Embedded Discriminative Attention Mechanism for Weakly Supervised Semantic Segmentation

Tong Wu[1]*, Junshi Huang[2]*, Guangyu Gao[1]†, Xiaoming Wei[2], Xiaolin Wei[2], Xuan Luo[2], Chi Harold Liu[1]
[1]Beijing Institute of Technology [2]Meituan
{3220190896, guangyugao, chiliu}@bit.edu.cn
{huangjunshi, weixiaoming, weixiaolin02, luoxuan03}@meituan.com

Abstract

Weakly Supervised Semantic Segmentation (WSSS) with image-level annotation uses class activation maps from the classifier as pseudo-labels for semantic segmentation. However, such activation maps usually highlight the local discriminative regions rather than the whole object, which deviates from the requirement of semantic segmentation. To explore more comprehensive class-specific activation maps, we propose an Embedded Discriminative Attention Mechanism (EDAM) by integrating the activation map generation into the classification network directly for WSSS. Specifically, a Discriminative Activation (DA) layer is designed to explicitly produce a series of normalized class-specific masks, which are then used to generate class-specific pixel-level pseudo-labels demanded in segmentation. For learning the pseudo-labels, the masks are multiplied with the feature maps after the backbone to generate the discriminative activation maps, each of which encodes the specific information of the corresponding category in the input images. Given such class-specific activation maps, a Collaborative Multi-Attention (CMA) module is proposed to extract the collaborative information of each given category from images in a batch. In inference, we directly use the activation masks from the DA layer as pseudo-labels for segmentation. Based on the generated pseudo-labels, we achieve the mIoU of 70.60% *on PASCAL VOC* 2012 *segmentation testset, which is the new state-of-the-art, to our best knowledge. Code and pre-trained models are available online soon.*

1. Introduction

Driven by Deep Neural Networks (DNNs), significant progress [32, 6, 7] has been made in fully supervised semantic segmentation, for which however, it is time-consuming and costly to attain pixel-level annotations. To deal with this problem, researchers seek to leverage weaker supervision, such as bounding boxes [34], scribbles [29], points [3], and even image-level labels [35], for semantic segmentation, namely, Weakly Supervised Semantic Segmentation (WSSS). In this paper, we mainly focus on the image-level labels based WSSS.

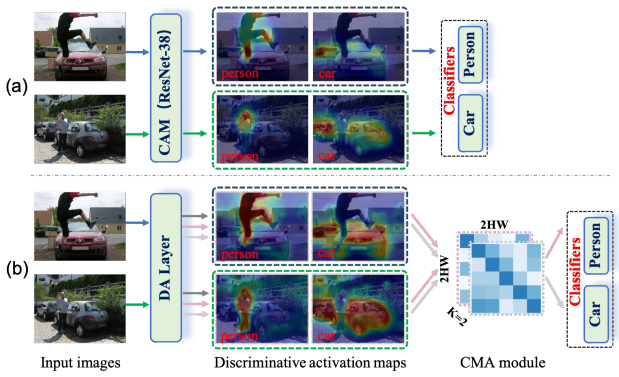

Figure 1. Illustration of our motivation. (a) The heatmap from CAM [52] tends to only highlight the local discriminative regions instead of the whole object. (b) In our model, the DA layer extracts class-specific activation maps, called discriminative activation maps, of multi-images. For each category, the CMA module exploits the collaborative information based on the intra-image and inter-image homogeneity of activation maps. Note that the *black*, *pink* and *grey* lines indicate the information flow of background, person and car, respectively. Best viewed in color.

Previous WSSS methods [21, 45, 40, 4] with image-level labels often use a classification network to generate the initial segmentation response, such as the Class Activation Map (CAM) [52], to highlight the corresponding foreground. However, the initial response from the classifier usually focuses on the discriminative object regions, rather than the whole object, which deviates from a desirable segmentation result. Some approaches improve quality of such initial responses by region erasing [46, 18], region grow-

*Equal contribution.

†Corresponding author. This work was supported mainly by the National Natural Science Foundation of China under Grant 61972036, in part by grants under Grant 91746210, and in part by Beijing Science and Technology Project. (No. Z181100008918018).

ing [20], or multi-scale feature map fusion [48, 21], which easily lead to the background regions incorrectly highlighted as well. The latest trend is to add auxiliary tasks, including consistency regularization [45], sub-category classification [4] and mining cross-image semantics [40], to train jointly with the classification network for refining object responses. However, these auxiliary tasks usually explore information in an *implicit* manner, leading to a complex training process and a non-optimal inference result.

In this work, we propose an Embedded Discriminative Attention Mechanism (EDAM) to *explicitly* infer class-specific masks for WSSS by exploring the intra-image and inter-image homogeneity. The EDAM consists of a Discriminative Activation layer (DA layer) and a Collaborative Multi-Attention module (CMA module), which is illustrated in Fig. 1. In the DA layer, we first predict the class-specific masks for foreground categories and background, which explicitly represent the probability of each pixel belonging to a specific category, and can directly serve as the initial segmentation responses for WSSS. These normalized masks are then multiplied with the original feature maps to generate discriminative activation maps, each of which encodes the information of each category or background. After that, the CMA module is used to explore the *collaborative information* of the corresponding foreground object by applying *self-attention* to the activation maps of *multi-images*. We then apply average pooling on the attended activation map of each image, and feed the pooling result into the binary classifier of the corresponding category for prediction. Note that the CMA module is only required in the training stage, as we can acquire the initial segmentation response via the class-specific masks as aforementioned. In tradition, post-processing is often applied, where saliency maps are widely used for result enhancement [11, 25, 44, 40]. However, due to the class-agnostic property, the saliency map usually wrongly highlights the non-target objects, or suppresses the insignificant target objects. Therefore, we also develop a new strategy for post-processing in our method as well for further improvement.

