Threshold Matters in WSSS: Manipulating the Activation for the Robust and Accurate Segmentation Model Against Thresholds

Minhyun Lee,* Dongseob Kim*, Hyunjung Shim†
Yonsei University
{lmh315, kou.k, kateshim}@yonsei.ac.kr

Abstract

Weakly-supervised semantic segmentation (WSSS) has recently gained much attention for its promise to train segmentation models only with image-level labels. Existing WSSS methods commonly argue that the sparse coverage of CAM incurs the performance bottleneck of WSSS. This paper provides analytical and empirical evidence that the actual bottleneck may not be sparse coverage but a global thresholding scheme applied after CAM. Then, we show that this issue can be mitigated by satisfying two conditions; 1) reducing the imbalance in the foreground activation and 2) increasing the gap between the foreground and the background activation. Based on these findings, we propose a novel activation manipulation network with a per-pixel classification loss and a label conditioning module. Per-pixel classification naturally induces two-level activation in activation maps, which can penalize the most discriminative parts, promote the less discriminative parts, and deactivate the background regions. Label conditioning imposes that the output label of pseudo-masks should be any of true image-level labels; it penalizes the wrong activation assigned to non-target classes. Based on extensive analysis and evaluations, we demonstrate that each component helps produce accurate pseudo-masks, achieving the robustness against the choice of the global threshold. Finally, our model achieves state-of-the-art records on both PASCAL VOC 2012 and MS COCO 2014 datasets. The code is available at https://github.com/gaviotas/AMN.

1. Introduction

Weakly-supervised semantic segmentation (WSSS) requires only weak supervision (e.g., image-level labels [35, 36], scribbles [31], bounding boxes [19]) as opposed to the fully supervised model, which involves costly pixel-level annotations. In this work, we address WSSS using image-
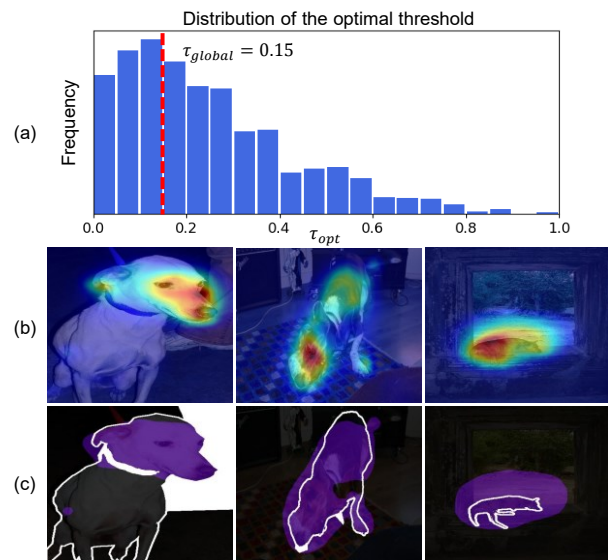

Figure 1. Motivating examples show that the optimal threshold per image (τ_{opt}) from the same *dog* class is quite different from each other. (a) The distribution of the optimal threshold on PASCAL VOC 2012 train set, (b) the activation maps, (c) the thresholded masks using a global threshold $\tau_{global} = 0.15$.

level labels because of its low labeling cost. The overall pipeline of WSSS consists of two stages. The pseudo-mask is first generated from an image classifier, and then it is used as supervision to train a segmentation network.

The prevalent technique for generating pseudo-masks is class activation mapping (CAM) [45]. It uses the intermediate classifier's activation to compute the class activation map corresponding to its image-level label. The common practice of WSSS is to apply a global threshold to the activation map (i.e., assigning the object class if the activation is greater than the threshold) for obtaining the pseudo-mask. Existing methods point out that the pseudo-mask obtained from CAM only captures the most discriminative parts of the object, incurring the performance bottleneck. Therefore, most existing studies have expanded object coverage by manipulating the image [29, 39] or feature map [18, 26].

However, we argue that the performance bottleneck of

---

*indicates an equal contribution.
†Hyunjung Shim is a corresponding author.

WSSS comes from a global threshold applied after CAM; the sparse object coverage does not explain all. This threshold partitions each activation map into the foreground (object class) and background. Then, the pseudo-mask is generated by combining all foreground regions. Here, the choice of threshold critically affects the performance of WSSS. i) We further observe that a global threshold cannot provide an optimal threshold per image. Figure 1(a) visualizes the distribution of optimal threshold on the PASCAL VOC 2012 train set (For analysis, we obtain the best threshold per image using its ground-truth segmentation map). It shows that the optimal threshold per image quite differs from each other, and the global threshold is often far from the optimal one. ii) Besides, a global threshold for CAM does not always lead to sparse coverage. Figure 1(b) and (c) show several CAM examples and the corresponding masks generated by a global threshold, respectively; the third row shows that CAM and the thresholded mask overly capture the target object. These results clearly motivate us to rethink that the performance bottleneck of WSSS is closely related to the usage of a global threshold.

To tackle this problem, we first investigate why this problem happens. By tracing the procedure of CAM, we realize that global average pooling (GAP) applied to the last layer invokes this issue; the global threshold largely differs from the optimal threshold per image. The first stage of the WSSS framework trains the image classifier, whose score is computed via GAP. While GAP facilitates deriving the activation map, it averages the feature maps into a single classification score. The same value can be from totally different activations. For example, the same score can be from 1) high activations only in the most discriminative region (low optimal threshold to cover more regions), 2) moderate activations distributed over the entire object, or 3) small activations covering even outside the object (high optimal threshold to cover small regions).

