MARS: Model-agnostic Biased Object Removal without Additional Supervision for Weakly-Supervised Semantic Segmentation

Sanghyun Jo[1] In-Jae Yu[2] Kyungsu Kim[3]*
[1]OGQ, Seoul, Korea [2]Samsung Electronics, Suwon, Korea
[3]Department of Data Convergence and Future Medicine, Sungkyunkwan University, Seoul, Korea
{shjo.april, ijyu.phd, kskim.doc}@gmail.com

Abstract

*Weakly-supervised semantic segmentation aims to reduce labeling costs by training semantic segmentation models using weak supervision, such as image-level class labels. However, most approaches struggle to produce accurate localization maps and suffer from false predictions in class-related backgrounds (*i.e.*, biased objects), such as detecting a railroad with the train class. Recent methods that remove biased objects require additional supervision for manually identifying biased objects for each problematic class and collecting their datasets by reviewing predictions, limiting their applicability to the real-world dataset with multiple labels and complex relationships for biasing. Following the first observation that biased features can be separated and eliminated by matching biased objects with backgrounds in the same dataset, we propose a fully-automatic/model-agnostic biased removal framework called MARS (**M**odel-**A**gnostic biased object **R**emoval without additional **S**upervision), which utilizes semantically consistent features of an unsupervised technique to eliminate biased objects in pseudo labels. Surprisingly, we show that MARS achieves new state-of-the-art results on two popular benchmarks, PASCAL VOC 2012 (val: 77.7%, test: 77.2%) and MS COCO 2014 (val: 49.4%), by consistently improving the performance of various WSSS models by at least 30% without additional supervision. Code is available at https://github.com/shjo-april/MARS.*

1. Introduction

Fully-supervised semantic segmentation (FSSS) [7, 8], which aims to classify each pixel of an image, requires time-consuming tasks and significant domain expertise in some applications [54] to prepare pixel-wise annotations. By contrast, weakly-supervised semantic segmentation (WSSS) with image-level supervision, which is the most economical among weak supervision, such as bounding boxes [12], scribbles [35], and points [4], reduces the labeling cost by more than 20× [4]. The multi-stage learning framework is the dominant approach for training WSSS models with image-level labels. Since this framework heavily relies on the quality of initial class activation maps (CAMs), numerous researchers [2, 49, 28, 10, 51, 21] moderate the well-known drawback of CAMs, highlighting the most discriminative part of an object to reduce the false negative (FN).

*Correspondence to

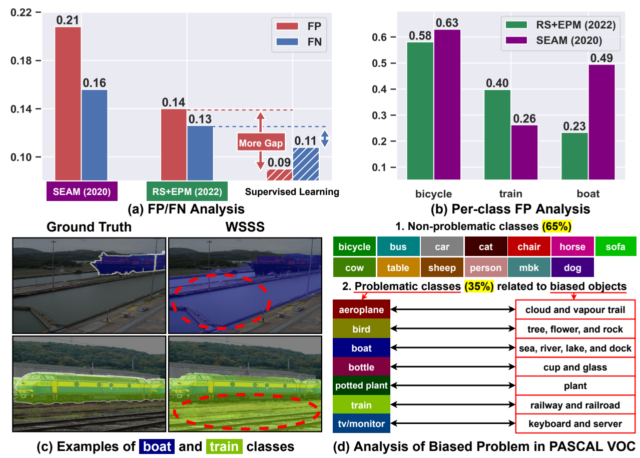

Figure 1. (a) Comparison with existing WSSS studies [49, 21] and FSSS. (b) Per-class FP analysis. (c) Examples of biased objects in boat and train classes. (d) Quantitative analysis of biased objects on the PASCAL VOC 2012 dataset. Red dotted circles illustrate the false activation of biased objects such as railroad and sea.

However, the false positive (FP) is the most crucial bottleneck to narrow the performance gap between WSSS and FSSS in Fig. 1(a). According to per-class FP analysis in Fig. 1(b), predicting target classes (*e.g.*, boat) with class-related objects (*e.g.*, sea) are factored into increasing FP in Fig. 1(c), besides incorrect annotations in the bicycle class. Moreover, 35% of classes in the PASCAL VOC 2012 dataset have biased objects in Fig. 1(d). These results show that the performance degradation of previous approaches depends on the presence or absence of problematic classes in the dataset. We call this issue a biased problem. We also add examples of all classes in the Appendix.

Table 1. Comparison with public datasets for WSSS. Since Open Images [24] does not provide pixel-wise annotations for all classes, existing methods employ PASCAL VOC 2012 [14] and MS COCO 2014 [36] for fair comparison and evaluation.

Dataset	Training images	Classes	GT
PASCAL VOC 2012 [14]	10,582	20	✓
MS COCO 2014 [36]	80,783	80	✓
Open Images [24]	9,011,219	19,794	✗

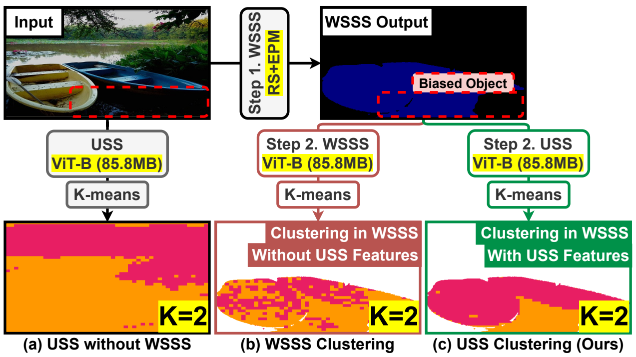

Figure 2. Integration with WSSS and USS. **(a)**: The USS method fails to detect biased objects without the WSSS output. **(b)**: The WSSS method cannot find biased objects due to using biased features. **(c)**: Thanks to the WSSS guidance, the USS method identifies biased objects on a limited area of the WSSS output.

Although two studies [51, 29] alleviate the biased problem, their requirements hinder WSSS applications in the real world having complex relationships between classes. For example, to apply them to train the Open Images dataset [24], which includes most real-world categories (19,794 classes) in Table 1, they need to not only analyze pairs of the WSSS prediction and image to find biased objects in 6,927 classes (35% of 19,794 classes) as referred to Fig. 1(d) but also confirm the correlation of biased objects and non-problematic classes to prevent decreasing performance of non-problematic classes, impeding the practical WSSS usage. Therefore, without reporting performance on MS COCO 2014 dataset, current debiasing methods [51, 29] have only shared results on the PASCAL VOC 2012 dataset.

To address the biased problem without additional dataset and supervision, we propose a novel fully-automatic biased removal called MARS (**M**odel-**A**gnostic biased object **R**emoval without additional **S**upervision), which first utilizes unsupervised semantic segmentation (USS) in WSSS. In particular, our method follows a model-agnostic manner by newly connecting existing WSSS and USS methods for biased removal, which have been only independently studied [21, 15]. Specifically, our method is based on two key observations related to the integration with USS and WSSS:

- (The first USS application to separate biased and target objects in WSSS) As the bias issue is intrinsically linked to image-level supervision, USS has fewer biased features than WSSS. In Fig. 2(a), without WSSS, the USS method must tune an optimal K per image to separate biased and target objects. Despite using the same ViT-B backbone, WSSS cannot find biased objects in Fig. 2(b). USS clustering successfully disentangles biased and target objects on a limited area of the WSSS output, as shown in Fig. 2(c).
- (The first USS-based distance metric to single out the biased object) As shown in Fig. 3, the shorter distance reflects the biased object among distances between two separated regions (pink and orange) and background regions of other images distinguished by the USS method (blue) because the minimum distance between the target and all background sample sets is greater than the minimum distance between the bias and all background sample sets. Accordingly, we show the biased object can exist in the background set, which is a set of classes excluding foreground classes.