We claim that the proposed EDAM enjoys two advantages for WSSS. First, using a class-specific activation map eliminates some impact of other irrelevant categories or background, and thus benefits the classification accuracy. Also, as the collaborative information for classification stems from the discriminative activation maps, optimizing the classifiers is able to directly enhance the prediction of class-specific masks.

Our main contributions can be summarized as follows:

- The Embedded Discriminative Attention Mechanism (EDAM) is proposed to seamlessly integrate the semantic segmentation task into the classification network. To our best knowledge, this work is the first trail of *explicitly* modeling the semantic segmentation in classification networks for WSSS.

- Different from the existing attention-based methods [40, 45], we explore the collaborative information based on the intra-image and inter-image homogeneity of discriminative activation maps simultaneously, which can make full use of the supervision information of image-level labels.

- We propose a new strategy for post-processing, including the foreground pop-up and background suppression, which further enhances the quality of pseudo-labels for training segmentation networks.

- On the PASCAL VOC 2012, our approach achieves the new state-of-the-art, with mIoU of 70.9% and 70.6% on validation and test sets respectively.

2. Related Work

We briefly review the prior works on WSSS, including pseudo-labels generation and refinement, followed by those on the self-attention and its applications.

2.1. Pseudo-label Generation

The common solution to WSSS with image-level supervision is to train the segmentation network with the pseudo-labels generated by the classification network using network visualization techniques. Especially, the Class Activation Map (CAM) [52] is the most widely used. The CAM is to combine the weight of the linear layer with the feature map before global average pooling to determine the contribution of each pixel to the classification result. However, CAM only highlights the most discriminative regions of the image since it is merely trained by classification tasks, and thus the regions of pseudo-labels generated by CAM are usually incomplete. A lot of works have been proposed to expand the activation regions of CAM. Among them, some make the network focus on more regions of the object by constantly erasing the most discriminative region of the object [18, 46]; some works aggregate multiple CAMs to expand the target regions using dilated convolution with different dilate rates [48, 25], different epochs' parameters [21] or different layers to generate multiple CAMs [26]. In addition, some latest works explore auxiliary tasks to automatically make the network focus on more pixels. For example, SEAM [45] adopts the consistency regularization on CAMs from transformed images; [4] explores the sub-category for classification; [40] exploits the cross-image semantic relations. Different from the above methods, our EDAM directly predicts the activation map for segmentation by seamlessly integrating it into the classification network for joint learning.

2.2. Pseudo-labels Refinement

Some methods focus on refining the segmentation pseudo-labels generated by the classification network through class-agnostic post-processing. The SEC [23] develops three principles of seed, expansion and constraining, to refine the CAM, which is followed by many other works. DSRG [20] inspired by seeded region growing uses CAM as seed cues and expands the regions of interest. AffinityNet [2] trains another network to learn the similarity between pixels, which generates a transition matrix and implements the semantic propagation by a random walk. OAA$^+$ [4] uses an integral attention model to strengthen the lower attention values of target object regions while constraining the excess expansion of attention regions to the background. Also, many methods [11, 20, 48, 25, 27, 44, 46, 40] use the CAMs as foreground cues, and class-agnostic saliency maps generated by other pre-trained networks as background cues.

2.3. Self-attention Mechanism

Recently, the attention mechanism [42] has become increasingly popular in many research fields including the semantic segmentation. The non-local convolution network [43] is firstly proposed to solve the short dependency caused by the convolution. After that, in [53] an asymmetric non-local network is proposed as a condensed version of the non-local network. [19] presents the CCNet which improves the efficiency by reducing the calculation flops in non-local blocks. In [12], a dual attention network consisting of one channel attention module and one spatial attention module is developed for scene segmentation. In Visual Question Answering area, co-attention [33, 50] is used to improve performance by introducing the question-guided image attention and image-guided question attention. In this work, we involve the self-attention of class-specific activation maps based on intra-image and inter-image homogeneity to explore the collaborative information for WSSS.

3. Our Approach

We elaborate on the proposed Embedded Discriminative Attention Mechanism (EDAM) for WSSS, including the DA layer and the CMA module, as well as a new post-processing strategy.

3.1. Overview of EDAM

Commonly, a classification network is trained with image-level labels to highlight the discriminative regions of the object as the initial response for semantic segmentation. To improve the quality of initial response, our Embedded Discriminative Attention Mechanism (EDAM) seamlessly integrates the response generation into the classification network for joint learning, as shown in Fig. 2.