Due to its averaging nature, GAP hinders achieving the accurate pseudo-mask via a global threshold. As a naïve solution, one might consider introducing a different threshold per image. However, this is prohibitive because finding a per-image threshold requires pixel-level annotation, violating the principle of weakly-supervised learning. Instead of controlling a threshold per image, our key idea is to manipulate the activation in a way that the resultant pseudo-mask is of high quality regardless of threshold values. To achieve robust performance, we can increase the activation gap between the foreground region and the background region; the thresholded masks are the same if the threshold value is within the gap. However, it can induce the model to capture the most discriminative parts only, resulting in consistent but poor quality.

To achieve high quality consistently, it is important to reduce the activation imbalance within the foreground and

keep the large activation gap between the foreground and background simultaneously. We can achieve the two factors by assigning the two-level activation for the entire foreground pixels and background pixels (e.g., 1 and 0). In this way, the high activation in the most discriminative parts is penalized, but the low activation in the less discriminative parts is promoted. Meanwhile, the background activation can be deactivated. Naturally, this strategy can guarantee a large gap, enabling us to achieve the robust performance even with a global threshold.

Specifically, we introduce a robust and accurate *activation manipulation network (AMN)*, which takes an image with its image-level label as the input and provides the high-quality pseudo-mask as the output. For that, we formulate a training objective using i) per-pixel classification with an effective constraint using ii) label conditioning. Since per-pixel classification does not rely on GAP, it bypasses the issue of having totally different activation maps for the same classification score. More importantly, it directly enforces the same large activation for the foreground (e.g., 1) and the same small activation for the background (e.g., 0). Here, it leads to reducing the activation imbalance inside the foreground and having a large gap between the foreground and background. Since we do not have pixel-level supervision to formulate per-pixel classification problems, the noisy pseudo-mask from CAM with conditional random field (CRF) [23] serves as the initial target for training AMN.

Moreover, we propose label conditioning to reduce the activation of non-target classes. The idea of label conditioning is to reformulate the label prediction problem by finding the best prediction out of the given K classes (K is the number of classes given by the ground-truth image-level label per image) and background, instead of $N + 1$ classes (i.e., a total of N foreground classes and a background class). K is always much less than N and thereby the range of possible answers is clearly reduced. It makes the problem better-posed. More importantly, the activation of non-target classes is strictly suppressed by mapping to 0. It helps strengthen the foreground activation. As a result, with the same global threshold as the previous studies [1, 25, 27], AMN largely improves the quality of pseudo-mask and eventually records state-of-the-art performances on the Pascal VOC 2012 and MS COCO 2014 benchmarks.

2. Related work

Most WSSS techniques utilize CAM [45] to obtain localization maps from image-level labels. Considering the sparse coverage of CAM as the bottleneck of WSSS, many studies focused on expanding the seed activation of CAM. Specifically, [8, 16, 29] suggested erasing the most discriminative regions. CIAN [13] utilized the cross-image affin-

ity. [5,38] devised a self-supervised task. [18,26] suggested a feature ensemble method. [30,37] developed a class-wise co-attention mechanism. AdvCAM [27] proposed an anti-adversarial image manipulation method. [1, 2] implicitly exploit the boundary information with pixel-level affinity information, naturally expanding the object coverage until boundaries.

Another approach exploits additional information to refine the object boundaries or distinguish co-occurring objects [12,40,42]. [6,41] combined saliency maps with class-specific attention cues to generate reliable pseudo-masks. EPS [28] utilized the saliency maps as the cues for boundaries and co-occurring pixels. DRS [20] suppressed the most discriminative parts to expand the object coverage and then refine the boundary with saliency map.

Several existing studies resolved the limitation of CAM-GAP by modifying the pooling methods [3, 22, 25]. SEC [22] argued that global max pooling (GMP) underestimates the object size and GAP sometimes overestimates it. Then, they proposed global weighted rank pooling. Araslanov et al. [3] claimed that CAM-GAP may penalize small segments and proposed normalized global weighted pooling (nGWP) instead of GAP. As a concurrent work, RIB [25] promotes less discriminative regions by collecting only non-discriminative regions for pooling. Unlike the methods suggested new pooling layers, we focus on the fact that GAP (in fact, any pooling methods will do) leads to having a different optimal threshold per image.

3. Preliminaries

Class activation mapping (CAM). In the WSSS framework, CAM is used to provide class activation maps corresponding to their image-level labels. Given a CNN f and the input $x \in \mathbb{R}^{H \times W \times 3}$, H and W indicate the height and width of the input. The feature maps are average pooled and then multiplied by the weights $\mathbf{w}_i^c$ for class c from the classifier, resulting in the classification score. By multiplying $\mathbf{w}_i^c$ back to the feature maps $f(x) \in \mathbb{R}^{H_{out} \times W_{out} \times Q}$, we can compute the class activation map $\mathrm{F}_c(x) \in \mathbb{R}^{H_{out} \times W_{out}}$ for class c as follows:

$$\mathrm{F}_c(x) = \sum_{i=1}^{Q} \mathbf{w}_i^{c\top} \cdot f_i(x), \tag{1}$$

where Q is the number of channels in feature maps.

All existing WSSS methods normalize F_c into the range of [0 1] and then apply a global threshold to separate the foreground and background pixels. In this way, we can generate pixel-level masks from image-level labels.

