FPR: False Positive Rectification for Weakly Supervised Semantic Segmentation

Liyi Chen[1,2] Chenyang Lei[2] Ruihuang Li[1] Shuai Li[1] Zhaoxiang Zhang[1,2,3*] Lei Zhang[1*]
[1]The Hong Kong Polytechnic University [2]Center for Artificial Intelligence and Robotics, HKISI, CAS
[3]State Key Laboratory of Multimodal Artificial Intelligence Systems, CASIA
liyi0308.chen@connect.polyu.hk, {leichenyang7,lishuai9401}@gmail.com,
zhaoxiang.zhang@ia.ac.cn, {csrhli, cslzhang}@comp.polyu.edu.hk

Abstract

*Many weakly supervised semantic segmentation (WSSS) methods employ the class activation map (CAM) to generate the initial segmentation results. However, CAM often fails to distinguish the foreground from its co-occurred background (*e.g.*, train and railroad), resulting in inaccurate activation from the background. Previous endeavors address this co-occurrence issue by introducing external supervision and human priors. In this paper, we present a False Positive Rectification (FPR) approach to tackle the co-occurrence problem by leveraging the false positives of CAM. Based on the observation that the CAM-activated regions of absent classes contain class-specific co-occurred background cues, we collect these false positives and utilize them to guide the training of CAM network by proposing a region-level contrast loss and a pixel-level rectification loss. Without introducing any external supervision and human priors, the proposed FPR effectively suppresses wrong activations from the background objects. Extensive experiments on the PASCAL VOC 2012 and MS COCO 2014 demonstrate that FPR brings significant improvements for off-the-shelf methods and achieves state-of-the-art performance. Code is available at https://github.com/mt-cly/FPR.*

1. Introduction

Semantic segmentation aims to assign a semantic label to each pixel of an input image. Benefiting from the strong capability of deep neural network (DNN) to extract semantic features, tremendous progress [41, 42] has been made on semantic segmentation with successful applications in autonomous driving, image editing, medical image analysis, *etc*. However, the training of a fully supervised segmentation model demands a large number of pixel-level annotations, which is labor-intensive and time-consuming. To reduce the cost, weakly supervised semantic segmentation (WSSS) has been proposed to employ weaker supervision, such as image labels [22, 5, 2, 54], bounding box [13, 46], scribbles [38] and points [3] to train the model. Among them, WSSS with image-level labels has attracted much attention since image labels can be easily obtained from popular datasets [40, 14, 16] or the Internet. In this work, we focus on the image-level labels based WSSS.

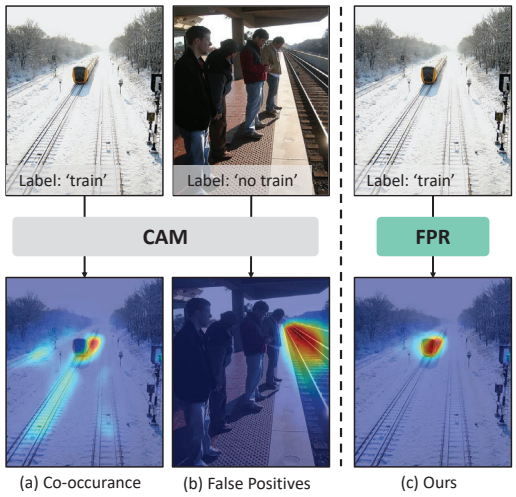


Figure 1. **The main idea of our approach.** (a) Conventional CAM often suffers from co-occurrence issue, *e.g.*, railroad and train. (b) The false positives of CAM contain the cues of co-occurred background, (c) which are leveraged in our FPR to distinguish foreground.

Most existing WSSS methods follow a two-stage pipeline: estimating the semantic localization maps by training a CAM [63] classification network with image-level labels, followed by a refinement stage. Unfortunately, CAM suffers from two critical problems, *i.e.*, *incomplete estimation* and *co-occurrence issue*. Incomplete estimation is caused by the fact that CAM only activates the most discriminative parts of objects, which has been well studied

*Corresponding author

in past years [22, 62, 2, 1, 17]. In opposite to the underestimation of incomplete estimation, co-occurrence issue refers to the overestimation of the foreground regions. Some background objects often appear together with foreground objects, which makes CAM confused to distinguish co-occurred pairs (*e.g.*, train and railroad, boat and water), resulting in co-occurred background falsely activated.

Several methods have been proposed to address the co-occurrence issue, EPS [33] trains the CAM classifier under the guidance of saliency maps, which helps to separate the foreground (*e.g.*, train) from the co-occurred background (*e.g.*, railroad). CLIMS [56] introduces the pre-trained CLIP [43] to locate and suppress the co-occurred background by feeding the corresponding textual prompts into the text encoder. In W-OoD [31], out-of-distribution images are manually collected as supplement training data to improve the CAM localization ability. However, these methods heavily rely on external supervision or require human priors to manually specify a list of classes suffering from co-occurrence issue [31, 56].

In this work, we propose a False Positive Rectification (FPR) approach to address the co-occurrence issue without introducing any external supervision (*e.g.*, saliency map [33], CLIP [56], and supplement images [31]) and human priors. The key insight of FPR is illustrated in Figure 1, we observe that the co-occurred background sometimes solely occurs in an image and will be falsely recognized as foreground class with relatively high probability in CAM, *e.g.*, the background railroad is recognized as train. Such wrong activation from CAM of absent class, termed false positives, are useful but neglected by previous methods. We argue that the WSSS co-occurrence issue can be alleviated by adequately leveraging false positives cues.

In FPR, false positives are fully exploited to guide network learning in a two-step manner: online prototype computing and training with prototypes. In the first step, all training images are fed into a trained CAM network to generate class-specific positive and negative prototypes according to given image-level labels. These dataset-level prototypes are able to comprehensively represent the semantics of foreground and the co-occurred background. In the second step, we propose a Region-level Contrast loss ($\mathcal{L}_{\text{RC}}$) and a Pixel-level Rectification loss ($\mathcal{L}_{\text{PR}}$) to train the network with prototypes. Specifically, $\mathcal{L}_{\text{RC}}$ pushes predicted region-level representations close to their positive prototypes and pulls them away from their negative prototypes. $\mathcal{L}_{\text{PR}}$ removes the pixels in the CAM-activated regions if their distances to negative prototypes are less than their distances to the positive prototypes in the representation space. These two loss functions work together to exclude co-occurred backgrounds without destroying the integrity of foreground objects. We perform the above two steps iteratively so that prototypes can be updated online and provide better guidance for training.

In summary, our main contributions:

- We experimentally demonstrate the co-occurrence issues of WSSS can be alleviated without introducing any additional external supervision and human priors.
- The design of FPR does not modify the architecture of CAM network, it can be seamlessly integrated into off-the-shelf WSSS methods to enhance performance.
- Extensive experiments on PASCAL VOC 2012 and MS COCO 2014 benchmarks demonstrate that FPR brings signification improvement and achieves state-of-the-art performance.

2. Related Work

In this section, we first review the existing WSSS methods using image-level labels, then review the relevant works using hard samples for learning.

Weakly supervised semantic segmentation. WSSS aims to generate high-quality pseudo segmentation masks, which enable the off-the-shelf segmentation models to achieve competitive performance with fully supervised methods. Most WSSS works take the outputs of the class activation map (CAM) [63] as the initial localization maps. However, CAM suffers from two critical issues: *incomplete estimation* and *co-occurrence*.

