong Huang. Comprehensive attention self-distillation for weakly-supervised object detection. In *NeurIPS*, pages 16797–16807, 2020. 3, 4

[33] Ching-Yi Hung, Cheng-Hao Tu, Cheng-En Wu, Chien-Hung Chen, Yi-Ming Chan, and Chu-Song Chen. Compacting, picking and growing for unforgetting continual learning. In *NeurIPS*, pages 13647–13657, 2019. 1

[34] Longlong Jing, Yucheng Chen, and Yingli Tian. Coarse-to-fine semantic segmentation from image-level labels. *IEEE Transactions on Image Processing*, 29:225–236, 2020. 2

[35] A. Khoreva, R. Benenson, J. Hosang, M. Hein, and B. Schiele. Simple does it: Weakly supervised instance and semantic segmentation. In *CVPR*, pages 1665–1674, 2017. 3

[36] Beomyoung Kim, Youngjoon Yoo, Chae Eun Rhee, and Junmo Kim. Beyond semantic to instance segmentation: Weakly-supervised instance segmentation via semantic knowledge transfer and self-refinement. In *CVPR*, pages 4268–4277, 2022. 3, 5, 6, 7, 8, 9

[37] James Kirkpatrick, Razvan Pascanu, Neil Rabinowitz, Joel Veness, Guillaume Desjardins, Andrei A Rusu, Kieran Milan, John Quan, Tiago Ramalho, Agnieszka Grabska-Barwinska, et al. Overcoming catastrophic forgetting in neural networks. *Proceedings of the National Academy of Sciences*, 114(13):3521–3526, 2017. 1

[38] Justin Lazarow, Weijian Xu, and Zhuowen Tu. Instance segmentation with mask-supervised polygonal boundary transformers. In *CVPR*, pages 4372–4381, 2022. 1

[39] Seungho Lee, Minhyun Lee, Jongwuk Lee, and Hyunjung Shim. Railroad is not a train: Saliency as pseudo-pixel supervision for weakly supervised semantic segmentation. In *CVPR*, pages 5495–5505, 2021. 3

[40] Yi Li, Zhanghui Kuang, Liyang Liu, Yimin Chen, and Wayne Zhang. Pseudo-mask matters in weakly-supervised semantic segmentation. In *ICCV*, pages 6944–6953, 2021. 3

[41] Zhizhong Li and Derek Hoiem. Learning without forgetting. *IEEE TPAMI*, 40(12):2935–2947, 2018. 3

[42] Di Lin, Jifeng Dai, Jiaya Jia, Kaiming He, and Jian Sun. Scribblesup: Scribble-supervised convolutional networks for semantic segmentation. In *CVPR*, pages 3159–3167, 2016. 3

[43] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C. Lawrence Zitnick. Microsoft COCO: Common objects in context. In *ECCV*, pages 740–755, 2014. 6

[44] Yaoyao Liu, Bernt Schiele, and Qianru Sun. Adaptive aggregation networks for class-incremental learning. In *CVPR*, pages 2544–2553, 2021. 1, 3

[45] Yaoyao Liu, Yuting Su, An-An Liu, Bernt Schiele, and Qianru Sun. Mnemonics training: Multi-class incremental learning without forgetting. In *CVPR*, June 2020. 1

[46] Yun Liu, Yu-Huan Wu, Peisong Wen, Yujun Shi, Yu Qiu, and Ming-Ming Cheng. Leveraging instance-, image- and dataset-level information for weakly supervised instance segmentation. *IEEE TPAMI*, 44(3):1415–1428, 2022. 3, 6

[47] Michal Lukasik, Srinadh Bhojanapalli, Aditya Menon, and Sanjiv Kumar. Does label smoothing mitigate label noise? In *ICML*, pages 6448–6458. PMLR, 2020. 4

[48] Maxime Oquab, Léon Bottou, Ivan Laptev, and Josef Sivic. Is object localization for free? - Weakly-supervised learning with convolutional neural networks. In *CVPR*, pages 685–694, 2015. 3