Threshold matters in WSSS. GAP allows different activations to be mapped to the same classification score. Thus, the resultant F_c can have various distributions of activations. As a result, as seen in Figure 1, no single threshold can be sufficient to derive the optimal pseudo-masks for
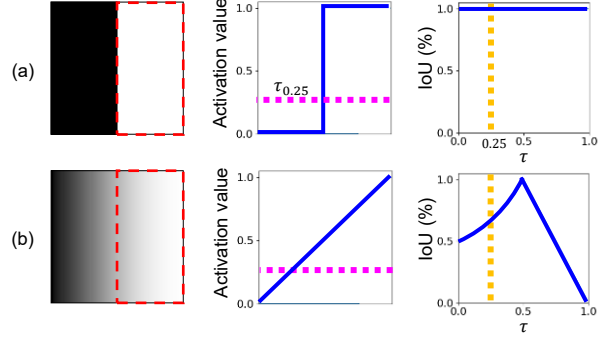

Figure 2. Two different activation maps and their IoU curves under different thresholds. The black with 0 activation for the background, the white with 1 activation for the foreground, and a red box for the ground-truth object region. A magenta and yellow indicate a threshold line at $\tau = 0.25$ and corresponding IoU of each activation map, respectively. The plots in the second column are the one-dimensional horizontal cross-section of the activation map.

different inputs. We investigate the conditions where the pseudo-mask is accurate and robust against different thresholds. The first condition (c1) is reducing the activation imbalance within the foreground as also pointed out in [24]. It guides the activation value cover the entire extent of the target object rather than focusing on the most discriminative part. The second condition (c2) is enforcing the large activation gap between the foreground and the background activation. It helps the pseudo-mask generation less sensitive against the threshold. By jointly satisfying c1 and c2, we argue that the activation is formed to distinguish the foreground and the background reasonably well with a global threshold. A simple toy example in Figure 2(a) illustrates that satisfying the two conditions can guarantee consistent and accurate performance regardless of threshold τ; the same pseudo-mask is generated by choosing any τ within the gap. Figure 2(b) shows the opposite scenario, where it satisfies neither c1 nor c2; the performance is extremely sensitive to the choice of τ. The two cases illustrate that satisfying both c1 and c2 allows us to obtain accurate and robust pseudo-masks.

4. Activation Manipulation Network

Our goal is to improve pseudo-mask quality by manipulating the activation map at the pixel-level, leading to robust performance against threshold choice. To this end, we propose an *activation manipulation network* (AMN) with two learning objectives. Firstly, we introduce a *per-pixel classification loss*, which reduces the activation imbalance inside foreground and provides the large gap between the foreground and the background (i.e., 1 when it is normalized into [0 1]). In addition, we propose a *label conditioning* module, which eliminates the activation from the non-target classes. It helps produce the foreground and background ac-
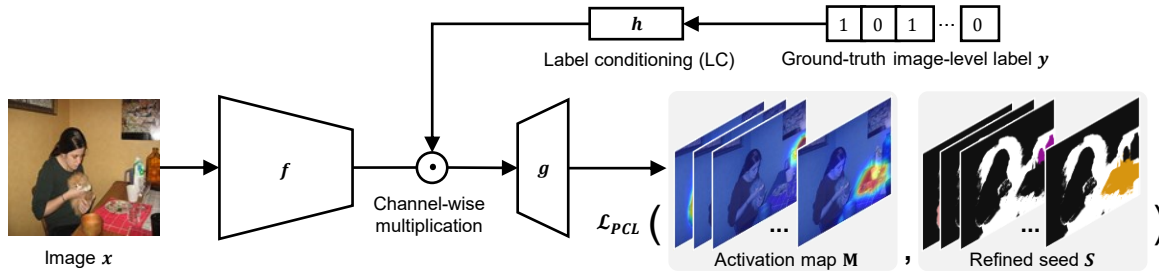
Figure 3. The overall framework of activation manipulation network (AMN). The refined seed from a classification network is used as noisy supervision to train AMN. The per-pixel classification loss (PCL) and label conditioning (LC) improve the pseudo-mask quality.

tivation more accurately by reallocating the activation.

4.1. Overall training procedure

The training of the proposed WSSS framework consists of three stages: 1) seed generation, 2) pseudo-mask generation with the proposed AMN, and 3) final segmentation. For seed generation, we obtain noisy pixel-level annotations from image-level labels by applying CAM. Then, we apply conditional random field (CRF) [23]. CRF is the prevalent post-processing method and refines the initial seed by assigning an undefined region for less confident pixels. Specifically, we follow the procedure of Ahn et al. [1] to generate the refined seed S_i. Secondly, given the image x_i and its image-level label y_i as inputs and the refined seed S_i as the target output, we train AMN via a per-pixel classification loss (PCL) with label conditioning (LC). The network architecture for AMN is identical to that of the classification network for CAM with a small modification; replacing the GAP layer and the last classification layer with convolutional layers to predict pixel-level mask. Specifically, we adopt the atrous spatial pyramid pooling (ASPP) scheme [7]. To generate the final pseudo-masks, we improve the predicted mask quality using the well-known refinement technique, IRN [1]. Finally, we train a segmentation network with the generated pseudo-masks as supervision. Figure 3 visualizes the overall framework of AMN.

4.2. Per-pixel classification

Based on a case study in Section 3, we concluded that jointly achieving the two conditions can resolve the issue caused by the global threshold: reducing the imbalance within the foreground activation and having a large gap between the foreground and the background activations. Then, we devise an activation manipulation network (AMN) that satisfies the above two conditions. To achieve this goal, we introduce the per-pixel classification loss (PCL) because it directly enforces the two-level activation (e.g., 0 or 1), manipulating the activation (before thresholding) at the pixel-level.

Specifically, the two-level activation as the target signal can reduce the activation imbalance inside the foreground

because the foreground should be assigned to the same activation value. Likewise, the two-level activation naturally retains the large activation gap between the foreground and background. Another advantage of PCL is that it does not rely on GAP. Since GAP yields different activation maps having the same classification score, discarding the GAP can be effective in handling a global threshold problem. To train the model with per-pixel classification, we need pixel-level supervision. Under the WSSS scenario, direct access to the pixel-level supervision is prohibited. We instead utilize the refined seed S_i as noisy supervisory for training AMN. Finally, the balanced cross-entropy loss [17] is adopted for a per-pixel classification loss (PCL).