Besides the backbone network, the EDAM includes a Discriminative Activation (DA) layer and a Collaborative Multi-Attention (CMA) module. Unlike most previous works [48, 2, 25, 21, 4] using a single activation map for classification, we add a DA layer after the backbone to generate the class-specific activation map for each category. Such an activation map encodes the specific information of a single category, and is thus named as class-specific *discriminative activation map*. Given the class-specific discriminative activation maps of multiple images in a training batch, the CMA module is used to explore the intra-image and inter-image homogeneity, which we call *collaborative information*, by applying a self-attention mechanism. Finally, the collaborative information of each image is independently fed into the classifier for final prediction.

We find through experiments (see Sec. 4.5.2) that the number of images for collaborative information extraction has little impact upon the quality of the initial response. Therefore, we simply use two images in the CMA module for training efficiency. In the inference stage, we remove the CMA module and directly use the class-specific masks from the DA layer as the initial response of foreground objects.

3.2. Discriminative Activation Layer

Denote the training data as $\mathcal{I} = \{(\boldsymbol{I}_n, \boldsymbol{l}_n)\}_N$, where $\boldsymbol{I}_n$ is the n-th image, and $\boldsymbol{l}_n \in \{0,1\}^K$ represents the corresponding image-level labels of K categories. The DA layer appended after the CNN backbone network takes the feature maps $\boldsymbol{F}_n \in \mathbb{R}^{C\times H\times W}$ of $\boldsymbol{I}_n$ as input, and outputs the class-specific activation mask $\boldsymbol{M}_n \in \mathbb{R}^{(K+1)\times H\times W}$ for the K object categories and the background. It should be noted that, to prevent the region of background scattering over the area of foreground objects, we explicitly model the background mask in $\boldsymbol{M}_n$.

Since the activation mask is required to indicate the probability of each pixel belonging to a corresponding category or the background, we apply L2-norm along the channel axis of $\boldsymbol{M}_n$ as follows:

$$\hat{\boldsymbol{M}}_n(i,j) = norm(|\boldsymbol{M}_n(i,j)|). \tag{1}$$

After that, $\hat{\boldsymbol{M}}_n(i,j)$ can be considered as the pixel-wise category distribution at position (i,j), and $\hat{\boldsymbol{M}}_n^k(i,j)$ is the k-th value of the distribution vector. Hitherto, we can easily obtain the class-specific activation map of the image $\boldsymbol{I}_n$ for each category by

$$\boldsymbol{F}_n^k = \boldsymbol{F}_n \cdot \hat{\boldsymbol{M}}_n^k \tag{2}$$

where $\boldsymbol{F}_n^k$ is the class-specific activation map of the image $\boldsymbol{I}_n$ on the category k, and $k \in [0, K]$. Note again that the background activation map is removed as it is useless in the following procedures.

Overall, in the DA layer, multi-head class-specific information is separated from the whole activation map. Each

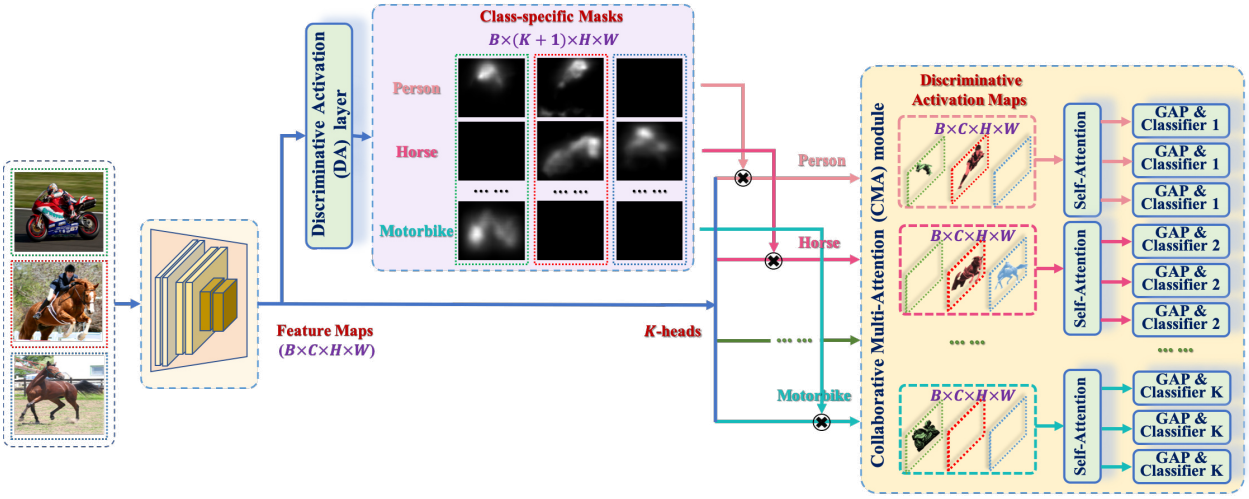

Figure 2. Overall architecture of the proposed EDAM for WSSS. EDAM consists of a Discriminative Activation (DA) layer and a Collaborative Multi-Attention (CMA) module. Note the CMA module is not required in inference. Best viewed in color.

head concentrates on the discriminative information of each category, which is why it is named the discriminative activation map. Since the noise from background or irrelevant categories is eliminated in the discriminative activation map, the performance of each head can be greatly enhanced, as we will see in the ablation study (see Sec. 4.5.1).

3.3. Collaborative Multi-Attention Module

Given a series of class-specific discriminative activation maps of an image set on a certain category, the CMA module is to highlight the similar regions, namely collaborative information, of these activation maps. Inspired by the attention matrix in self-attention that indicates the compatibility of query and key [42], we propose to use the self-attention mechanism directly in our CMA module to explore the collaborative information.