Incomplete estimation means that only the most discriminative parts of the object (*e.g.*, dog head), rather than a complete foreground region, will be activated in CAM. SEC [26] and DSRG [22] pick the high-probability pixels as the certain regions, and compare the similarity of ambiguous pixels with them to expand the activated regions. Adversarial based methods in [53, 21, 62, 49] force the network to focus on non-discriminative pixels by erasing the activated regions in the feature maps. IRN [1], BES [7], and SANCE [34] compute the pixel-pair affinity matrix to guide the random walk so that incomplete activation can be propagated to semantically similar areas. In summary, most WSSS methods [32, 48, 60, 61, 58] solve the incomplete estimation issue by expanding the activated regions in CAM.

Co-occurrence refers to that co-occurred semantic pairs (*e.g.*, train and railroad, boat and water) confuse the CAM and result in the wrong activation from the co-occurred background. This issue has attracted lots of attention recently, EPS [33] utilizes the saliency map as pseudo-pixel feedback to distinguish the foreground objects from the co-occurred background. CLIMS [56] relies on the cross-modality ability of pre-trained CLIP network to suppress background objects by feeding a predefined set of background text descriptions into the text encoder of CLIP. W-OoD [31] manually collects the supplementary training im-

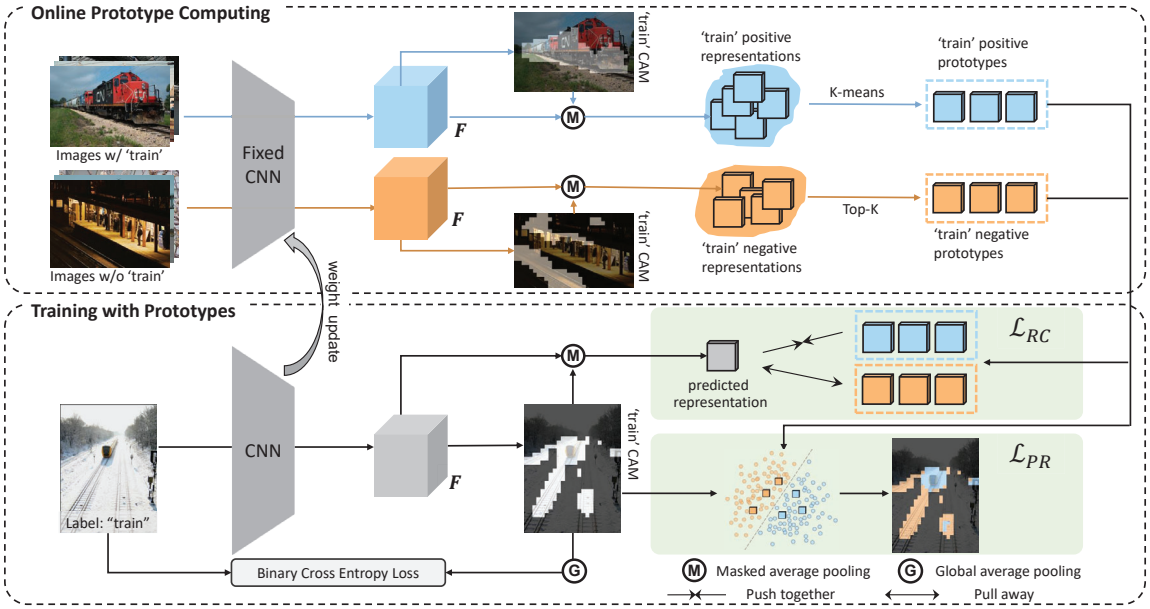


Figure 2. **The overview of the FPR approach for generating high-quality class localization maps.** The class-specific positive and negative prototypes containing co-occurred background cues are generated in the *online prototypes computing* stage, which are then used to calculate the losses $\mathcal{L}_{\text{RC}}$ and $\mathcal{L}_{\text{PR}}$ in the stage of *training with prototypes*. These two steps are performed iteratively.

ages that contain co-occurred background objects to facilitate the discrimination capability of the network. Different from the above methods, we address this issue without introducing external supervision and human prior.

Learning from hard samples. Previous works have demonstrated the importance of hard samples for deep model training. In fully supervised detection task [51, 18], OHEM [44] performs online hard sample mining to reduce the gap of sample imbalance. RetinaNet [39] proposes the focal loss to increase the contribution of hard negative samples, which significantly improves the detection performance. Cascade R-CNN [4] designs a cascade architecture so that easy samples and hard samples can be optimized in different stages. In fully supervised segmentation task, PointRend [25] introduces a rendering head to the segmentation model, which receives the uncertain points (*i.e.*, hard samples) as input and predicts their labels.

In WSSS, the activation in the CAM of absent class (*i.e.*, false positives) can be regarded as hard negative samples, which is ignored in previous WSSS methods. A contemporaneous study [12] employs false positives to guide fully supervised learning as well. However, in this paper, we focus on generating high-quality localization maps.

3. False Positive Rectification for WSSS

In this section, we first review the computation of conventional CAM in Section 3.1, then elaborate on our proposed False Positive Rectification (FPR) framework in Section 3.2. Finally, Section 3.3 applies the generated localization maps for WSSS.

3.1. Class Activation Map (CAM)

CAM is proposed to locate the foreground objects by training a classification network. For each training image I and its image-level label $y = [y_1, y_2, ..., y_C] \in \{0,1\}^C$, where C is the number of classes, CAM takes I as the input to extract high-level feature maps $F \in \mathbb{R}^{D\times H\times W}$, with D channels and $H \times W$ spatial size. To bridge the gap between the classification task and the segmentation task, a 1 $\times$ 1 convolution layer and a global average pooling (GAP) layer are employed to produce the logits prediction $\hat{y} \in \mathbb{R}^C$. During training, binary cross entropy loss is used as follows:

$$\mathcal{L}_{\text{BCE}} = -\frac{1}{C}\sum_{c=1}^{C}[y_c \cdot \log(\sigma(\hat{y_c})) + (1-y_c)\cdot\log(1-\sigma(\hat{y_c}))], \tag{1}$$

where $\sigma(\cdot)$ is the sigmoid function.

During inference, CAM feeds the feature map F into the trained 1 $\times$ 1 convolution layer to get logits output $M_c \in \mathbb{R}^{H\times W}$ for each class c. The binary class localization map $A_c \in \{0,1\}^{H\times W}$ can be obtained as:

$$A_c = \mathbb{1}(\frac{\text{RELU}(M_c)}{\max(\text{RELU}(M_c))} > \theta), \tag{2}$$

where $\mathbb{1}(\cdot)$ is the indicator function, the pixels with normalized values larger than the threshold θ are regarded as the foreground of class c.

3.2. The Proposed FPR Approach

The framework of our proposed FPR approach is illustrated in Figure 2. It consists of two steps: *online prototype computing* and *training with prototypes*. In the first step, we feed all training images into a trained classification network to derive representations of CAM-activated regions, which are used to generate class-specific positive/negative prototypes. In the second step, we train the CAM network with our proposed region-level contrast loss and pixel-level rectification loss under the guidance of prototypes. These two steps are iteratively performed to promote the capability of the network to distinguish co-occurred pairs.