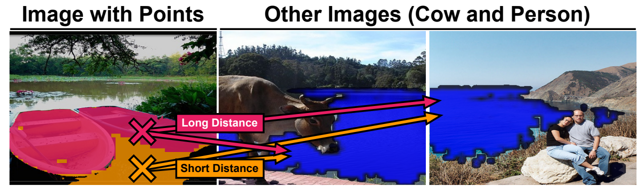

Figure 3. Correspondence between biased objects and backgrounds. We measure the distance between each separated object (crosses in the left image) and the background regions of other images (middle and right) within the same dataset. As a result, the long and short distances reflect target and biased objects, respectively. Therefore, the distance of USS features can be used as a criterion to remove biased objects after clustering features.

Therefore, MARS produces debiased labels using the USS-based distance metric after separating biased and target objects in all training images. To prevent increasing FN of non-problematic classes, MARS then complements debiased labels with online predictions in the training time. Our main contributions are summarized as follows.

- We first introduce two observations of applying USS in WSSS to find biased objects automatically: the USS-based feature clustering for separating biased and target objects and a new distance metric to select the biased object among two isolated objects.
- We propose a novel fully-automatic/model-agnostic method, MARS, which leverages semantically consistent features learned through USS to eliminate biased objects without additional supervision and dataset.
- Unlike current debiasing methods [51, 29] that validated only in the PASCAL VOC 2012 dataset with fewer labels, we have also verified the validity of MARS in the more practical case with larger and complex labels such as MS COCO 2014; MARS achieves new state-of-the-art results on two benchmarks (VOC: 77.7%, COCO: 49.4%) and consistently improves representative WSSS methods [1, 49, 28, 21] by at least 3.4%, newly validating USS grafting on WSSS.

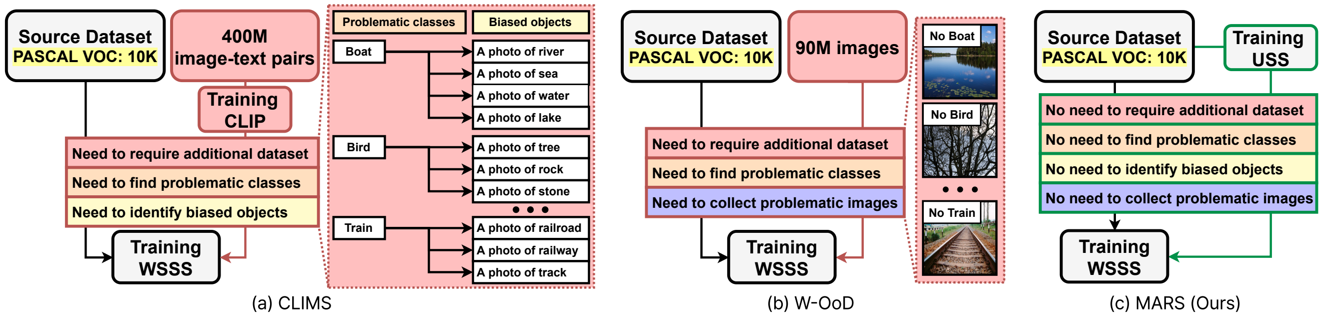

Figure 4. Conceptual comparison of three WSSS requirements. **(a)**: Using the CLIP's knowledge trained on image-text pairs dataset alleviates the biased problem by finding problematic classes and identifying biased objects. **(b)**: Human annotators manually collect problematic images from the Open Images dataset [24] to train biased objects directly. **(c)**: The proposed MARS first applies an existing USS approach to remove biased objects without additional supervision, achieving the fully-automatic biased removal.

Table 2. Comparison with our method and its related works. With the CLIP model trained on a 400M image-and-text dataset, CLIMS [51] removes biased objects after finding problematic classes and identifying biased objects for each class (*i.e.*, a railroad for the train class). W-OoD [29] requires human annotators manually collect problematic images (*i.e.*, only including railroad in an image). Unlike previous approaches, our method removes biased objects without additional datasets and human supervision.

Properties	CLIMS [51]	W-OoD [29]	**Ours**
For removing biased objects	✓	✓	✓
Use model-agnostic manner	✗	✓	✓
Need to require additional dataset	✓	✓	✗
Need to find problematic classes	✓	✓	✗
Need to identify biased objects	✓	✗	✗
Need to collect problematic images	✗	✓	✗

2. Related Work

2.1. Weakly-Supervised Semantic Segmentation

Most WSSS approaches [55, 28, 34, 25, 44, 26, 41, 53, 30] aim to enlarge insufficient foregrounds of initial CAMs. Some studies apply the feature correlation, such as SEAM [49], CPN [56], PPC [56], SIPE [9], and RS+EPM [21], or patch-based dropout principles, such as FickleNet [27], Puzzle-CAM [20], and L2G [19]. Other methods exploit cross-image information, such as MCIS [45], EDAM [50], RCA [57], and C^2AM [52], or global information, such as MCTformer [53] and AFA [43]. SANCE [32] and ADELE [38] propose advanced pipelines to only remove minor noise in pseudo labels. In addition, some studies [31, 22, 13] employ saliency supervision to remove FP in pseudo labels. However, saliency supervision requires class-agnostic pixel-wise annotations and ignores small and low-prominent objects. All studies mentioned above are independent of our method. We demonstrate consistent improvements of some WSSS approaches [1, 49, 28, 21] in Table 5.

Similar to our approach, several studies [29, 51] have focused on removing biased objects in pseudo labels. Table 2 compares the essential properties of our method with those of related studies. We also illustrate the conceptual difference with existing WSSS methods [51, 29] and the proposed MARS in Fig. 4. CLIMS [51] utilizes the Contrastive Language-Image Pre-training (CLIP) model [42], which is trained on a large-scale dataset of 400 million image-text pairs (*i.e.*, using text supervision), and needs to identify biased objects (*e.g.*, railroad and sea) in all problematic classes (*e.g.*, train and boat classes), as shown in Fig. 1(d). W-OoD [29] needs human annotators to collect additional images, which only include biased objects (*e.g.*, railroad and sea), from the Open Images dataset [24] to train the classification network directly with problematic images. Our method first removes biased objects by leveraging the semantic consistency of the trained USS method from scratch without additional human supervision and dataset.

2.2. Unsupervised Semantic Segmentation

USS focuses on training semantically meaningful features within image collection without any form of annotations. Therefore, all USS methods [5, 18, 39, 11, 47, 48, 58, 15] are used as the pre-training strategy because they cannot produce class-aware predictions only by grouping features. IIC [18], AC [39], and PiCIE [11] maximize the mutual information between different views. Leopart [58], and STEGO [15] utilize the self-supervised vision transformer to learn spatially structured image representations, resulting in accurate object masks without additional supervision. Notably, STEGO [15] enriches correlations between unsupervised features with training a simple feed-forward network, leading to efficient training without re-training or fine-tuning weights initialized by DINO [6]. Our method is agnostic to the underlying USS methods, utilizing pixel-wise semantic features only. Hence, all USS methods are independent of our approach. We show consistent improvements in recent USS methods [58, 15], verifying the flexibility of our method and the potential for integrating future advances in USS into our method.