4.3. Label conditioning

The original per-pixel classifier maps each pixel into one out of $N+1$ classes (i.e., a total of N foreground classes and a background class). Meanwhile, label conditioning (LC) imposes that each pixel should be mapped into one out of $K+1$ classes, meaning K-number of classes in the ground-truth image-level label per image and 1-background class. LC is effective in two aspects. Firstly, it helps distinguish objects with similar appearances unless they really appear together in the image. It prevents false predictions due to confusing textures (e.g., among the skin of horse, cow, or dog) by allowing activation only if its class is corresponding to any of the input ground-truth image-level label. Next, noisy pseudo-masks S_i often include a considerable number of undefined regions. The model is thus data-hungry due to lack of supervisory signals. LC can act as auxiliary supervision, providing rich learning signals. As a result, adopting LC leads to reallocating the non-target class activation to the target class activation, increasing the overall activation of the foreground. This is particularly useful to promote the less discriminative regions of the foreground.

Here, we introduce an additional layer for LC such that the effects of LC only influence high-level features. This is because limiting the choice of classes at low-level features may add to unwanted bias to the representation. Instead, we encode the ground-truth image-level labels as a feature vector and then directly multiply this vector to the feature

CAM	PCL	LC	w/ CRF	w/ IRN [1]
✓			54.3	66.3
✓	✓		62.1	69.1
✓	✓	✓	**65.3**	**72.2**

Table 1. Ablation study of the proposed modules. The accuracy (mIoU) of pseudo-masks on PASCAL VOC 2012 train set is reported. The best score is in **bold** throughout all experiments.

map $f(x)$. Finally, the activation map $\mathbf{M}$ is computed as:

$$\mathbf{M} = \sigma(g(f(x) \cdot h(\mathbf{y}_{gt})), \tag{2}$$

where f, g, and h indicate a CNN backbone, convolutional layers to predict pixel-level masks, and a linear layer to map the label to the feature vector, respectively. By doing so, the feature vector of ground-truth image-level labels directly constrains the final map.

5. Experiments

5.1. Experimental setup

Dataset & evaluation metric. For performance evaluation, we use both PASCAL VOC 2012 [10] (CC-BY 4.0) and MS COCO 2014 [4] (CC-BY 4.0) datasets, the most popular benchmarks in the semantic segmentation task. PASCAL VOC 2012 contains 20 foreground object categories and one background category with 10,582 training images expanded by SBD [14], 1,449 validation images, and 1,456 test images. MS COCO 2014 dataset consists of 81 classes, including a background, with 82,783 and 40,504 images for training and validation. In all experiments, we only used image-level class labels for training. For an evaluation metric, we used mean Intersection over Union (mIoU), which is widely used to measure segmentation performance.

Implementation details. We train the classification network to extract the seed activation map via CAM. Here, we adopt ResNet50 [15] pre-trained on ImageNet [9] as a backbone classification network, except for the additional layers. The CAM implementation follows the configuration from Ahn et al. [1]. For training AMN, we used an Adam [21] optimizer and the learning rate of 5e-6 for updating the backbone parameters and 1e-4 for updating parameters associated with a per-pixel classification head. We adopt label smoothing as a training technique to subside the noise in initial seed, as discussed in [34]. The additional hyper-parameters are found in supplementary material. For the segmentation network, we experimented with DeepLab-v2 with the ResNet101 backbone [7] and followed the default training settings of AdvCAM [27].

5.2. Ablation study

We investigate whether each component of AMN is effective. Considering the CAM with IRN as the baseline, we

Figure 4. The effects of label conditioning on the *cow* image. The wrong activations of the *horse* are reallocated into the correct class when LC is applied. (a) and (b) are activation maps corresponding to *cow* and *horse*, respectively.

add a per-pixel classification loss (PCL) and a label conditioning (LC) in sequence and report their performances in Table 1. These results are from PASCAL VOC 2012 train set, thereby implying the quality of pseudo-masks. Compared to the baseline, adding PCL improves the mIoU by 2.8%. By additionally applying LC, the performance increases by 3.1%. Considering that the performance is already high, the additional gain by LC is impressive. Since LC suppresses any activation for non-target classes, it implicitly increases the activation of the target objects.

To confirm the effects of LC, we visualize the activation map with and without LC in Figure 4. The two classes having a similar appearance, such as *cow* and *horse*, can be hardly distinguishable using AMN without LC. When the image only has *cow*, Figure 4 shows that the results without LC are activated for both the *cow* and *horse* (the first image of (a) and (b)). Meanwhile, after adopting LC, it is clearly seen that only *cow* pixels are activated, but *horse* pixels are deactivated, as shown in the second images of Figures 4(a) and (b). From these results, we support that LC not only reduces non-target activations but also increases the foreground activations of the target objects. It can be interpreted that using the ground-truth image-level labels can subside the noise in our initial target (CAM with CRF), greatly increasing the performance.

We stress that LC is not applicable to the conventional image classifier because its target is already the image-level labels; LC on the image classifier can yield the model to return a trivial solution. Since AMN learns to manipulate pixel-level activation, adopting LC behaves as auxiliary supervision and leads to performance improvement.

In addition, we experimented to determine which layer of AMN is best for LC to affect. We use ResNet50 as the backbone, consisting of 4 resblocks. We applied LC after each residual block sequentially, and computed the feature vectors. The accuracy of pseudo-masks by applying LC differently is summarized in Table 2. The best performance was achieved when LC was applied to high-level features. Likewise, the performance deteriorated when applied to low-level features. Also, applying LC on multiple layers, including the last layer, did not help the performance either. These results show that the idea of LC should be carefully
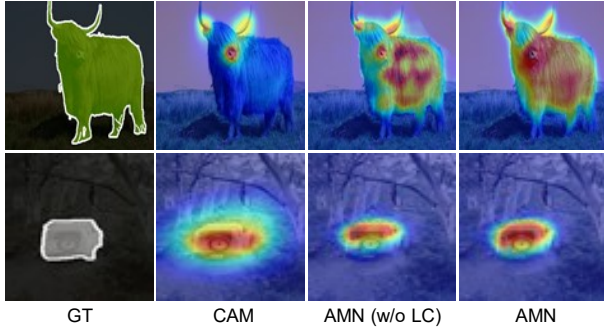
Figure 5. The effects of each component of AMN on PASCAL VOC 2012 train set. All activation maps are normalized.