3.2.1 Online Prototype Computing

Given an input image I and its image-level label y, we first compute its feature map F and normalized CAM A_c for each class c, as described in Section 3.1. Then we calculate the regional representation $f_c \in \mathbb{R}^D$, which is a compact feature embedding obtained by applying masked average pooling [64, 45] on CAM-activated regions as follows:

$$f_c = \frac{\sum_i A_c^i \cdot F^i}{\sum_i A_c^i}, \tag{3}$$

where i is the pixel over the class localization map A_c and feature map F.

To store the representations, we set up a positive representation pool $R_{\text{pos}} = [R_{\text{pos}}^1, R_{\text{pos}}^2, ..., R_{\text{pos}}^C]$ and a negative representation pool $R_{\text{neg}} = [R_{\text{neg}}^1, R_{\text{neg}}^2, ..., R_{\text{neg}}^C]$. Each entry R_{pos}^c or R_{neg}^c is a list of representations f_c belonging to the c-th class. In particular, a f_c will be added to R_{pos}^c if the c-th class appears in image I (*i.e.*, $y_c = 1$), otherwise, it is assigned to R_{neg}^c. With the dataset-level collection, R_{pos} captures the intrinsic properties of foreground classes, while R_{neg} collects the representations from class-specific false positives, which contain the co-occurred background cues.

Directly utilizing the collected representations in R_{pos} and R_{neg} to guide network training will cost massive computational resources. Therefore, for each class c, we build class-specific positive prototypes $P_{\text{pos}}^c = [P_{\text{pos}}^{c,1}, P_{\text{pos}}^{c,2}, ..., P_{\text{pos}}^{c,K}]$ and negative prototypes $P_{\text{neg}}^c = [P_{\text{neg}}^{c,1}, P_{\text{neg}}^{c,2}, ..., P_{\text{neg}}^{c,K}]$ from R_{pos}^c and R_{neg}^c, respectively, where K is the number of prototypes and it is set to a small number (*e.g.*, 10) to reduce the computational cost.

The generation processes of prototypes P_{pos}^c and P_{neg}^c are different. For P_{pos}^c, we set them as the cluster centroids obtained by performing k-means clustering on R_{pos}^c. However, k-means clustering is inapplicable for P_{neg}^c, since R_{neg}^c includes representations from various undefined semantics. For example, the representations of railroad, traffic light, or station would be collected as false positives of foreground train. Using k-means clustering to aggregate these semantic representations results in ambiguous negative prototypes. Therefore, we simply select the K representations with the highest class probability as the negative prototypes. In detail, we sort the representations f_c in R_{neg}^c in descending order according to the predicted probability $\hat{y}_c$, the top-K representations are selected as the negative prototypes P_{neg}^c.

Remarks on difference with previous works. The prototype-based methods has been well studied and successfully applied to WSSS [64, 15]. Different from previous methods only leveraging prototypes to reduce intra-class differences, our approach additionally introduces class-specific negative prototypes from false positives to enable network to distinguish co-occurred pairs.

3.2.2 Training with Prototypes

After obtaining the class-specific positive/negative prototypes, we use them to train the network. Similar to the representation collection process, we feed the training image I to CAM backbone to predict representation $f_c \in \mathbb{R}^D$, but here we only care about the classes appearing in the image (*i.e.*, $y_c = 1$). Region-level contrast loss $\mathcal{L}_{\text{RC}}$ is proposed to guide the learning of f_c. On the one hand, f_c is pulled to the center of positive prototypes to reduce intra-class variance. On the other hand, f_c is pushed away from its closest negative prototype to enlarge the difference between f_c and co-occurred background. $\mathcal{L}_{\text{RC}}$ is defined with cross entropy as follows:

$$\begin{aligned}\mathcal{L}_{\text{RC}} = -\frac{1}{\sum_{c=1}^{C} y_c} \sum_{c=1}^{C} y_c \cdot [&\log(S_{\text{pos}}(f_c, P_{\text{pos}}^c) \\ &+ \log(1 - S_{\text{neg}}(f_c, P_{\text{neg}}^c)],\end{aligned} \tag{4}$$

where S_{pos} and S_{neg} are functions to measure the similarity between representation f_c and prototypes based on their Euclidean Distance:

$$\begin{aligned} S_{\text{pos}}(f_c, P_{\text{pos}}^c) &= \exp\{-\frac{||f_c - \overline{P}_{\text{pos}}^c||_2}{\mathcal{T}}\}, \\ S_{\text{neg}}(f_c, P_{\text{neg}}^c) &= \exp\{-\frac{\min(||f_c - P_{\text{neg}}^c||_2)}{\mathcal{T}}\}, \end{aligned} \tag{5}$$

where $\mathcal{T}$ is the temperature to scale the Euclidean Distance.

$\mathcal{L}_{\text{RC}}$ makes foreground representations and co-occurred background representations separable by enlarging the distance between them. However, the benefit is limited, because the last 1×1 convolution layer of the network does not receive gradients from $\mathcal{L}_{\text{RC}}$, this linear classifier is not aware that which representation should be filtered out. Therefore, we propose a pixel-level rectification loss $\mathcal{L}_{\text{PR}}$ to suppress the logit outputs of pixels that are similar to the background. Specifically, we check each pixel activated in CAM and suppress the pixel if the distance from its representation to the closest negative prototype is less than the distance to the closest positive prototype. Let Φ_c denote the

set of pixels to be suppressed in the c-th CAM:

$$\Phi_c = \{i | \min(||F^i - P^c_{\text{neg}}||_2) < \min(||F^i - P^c_{\text{pos}}||_2), A^i_c = 1\}. \tag{6}$$

The loss $\mathcal{L}_{\text{PR}}$ is defined as follows:

$$\mathcal{L}_{\text{PR}} = \frac{1}{\sum_{c=1}^{C} y_c \cdot |\Phi_c|} \sum_{c=1}^{C} y_c \cdot \sum_{i \in \Phi_c} M^i_c, \tag{7}$$

where M^i_c is the logits value of pixel i at M_c.

Although $\mathcal{L}_{\text{PR}}$ works well in suppressing the co-occurred background, parts of the foreground regions like the car wheels are suppressed as well. The reason is that some negative prototypes are from regions shared by different objects. For example, the wheels from a bus image would be collected as false positives of car, which misleads the network to discard the wheel parts of the car. To alleviate this problem, we propose to filter out negative prototypes of shared regions by leveraging the super-class information provided by dataset, which is usually utilized to build the tree-structured class dependencies [36]. When calculating the $\mathcal{L}_{\text{PR}}$ for a class (*e.g.*, train), we only remain the negative prototypes from images where all appeared classes do not belong to the same super-class (*e.g.*, vehicle). By filtering out those negative prototypes from shared regions in P^c_{neg}, we get trimmed $P^{c\prime}_{\text{neg}}$ and update the pixel set Φ_c as follows:

$$\Phi_c = \{i | \min(||F^i - P^{c\prime}_{\text{neg}}||_2) < \min(||F^i - P^c_{\text{pos}}||_2), A^i_c = 1\}. \tag{8}$$

3.2.3 Network Optimization

The final loss to train the network is the combination of the binary cross entropy loss and the two proposed losses:

$$\mathcal{L} = \mathcal{L}_{\text{BCE}} + \lambda_1 \cdot \mathcal{L}_{\text{RC}} + \lambda_2 \cdot \mathcal{L}_{\text{PR}}, \tag{9}$$

where the λ_1 and λ_2 are the loss weight parameters.