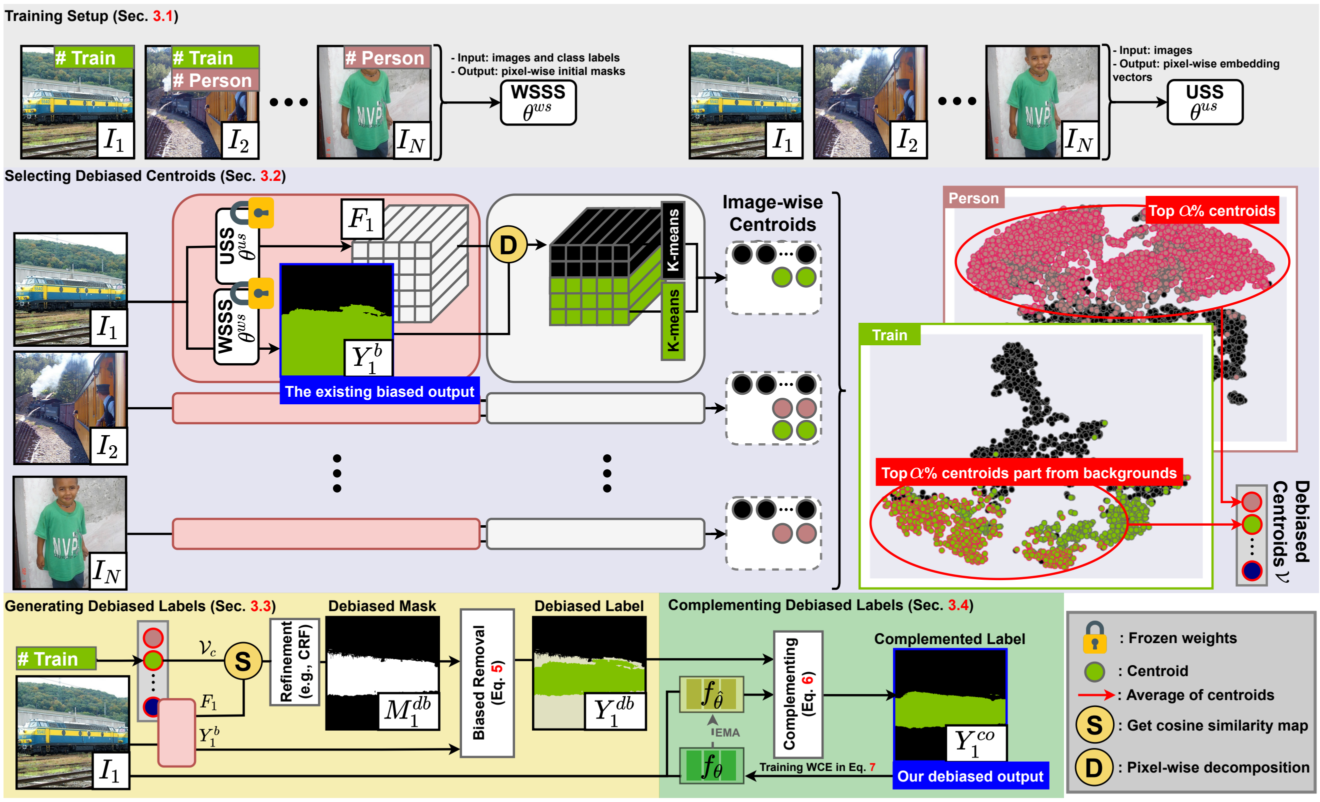

Figure 5. Overview of MARS. The USS and WSSS methods, which are trained from scratch, produce pixel-wise embedding vectors F_i and the pseudo label Y_i^b, including biased objects, respectively. Based on our observations, K-means clustering generates image-wise centroids (*i.e.*, biased and target objects) from decomposed vectors per class. Then, the debiased centroid $\mathcal{V}^c$ per class is derived as the average of the top α% centroids from $\{v_i^c\}_{i=1}^{N_c \cdot K_{fg}}$, the most apart from background centroids of all training images in (2). To generate the debiased label Y_i^{db}, we calculate the similarity map using debiased centroids and embedding vectors of the USS method in (4). The segmentation network then trains the debiased labels Y_i^{db} with the proposed weighted cross-entropy loss function (WCE) in (7). Thus, our MARS provides the final debiased label as Y_i^{co}.

3. Method

The proposed MARS consists of four sections/stages: (a) training WSSS and USS methods for the model-agnostic manner, (b) selecting debiased centroids, (c) generating debiased labels, and (d) complementing debiased labels during the learning process. The overall framework of MARS is illustrated in Fig. 5.

3.1. Training Setup

This section describes the training setup for existing WSSS and USS models. Unlike [51, 29], our model-agnostic approach does not require additional datasets for training these models. For a fair comparison, we train all WSSS and USS models from scratch on the PASCAL VOC 2012 or MS COCO 2014 datasets, following the standard setup of WSSS methods [1, 49, 28, 21]. Each training image $I_i \in \mathbb{R}^{3 \times H \times W}$ in the dataset is associated with a set of image-level class labels $L_i \in \{0, 1\}^C$, where C is the number of categories/classes. In detail, the classification network generates initial CAMs after training images and image-level class labels. Then, the conventional propagating method [1] refines initial CAMs to produce pseudo labels. Finally, USS methods [58, 15] are trained only on the images, following each pretext task. For the following sections, our method utilizes pseudo masks and semantic features produced from the frozen weights of the WSSS and USS methods, respectively.

3.2. Selecting Debiased Centroids

This section describes how our approach separates biased and target objects using trained WSSS and USS methods. For a target input image I_i, the trained USS method generates pixel-wise embedding vectors $F_i \in \mathbb{R}^{D \times H \times W}$, not including class-specific information. Meanwhile, the trained WSSS method produces pseudo labels $Y_i^b \in \{0, 1, ..., C\}^{H \times W}$, including both biased and target objects. We group pixel-wise embedding vectors F_i under Y_i^b's prediction region $\{(y, x) | Y_i^b(y, x) = c\}$ for each class c, and apply K-means clustering to generate image-wise centroids $v_{i \cdot K+j}^c \in \mathbb{R}^D$ per class c for $j \in \{1, 2, ..., K\}$. Here, the number K of clusters for foreground ($c > 0$) and background ($c = 0$) classes are K_{fg} and K_{bg}, respectively. We set K_{fg} to 2 to separate biased and target objects, and K_{bg} can be varied. Although our aforementioned simple clustering isolates biased and target objects, it cannot identify which one is the target or biased object among both candidate objects. To single out the biased object, we propose a new following distance metric between each candidate object and background centroids in all training images in (1):

$$dist_k^c = \frac{1}{N^{bg}} \sum_{j=0}^{N^{bg}} D(v_k^c, v_j^0) \tag{1}$$

where 0 and c denote the index of the background and foreground classes, respectively, i denotes the index of the foreground centroid, and $N^{bg} := N \cdot K_{bg}$ denotes the number of background centroids from all N training images. $S(\cdot)$ and $D(\cdot)$ mean the cosine similarity (*i.e.*, $v \cdot v' / \|v\| \, \|v'\|$) and distance (*i.e.*, $(1 - S(v, v'))/2$), respectively. For instance, long and short distances mean target and biased centroids, respectively, since each distance reflects the degree of whether to include the biased object as shown in Fig. 3. We sort all foreground centroids per class in descending order by the distance using background centroids. Thus, for each class c, we aggregate the average of top $\alpha\%$ centroids most apart from background centroids to get a single vector representing the debiased centroid $\mathcal{V}^c \in \mathbb{R}^D$ as follows:

$$\mathcal{V}^c = \frac{1}{\lceil N_c^{fg} \cdot \alpha \rceil} \sum_{j \in \{k_1, k_2, \ldots, k_{\lceil N_c^{fg} \cdot \alpha \rceil}\}} v_j^c, \tag{2}$$

$$dist_{k_1}^c \geq dist_{k_2}^c \geq \ldots \geq dist_{k_{N_c^{fg}}}^c \tag{3}$$

where $N_c^{fg} := N_c \cdot K_{fg}$ denotes the number of centroids from N_c images having class c, $\alpha \in [0, 1]$ is the ratio of selecting target centroids, and $\{k_i\}_{i \in \{1:N_c^{fg}\}}$ is the ordered index set satisfying (3) (*e.g.*, $v_{k_1}^c$ is the centroid having the largest distance from all background centroids). In other words, when we identify the biased or debiased object in the given image I_i, we improve its identification performance by using information from other training images together; its analysis is detailed in Sec. 4.3.