Denoting $\mathcal{F}^k = [\boldsymbol{F}_1^k, \boldsymbol{F}_2^k, ..., \boldsymbol{F}_B^k] \in \mathbb{R}^{B\times C\times H\times W}$ as the activation maps of B images on the k-th category, we first reduce the channel number of $\mathcal{F}^k$ to d by a 1×1 convolutional layer and permute it to $\hat{\mathcal{F}}^k \in \mathbb{R}^{1\times(B\times H\times W)\times d}$, which can be considered as a sequence of $B\times H\times W$ tokens. Similar to self-attention, we add two kinds of positional encodings onto $\hat{\mathcal{F}}^k$. Specifically, within each subsequence $\hat{\boldsymbol{F}}_i^k \in \mathbb{R}^{(H\times W)\times d}$, a shared 1-D positional encoding of length $H\times W$ is embedded for each activation map. Across the activation maps in $\hat{\mathcal{F}}^k$, B positional encoding vectors are injected, each of which is repeated for $H\times W$ times to fit the input dimension.

With the input embedding, the self-attention module is directly used to explore the collaborative information of class-specific activation maps. Different from the co-attention manner, the self-attention mechanism considers the attention of descriptors within an image and among images simultaneously, and thus is more effective for exploring collaborative information. Moreover, the labels of images in the same batch are not required to be similar, as the inter-image homogeneity is able to provide enough information for learning. More details can be found in ablation study (see Sec. 4.5.2).

As the output of the self-attention module has the same size to its input activation maps, we perform global average pooling on the output activation map of each image for each specific category, and use the corresponding class-specific classifier for label prediction. As the input image may have multiple categories, we solve them as multiple binary classification tasks, and the loss function can be written as

$$\mathcal{L}_{cls} = \frac{1}{B\times K}\sum_{n=1}^{B}\sum_{k=1}^{K}\mathcal{L}_{BCE}(Linear(GAP(\boldsymbol{A}_n^k)), \boldsymbol{l}_n^k) \tag{3}$$

$$[\boldsymbol{A}_1^k, \boldsymbol{A}_2^k, ..., \boldsymbol{A}_B^k] = SelfAttention(\hat{\mathcal{F}}^k) \tag{4}$$

where $\boldsymbol{A}_n^k$ is the output activation map of the image $\boldsymbol{I}_n$ on the k-th category after the self-attention module, and $\boldsymbol{l}_n^k \in [0,1]$ is the ground-truth label of the image $\boldsymbol{I}_n$ on the k-th category. Finally, since the self-attention module receives as input the class-specific activation maps of a single category, K independent self-attentions are integrated into CMA module for all foreground categories, respectively.

3.4. Post-processing

During inference, we use the normalized activation mask $\hat{\boldsymbol{M}}$ from the DA layer as the foreground object cues, and follow the popular pipeline of WSSS [11, 20, 48, 25, 27, 44, 46, 40] by using the saliency map [31, 17] to refine the background regions. Specifically, the foreground regions and object categories are extracted by selecting the pixel-wise label map according to the maximum value of the ac-

Algorithm 1 Pseudo-labels Generation

Input: Normalized Activation Mask $\hat{\boldsymbol{M}}$; Saliency map $\boldsymbol{S}$; Category Number K; Thresholds θ, α, β
Output: Pseudo Label Map $\boldsymbol{P}$

// Assume the background label is 0

$$\boldsymbol{P} = argmax_k(\hat{\boldsymbol{M}}^{\boldsymbol{k}}), \text{ where } k \in [0, K]$$

$$\boldsymbol{S}_{i,j} = \begin{cases} 0, \text{if } \boldsymbol{S}_{i,j} < \theta \\ 1, \text{otherwise} \end{cases}$$

$$\boldsymbol{P}_{i,j} = \begin{cases} 0, \text{if } \boldsymbol{S}_{i,j} == 1 \&\& \hat{\boldsymbol{M}}_{\boldsymbol{i,j}}^{\boldsymbol{P}_{i,j}} < \alpha \\ \boldsymbol{P}_{i,j}, \text{elif } \boldsymbol{S}_{i,j} == 0 \&\& \hat{\boldsymbol{M}}_{\boldsymbol{i,j}}^{\boldsymbol{P}_{i,j}} > \beta \\ \boldsymbol{P}_{i,j} * \boldsymbol{S}_{i,j}, \text{otherwise} \end{cases}$$

tivation mask along the channel dimension. Meanwhile, the saliency detection [31] is used to extract the saliency map of an image, whose value varies within $[0, 255]$, and the region of the label map is considered as background if the saliency value is less than the threshold θ.

However, the saliency map is class-agnostic and tends to highlight the most discriminative objects in the image, in which case the non-target objects would be treated as foreground. Meanwhile, due to the diversity of input images, some foreground objects may locate at the image corner, and are usually treated as background by the saliency map. Therefore, we arbitrarily set the salient regions as background if the maximum pixel-wise value of the corresponding region in the activation mask is less than the predefined threshold α; otherwise we consider the insignificant parts of the saliency map as foreground if the maximum pixel-wise value of the corresponding region in the activation mask exceeds the predefined threshold β. The whole process of post-processing is shown in Alg. 1.