The two steps of online prototype computing and training with prototypes benefit each other. A trained network produces more discriminative representations for prototypes, while these prototypes could provide better guidance for network training. Therefore, we iteratively perform these two steps by updating the network with trained weight for online prototype computing before each epoch.

3.3. Training Segmentation Networks

Our FPR is able to benefit the performance of CAM. Following the popular practice [6, 30, 59, 52] in weakly supervised semantic segmentation, we refine the high-quality localization maps obtained by proposed FPR with established methods (*e.g.* IRN[1]) to generate pseudo segmentation masks, which are subsequently used to train the segmentation network [9] under the fully supervised setting.

Methods	Prec.	Recall	Seed	+CRF
CAM[63]$_{\text{CVPR16}}$	61.9	72.7	49.5	54.3
+FPR (Ours)	**66.7**	**74.7**	**54.3** (+4.8)	**59.6** (+5.3)
SEAM[52]$^*_{\text{CVPR20}}$	67.2	76.5	54.8	-
+FPR (Ours)	**68.9**	**77.1**	**57.0** (+2.2)	**60.8**
AdvCAM[30]$_{\text{CVPR21}}$	66.8	77.6	55.5	62.1
+FPR(Ours)	**71.5**	**78.1**	**59.7** (+4.2)	**65.5** (+3.4)
MCTformer[57]$_{\text{CVPR22}}$	-	-	61.7	-
+FPR(Ours)	**75.0**	**81.0**	**63.8** (+2.1)	**66.4**

Table 1. **Comparison of the quality of localization maps evaluated on PASCAL VOC 2012 *train* set.** $*$ means the reproduced results from W-OoD.

4. Experiments

4.1. Experimental Setup

Dataset and evaluation metric. We conduct the experiments on the PASCAL VOC 2012 [16] and MS COCO 2014 [40] benchmarks. VOC 2012 consists of 21 classes (*i.e.*, 20 foreground objects plus background), following the common protocol in WSSS, the augmented training images of 10,582 images [19] and associated image-level labels are used for training. A total of 1,464 *val* images and 1,449 *test* images with pixel-wise segmentation masks are used for evaluation. COCO 2014 includes 81 classes (80 foregrounds and background), it has around 80*k* and 40*k* images in *train* set and *val* set, respectively. The evaluation metric is the mean Intersection-over-Union (mIoU) [41].

Implementation details. In our experiments, we adopt the ResNet50 [20] pre-trained on ImageNet [14] as the backbone of the classification network with the output stride of 16, the optimizer and data augmentation are the same as [30, 31]. For PASCAL VOC 2012, the θ in Equation 2 is set to the default background threshold 0.1, and the number of prototypes K described in Section 3.2.1 is set to 10. The temperature parameter $\mathcal{T}$ in Equation 5, and the two loss weight parameters of λ_1 and λ_2 in Equation 9 are set to 13, 12e-2, and 15e-5, respectively. For the segmentation network, we leverage generated pseudo segmentation masks to train models (DeepLab-v1 [8] with WResNet38 backbone and Deeplab-v2 [9] with ResNet101 backbone) to achieve final segmentation results.

4.2. Experimental Results

Quality of localization maps. Since the proposed FPR does not modify the architecture of CAM network, it can be seamlessly integrated into off-the-shelf methods. Table 1 presents the VOC 2012 performance of localization maps obtained by applying FPR to WSSS methods: baseline CAM[1], SEAM[52], and AdvCAM [30]. We observe that FPR brings significant improvements for all three methods. It is worth mentioning that the gain in CAM (+4.8%) is preserved in AdvCAM (+4.2%). We argue that this gain consistency is caused by different refinement aims: Adv-

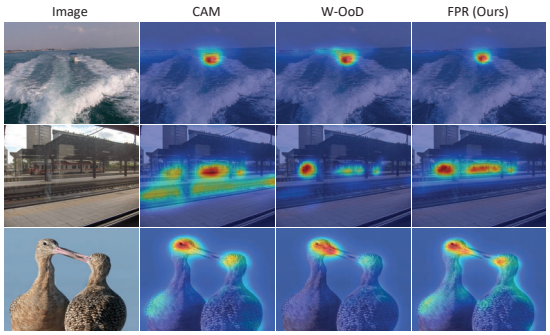


Figure 3. **Visualization of localization maps on VOC 2012 *train* set.** Our FPR accurately highlights the foreground regions without hurting the integration of objects.

Methods	Ext.	Backbone	*val*	*test*
FickleNet[29]$_{\text{CVPR19}}$	Saliency	ResNet101	64.9	65.3
OAA+[23]$_{\text{ICCV19}}$	Saliency	ResNet101	65.2	66.4
ICD[17]$_{\text{CVPR20}}$	Saliency	ResNet101	67.8	68.0
EDAM[55]$_{\text{CVPR21}}$	Saliency	ResNet101	70.9	70.6
EPS[33]$_{\text{CVPR21}}$	Saliency	ResNet101	71.0	71.8
CLIMS[56]$_{\text{CVPR22}}$	CLIP	ResNet101	70.4	70.0
W-OoD[31]$_{\text{CVPR22}}$	Images	WResNet38	70.7	70.1
L2G[24]$_{\text{CVPR22}}$	Saliency	ResNet101	**72.1**	**71.7**
IRN[1]$_{\text{CVPR19}}$	-	ResNet50	63.5	64.8
SEAM[52]$_{\text{CVPR20}}$	-	WResNet38	64.5	65.7
SCCAM[6]$_{\text{CVPR20}}$	-	ResNet101	66.1	65.9
BES[7]$_{\text{ECCV20}}$	-	ResNet101	65.7	66.6
CONTA[59]$_{\text{NIPS20}}$	-	ResNet101	66.1	66.7
CSE[27]$_{\text{ICCV21}}$	-	WResNet38	68.3	68.0
PMM[37]$_{\text{ICCV21}}$	-	WResNet38	68.5	69.0
ReCAM[11]$_{\text{CVPR22}}$	-	ResNet101	68.5	68.4
SIPE[10]$_{\text{CVPR22}}$	-	ResNet101	68.8	69.7
ESOL[35]$_{\text{NIPS22}}$	-	ResNet101	69.9	69.3
AdvCAM[30]$^{*}_{\text{CVPR21}}$	-	ResNet101	67.5	67.1
+FPR(Ours)	-	ResNet101	**70.3** (+2.8)	70.1 (+3.0)
+FPR(Ours)	-	WResNet38	70.0 (+2.5)	**70.6** (+3.5)

Table 2. **Comparison with other state-of-the-art methods on PASCAL VOC 2012 *val* and *test* set.** Ext. means external supervision. $*$ means the reproduced results from W-OoD.

CAM expands foreground regions to tackle the incomplete issue, FPR prevents the co-occurred background from being activated. The combination with AdvCAM builds a better performance in a complementary way.

Figure 3 illustrates the visualization comparison with baseline CAM, we additionally present W-OoD results since it aims to tackle co-occurrence as well. One can observe that both W-OoD and FPR are capable of distinguishing the co-occurred pairs (*e.g.*, train and station). However, parts of foreground object like bird bodies are suppressed wrongly in W-OoD, while FPR accurately captures the foreground without hurting the object integration.