[49] Minh Hieu Phan, The-Anh Ta, Son Lam Phung, Long Tran-Thanh, and Abdesselam Bouzerdoum. Class similarity weighted knowledge distillation for continual semantic segmentation. In *CVPR*, pages 16845–16854, 2022. 3

[50] Jordi Pont-Tuset, Pablo Arbeláez, Jonathan T. Barron, Ferran Marques, and Jitendra Malik. Multiscale combinatorial grouping for image segmentation and object proposal generation. *IEEE TPAMI*, 39(1):128–140, 2017. 3

[51] Ziniu Qian, Kailu Li, Maode Lai, Eric I-Chao Chang, Bingzheng Wei, Yubo Fan, and Yan Xu. Transformer based multiple instance learning for weakly supervised histopathology image segmentation. In *MICCAI*, pages 160–170, 2022. 4

[52] Sylvestre-Alvise Rebuffi, Alexander Kolesnikov, Georg Sperl, and Christoph H. Lampert. iCaRL: Incremental classifier and representation learning. In *CVPR*, pages 5533–5542, 2017. 3

[53] Zhongzheng Ren, Zhiding Yu, Xiaodong Yang, Ming-Yu Liu, Yong Jae Lee, Alexander G. Schwing, and Jan Kautz. Instance-aware, context-focused, and memory-efficient weakly supervised object detection. In *CVPR*, pages 10598–10607, 2020. 3, 5

[54] Amir Rosenfeld and John K. Tsotsos. Incremental learning through deep adaptation. *IEEE Trans. Pattern Anal. Mach. Intell.*, 42(3):651–663, 2018. 1

[55] Lixiang Ru, Yibing Zhan, Baosheng Yu, and Bo Du. Learning affinity from attention: end-to-end weakly-supervised semantic segmentation with transformers. In *CVPR*, pages 16846–16855, 2022. 4

[56] Jinhwan Seo, Wonho Bae, Danica J Sutherland, Junhyug Noh, and Daijin Kim. Object discovery via contrastive learning for weakly supervised object detection. In *ECCV*, pages 312–329, 2022. 4

[57] Krishna Kumar Singh and Yong Jae Lee. Hide-and-seek: Forcing a network to be meticulous for weakly-supervised object and action localization. In *ICCV*, pages 3544–3553, 2017. 5

[58] Peng Tang, Xinggang Wang, Xiang Bai, and Wenyu Liu. Multiple instance detection network with online instance classifier refinement. In *CVPR*, pages 2843–2851, 2017. 3

[59] Tz-Ying Wu, Gurumurthy Swaminathan, Zhizhong Li, Avinash Ravichandran, Nuno Vasconcelos, Rahul Bhotika, and Stefano Soatto. Class-incremental learning with strong pre-trained models. In *CVPR*, pages 9601–9610, June 2022. 1

[60] Zifeng Wu, Chunhua Shen, and Anton van den Hengel. Wider or deeper: Revisiting the resnet model for visual recognition. *PR*, pages 119–133, 2019. 7

[61] 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 TPAMI*, 2022. 3

[62] Chang-Bin Zhang, Jia-Wen Xiao, Xialei Liu, Ying-Cong Chen, and Ming-Ming Cheng. Representation compensation networks for continual semantic segmentation. In *CVPR*, pages 7043–7054, 2022. 3

[63] Bolei Zhou, Aditya Khosla, Agata Lapedriza, Aude Oliva, and Antonio Torralba. Learning deep features for discriminative localization. In *CVPR*, pages 2921–2929, 2016. 3, 4

[64] Yanzhao Zhou, Xin Wang, Jianbin Jiao, Trevor Darrell, and Fisher Yu. Learning saliency propagation for semi-supervised instance segmentation. In *CVPR*, pages 10307–10316, 2020. 3

[65] Yanzhao Zhou, Yi Zhu, Qixiang Ye, Qiang Qiu, and Jianbin Jiao. Weakly supervised instance segmentation using class peak response. In *CVPR*, pages 3791–3800, 2018. 3, 5