GT   CAM   AMN (w/o LC)   AMN

layer1	layer2	layer3	layer4	mIoU
✓				48.2
	✓			51.7
		✓		61.2
			✓	**62.1**
		✓	✓	61.0
	✓	✓	✓	51.4
	✓	✓		50.9

Table 2. Accuracy (mIoU) of pseudo-masks from AMN without the boundary refinement on PASCAL VOC 2012 train set. The accuracy varies depending on where to apply LC.

implemented, because class-specific information is not always useful for feature engineering. We conjecture that the last layer handles the semantics, thus it can effectively utilize the LC for improving the final decision.

5.3. Sensitivity to threshold

Quantitative evaluation. In this section, we evaluate the robustness of different methods under various thresholds. We apply different thresholds to the activation map, obtain the pseudo-mask accordingly, and then measure its accuracy (mIoU). For comparison, the baseline CAM [45], RIB [25], DRS [20] without saliency map, AMN without LC, and AMN are selected. Figure 6 presents mIoU.

In the case of CAM and RIB, the accuracy decreases even by adding a small perturbation to their global threshold; the curves fluctuate rapidly upon τ. This is expected because GAP yields an imbalance in activation, thus the pseudo-mask is highly sensitive to τ. On the other hand, DRS and AMN without LC exhibit relatively gentle slopes, indicating robust performances to the changes in threshold. Note that DRS tends to decrease the high activation on the most discriminative parts, thereby it partially shares the philosophy of PCL. However, DRS only focuses on suppressing foreground activations while PCL i) promotes the less discriminative parts of the foreground and ii) reduces the background activation. For this reason, activation of PCL tends to distinguish foreground and background more accurately, whereas DRS always tends to have excessive fore-
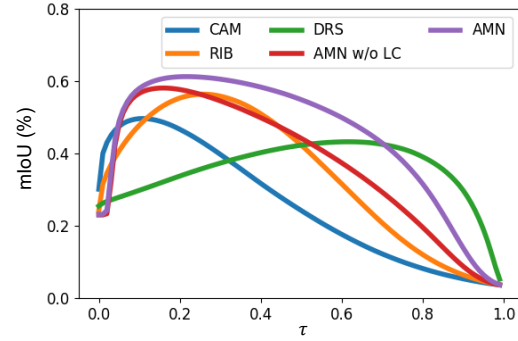

Figure 6. Accuracy(mIoU) of pseudo-masks depending on thresholds. The results are before boundary refinement thus differ from the final mIoU score. AMN shows more accurate and robust performance than others.

Method	mIoU
IRN [1]$_{CVPR'16}$	66.3
SEAM [38]$_{CVPR'20}$	63.6
MBMNet [32]$_{ACMMM'20}$	66.8
CONTA [44]$_{NeurIPS'20}$	67.9
AdvCAM [27]$_{CVPR'21}$	69.9
RIB [25]$_{NeurIPS'21}$	70.6
AMN (ours)	**72.2**

Table 3. Accuracy (mIoU) of pseudo-masks evaluated on PASCAL VOC 2012 train set.

ground coverage. These side effects of DRS finally result in relatively low accuracy in general. The original DRS utilizes a saliency map as additional supervision to compensate for this issue, thus the disadvantage was well mitigated. AMN without LC alleviated the imbalance in activation for both the foreground and background. Consequently, we achieve high mIoU and robust performance against the threshold at the same time.

Final AMN uses LC, and it increases the prior probability of a class belonging to the input image-level label. It helps reduce wrong activation for wrong classes, reallocating them to the correct class. This effectively promotes the less discriminative part of the foreground. As a result, we can achieve more accurate and robust performance.

Qualitative evaluation. Figure 5 shows the effect of each component of AMN qualitatively. Since PCL imposes each pixel to map either the foreground or background, it penalizes the high activation in the most discriminative parts as well as the noisy activation in the background. Meanwhile, PCL can increase the moderate activation in the less discriminative part. As a natural consequence, we observe that the map generated by PCL alone (AMN without LC) covers the full extent of the object more than the original CAM. Concretely, the result from CAM concentrated on the most discriminative regions, such as a *cow*'s face. Meanwhile, the resultant map by AMN without LC can capture the en-

Method	Backbone	Sup.	val	test
FickleNet [26]CVPR'19	ResNet101	I.+S.	64.9	65.3
OAA [18]ICCV'19	ResNet101	I.+S.	65.2	66.4
Multi-Est. [12]ECCV'19	ResNet101	I.+S.	67.2	66.7
MCIS [37]ECCV'20	ResNet38	I.+S.	66.2	66.9
SGAN [41]ACCESS'20	ResNet101	I.+S.	67.1	67.2
EPS [28]CVPR'21	ResNet101	I.+S.	70.9	70.8
DRS [20]AAAI'21	ResNet101	I.+S.	71.2	71.4
ICD [11]CVPR'20	ResNet101	I.	64.1	64.3
SEAM [38]CVPR'20	ResNet38	I.	64.5	65.7
SC-CAM [5]CVPR'20	ResNet101	I.	66.1	65.9
RRM [43]AAAI'20	ResNet101	I.	66.3	66.5
BES [33]ECCV'20	ResNet101	I.	65.7	66.6
CONTA [44]NeurIPS'20	ResNet50	I.	66.1	66.7
AdvCAM [27]CVPR'21	ResNet101	I.	68.1	68.0
RIB [25]NeurIPS'21	ResNet101	I.	68.3	68.6
AMN (ours)	ResNet101	I.	69.5†	69.6†
	ResNet101	I.	**70.7**‡	**70.6**‡

Table 4. Segmentation results (mIoU) on PASCAL VOC 2012. I. and S. denotes image-level labels and the external saliency maps used for supervision, respectively. The best score is underlined for I.+S. and in **bold** for I. † for the ImageNet pre-trained model and ‡ for the MS COCO pre-trained model throughout all experiments.