Quality of final segmentation maps. Following the common practice [30, 31, 10], the generated high-quality localization maps are refined by IRN to get pseudo masks for DeepLab segmentation models [8, 9] training. Table 2 presents the performance comparison on the PASCAL VOC 2012 *val* and *test* sets. We find that FPR brings significant improvements for baseline AdvCAM (+3% mIoU on *test*). In addition, such performance is competitive against W-OoD, EPS, and CLIMS, which rely on external supervision to tackle the co-occurrence issue. The segmentation results on COCO 2014 are summarized in Table 3, one can observe that baseline CAM improves the segmentation performance by integrating FPR, introducing IRN refinement help FPR to achieve 43.9% mIoU on *val* set.

Methods	*train*	Backbone	*val*
SEAM[52]$_{\text{CVPR20}}$	-	VGG16	31.9
CONTA[59]$_{\text{NIPS20}}$	-	WResNet38	32.8
CDA[47]$_{\text{ICCV21}}$	-	WResNet38	33.2
EPS[33]$_{\text{CVPR21}}$	-	VGG16	35.7
ReCAM[11]$_{\text{CVPR22}}$	34.6	ResNet101	36.5
MCTformer[57]$_{\text{CVPR22}}$	-	WResNet38	42.0
ESOL[35]$_{\text{NIPS22}}$	-	ResNet101	42.6
RIB[28]$_{\text{NIPS21}}$	-	ResNet101	43.8
CAM[63]$^{*}_{\text{CVPR16}}$	33.1	ResNet101	35.7
+FPR(Ours)	33.9 (+0.8)	ResNet101	36.6 (+0.9)
IRN[1]$^{*}_{\text{CVPR19}}$	42.4	ResNet101	42.0
+FPR(Ours)	**44.0** (+1.6)	ResNet101	**43.9** (+1.9)

Table 3. **Performance comparison with other methods on MS COCO 2014 benchmark.** *Train* is the mIoU of pseudo masks, *val* is the performance of DeepLab trained with pseudo masks, $*$ means the reproduced results from ReCAM.

Methods	train IoU	boat IoU	mIoU
CAM [63]	49.8	33.3	49.5
W-OoD [31]	61.4	41.0	53.3
AdvCAM+W-OoD*	65.4	41.8	58.8
CLIMS [56]	63.9	**58.2**	56.6
FPR(Ours)	65.1 (+15.3)	44.7 (+11.4)	54.3 (+4.8)
AdvCAM+FPR(Ours)	**68.5** (+18.7)	49.0 (+15.7)	**59.7** (+10.2)

Table 4. **Comparison of the quality of generated localization maps on the PASCAL VOC 2012 *train* set**. Note that both W-OoD and CLIMS rely on external supervision and human priors. $*$ means the reproduced results.

The visualized segmentation maps of our method as well as baselines are presented in Figure 4. FPR shows better localization and segmentation capability, especially for classes burdened with co-occurrence issue. For example, FPR is capable of filtering out waterfront backgrounds and activating boats accurately.

Comparison with methods using external supervision. Previous works W-OoD [31] and CLIMS [56] introduce external supervision to refine a manually-defined class list suffering from co-occurrence. The performance comparison is shown in Table 4. We report the train, boat, and mean IoU of localization maps on the PASCAL VOC 2012 *train* set. Our FPR outperforms W-OoD in both settings of without and with AdvCAM. Compared to CLIMS, FPR achieves higher mIoU when integrating with AdvCAM (59.7% vs. 56.6%), although the IoU of the boat is not satisfied, this is mainly because the CAM, which is used as the starting

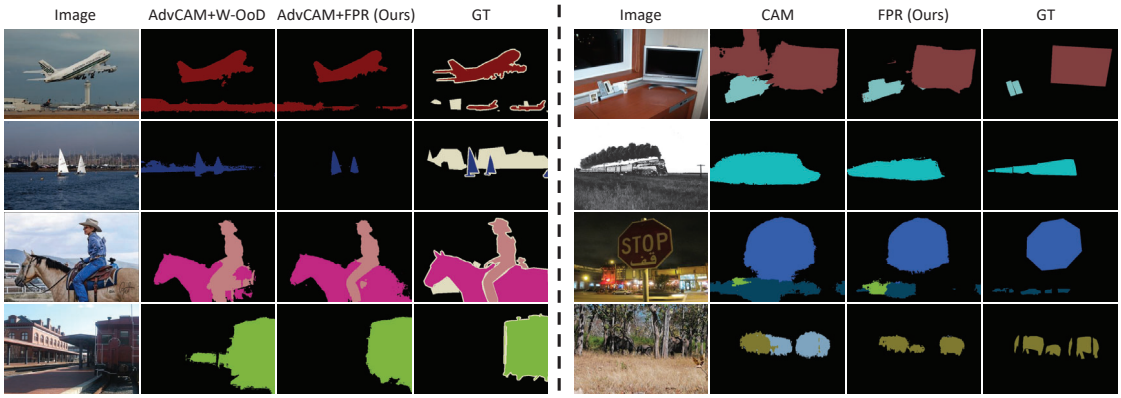


Figure 4. **Qualitative segmentation results on (a) PASCAL VOC 2012 and (b) MS COCO 2014 *val* sets.**

point in our method, achieves only 33.3% IoU on the boat.

4.3. Diagnostic Experiment

We conduct experiments on VOC 2012 datasets to perform a sanity check for our method. Specifically, we provide quantitative evaluation and qualitative visualization for two proposed losses respectively to illustrate how the co-occurrence issue is alleviated. In addition, we present the sensitivity analyses for parameters.

Ablation study. Table 5 presents the ablation study to evaluate the effect of the two proposed losses $\mathcal{L}_{\text{RC}}$ and $\mathcal{L}_{\text{PR}}$ for localization maps. Besides the baseline CAM trained with $\mathcal{L}_{\text{BCE}}$, we evaluate the performance of an intuitive idea leveraging the false positives for comparison, *i.e.*, we introduce the focal loss [39] to improve the contribution weights of hard negative samples. Based on Equation 1, the Focal $\mathcal{L}_{\text{BCE}}$ is defined as follows:

$$\begin{aligned}\text{Focal } \mathcal{L}_{\text{BCE}} = -\frac{1}{C}\sum_{c=1}^{C}[y_c \cdot \log(\sigma(\hat{y_c})) \\ +(\sigma(\hat{y_c}))^2 \cdot (1-y_c) \cdot \log(1-\sigma(\hat{y_c}))].\end{aligned} \tag{10}$$

From Table 5, we notice that simply amplifying the loss weights of hard negative samples (*i.e.*, false positives) only brings 0.6% mIoU improvement. In the proposed FPR, $\mathcal{L}_{\text{RC}}$ brings 2.9% improvement over the original CAM in mIoU score. By combining $\mathcal{L}_{\text{RC}}$ and $\mathcal{L}_{\text{PR}}$, FPR performs significantly better than CAM (54.3% vs. 49.5%).

Qualitative visualization of $\mathcal{L}_{\text{RC}}$. $\mathcal{L}_{\text{RC}}$ aims to enlarge the distance from the foreground to its corresponding negative prototypes in the representation space. Figure 5 uses t-SNE [50] to visualize the prototype localization of randomly selected three classes (*i.e.*, train, boat, and horse) in representation space. At epoch 0, class-specific positive prototypes contain the pixels from backgrounds, so that they are close to the negative prototypes, which makes the co-occurred pairs indistinguishable from the classification network. During FPR training, co-occurred pairs become distinct because the predicted representations are pulled away from the closest negative prototype. As shown in the figure, the new generated positive prototypes gradually move away from the initial negative prototypes.