4. Experiments

4.1. Datasets and Evaluation Metric

We evaluate our approach on the PASCAL VOC 2012 dataset [9] and the COCO-Stuff 10k dataset [30]. As implemented in most previous works [11, 48, 25, 44, 40], we use the augmented training set [14] which includes 10,582 images in VOC for model training. We only use image-level labels during training, and each image contains one or multiple categories. To compare our approach with the competitors, we evaluate our results on both validation and test sets. For all experiments, the mean Intersection over Union (mIoU) is used as the evaluation metric. Since the ground-truths of the test set are not publicly available, the mIoU of test set is obtained by submitting the results to the official evaluation website of PASCAL VOC.

4.2. Implementation Details

In our experiments, we use ResNet38 [49] as backbone for our EDAM architecture. It is pre-trained on the ImageNet dataset [8] and is fine-tuned on the target dataset, including PASCAL VOC 2012 and COCO-Stuff. The input images are randomly re-scaled and cropped to 368×368. We also use random horizontal flip and color jitter as data augmentation during training. In pseudo label generation, the bilinear interpolation is used to expand the output of the DA layer to the size of the original images. During segmentation, DeepLab-LargeFOV [6] is used as our segmentation network with the ResNet101 [15] as backbone.

Our model is implemented with Pytorch and trained on 4 Tesla V100 GPUs with 16 GB memory. We use the SGD optimizer with learning rate $1e-3$, and warm up the training in first $2k$ iterations with learning rate $1e-4$, which is then gradually reduced in each iteration. We run the classification network for $30k$ iterations totally. Other hyper-parameters are set as follows: the batch-size is 8, the weight decay is 0.0002, and the momentum is 0.9. Unless otherwise specified, the default values of θ, α and β are 20, 0.25 and 0.95, which are tuned by grid-search on evaluation set.

4.3. Comparisons to State-of-the-arts

In the experiments, the post-processed pseudo-labels from the DA layer are used to fully-supervised train the DeepLab-LargeFOV to get the final semantic segmentation results. The accuracy of each category is shown in Table 1, and the comparison with some latest methods is shown in Table 2. As we can see, our method not only achieves significantly better performance on both validation set and test set (with overall 3.1% and 2.6% enhancement of mIoU comparing to [10]), but also dominates on most of the categories (with the highest mIoU in 16 of the 21 categories). Moreover, it should be noted that our initial response is directly generated by the DA layer without too many bells and whistles. In Fig. 3, we present some examples of [40] and our final semantic segmentation results. It can be seen that our framework generates good segmentation, even though the images contain multiple objects of different categories or different sizes.

4.4. Learning WSSS with Extra Data

Besides the PASCAL VOC training set, we explore the performance of our EDAM method for WSSS by using extra training data. Following [28, 40], we train our network with extra single-label images from Caltech-256 [13] and web data [37]. Specifically, we manually select 3,995 extra images across 20 categories of PASCAL VOC 2012 from Caltech-256, and use the full set of 76,683 images from web data for training data expansion.

As shown in Table 3, compared to previous works [28, 40] under similar settings, our results can be greatly im-

Method	bkg	aero	bike	bird	boat	bottle	bus	car	cat	chair	cow	table	dog	horse	motor	person	plant	sheep	sofa	train	tv	mIoU
MCOF [44]	87.0	78.4	29.4	68.0	44.0	67.3	80.3	74.1	82.2	21.1	70.7	28.2	73.2	71.5	67.2	53.0	47.7	74.5	32.4	71.0	45.8	60.3
AffinityNet [2]	88.2	68.2	30.6	81.1	49.6	61.0	77.8	66.1	75.1	29.0	66.0	40.2	80.4	62.0	70.4	73.7	42.5	70.7	42.6	68.1	51.6	61.7
Zeng *et al*. [51]	90.0	77.4	37.5	80.7	61.6	67.9	81.8	69.0	83.7	13.6	79.4	23.3	78.0	75.3	71.4	68.1	35.2	78.2	32.5	75.5	48.0	63.3
SEAM [45]	88.8	68.5	33.3	**85.7**	40.4	67.3	78.9	76.3	81.9	29.1	75.5	48.1	79.9	73.8	71.4	75.2	48.9	79.8	40.9	58.2	53.0	64.5
FickleNet [25]	89.5	76.6	32.6	74.6	51.5	71.1	83.4	74.4	83.6	24.1	73.4	47.4	78.2	74.0	68.8	73.2	47.8	79.9	37.0	57.3	**64.6**	64.9
Chang *et al*. [4]	88.8	51.6	30.3	82.9	53.0	75.8	88.6	74.8	86.6	**32.4**	79.9	**53.8**	82.3	78.5	70.4	71.2	40.2	78.3	**42.9**	66.8	58.8	66.1
Ours	**92.0**	**87.0**	**42.4**	83.0	**70.0**	**76.4**	**89.5**	**79.5**	**88.6**	29.0	**87.2**	24.7	**83.8**	**83.0**	**81.0**	**82.1**	**51.9**	**83.6**	35.1	**82.0**	58.0	**70.9**

Table 1. Comparison of per-category performance on PASCAL VOC 2012 validation set.