Method	Backbone	Sup.	val
SGAN [41]ACESS'20	VGG16	I.+S.	33.6
EPS [28]CVPR'21	VGG16	I.+S.	35.7
ADL [8]TPAMI'20	VGG16	I.	30.8
CONTA [44]NeurIPS'20	ResNet50	I.	33.4
IRN [1]CVPR'19	ResNet101	I.	41.4
RIB [25]NeurIPS'21	ResNet101	I.	43.8
AMN (ours)	ResNet101	I.	**44.7**†

Table 5. Accuracy (mIoU) of semantic segmentation evaluated on MS COCO 2014 val set.

tire extent of the object. Depending on the object size, the CAM occasionally covers the object excessively, as seen in Figure 5 (the bottom image for CAM). Since PCL regularizes both the foreground and background activation, it achieves reasonable coverage of object extent.

LC provides additional supervision to undefined areas (originally less confident) and promotes the less discriminative parts of the foreground. Figure 5 shows that activation of AMN is evenly spread inside the foreground and reasonably covers the object, such as *cow* and *car* region. Besides, it shows a large activation gap between the object region and the background by promoting the less discriminative regions of the foreground. As we intended, LC helps reduce activation imbalance and increases the gap between the foreground and the background activation.

5.4. Comparisons with the state-of-the-arts

Accuracy of pseudo-masks. Similar to existing WSSS methods, we aim to improve the pseudo-mask quality and expect that it will eventually increase the accuracy of WSSS. We first evaluate the quality of pseudo-masks by comparing them with ground-truth masks. Table 3 compares mIoU of the proposed AMN with that of other state-of-the-art WSSS methods. For a fair comparison, we apply the best refinement scheme reported by each method for pseudo-mask generation. Our results achieve a gain of 5.9% over that of IRN [1], which can be regarded as a baseline, and the gain of 1.6% over RIB, the state-of-the-art method among the WSSS methods only with image-level labels.

Specifically, the accuracy (mIoU) in *dining table / tv* was 41.9 / 54.2 with RIB, but 62.8 / 63.1 with ours on PASCAL VOC 2012 train set. *dining table* usually exhibits extremely strong activation on the most discriminative parts (i.e., an extreme imbalance in activation), thus the optimal threshold for this class is much smaller than the global threshold. Although existing WSSS methods aim at expanding the object coverage, their effects are designed at image-level, thus cannot suppress strong activation at pixel-level. Meanwhile, AMN explicitly regularized the pixel-level activation, therefore capable of handling extreme activation.

On the other hand, the optimal threshold for *tv* significantly varies depending on the image. That means, no single threshold is meaningful. Thanks to the robust nature of AMN, we could achieve considerable gain on *tv* regardless of images. These results are consistent with our motivation; threshold matters in WSSS and AMN can effectively resolve this issue. A quantitative evaluation of the pseudo-mask for each class is provided in supplementary material.

Accuracy of segmentation maps. For quantitative comparison, we report the mIoU scores of our method and recent WSSS methods on PASCAL VOC 2012 validation and test images. The competitors are chosen to represent the best-performing models in the last three years. On PASCAL VOC 2012 benchmark, we achieved 69.5% and 69.6% mIoU using the ImageNet pretrained backbone, and 70.7% and 70.6% mIoU with MS COCO pretrained backbone. This is a new state-of-the-art record for WSSS methods only using the image-level labels; comparable to EPS [28] using both image-level labels and saliency maps.

Analogous to the observation in the pseudo-mask, our achievement is particularly affected by a large improvement in several classes that have performed poorly in the past. Specifically, the segmentation result (mIoU) in *dining table / tv* was 37.5 / 54.9 with RIB, but 53.8 / 57.5 with ours on PASCAL VOC 2012 validation set. These results are consistent with the pseudo-mask accuracy. Our method handles the strong imbalance in activation at the pixel-level (*dining table*) and is robust against the threshold choice (*tv*). More results on per-class mIoU scores are provided in supplementary material. Figure 7 shows the qualitative examples of
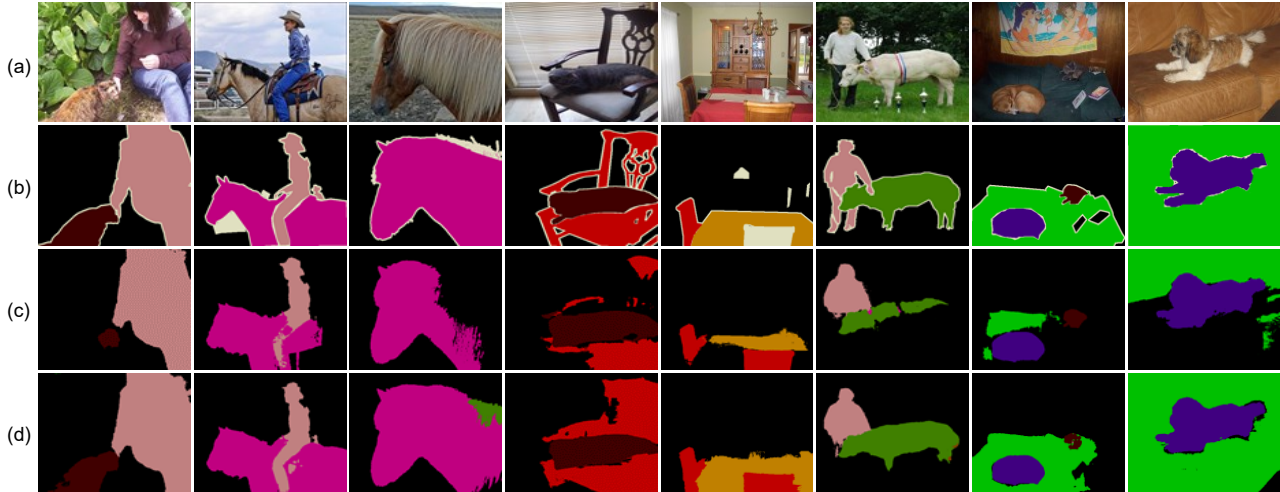
Figure 7. Qualitative examples of segmentation results on PASCAL VOC 2012 val set. (a) Input, (b) Ground-truth, (c) IRN, and (d) AMN.
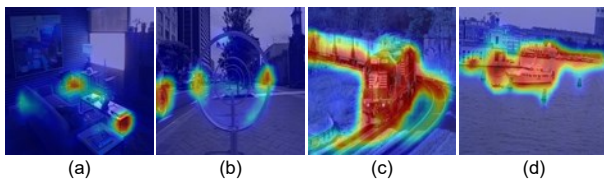