Qualitative visualization of $\mathcal{L}_{\text{PR}}$. To check if the negative prototypes and pixel to be suppressed are semantically con-

$\mathcal{L}_{\text{BCE}}$	Focal $\mathcal{L}_{\text{BCE}}$	$\mathcal{L}_{\text{RC}}$	$\mathcal{L}_{\text{PR}}$	mIoU
✓				49.5
	✓			50.1
✓		✓		52.4
✓		✓	✓	**54.3**

Table 5. **Ablation study of loss terms in FPR.** The mIoU values are evaluated on PASCAL VOC 2012 *train* set. Proposed two losses brings +4.8% mIoU improvements.

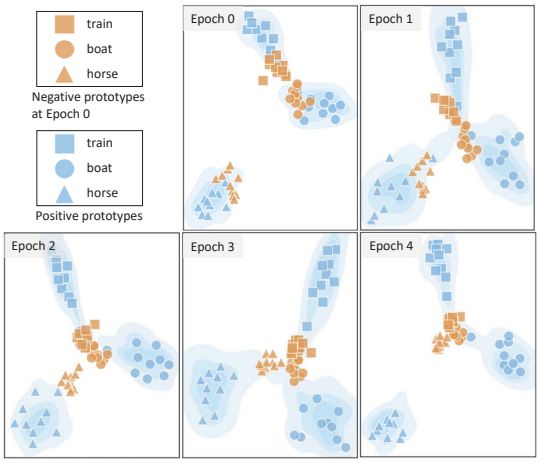


Figure 5. **Visualization of the effectiveness of $\mathcal{L}_{\text{RC}}$ with t-SNE [50].** We present the positive prototypes of three classes of horse, train, and boat, as well as their negative prototypes.

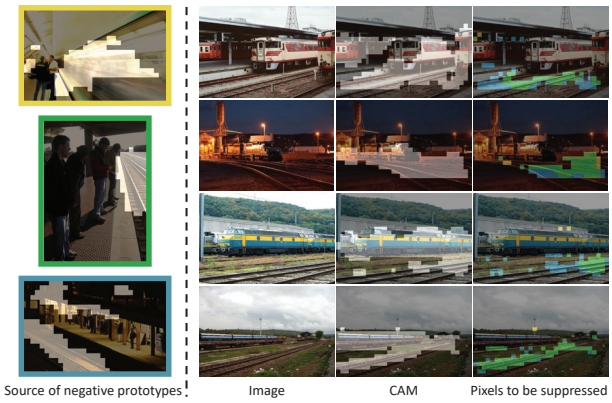
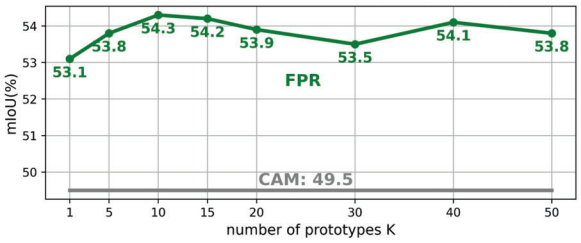


Figure 6. **Visualization of the effectiveness of $\mathcal{L}_{\text{PR}}$ for train class.** The pixels to be suppressed by $\mathcal{L}_{\text{PR}}$ are colorized. According to their closest negative prototypes in the representation space, the pixels are annotated with different colors.

λ_2 (e^{-5})	$\lambda_1(e^{-2})$ 10	12	15	17	20
10	53.3±0.3	53.6±0.2	53.8±0.4	54.2±0.3	54.1±0.6
15	53.4±0.4	54.3±0.3	54.3±0.2	*54.5±0.5*	54.2±0.6
20	53.2±0.5	53.3±0.5	53.6±0.6	53.8±0.6	53.4±0.4

(a) Results with different values of λ_1 and λ_2.



(b) Results with different numbers of prototypes K.

Figure 7. **The effect of (a) loss weight parameters and (b) number of prototypes K on PASCAL VOC 2012 *train* set.** The FPR performance is relatively stable with variations of parameters.

sistent, we colorize the pixels suppressed by $\mathcal{L}_{\text{PR}}$ in terms of train class. Particularly, we list three source of negative prototypes and use different colors for suppressed pixels to represent the source of its closest negative prototype. As illustrated in Figure 6, $\mathcal{L}_{\text{PR}}$ is able to accurately locate the position of the co-occurred background without hurting the integration of foreground objects. For the three presented negative prototypes, the last two prototypes from railroad regions make the major contribution to excavating background, which demonstrates that only a few false positives samples are needed to tackle the co-occurrence issue.

Parameters sensitivity analyses. We analyze the effect of parameters λ_1 and λ_2 in Equation 9 on the quality of localization maps. The results on VOC 2012 *train* set are summarized in Figure 7(a). We find that the mIoU of localization maps are relatively stable across different λ_1 when $\lambda_2 = 15e^{-5}$. We select the parameter values $\lambda_1 = 12e^{-2}$ and $\lambda_2 = 15e^{-5}$ as default setting.

We then study the impact of parameter K on the localization maps. As shown in Figure 7(b), when $K = 10$, FPR achieves the best performance with 54.3% mIoU. As K increases from 10 to 50, the performance slightly degenerates to 53.8% mIoU. Interestingly, FPR shows a considerable promotion (+3.6% mIoU) even with $K = 1$, which demonstrates that our approach is insensitive to the number of prototypes.

Quantitative analyses on false positives. To validate the ability of FPR to suppress the activation of false positives, we present three quantitative comparisons between CAM baseline and FPR on VOC *train* set to demonstrate false positives reduction. (1) BCE loss: (CAM: 0.0300 $vs.$ FPR: 0.0248); (2) Average predicted probability of absent classes, *i.e.*, $\frac{\sum_{c=1}^{C}(1-y_c)\sigma(\hat{y_c})}{\sum_{c=1}^{C}(1-y_c)}$: (CAM: 0.0120 $vs.$ FPR: 0.0086); (3) The False Discovery Rate, *i.e.*, FP/(TP+FP) for 20 classes are shown in Figure 8. One can observe that FPR significantly decreases the false positives occurred in CAM based on numerical comparison results.

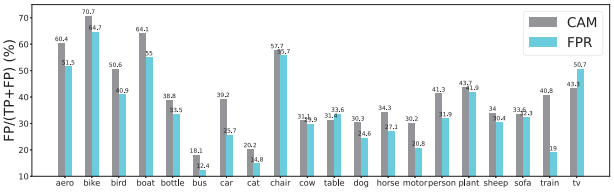


Figure 8. **The False Discovery Rate (FDR) comparison of CAM and FPR on PASCAL VOC 2012 *train* set.**

4.4. Discussion and Limitation

Discussion. Our method provides an effective way to tackle the co-occurrence issue by adequately exploiting the internal cues of false positives. Instead of only refining some specific classes as in W-OoD [31] and CLIMS [56], our proposed FPR applies the proposed losses to all classes equally. Interestingly, FPR could benefit most classes even those classes that do not struggle with co-occurrence issue, 19 out of 20 classes on VOC 2012 get performance improvement **(please refer to supplemental materials)**. We suspect that this is because the negative prototypes are usually close to the decision boundary in representation space, *i.e.*, most negative prototypes are in the figure center as shown in Figure 5. Applying $\mathcal{L}_{\text{RC}}$ to push the predicted representations away from the decision boundary can prevent ambiguous outputs from the network. Such an optimization strategy is applicable to all classes.