3.3. Generating Debiased Labels

We present our approach for finding and removing biased pixels in pseudo labels Y_i^b. We first compute the similarity map between each debiased centroid $\mathcal{V}^c$ and embedding vectors F_i for per-pixel biased removal. However, we observe that the trained USS method cannot separate some classes if two categories (*e.g.*, horse and sheep) have the same super-category (*e.g.*, animals). This issue is also present in current USS methods [11, 58, 15] and is caused by the inability to distinguish between objects within the same supercategory. To address this shortcoming, we introduce a debiasing process that generates the debiased mask $\hat{M}_i^{db}$ using the pixel-wise maximum function as follows:

$$\hat{M}_i^{db}(y, x) = ReLU\Big(\max_{c \in \mathcal{C}_{I_i}} S(F_i[:, y, x], \mathcal{V}^c)\Big) \tag{4}$$

where (x, y) indicates x, y-th pixel position, $F_i(:, y, x) \in \mathbb{R}^D$ is the pixel-wise embedding vector, $\mathcal{V}^c \in \mathbb{R}^D$ denotes the debiased/target centroid for each class c, $\mathcal{C}_{I_i}$ is corresponding class indices of each image I_i, and the ReLU activation removes negative values in $\hat{M}_i^{db} \in [-1, 1]^{H \times W}$. After applying a typical post-processing refinement (*e.g.*, CRF [23]) to $\hat{M}_i^{db}$, we generate the binary debiased mask $M_i^{db} \in \{0, 1\}^{H \times W}$, which produces the debiased label $Y_i^{db} = \{-1, 0, 1, ..., c\}^{H \times W}$ using the binary debiased mask M_i^{db} and the WSSS label Y_i^b as follows:

$$Y_i^{db}(y, x) = \begin{cases} -1, & \text{if } Y_i^b(y, x) > 0 \text{ and } M_i^{db}(y, x) = 0, \\ Y_i^b(y, x), & \text{otherwise} \end{cases} \tag{5}$$

where -1 indicates the new biased class for the next section 3.4. The pixel value in the debiased label Y_i^{db} is only replaced with the biased class (-1) if our debiased mask M_i^{db} and the WSSS mask Y_i^b provide the label 0 and the foreground class (> 0), respectively. Namely, we remove biased predictions of WSSS by computing the per-pixel similarity of debiased centroids within the embedding space.

3.4. Complementing Debiased Labels

This last section proposes a new training strategy to complement biased pixels in debiased labels. As shown in Fig. 7, although biased objects in our debiased labels are successfully removed for problematic classes (*i.e.*, classes including biased objects, such as train and boat classes), we observe non-biased objects (*e.g.*, people's clothes) are also eliminated, increasing FN of non-problematic classes, *e.g.*, the dog class. To complement non-biased objects, we utilize online predictions $\hat{P}_i$ from a teacher network during its learning process with certain masks.

We illustrate the complementing process as shown in Fig. 6. Here, θ denotes weights of the student network, and we update a teacher network $\hat{\theta}$ using an exponential moving average (EMA). The student and teacher networks predict segmentation outputs $P_i, \hat{P}_i \in [0, 1]^{C \times H \times W}$ after applying the softmax function. We then employ the refinement R (*e.g.*, CRF [23]) and argmax operator to produce the teacher's label $Y_i^{te} = \{0, 1, ..., c\}^{H \times W}$. Finally, we generate complemented labels $Y_i^{co} \in \{0, 1, ..., c\}^{H \times W}$ by filling biased classes (-1) in debiased labels $Y_i^{db} \in \{-1, 0, 1, ..., c\}^{H \times W}$ with the teacher's prediction Y_i^{te}.

However, when updating the teacher network in early epochs, the complemented label Y_i^{co} includes incorrect predictions in smooth probabilities (*i.e.*, uncertain predictions),

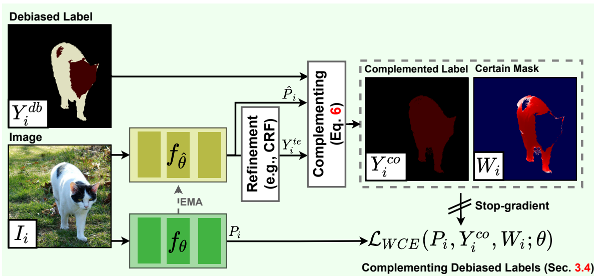

Figure 6. Illustration of the proposed complementing process. With the refinement, the teacher network produces the teacher's label Y_i^{te}. To prevent increasing FN of non-problematic classes, biased pixels in debiased labels Y_i^{db} are complemented with the teacher's prediction. To avoid training uncertain labels, the student network is updated using the proposed WCE in (7) with complemented labels Y_i^{co} and certain masks W_i, resulting in the final predictions similar to ground truths.

covering biased objects in the complementing process. To address this issue in uncertain pixels, we propose a concept of a certain mask $W_i \in [0,1]^{H \times W}$, which is the matrix of pixel-wise maximum probabilities for all foreground classes, and its ablation analysis is detailed in Sec. 4.3:

$$W_i(y,x) = \begin{cases} \max_{c \in \mathcal{C}_{I_i}} \hat{P}_i(c,y,x), & \text{if } Y_i^{db}(y,x) = -1, \\ 1, & \text{otherwise} \end{cases} \tag{6}$$

where $\mathcal{C}_{I_i} := \{k \mid L_i(k) = 1\}$ is an index set of truth classes for each image I_i and -1 denotes the complemented/biased class. To train the segmentation network with complemented labels Y_i^{co} and certain masks W_i, we propose the weighted cross entropy (WCE) loss that multiplies the certain mask W_i with the per-pixel cross-entropy loss to reflect the uncertainty ratio:

$$\begin{aligned} &\mathcal{L}_{WCE}(P_i, Y_i^{co}, W_i; \theta) \qquad (7) \\ &= -\sum_{c \in \mathcal{C}} \sum_{y,x \in \mathcal{W}} W_i(y,x) \cdot O[Y_i^{co}](c,y,x) \log P_i^{\theta}(c,y,x) \end{aligned}$$

where $O[\cdot]$ means one-hot encoding for the per-pixel cross-entropy loss function. As a result, the proposed MARS successfully removes biased objects without performance degradation of non-problematic classes by complementing biased pixels in debiased labels with the teacher's predictions in its learning process (the bottom results in Fig. 7).

In summary, Fig. 7 illustrates the effect of the proposed components on the WSSS performance, following examples in Fig. 1(c) (see examples of other classes in Appendix): After training WSSS and USS methods in Sec. 3.1, the first component (Sec. 3.2) extracts debiased centroids $\{\mathcal{V}^c\}_{c=1}^{C}$ based on the distance of all background centroids to each foreground centroid. The second component (Sec. 3.3) generates debiased labels Y_i^{db} using debiased centroids and previous WSSS labels. The last component (Sec. 3.4) trains the segmentation network by complementing biased pixels to provide the final debiased label as Y_i^{co}. We provide a detailed analysis of our method in Sec. 4.3.