Figure 8. Failure cases of AMN. The results are normalized activation maps of (a) *chair*, (b) *bicycle*, (c) *train*, and (d) *boat*.

segmentation results on PASCAL VOC 2012 validation set. These results confirm that our method covers the full extent of the objects correctly, especially *dining table*, which was not possible by previous methods.

To investigate our performance on the large-scale benchmark, we adopt the MS COCO 2014 dataset. Not all competitors provide their evaluation on MS COCO 2014. For this reason, we only compare our method with five competitors. Table 5 summarizes the comparison results on MS COCO 2014 validation set. AMN achieves 44.7% mIoU, breaking a new state-of-the-art record. This demonstrates that AMN is also effective on large-scale benchmarks. More qualitative comparisons and results on per-class segmentation mIoU scores for MS COCO 2014 are in supplementary material.

5.5. Discussion

Limitation. Although LC helps disambiguate the confusing foreground classes (e.g., visually similar to each other), it cannot handle the case where they appear together in the input image or the background is similar to the foreground object. For example, the table in Figure 8(a) is misclassified as chair pixels upon similar appearance. Similarly, the metal ring in Figure 8(b) is mispredicted as bicycle pixels due to its shared shape. In addition, our method cannot overcome the contextual bias (i.e., co-occurrence) and inaccurate boundary problem, which is inherited by CAM.

Since the classifier is not designed to separate the foreground and background, activation maps from the classifier do not capture precise object boundaries, especially for complex shapes (e.g., the rough boundary of bicycles and chairs). Figures 8(c) and (d) show that our method cannot distinguish the co-occurring pixels in a railroad-train pair and a boat-water pair, respectively.

Negative societal impact. Since our framework consists of three training stages, it incurs more carbon emissions and power consumption. In future work, we plan to reduce the training stages, integrating the classifier and AMN.

6. Conclusions

In this paper, we identified that the optimal thresholds largely vary in the images, and this issue can significantly affect the performance of WSSS. To address this issue, we devised a new activation manipulation strategy for achieving robust and accurate performances. Toward this goal, we showed that jointly satisfying the two conditions can sufficiently resolve this problem. That is, we should reduce the imbalance in activation and increase the gap between the foreground and the background activation at the same time. For that, we developed an activation manipulation network (AMN) with a per-pixel classification loss and an image-level label conditioning module. Extensive experiments show that each component of AMN is effective, AMN helps induce robust pseudo-masks against the threshold, and finally achieved a new state-of-the-art performance in both PASCAL VOC 2012 and MS COCO 2014 datasets.

References

[1] Jiwoon Ahn, Sunghyun Cho, and Suha Kwak. Weakly supervised learning of instance segmentation with inter-pixel relations. In *CVPR*, 2019. 2, 3, 4, 5, 6, 7

[2] Jiwoon Ahn and Suha Kwak. Learning pixel-level semantic affinity with image-level supervision for weakly supervised semantic segmentation. In *CVPR*, 2018. 3

[3] Nikita Araslanov and Stefan Roth. Single-stage semantic segmentation from image labels. In *CVPR*, 2020. 3

[4] Holger Caesar, Jasper Uijlings, and Vittorio Ferrari. Cocostuff: Thing and stuff classes in context. In *CVPR*, 2018. 5

[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 *CVPR*, 2020. 3, 7

[6] Arslan Chaudhry, Puneet K. Dokania, and Philip H. S. Torr. Discovering class-specific pixels for weakly-supervised semantic segmentation. In *BMVC*, 2017. 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. *TPAMI*, 2017. 4, 5

[8] Junsuk Choe, Seungho Lee, and Hyunjung Shim. Attention-based dropout layer for weakly supervised single object localization and semantic segmentation. *TPAMI*, 2020. 2, 7

[9] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *CVPR*, 2009. 5

[10] Mark Everingham, SM Ali Eslami, Luc Van Gool, Christopher KI Williams, John Winn, and Andrew Zisserman. The pascal visual object classes challenge: A retrospective. *IJCV*, 2015. 5

[11] Junsong Fan, Zhaoxiang Zhang, Chunfeng Song, and Tieniu Tan. Learning integral objects with intra-class discriminator for weakly-supervised semantic segmentation. In *CVPR*, 2020. 7

[12] Junsong Fan, Zhaoxiang Zhang, and Tieniu Tan. Employing multi-estimations for weakly-supervised semantic segmentation. In *ECCV*, 2020. 3, 7