Limitation. We noticed that both in PASCAL VOC 2012 and MS COCO 2014 datasets, a small number of the classification labels are incorrect, which hurt the FPR perfor-

mance. Consider the following case, an appeared class is mistakenly labeled as absent, the representation of which may be taken as the negative prototype. At the training phase, pulling away prototypes belonging to the same semantic category with two proposed losses is meaningless and misguides the FPR learning. We believe that if all wrong classification labels are revised, the effectiveness of FPR will be further improved.

5. Conclusion

In this paper, we focused on the co-occurrence issue inherited in weakly supervised semantic segmentation. Inspired by the fact that the false positives of CAM contain class-specific co-occurred background cues, we proposed an FPR approach to leverage representations of false positives as guidance for CAM network training. We first collected the representations of CAM-activated regions to build class-specific positive and negative prototypes, which were then utilized to calculate region-level contrast loss and pixel-level rectification loss for network learning. Different from previous methods such as EPS [33], W-OoD [31] and CLIMS [56], which introduced external supervision to tackle the co-occurrence problem, our proposed FPR fully exploited the dataset internal information. Experiments on PASCAL VOC 2012 and MS COCO 2014 benchmarks demonstrated that FPR achieved competitive performance with the aforementioned methods, and outperformed other WSSS methods using only image-level labels.

References

[1] Jiwoon Ahn, Sunghyun Cho, and Suha Kwak. Weakly supervised learning of instance segmentation with inter-pixel relations. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 2209–2218, 2019. 2, 5, 6

[2] Jiwoon Ahn and Suha Kwak. Learning pixel-level semantic affinity with image-level supervision for weakly supervised semantic segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4981–4990, 2018. 1, 2

[3] Amy Bearman, Olga Russakovsky, Vittorio Ferrari, and Li Fei-Fei. What's the point: Semantic segmentation with point supervision. In *European conference on computer vision*, pages 549–565. Springer, 2016. 1

[4] Zhaowei Cai and Nuno Vasconcelos. Cascade r-cnn: Delving into high quality object detection. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 6154–6162, 2018. 3

[5] Yu-Ting Chang, Qiaosong Wang, Wei-Chih Hung, Robinson Piramuthu, Yi-Hsuan Tsai, and Ming-Hsuan Yang. Weakly-supervised semantic segmentation via sub-category exploration. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8991–9000, 2020. 1

[6] Yu-Ting Chang, Qiaosong Wang, Wei-Chih Hung, Robinson Piramuthu, Yi-Hsuan Tsai, and Ming-Hsuan Yang. Weakly-supervised semantic segmentation via sub-category exploration. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8991–9000, 2020. 5, 6

[7] Liyi Chen, Weiwei Wu, Chenchen Fu, Xiao Han, and Yuntao Zhang. Weakly supervised semantic segmentation with boundary exploration. In *European Conference on Computer Vision*, pages 347–362. Springer, 2020. 2, 6

[8] Liang-Chieh Chen, George Papandreou, Iasonas Kokkinos, Kevin Murphy, and Alan L Yuille. Semantic image segmentation with deep convolutional nets and fully connected crfs. *arXiv preprint arXiv:1412.7062*, 2014. 5, 6

[9] Liang-Chieh Chen, George Papandreou, Iasonas Kokkinos, Kevin Murphy, and Alan L Yuille. Deeplab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected crfs. *IEEE transactions on pattern analysis and machine intelligence*, 40(4):834–848, 2017. 5, 6

[10] Qi Chen, Lingxiao Yang, Jian-Huang Lai, and Xiaohua Xie. Self-supervised image-specific prototype exploration for weakly supervised semantic segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4288–4298, 2022. 6

[11] Zhaozheng Chen, Tan Wang, Xiongwei Wu, Xian-Sheng Hua, Hanwang Zhang, and Qianru Sun. Class re-activation maps for weakly-supervised semantic segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 969–978, 2022. 6

[12] Zesen Cheng, Pengchong Qiao, Kehan Li, Siheng Li, Pengxu Wei, Xiangyang Ji, Li Yuan, Chang Liu, and Jie Chen. Out-of-candidate rectification for weakly supervised semantic segmentation. *arXiv preprint arXiv:2211.12268*, 2022. 3

[13] Jifeng Dai, Kaiming He, and Jian Sun. Boxsup: Exploiting bounding boxes to supervise convolutional networks for semantic segmentation. In *Proceedings of the IEEE international conference on computer vision*, pages 1635–1643, 2015. 1

[14] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *2009 IEEE conference on computer vision and pattern recognition*, pages 248–255. Ieee, 2009. 1, 5

[15] Ye Du, Zehua Fu, Qingjie Liu, and Yunhong Wang. Weakly supervised semantic segmentation by pixel-to-prototype contrast. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4320–4329, 2022. 4

[16] Mark Everingham, Luc Van Gool, Christopher KI Williams, John Winn, and Andrew Zisserman. The pascal visual object classes (voc) challenge. *International journal of computer vision*, 88(2):303–338, 2010. 1, 5

[17] Junsong Fan, Zhaoxiang Zhang, Chunfeng Song, and Tieniu Tan. Learning integral objects with intra-class discriminator for weakly-supervised semantic segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4283–4292, 2020. 2, 6

[18] Pedro F Felzenszwalb, Ross B Girshick, and David McAllester. Cascade object detection with deformable part models. In *2010 IEEE Computer society conference on computer vision and pattern recognition*, pages 2241–2248. Ieee, 2010. 3

[19] Bharath Hariharan, Pablo Arbeláez, Lubomir Bourdev, Subhransu Maji, and Jitendra Malik. Semantic contours from inverse detectors. In *2011 international conference on computer vision*, pages 991–998. IEEE, 2011. 5

[20] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016. 5

[21] Qibin Hou, PengTao Jiang, Yunchao Wei, and Ming-Ming Cheng. Self-erasing network for integral object attention. *Advances in Neural Information Processing Systems*, 31, 2018. 2

[22] Zilong Huang, Xinggang Wang, Jiasi Wang, Wenyu Liu, and Jingdong Wang. Weakly-supervised semantic segmentation network with deep seeded region growing. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 7014–7023, 2018. 1, 2

[23] Peng-Tao Jiang, Qibin Hou, Yang Cao, Ming-Ming Cheng, Yunchao Wei, and Hong-Kai Xiong. Integral object mining via online attention accumulation. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 2070–2079, 2019. 6

[24] 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 *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 16886–16896, 2022. 6

[25] Alexander Kirillov, Yuxin Wu, Kaiming He, and Ross Girshick. Pointrend: Image segmentation as rendering. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 9799–9808, 2020. 3

[26] Alexander Kolesnikov and Christoph H Lampert. Seed, expand and constrain: Three principles for weakly-supervised image segmentation. In *European conference on computer vision*, pages 695–711. Springer, 2016. 2

[27] 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 *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 6994–7003, 2021. 6

[28] Jungbeom Lee, Jooyoung Choi, Jisoo Mok, and Sungroh Yoon. Reducing information bottleneck for weakly supervised semantic segmentation. *Advances in Neural Information Processing Systems*, 34:27408–27421, 2021. 6