Method	Backbone	Val	Test
SEC [23]	VGG16	50.7	51.1
AE-PSL [46]	VGG16	55.0	55.7
MDC [48]	VGG16	60.4	60.8
MCOF [44]	ResNet101	60.3	61.2
DCSP [5]	ResNet101	60.8	61.9
SeeNet [18]	ResNet101	63.1	62.8
DSRG [20]	ResNet101	61.4	63.2
AffinityNet [2]	ResNet38	61.7	63.2
IRNet [1]	ResNet50	63.5	64.8
FickleNet [25]	ResNet101	64.9	65.3
SSDD [39]	ResNet101	64.9	65.5
OAA$^+$ [21]	ResNet101	65.2	66.4
Cian [11]	ResNet101	64.3	65.3
SEAM [45]	ResNet38	64.5	65.7
Chang *et al*. [4]	ResNet101	66.1	65.9
MCIS [40]	ResNet101	66.2	66.9
ICD [10]	ResNet101	67.8	68.0
Ours	ResNet101	**70.9**	**70.6**

Table 2. Comparison to previous state-of-the-art approaches of weakly-supervised semantic segmentation on PASCAL VOC 2012 validation and test sets.

Method	Val	Test
MCNN [41]	-	36.9
MIL-seg [35]	42.0	40.6
AttnBN [28]	66.1	65.9
MCIS [40]	67.1	67.2
Ours	**72.0**	**71.4**

Table 3. Results of WSSS with extra simple single-label images.

proved with the expanded dataset. Particularly, note that, different from the previous best-performing work [40] using 20k extra images from both Caltech-256 and ImageNet CLS-LOC [36], we use only 4k extra images from Caltech-256 while achieving better performance. Table 4 shows further improved performance of our method with extra web data on test set, even though there exists noise.

4.5. Ablation Studies

In this subsection, we conduct extensive ablation experiments to prove the effectiveness of our method design. Since our method focuses on the improvement of pseudo-labels, we train the network on the augmented training images for PASCAL VOC segmentation task with only image-level labels, and evaluate the mIoU of pseudo-labels on the images for fully-supervised segmentation task.

Method	Val	Test
MCNN [41]	38.1	39.8
STC [47]	49.8	51.2
WebS-i2 [22]	53.4	55.3
Hong *et al*. [16]	58.1	58.7
BDWSS [38]	63.0	63.9
MCIS [40]	67.7	67.5
Ours	**71.3**	**71.6**

Table 4. Results of WSSS with extra noisy web images/videos.

CAM	DA	CMA	dCRF	mIoU
✓				47.55%
✓	✓			51.58%
✓	✓	✓		52.83%
✓	✓	✓	✓	58.18%

Table 5. Ablation study of each component of our network.

4.5.1 Contribution of Components

In the experiments, we investigate the contribution of different components in our method, including the DA layer, the CMA module, and the dense Conditional Random Field (dCRF) [24]. Specifically, we use ResNet-38 as the classification backbone, and the CAMs generated by the fine-tuned ResNet-38 model as our baseline. To select the background region, we carefully tune the threshold for different results, and report the best of them. It is worth noting that, to conduct fair comparison, the post-processing is not used in these experiments.

As shown in Table 5, compared with the baseline, the DA layer improves the mIoU of pseudo-labels by over 4%. By introducing the CMA module for collaborative information exploration, we achieve another 1.2% enhancement, reaching up to 52.83% mIoU. For efficiency, the class-specific activation maps of two images with similar labels are fed into CMA module. Actually such a constraint on similar labels is unnecessary, as proved in Sec. 4.5.2. We also eval-

Figure 3. Qualitative segmentation results on PASCAL VOC 2012 validation set. (a) Original images. (b) Ground truth. (c) Segmentation results by [40]. (d) Segmentation results by DeepLab-LargeFOV model retrained on our pseudo-labels.

Input Data	mIoU
Single image	52.14%
Two random images	52.77%
Two similar images	52.83%
Three similar images	52.77%
Four similar images	52.52%

Table 6. The influence of different inputs on the mIoU.

uate the results after dCRF that is for further refinement, and achieve mIoU of 58.18%, which may be benefited from taking our results as high-quality initialization.

In Figure 4, we compare the heatmaps generated by baseline and our method without dCRF. It can be observed that our network can better extract the complete object instead of discriminative object parts. Especially, for the images with multiple instances, EDAM can highlight all the instances, while CAM tends to focus on some of them.

4.5.2 Data Configuration for CMA Module

Here we evaluate the impact of different datasets on the CMA module without post-processing. As we know, the CMA can explore the collaborative information from multiple images, and we conduct most experiments by feeding two images with similar labels into CMA module, namely "Two similar images". By using more similar images, we achieve comparable results to those of "Two similar images", as shown in the 3rd and 4th row of Table 6. The performance drop of "Four similar images" may be caused by our using smaller images due to memory limit. Meanwhile, we also randomly feed two images into the CMA module, and achieve mIoU of 52.77% (see "Two random images" in

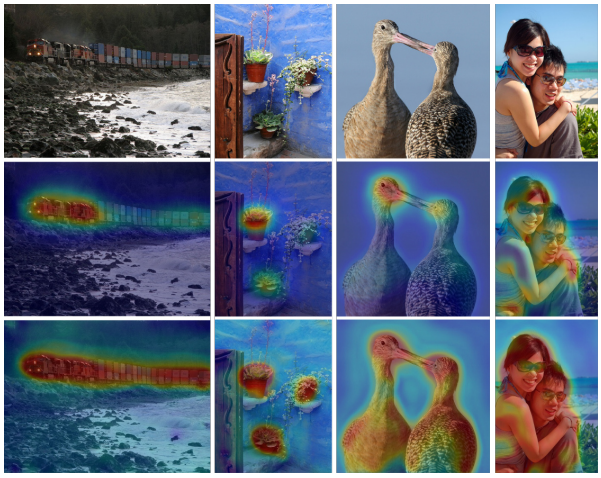

Figure 4. Visualization of heat maps. The first row is the original image, the second row is the heatmap from CAM, and the final row is the heatmap of EDAM without dCRF.