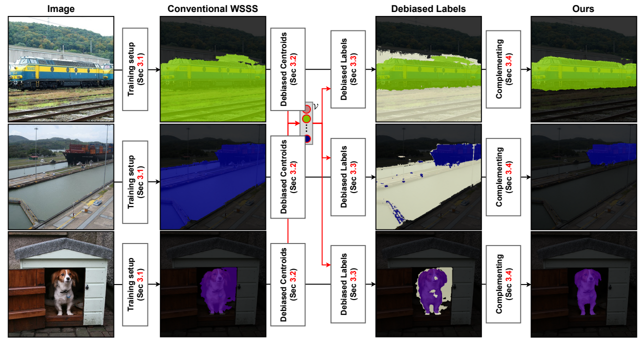

Figure 7. Effect of the proposed components. For problem classes including the biased objects, *e.g.*, boat and train classes, second and third components (Secs. 3.2 and 3.3) remove biased objects in debiased labels Y_i^{db} and then the fourth component (Sec. 3.4) preserves removed objects (the first and second samples). For non-problematic classes not containing biased objects, *e.g.*, the dog class, the fourth component accurately restores non-biased objects (the third sample). In addition, the red line denotes applying debiased centroids to produce debiased labels.

4. Experiments

4.1. Experimental Setup

Datasets. We conduct all experiments on the PASCAL VOC 2012 [14] and MS COCO 2014 [36] datasets, both of which contain image-level class labels, bounding boxes, and pixel-wise annotations. Despite the difficulty of MS COCO 2014 dataset [36], *e.g.*, small-scale objects and imbalance class labels, our method significantly improves all benchmarks. PASCAL VOC 2012 [14] and MS COCO 2014 [36] datasets have 21 and 81 classes, respectively.

Implementation details. From scratch, all USS models are exclusively trained on each dataset without extra data in our experiments. To demonstrate the scalability of our method, we utilize four WSSS methods [1, 49, 28, 21] on PASCAL VOC 2012 dataset [14]. We strictly follow the training details in all USS and WSSS papers for a fair comparison. Thus, our method has the same runtime as other methods in evaluation. We employ RS+EPM as the initial WSSS method for the final result and only use two hyperparameters to select debiased centroids: K_{bg} is set to 2, and α is set to 0.40. In addition, we use multi-scale inference and CRF [23] with conventional settings to evaluate the segmentation network's performance. We conduct all experiments on a single RTX A6000 GPU and implement all WSSS and USS methods in PyTorch.

Evaluation metrics. We evaluate our method using mIoU, following the typical evaluation metric of existing WSSS studies [2, 1, 49, 28, 21]. We also follow FP and FN metrics proposed by the gold standard [49]. We obtain all results for the PASCAL VOC 2012 *val* and *test* sets from the official PASCAL VOC online evaluation server.

Table 3. Performance comparison of WSSS methods regarding mIoU (%) on PASCAL VOC 2012 and COCO 2014. * and † indicate the backbone of VGG-16 and ResNet-50, respectively. Sup., supervision; $\mathcal{I}$, image-level class labels; $\mathcal{S}$, saliency supervision; $\mathcal{D}$, using the external dataset; $\mathcal{F}$, pixel-wise annotations (i.e, fully-supervised semantic segmentation).

Method	Backbone	Sup.	VOC *val*	VOC *test*	COCO *val*
DSRG CVPR'18 [17]	R101	$\mathcal{I}$+$\mathcal{S}$	61.4	63.2	26.0*
FickleNet CVPR'19 [27]	R101	$\mathcal{I}$+$\mathcal{S}$	64.9	65.3	-
CLIMS CVPR'22 [51]	R50	$\mathcal{I}$+$\mathcal{D}$	69.3	68.7	-
W-OoD CVPR'22 [29]	R101	$\mathcal{I}$+$\mathcal{D}$	69.8	69.9	-
EDAM CVPR'21 [50]	R101	$\mathcal{I}$+$\mathcal{S}$	70.9	70.6	-
EPS CVPR'21 [31]	R101	$\mathcal{I}$+$\mathcal{S}$	70.9	70.8	35.7*
DRS AAAI'21 [22]	R101	$\mathcal{I}$+$\mathcal{S}$	71.2	71.4	-
L2G CVPR'22 [19]	R101	$\mathcal{I}$+$\mathcal{S}$	72.1	71.7	44.2
RCA CVPR'22 [57]	R101	$\mathcal{I}$+$\mathcal{S}$	72.2	72.8	36.8*
PPC CVPR'22 [13]	R101	$\mathcal{I}$+$\mathcal{S}$	72.6	73.6	-
PSA CVPR'18 [2]	WR38	$\mathcal{I}$	61.7	63.7	-
IRNet CVPR'19 [1]	R50	$\mathcal{I}$	63.5	64.8	-
SSSS CVPR'20 [3]	WR38	$\mathcal{I}$	62.7	64.3	-
RRM AAAI'20 [55]	R101	$\mathcal{I}$	66.3	65.5	-
SEAM CVPR'20 [49]	WR38	$\mathcal{I}$	64.5	65.7	31.9
CDA ICCV'21 [44]	WR38	$\mathcal{I}$	66.1	66.8	33.2
AdvCAM CVPR'21 [28]	R101	$\mathcal{I}$	68.1	68.0	-
CSE ICCV'21 [25]	WR38	$\mathcal{I}$	68.4	68.2	36.4
ReCAM CVPR'22 [10]	R101	$\mathcal{I}$	68.5	68.4	-
CPN ICCV'21 [56]	WR38	$\mathcal{I}$	67.8	68.5	-
RIB NeurIPS'21 [26]	R101	$\mathcal{I}$	68.3	68.6	43.8
ADELE CVPR'22 [38]	WR38	$\mathcal{I}$	69.3	68.8	-
PMM ICCV'21 [34]	WR38	$\mathcal{I}$	68.5	69.0	36.7
AMR AAAI'22 [41]	R101	$\mathcal{I}$	68.8	69.1	-
URN AAAI'22 [33]	R101	$\mathcal{I}$	69.5	69.7	40.7
SIPE CVPR'22 [9]	R101	$\mathcal{I}$	68.8	69.7	40.6
MCTformer CVPR'22 [53]	WR38	$\mathcal{I}$	71.9	71.6	42.0
SANCE CVPR'22 [32]	R101	$\mathcal{I}$	70.9	72.2	44.7†
RS+EPM Arxiv'22 [21]	R101	$\mathcal{I}$	74.4	73.6	46.4
MARS (Ours)	R101	$\mathcal{I}$	**77.7**	**77.2**	**49.4**
FSSS	R101	$\mathcal{F}$	80.6	81.0	61.8

4.2. Comparison with state-of-the-art approaches

We compare our method with other WSSS methods in Table 3. Recent state-of-the-art methods exploit additional supervision to reduce the number of FP in pseudo labels, such as saliency supervision [16, 37, 40], the external dataset to collect biased images [29], and text supervision from an image-to-text dataset (*e.g.*, CLIP [42]). By contrast, without additional supervision and dataset, we mitigate the biased problem by leveraging the inherent advantage of USS, outperforming previous state-of-the-art methods by at least 3.3%. We also refer to Appendix for the qualitative comparison with existing WSSS methods and ours.