[13] Junsong Fan, Zhaoxiang Zhang, Tieniu Tan, Chunfeng Song, and Jun Xiao. Cian: Cross-image affinity net for weakly supervised semantic segmentation. In *AAAI*, 2020. 2

[14] Bharath Hariharan, Pablo Arbeláez, Lubomir Bourdev, Subhransu Maji, and Jitendra Malik. Semantic contours from inverse detectors. In *ICCV*, 2011. 5

[15] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *CVPR*, 2016. 5

[16] Qibin Hou, PengTao Jiang, Yunchao Wei, and Ming-Ming Cheng. Self-erasing network for integral object attention. In *NeurIPS*, 2018. 2

[17] Zilong Huang, Xinggang Wang, Jiasi Wang, Wenyu Liu, and Jingdong Wang. Weakly-supervised semantic segmentation network with deep seeded region growing. In *CVPR*, 2018. 4

[18] Peng-Tao Jiang, Qibin Hou, Yang Cao, Ming-Ming Cheng, Yunchao Wei, and Hong-Kai Xiong. Integral object mining via online attention accumulation. In *ICCV*, 2019. 1, 3, 7

[19] Anna Khoreva, Rodrigo Benenson, Jan Hosang, Matthias Hein, and Bernt Schiele. Simple does it: Weakly supervised instance and semantic segmentation. In *CVPR*, 2017. 1

[20] Beomyoung Kim, Sangeun Han, and Junmo Kim. Discriminative region suppression for weakly-supervised semantic segmentation. In *AAAI*, 2021. 3, 6, 7

[21] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014. 5

[22] Alexander Kolesnikov and Christoph H Lampert. Seed, expand and constrain: Three principles for weakly-supervised image segmentation. In *ECCV*, 2016. 3

[23] Philipp Krähenbühl and Vladlen Koltun. Efficient inference in fully connected crfs with gaussian edge potentials. In *NeurIPS*, 2011. 2, 4

[24] 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 *ICCV*, 2021. 3

[25] Jungbeom Lee, Jooyoung Choi, Jisoo Mok, and Sungroh Yoon. Reducing information bottleneck for weakly supervised semantic segmentation. In *NeurIPS*, 2021. 2, 3, 6, 7

[26] Jungbeom Lee, Eunji Kim, Sungmin Lee, Jangho Lee, and Sungroh Yoon. Ficklenet: Weakly and semi-supervised semantic image segmentation using stochastic inference. In *CVPR*, 2019. 1, 3, 7

[27] Jungbeom Lee, Eunji Kim, and Sungroh Yoon. Anti-adversarially manipulated attributions for weakly and semi-supervised semantic segmentation. In *CVPR*, 2021. 2, 3, 5, 6, 7

[28] Seungho Lee, Minhyun Lee, Jongwuk Lee, and Hyunjung Shim. Railroad is not a train: Saliency as pseudo-pixel supervision for weakly supervised semantic segmentation. In *CVPR*, 2021. 3, 7

[29] Kunpeng Li, Ziyan Wu, Kuan-Chuan Peng, Jan Ernst, and Yun Fu. Tell me where to look: Guided attention inference network. In *CVPR*, 2018. 1, 2

[30] Xueyi Li, Tianfei Zhou, Jianwu Li, Yi Zhou, and Zhaoxiang Zhang. Group-wise semantic mining for weakly supervised semantic segmentation. In *AAAI*, 2021. 3

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

[32] Weide Liu, Chi Zhang, Guosheng Lin, Tzu-Yi Hung, and Chunyan Miao. Weakly supervised segmentation with maximum bipartite graph matching. In *ACMMM*, 2020. 6

[33] Chen Liyi, Wu Weiwei, Chenchen Fu, Xiao Han, and Yuntao Zhang. Weakly supervised semantic segmentation with boundary exploration. In *ECCV*, 2020. 7

[34] Michal Lukasik, Srinadh Bhojanapalli, Aditya Menon, and Sanjiv Kumar. Does label smoothing mitigate label noise? In *ICML*, 2020. 5

[35] Deepak Pathak, Philipp Krahenbuhl, and Trevor Darrell. Constrained convolutional neural networks for weakly supervised segmentation. In *ICCV*, 2015. 1

[36] Pedro O Pinheiro and Ronan Collobert. From image-level to pixel-level labeling with convolutional networks. In *CVPR*, 2015. 1

[37] Guolei Sun, Wenguan Wang, Jifeng Dai, and Luc Van Gool. Mining cross-image semantics for weakly supervised semantic segmentation. In *ECCV*, 2020. 3, 7

[38] Yude Wang, Jie Zhang, Meina Kan, Shiguang Shan, and Xilin Chen. Self-supervised equivariant attention mechanism for weakly supervised semantic segmentation. In *CVPR*, 2020. 3, 6, 7

[39] 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 *CVPR*, 2017. 1

[40] 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. *TPAMI*, 2016. 3

[41] Qi Yao and Xiaojin Gong. Saliency guided self-attention network for weakly and semi-supervised semantic segmentation. *IEEE Access*, 2020. 3, 7

[42] Yu Zeng, Yunzhi Zhuge, Huchuan Lu, and Lihe Zhang. Joint learning of saliency detection and weakly supervised semantic segmentation. In *ICCV*, 2019. 3

[43] 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*, 2020. 7

[44] Dong Zhang, Hanwang Zhang, Jinhui Tang, Xian-Sheng Hua, and Qianru Sun. Causal intervention for weakly-supervised semantic segmentation. In *NeurIPS*, 2020. 6, 7

[45] Bolei Zhou, Aditya Khosla, Agata Lapedriza, Aude Oliva, and Antonio Torralba. Learning deep features for discriminative localization. In *CVPR*, 2016. 1, 2, 6