Table 6), which indicates the CMA module could explore the collaborative information in an inter-image manner, and the requirement for similar images in a batch is not compulsory. Besides, we present the result of using only a single image during training. It shows that the results of all multiple images configurations are better than using a single image, which proves the effectiveness of inter-image context.

4.5.3 Thresholds in Post-processing

In Table 7, we show the ablative results for different strategies in post-processing by carefully tuning the threshold θ,

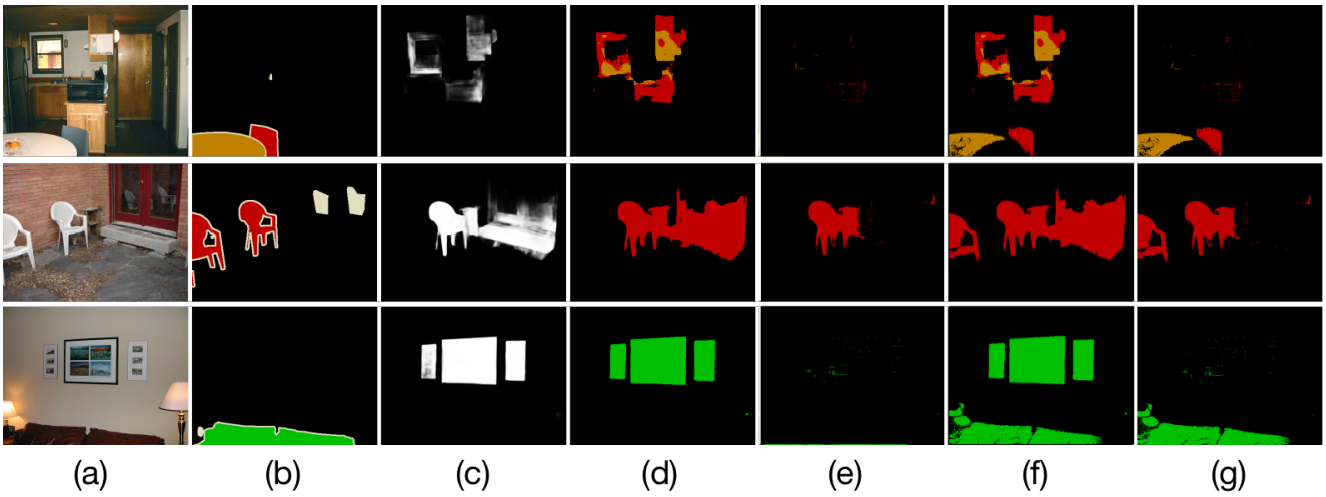

Figure 5. Visualization of strategies in post-processing. (a) Original images. (b) Ground-truths. (c) Saliency maps. (d) Pseudo-labels with saliency map fusion. (e) Suppressing non-target regions. (f) Popup high-confident regions. (g) Final fusion result.

Baseline	θ	α	β	mIoU
✓				58.18%
✓	✓			66.10%
✓	✓	✓		66.98%
✓	✓		✓	67.22%
✓	✓	✓	✓	68.11%

Table 7. Ablation study for α and β in post-processing.

α, and β respectively. We use the final label map of EDAM after dCRF as the baseline. By using the saliency map and tuning the threshold θ, the mIoU of pseudo-label reaches 66.11%. Like most previous works, this strategy can greatly improve the quality of initial responses. However, there are still some drawbacks with the saliency map. More details can be found in Sec. 3.4, where we propose two strategies to improve the results. By introducing the α and β for map fusion, the mIoU can be increased by 0.88% and 1.12%, respectively. Moreover, the fusion of all strategies can further improve the quality of pseudo-labels, reaching up to 68.11% in our experiments.

We visualize some results to reveal why the post-processing can improve final performance. In Fig. 5 (c) and (d), we can find that the saliency map may focus on some non-target objects, such as the windows and wall painting. By suppressing the incorrect foreground with α, some non-target regions are removed in Fig. 5 (e). Also, we can popup the high-confident regions generated by the DA layer as shown in Fig. 5 (f). The final fusion result of those strategies can be found in Fig. 5 (g), which greatly enhances the quality of the initial segmentation response.

4.6. Learning WSSS on COCO-Stuff 10k dataset

In addition to the PASCAL VOC 2012, we also explore the effectiveness of EDAM on COCO-Stuff 10k [30]. Specifically, we select 9,000 images of 20 categories the same as PASCAL VOC for training, and use remaining categories as background. The final mIoU of DeepLab-LargeFOV trained with pseudo-labels generated by EDAM reaches mIoU of 51.44% on test set, while the mIoU of DeepLab-LargeFOV trained with fully-supervised labels is 55.90%. Even in more complex scenarios, our EDAM can achieve comparable results to those with full-supervision.