4.3. Analysis

Flexibility. We demonstrate the flexibility of our method by comparing it to various WSSS and USS methods. As shown in Table 4, our method consistently outperforms existing WSSS methods regardless of applying Leopart [58] or STEGO [15] for our method. In Table 5, we compare our method to two flexible WSSS methods [38, 29] based on four WSSS methods [1, 49, 28, 21]. For the WSSS experiment, we utilize STEGO [15] because this USS method performs best in Table 4. We employ the same backbone

Table 4. Comparison with two USS methods [58, 15] in terms of mIoU (%) on PASCAL VOC 2012 dataset.

Method	USS	Backbone	mIoU (*val*)	mIoU (*test*)
IRNet [1]	✗	R50	63.5	64.8
+ Ours	Leopart [58]	R50	68.1	68.8
+ Ours	STEGO [15]	R50	**69.8**	**70.9**
RS+EPM [21]	✗	R101	74.4	73.6
+ Ours	Leopart [58]	R101	75.4	75.8
+ Ours	STEGO [15]	R101	**77.7**	**77.2**

Table 5. Comparison with four WSSS methods [1, 49, 28, 21] in terms of mIoU (%) on PASCAL VOC 2012 dataset. FSSS means training the dataset with pixel-wise annotations. $(\cdot)$ means the percentage improvement in the gap between WSSS and FSSS.

Method	Backbone	Segmentation	mIoU (*val*)	mIoU (*test*)
IRNet [1]	R50	DeepLabv2	63.5	64.8
+ Ours	R50	DeepLabv2	**69.8 (49%)**	**70.9 (52%)**
FSSS	R50	DeepLabv2	76.3	76.5
SEAM [49]	WR38	DeepLabv1	64.5	65.7
+ ADELE [38]	WR38	DeepLabv1	69.3 (35%)	68.8 (25%)
+ Ours	WR38	DeepLabv1	**70.8 (46%)**	**71.4 (46%)**
FSSS	WR38	DeepLabv1	78.1	78.2
AdvCAM [28]	R101	DeepLabv2	68.1	68.0
+ W-OoD [29]	R101	DeepLabv2	69.8 (17%)	69.9 (18%)
+ Ours	R101	DeepLabv2	**70.3 (22%)**	**71.2 (30%)**
FSSS	R101	DeepLabv2	78.0	78.6
RS+EPM [21]	R101	DeepLabv3+	74.4	73.6
+ Ours	R101	DeepLabv3+	**77.7 (53%)**	**77.2 (49%)**
FSSS	R101	DeepLabv3+	80.6	81.0

Table 6. Effect of key components in terms of mIoU (%) on PASCAL VOC 2012 *train* set.

	Complementing	WCE (7)	mIoU	FP	FN
1	✗	✗	77.4	0.123	0.108
2	✓	✗	80.9	0.122	0.075
3	✓	✓	**81.8**	**0.099**	**0.090**

and segmentation model to ensure a fair comparison. Surprisingly, our method improves each performance by 6.3%, 6.3%, 2.2%, and 3.3% for IRNet [1], SEAM [49], AdvCAM [28], and RS+EPM [21], respectively, as shown in Table 5. The qualitative improvements with ADELE [38], W-OoD [29], and ours are given in Appendix. Although W-OoD [29] addresses the biased problem, it requires the manual collection of images, only including biased objects from an additional dataset (*e.g.*, Open Images [24]). The proposed MARS first removes biased objects without additional human supervision, verifying the flexibility of our method.

Effect of complementing. Table 6 shows an ablation study of the proposed complementing process to remove biased objects and prevent increasing FN of non-problematic classes (*i.e.*, classes not including the biased problem). The first row is our baseline (*i.e.*, RS+EPM [21]). Training a segmentation network with debiased labels improves at least 3.5% of mIoU compared to our baseline RS+EPM [21] (rows 2 and 3). However, in row 2, the complementing process without the proposed WCE in (7) significantly decreases FN but increases FP due to incorrect labels when complementing with the model's predictions. The last row achieves the best performance with considering certain masks, proving the validity of the proposed components.

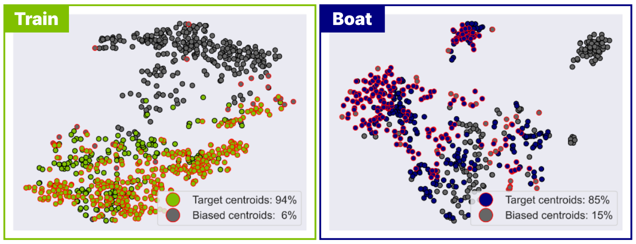

Figure 8. Visualization of selecting debiased centroids. We quantify the ratio of selecting target centroids by using pixel-wise annotations. The left and right results indicate train and boat classes, respectively. The percentage of target centroids is more than 85%, proving the validity of the proposed selection.

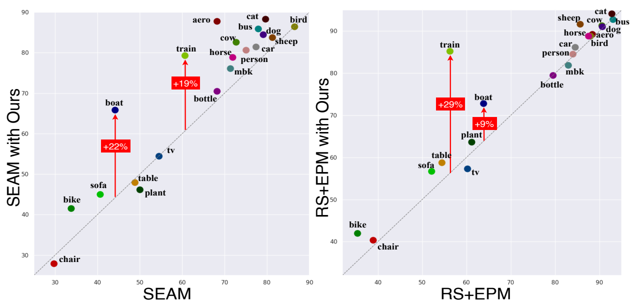

Figure 9. Category-wise comparison with SEAM [49], RS+EPM [21], and ours in terms of the IoU (%) on PASCAL VOC 2012 set.

Reasoning of debiased centroids. We quantify the ratio of target centroids in debiased centroids on the PASCAL VOC 2012 *train* set. Fig. 3 shows that K-means clustering separates two centroids (pink and orange) from decomposed embedding vectors for each class. We then measure each IoU score per centroid using pixel-wise annotations (each color has the IoU score). For simplicity, we classify all target and biased centroids based on their IoU scores, with target centroids having an IoU score above 0.3, biased centroids below 0.1, and others not visualized. Fig. 8 shows the visualization of target and biased centroids per class after dimensional reduction using T-SNE [46]. The ratio of target centroids selected for all foreground classes is more than 85% on the PASCAL VOC 2012 dataset (see other visualizations for all foreground classes in Appendix), validating the effectiveness of the proposed selection.

Category-wise improvements. Fig. 9 presents a class-wise comparison of our method with existing WSSS methods [49, 21] on the PASCAL VOC validation set. Our method improves the mIoU scores of most categories. However, the performance of a few categories (*e.g.*, tv/monitor) marginally decreases due to the poor quality of pseudo masks produced from the WSSS method. Notably, our method achieves significant improvements in the boat (+9%) and train (+29%) classes over RS+EPM [21], demonstrating the superiority of our method in removing biased objects without additional supervision. We also provide class-wise improvements for other WSSS methods [1, 28] in Appendix.

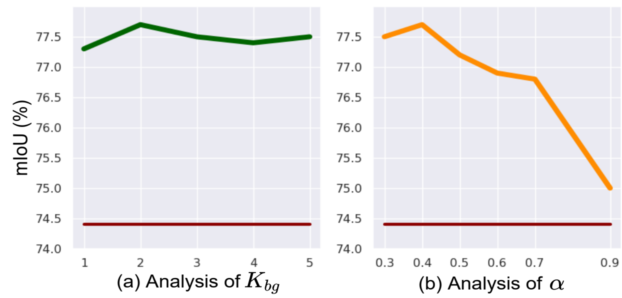

Figure 10. Sensitivity analysis of two hyperparameters K_{bg} and α. The mIoU scores are calculated on PASCAL VOC 2012 *val* set. The red line is our baseline RS+EPM [21].