[29] Jungbeom Lee, Eunji Kim, Sungmin Lee, Jangho Lee, and Sungroh Yoon. Ficklenet: Weakly and semi-supervised semantic image segmentation using stochastic inference. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5267–5276, 2019. 6

[30] Jungbeom Lee, Eunji Kim, and Sungroh Yoon. Anti-adversarially manipulated attributions for weakly and semi-supervised semantic segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4071–4080, 2021. 5, 6

[31] Jungbeom Lee, Seong Joon Oh, Sangdoo Yun, Junsuk Choe, Eunji Kim, and Sungroh Yoon. Weakly supervised semantic segmentation using out-of-distribution data. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 16897–16906, 2022. 2, 5, 6, 8, 9

[32] Minhyun Lee, Dongseob Kim, and Hyunjung Shim. Threshold matters in wsss: Manipulating the activation for the robust and accurate segmentation model against thresholds. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4330–4339, 2022. 2

[33] 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 *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 5495–5505, 2021. 2, 6, 9

[34] Jing Li, Junsong Fan, and Zhaoxiang Zhang. Towards noiseless object contours for weakly supervised semantic segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 16856–16865, 2022. 2

[35] Jinlong Li, Zequn Jie, Xu Wang, Xiaolin Wei, and Lin Ma. Expansion and shrinkage of localization for weakly-supervised semantic segmentation. *arXiv preprint arXiv:2209.07761*, 2022. 6

[36] Liulei Li, Tianfei Zhou, Wenguan Wang, Jianwu Li, and Yi Yang. Deep hierarchical semantic segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1246–1257, 2022. 5

[37] Yi Li, Zhanghui Kuang, Liyang Liu, Yimin Chen, and Wayne Zhang. Pseudo-mask matters in weakly-supervised semantic segmentation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 6964–6973, 2021. 6

[38] Di Lin, Jifeng Dai, Jiaya Jia, Kaiming He, and Jian Sun. Scribblesup: Scribble-supervised convolutional networks for semantic segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3159–3167, 2016. 1

[39] Tsung-Yi Lin, Priya Goyal, Ross Girshick, Kaiming He, and Piotr Dollár. Focal loss for dense object detection. In *Proceedings of the IEEE international conference on computer vision*, pages 2980–2988, 2017. 3, 7

[40] 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 *European conference on computer vision*, pages 740–755. Springer, 2014. 1, 5

[41] Jonathan Long, Evan Shelhamer, and Trevor Darrell. Fully convolutional networks for semantic segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3431–3440, 2015. 1, 5

[42] Shervin Minaee, Yuri Y Boykov, Fatih Porikli, Antonio J Plaza, Nasser Kehtarnavaz, and Demetri Terzopoulos. Image segmentation using deep learning: A survey. *IEEE transactions on pattern analysis and machine intelligence*, 2021. 1

[43] 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 *International Conference on Machine Learning*, pages 8748–8763. PMLR, 2021. 2

[44] Abhinav Shrivastava, Abhinav Gupta, and Ross Girshick. Training region-based object detectors with online hard example mining. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 761–769, 2016. 3

[45] Mennatullah Siam, Boris N Oreshkin, and Martin Jagersand. Amp: Adaptive masked proxies for few-shot segmentation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 5249–5258, 2019. 4

[46] Chunfeng Song, Yan Huang, Wanli Ouyang, and Liang Wang. Box-driven class-wise region masking and filling rate guided loss for weakly supervised semantic segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3136–3145, 2019. 1

[47] Yukun Su, Ruizhou Sun, Guosheng Lin, and Qingyao Wu. Context decoupling augmentation for weakly supervised semantic segmentation. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 7004–7014, 2021. 6

[48] Guolei Sun, Wenguan Wang, Jifeng Dai, and Luc Van Gool. Mining cross-image semantics for weakly supervised semantic segmentation. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part II 16*, pages 347–365. Springer, 2020. 2

[49] Kunyang Sun, Haoqing Shi, Zhengming Zhang, and Yongming Huang. Ecs-net: Improving weakly supervised semantic segmentation by using connections between class activation maps. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 7283–7292, 2021. 2

[50] Laurens Van der Maaten and Geoffrey Hinton. Visualizing data using t-sne. *Journal of machine learning research*, 9(11), 2008. 7

[51] Paul Viola and Michael Jones. Rapid object detection using a boosted cascade of simple features. In *Proceedings of the 2001 IEEE computer society conference on computer vision and pattern recognition. CVPR 2001*, volume 1, pages I–I. Ieee, 2001. 3

[52] Yude Wang, Jie Zhang, Meina Kan, Shiguang Shan, and Xilin Chen. Self-supervised equivariant attention mechanism for weakly supervised semantic segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 12275–12284, 2020. 5, 6

[53] Yunchao Wei, Jiashi Feng, Xiaodan Liang, Ming-Ming Cheng, Yao Zhao, and Shuicheng Yan. Object region mining with adversarial erasing: A simple classification to semantic segmentation approach. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1568–1576, 2017. 2

[54] Yunchao Wei, Xiaodan Liang, Yunpeng Chen, Xiaohui Shen, Ming-Ming Cheng, Jiashi Feng, Yao Zhao, and Shuicheng Yan. Stc: A simple to complex framework for weakly-supervised semantic segmentation. *IEEE transactions on pattern analysis and machine intelligence*, 39(11):2314–2320, 2016. 1

[55] 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 *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 16765–16774, 2021. 6

[56] Jinheng Xie, Xianxu Hou, Kai Ye, and Linlin Shen. Clims: Cross language image matching for weakly supervised semantic segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4483–4492, 2022. 2, 6, 8, 9

[57] Lian Xu, Wanli Ouyang, Mohammed Bennamoun, Farid Boussaid, and Dan Xu. Multi-class token transformer for weakly supervised semantic segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4310–4319, 2022. 5, 6

[58] Bingfeng Zhang, Jimin Xiao, Yunchao Wei, Mingjie Sun, and Kaizhu Huang. Reliability does matter: An end-to-end weakly supervised semantic segmentation approach. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pages 12765–12772, 2020. 2

[59] Dong Zhang, Hanwang Zhang, Jinhui Tang, Xian-Sheng Hua, and Qianru Sun. Causal intervention for weakly-supervised semantic segmentation. *Advances in Neural Information Processing Systems*, 33:655–666, 2020. 5, 6

[60] Fei Zhang, Chaochen Gu, Chenyue Zhang, and Yuchao Dai. Complementary patch for weakly supervised semantic segmentation. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 7242–7251, 2021. 2

[61] Tianyi Zhang, Guosheng Lin, Weide Liu, Jianfei Cai, and Alex Kot. Splitting vs. merging: Mining object regions with discrepancy and intersection loss for weakly supervised semantic segmentation. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XXII 16*, pages 663–679. Springer, 2020. 2

[62] Xiaolin Zhang, Yunchao Wei, Jiashi Feng, Yi Yang, and Thomas S Huang. Adversarial complementary learning for weakly supervised object localization. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1325–1334, 2018. 2

[63] Bolei Zhou, Aditya Khosla, Agata Lapedriza, Aude Oliva, and Antonio Torralba. Learning deep features for discriminative localization. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2921–2929, 2016. 1, 2, 5, 6

[64] Tianfei Zhou, Meijie Zhang, Fang Zhao, and Jianwu Li. Regional semantic contrast and aggregation for weakly supervised semantic segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4299–4309, 2022. 4