Hyperparameters. We conduct the sensitivity analysis on two hyperparameters of our method, K_{bg} and α, using the PASCAL VOC 2012 validation set. Fig. 10 illustrates evaluation results. Our method improves performance across all hyperparameter settings compared to our baseline RS+EPM [21] (the red line). Varying K_{bg} from 1 to 5 does not significantly affect our method's performance, indicating this hyperparameter's stability. On the other hand, larger values of α (> 0.5) result in only marginal improvements due to the difficulty in disentangling biased and target centroids. Conversely, smaller values of α (< 0.5) show sufficient improvements, demonstrating the validity of this hyperparameter to select debiased centroids based on the distance of all background centroids. These results further support the effectiveness of our method and provide insights for setting hyperparameters.

Latency. When tested on an Intel Xeon Gold 6130 CPU with 64 cores, our clustering and refinement steps in Secs. 3.2 and 3.3 take 10 and 9 minutes on the PASCAL VOC training set, respectively.

5. Conclusion

In this work, we present MARS, a novel model-agnostic approach that addresses the biased problem in WSSS simply by exploiting the principle that USS-based information of biased objects can be easily matched with that of backgrounds of other samples. Accordingly, our approach significantly reduces FP due to WSSS bias, which is the primary reason that WSSS performance is limited compared to FSSS, achieves the fully-automatic biased removal without additional human resources, and complements debiased pixels with online predictions to avoid possible FN increases due to that removal. Thanks to following a model-agnostic manner, our approach yields consistent improvements when integrated with previous WSSS methods, narrowing the performance gap of 53% between WSSS and FSSS. We believe the simplicity and effectiveness of our system will benefit future research in weakly- and semi-supervised tasks under the real industry with multiple labels and complex relationships.

References

[1] Jiwoon Ahn, Sunghyun Cho, and Suha Kwak. Weakly supervised learning of instance segmentation with inter-pixel relations. In *IEEE CVPR*, pages 2209–2218, 2019. 2, 3, 4, 6, 7, 8

[2] Jiwoon Ahn and Suha Kwak. Learning pixel-level semantic affinity with image-level supervision for weakly supervised semantic segmentation. In *IEEE CVPR*, pages 4981–4990, 2018. 1, 6, 7

[3] Nikita Araslanov and Stefan Roth. Single-stage semantic segmentation from image labels. In *IEEE CVPR*, pages 4253–4262, 2020. 7

[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. Springer, 2016. 1

[5] Mathilde Caron, Piotr Bojanowski, Armand Joulin, and Matthijs Douze. Deep clustering for unsupervised learning of visual features. In *ECCV*, pages 132–149, 2018. 3

[6] Mathilde Caron, Hugo Touvron, Ishan Misra, Hervé Jégou, Julien Mairal, Piotr Bojanowski, and Armand Joulin. Emerging properties in self-supervised vision transformers. In *IEEE ICCV*, pages 9650–9660, 2021. 3

[7] Liang-Chieh Chen, George Papandreou, Iasonas Kokkinos, Kevin Murphy, and Alan L Yuille. Deeplab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected CRFs. *IEEE TPAMI*, 40(4):834–848, 2017. 1

[8] 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. 1

[9] Qi Chen, Lingxiao Yang, Jian-Huang Lai, and Xiaohua Xie. Self-supervised image-specific prototype exploration for weakly supervised semantic segmentation. In *IEEE CVPR*, pages 4288–4298, 2022. 3, 7

[10] Zhaozheng Chen, Tan Wang, Xiongwei Wu, Xian-Sheng Hua, Hanwang Zhang, and Qianru Sun. Class re-activation maps for weakly-supervised semantic segmentation. In *IEEE CVPR*, pages 969–978, 2022. 1, 7

[11] Jang Hyun Cho, Utkarsh Mall, Kavita Bala, and Bharath Hariharan. Picie: Unsupervised semantic segmentation using invariance and equivariance in clustering. In *IEEE CVPR*, pages 16794–16804, 2021. 3, 5

[12] Jifeng Dai, Kaiming He, and Jian Sun. Boxsup: Exploiting bounding boxes to supervise convolutional networks for semantic segmentation. In *IEEE ICCV*, pages 1635–1643, 2015. 1

[13] Ye Du, Zehua Fu, Qingjie Liu, and Yunhong Wang. Weakly supervised semantic segmentation by pixel-to-prototype contrast. In *IEEE CVPR*, pages 4320–4329, 2022. 3, 7

[14] Mark Everingham, Luc Van Gool, Christopher KI Williams, John Winn, and Andrew Zisserman. The pascal visual object classes (VOC) challenge. *IJCV*, 88(2):303–338, 2010. 2, 6

[15] Mark Hamilton, Zhoutong Zhang, Bharath Hariharan, Noah Snavely, and William T. Freeman. Unsupervised semantic segmentation by distilling feature correspondences. In *ICLR*, 2022. 2, 3, 4, 5, 7

[16] Qibin Hou, Ming-Ming Cheng, Xiaowei Hu, Ali Borji, Zhuowen Tu, and Philip HS Torr. Deeply supervised salient object detection with short connections. In *IEEE CVPR*, pages 3203–3212, 2017. 7

[17] Zilong Huang, Xinggang Wang, Jiasi Wang, Wenyu Liu, and Jingdong Wang. Weakly-supervised semantic segmentation network with deep seeded region growing. In *IEEE CVPR*, pages 7014–7023, 2018. 7

[18] Xu Ji, Joao F Henriques, and Andrea Vedaldi. Invariant information clustering for unsupervised image classification and segmentation. In *IEEE ICCV*, pages 9865–9874, 2019. 3

[19] Peng-Tao Jiang, Yuqi Yang, Qibin Hou, and Yunchao Wei. L2g: A simple local-to-global knowledge transfer framework for weakly supervised semantic segmentation. In *IEEE CVPR*, pages 16886–16896, 2022. 3, 7

[20] Sanghyun Jo and In-Jae Yu. Puzzle-cam: Improved localization via matching partial and full features. In *IEEE ICIP*, pages 639–643. IEEE, 2021. 3

[21] Sanghyun Jo, In-Jae Yu, and Kyungsu Kim. Recurseed and edgepredictmix: Single-stage learning is sufficient for weakly-supervised semantic segmentation. *arXiv preprint arXiv:2204.06754*, 2022. 1, 2, 3, 4, 6, 7, 8

[22] Beomyoung Kim, Sangeun Han, and Junmo Kim. Discriminative region suppression for weakly-supervised semantic segmentation. In *AAAI*, volume 35, pages 1754–1761, 2021. 3, 7

[23] Philipp Krähenbühl and Vladlen Koltun. Efficient inference in fully connected CRFs with gaussian edge potentials. *NeurIPS*, 24:109–117, 2011. 5, 6

[24] Alina Kuznetsova, Hassan Rom, Neil Alldrin, Jasper Uijlings, Ivan Krasin, Jordi Pont-Tuset, Shahab Kamali, Stefan Popov, Matteo Malloci, Alexander Kolesnikov, et al. The open images dataset v4: Unified image classification, object detection, and visual relationship detection at scale. *IJCV*, 128(7):1956–1981, 2020. 2, 3, 7

[25] Hyeokjun Kweon, Sung-Hoon Yoon, Hyeonseong Kim, Daehee Park, and Kuk-Jin Yoon. Unlocking the potential of ordinary classifier: Class-specific adversarial erasing framework for weakly supervised semantic segmentation. In *IEEE ICCV*, pages 6994–7003, 2021. 3, 7

[26] Jungbeom Lee, Jooyoung Choi, Jisoo Mok, and Sungroh Yoon. Reducing information bottleneck for weakly supervised semantic segmentation. *NeurIPS*, 34, 2021. 3, 7

[27] Jungbeom Lee, Eunji Kim, Sungmin Lee, Jangho Lee, and Sungroh Yoon. Ficklenet: Weakly and semi-supervised semantic image segmentation using stochastic inference. In *IEEE CVPR*, pages 5267–5276, 2019. 3, 7

[28] Jungbeom Lee, Eunji Kim, and Sungroh Yoon. Anti-adversarially manipulated attributions for weakly and semi-supervised semantic segmentation. In *IEEE CVPR*, pages 4071–4080, 2021. 1, 2, 3, 4, 6, 7, 8

[29] Jungbeom Lee, Seong Joon Oh, Sangdoo Yun, Junsuk Choe, Eunji Kim, and Sungroh Yoon. Weakly supervised semantic segmentation using out-of-distribution data. In *IEEE CVPR*, pages 16897–16906, 2022. 2, 3, 4, 7

[30] Minhyun Lee, Dongseob Kim, and Hyunjung Shim. Threshold matters in wsss: Manipulating the activation for the robust and accurate segmentation model against thresholds. In *IEEE CVPR*, pages 4330–4339, 2022. 3

[31] 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 *IEEE CVPR*, pages 5495–5505, 2021. 3, 7

[32] Jing Li, Junsong Fan, and Zhaoxiang Zhang. Towards noiseless object contours for weakly supervised semantic segmentation. In *IEEE CVPR*, pages 16856–16865, 2022. 3, 7

[33] Yi Li, Yiqun Duan, Zhanghui Kuang, Yimin Chen, Wayne Zhang, and Xiaomeng Li. Uncertainty estimation via response scaling for pseudo-mask noise mitigation in weakly-supervised semantic segmentation. In *AAAI*, volume 36, pages 1447–1455, 2022. 7

[34] Yi Li, Zhanghui Kuang, Liyang Liu, Yimin Chen, and Wayne Zhang. Pseudo-mask matters in weakly-supervised semantic segmentation. In *IEEE ICCV*, pages 6964–6973, 2021. 3, 7

[35] Di Lin, Jifeng Dai, Jiaya Jia, Kaiming He, and Jian Sun. Scribblesup: Scribble-supervised convolutional networks for semantic segmentation. In *IEEE CVPR*, pages 3159–3167, 2016. 1

[36] 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. Springer, 2014. 2, 6

[37] Jiang-Jiang Liu, Qibin Hou, Ming-Ming Cheng, Jiashi Feng, and Jianmin Jiang. A simple pooling-based design for real-time salient object detection. In *IEEE CVPR*, pages 3917–3926, 2019. 7

[38] Sheng Liu, Kangning Liu, Weicheng Zhu, Yiqiu Shen, and Carlos Fernandez-Granda. Adaptive early-learning correction for segmentation from noisy annotations. In *IEEE CVPR*, pages 2606–2616, 2022. 3, 7

[39] Yassine Ouali, Céline Hudelot, and Myriam Tami. Autoregressive unsupervised image segmentation. In *ECCV*, pages 142–158. Springer, 2020. 3

[40] Youwei Pang, Xiaoqi Zhao, Lihe Zhang, and Huchuan Lu. Multi-scale interactive network for salient object detection. In *IEEE CVPR*, pages 9413–9422, 2020. 7

[41] Jie Qin, Jie Wu, Xuefeng Xiao, Lujun Li, and Xingang Wang. Activation modulation and recalibration scheme for weakly supervised semantic segmentation. In *AAAI*, volume 36, pages 2117–2125, 2022. 3, 7

[42] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In *ICML*, pages 8748–8763. PMLR, 2021. 3, 7

[43] Lixiang Ru, Yibing Zhan, Baosheng Yu, and Bo Du. Learning affinity from attention: end-to-end weakly-supervised semantic segmentation with transformers. In *IEEE CVPR*, pages 16846–16855, 2022. 3

[44] Yukun Su, Ruizhou Sun, Guosheng Lin, and Qingyao Wu. Context decoupling augmentation for weakly supervised semantic segmentation. In *IEEE ICCV*, pages 7004–7014, 2021. 3, 7

[45] Guolei Sun, Wenguan Wang, Jifeng Dai, and Luc Van Gool. Mining cross-image semantics for weakly supervised semantic segmentation. In *ECCV*, pages 347–365. Springer, 2020. 3

[46] Laurens Van der Maaten and Geoffrey Hinton. Visualizing data using t-sne. *Journal of machine learning research*, 9(11), 2008. 8

[47] Wouter Van Gansbeke, Simon Vandenhende, Stamatios Georgoulis, and Luc Van Gool. Unsupervised semantic segmentation by contrasting object mask proposals. In *IEEE ICCV*, pages 10052–10062, 2021. 3

[48] Wouter Van Gansbeke, Simon Vandenhende, and Luc Van Gool. Discovering object masks with transformers for unsupervised semantic segmentation. *arXiv preprint arXiv:2206.06363*, 2022. 3

[49] Yude Wang, Jie Zhang, Meina Kan, Shiguang Shan, and Xilin Chen. Self-supervised equivariant attention mechanism for weakly supervised semantic segmentation. In *IEEE CVPR*, pages 12275–12284, 2020. 1, 2, 3, 4, 6, 7, 8

[50] Tong Wu, Junshi Huang, Guangyu Gao, Xiaoming Wei, Xiaolin Wei, Xuan Luo, and Chi Harold Liu. Embedded discriminative attention mechanism for weakly supervised semantic segmentation. In *IEEE CVPR*, pages 16765–16774, 2021. 3, 7

[51] Jinheng Xie, Xianxu Hou, Kai Ye, and Linlin Shen. Clims: Cross language image matching for weakly supervised semantic segmentation. In *IEEE CVPR*, pages 4483–4492, 2022. 1, 2, 3, 4, 7

[52] Jinheng Xie, Jianfeng Xiang, Junliang Chen, Xianxu Hou, Xiaodong Zhao, and Linlin Shen. C2am: Contrastive learning of class-agnostic activation map for weakly supervised object localization and semantic segmentation. In *IEEE CVPR*, pages 989–998, 2022. 3

[53] Lian Xu, Wanli Ouyang, Mohammed Bennamoun, Farid Boussaid, and Dan Xu. Multi-class token transformer for weakly supervised semantic segmentation. In *IEEE CVPR*, pages 4310–4319, 2022. 3, 7

[54] Hongshan Yu, Zhengeng Yang, Lei Tan, Yaonan Wang, Wei Sun, Mingui Sun, and Yandong Tang. Methods and datasets on semantic segmentation: A review. *Neurocomputing*, 304:82–103, 2018. 1

[55] Bingfeng Zhang, Jimin Xiao, Yunchao Wei, Mingjie Sun, and Kaizhu Huang. Reliability does matter: An end-to-end weakly supervised semantic segmentation approach. In *AAAI*, volume 34, pages 12765–12772, 2020. 3, 7

[56] Fei Zhang, Chaochen Gu, Chenyue Zhang, and Yuchao Dai. Complementary patch for weakly supervised semantic segmentation. In *IEEE ICCV*, pages 7242–7251, 2021. 3, 7

[57] Tianfei Zhou, Meijie Zhang, Fang Zhao, and Jianwu Li. Regional semantic contrast and aggregation for weakly supervised semantic segmentation. In *IEEE CVPR*, pages 4299–4309, 2022. 3, 7

[58] Adrian Ziegler and Yuki M Asano. Self-supervised learning of object parts for semantic segmentation. In *IEEE CVPR*, pages 14502–14511, 2022. 3, 4, 5, 7