5. Conclusion

In this paper, we propose a simple yet effective framework, namely EDAM, for weakly-supervised semantic segmentation (WSSS). This framework is implemented by seamlessly integrating the segmentation task into the classification network to reduce the annotation gap. Specifically, the DA layer predicts the class-specific mask and generates the discriminative activation map for each category. As the region of background and irrelevant foreground objects are removed in the class-specific activation map, to some extent, the performance of classification can be greatly improved. After that, the CMA module is used to explore the collaborative information by concatenating multiple class-specific activation maps of the images. Based on the idea of self-attention, the intra-image and inter-image homogeneity can be explored simultaneously. In inference, we can directly use the activation mask from the DA layer to generate the pseudo-labels without any additional computational cost. Finally, a new post-processing strategy is proposed to further improve the quality of the initial segmentation response. Based on the improved initial response of our method, the segmentation network achieves new state-of-the-art performance of WSSS on the PASCAL VOC 2012 segmentation dataset.

References

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

[2] Jiwoon Ahn and Suha Kwak. Learning pixel-level semantic affinity with image-level supervision for weakly supervised semantic segmentation. In *Proc. CVPR*, 2018. 3, 6

[3] Amy Bearman, Olga Russakovsky, Vittorio Ferrari, and Li Fei-Fei. What's the point: Semantic segmentation with point supervision. In *Proc. ECCV*, 2016. 1

[4] 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 *Proc. CVPR*, 2020. 1, 2, 3, 6

[5] Arslan Chaudhry, Puneet K Dokania, and Philip HS Torr. Discovering class-specific pixels for weakly-supervised semantic segmentation. *arXiv:1707.05821*, 2017. 6

[6] Liang-Chieh Chen, George Papandreou, Iasonas Kokkinos, Kevin Murphy, and Alan L Yuille. Deeplab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected crfs. *IEEE TPAMI*, 40(4):834–848, 2017. 1, 5

[7] Liang-Chieh Chen, Yukun Zhu, George Papandreou, Florian Schroff, and Hartwig Adam. Encoder-decoder with atrous separable convolution for semantic image segmentation. In *Proc. ECCV*, 2018. 1

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

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

[10] Junsong Fan, Zhaoxiang Zhang, Chunfeng Song, and Tieniu Tan. Learning integral objects with intra-class discriminator for weakly-supervised semantic segmentation. In *cvpr*, June 2020. 5, 6

[11] Junsong Fan, Zhaoxiang Zhang, Tieniu Tan, Chunfeng Song, and Jun Xiao. Cian: Cross-image affinity net for weakly supervised semantic segmentation. In *Proc. AAAI* , 2020. 2, 3, 4, 5, 6

[12] Jun Fu, Jing Liu, Haijie Tian, Yong Li, Yongjun Bao, Zhiwei Fang, and Hanqing Lu. Dual attention network for scene segmentation. In *Proc. CVPR*, pages 3146–3154, 2019. 3

[13] Gregory Griffin, Alex Holub, and Pietro Perona. Caltech-256 object category dataset. 2007. 5

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

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

[16] Seunghoon Hong, Donghun Yeo, Suha Kwak, Honglak Lee, and Bohyung Han. Weakly supervised semantic segmentation using web-crawled videos. In *cvpr*, pages 7322–7330, 2017. 6

[17] Qibin Hou, Ming-Ming Cheng, Xiaowei Hu, Ali Borji, Zhuowen Tu, and Philip HS Torr. Deeply supervised salient object detection with short connections. In *Proc. CVPR*, 2017. 4

[18] Qibin Hou, PengTao Jiang, Yunchao Wei, and Ming-Ming Cheng. Self-erasing network for integral object attention. In *Proc. NeurIPS*, 2018. 1, 2, 6

[19] Zilong Huang, Xinggang Wang, Lichao Huang, Chang Huang, Yunchao Wei, and Wenyu Liu. Ccnet: Criss-cross attention for semantic segmentation. In *Proc. ICCV*, pages 603–612, 2019. 3

[20] Zilong Huang, Xinggang Wang, Jiasi Wang, Wenyu Liu, and Jingdong Wang. Weakly-supervised semantic segmentation network with deep seeded region growing. In *Proc. CVPR*, 2018. 2, 3, 4, 6

[21] Peng-Tao Jiang, Qibin Hou, Yang Cao, Ming-Ming Cheng, Yunchao Wei, and Hong-Kai Xiong. Integral object mining via online attention accumulation. In *Proc. ICCV*, 2019. 1, 2, 3, 6

[22] Bin Jin, Maria V Ortiz Segovia, and Sabine Susstrunk. Webly supervised semantic segmentation. In *cvpr*, pages 3626–3635, 2017. 6

[23] Alexander Kolesnikov and Christoph H. Lampert. Seed, expand and constrain: Three principles for weakly-supervised image segmentation. In *Proc. ECCV*, 2016. 3, 6

[24] Philipp Krähenbühl and Vladlen Koltun. Efficient inference in fully connected crfs with gaussian edge potentials. In *Proc. NeurIPS*, 2011. 6

[25] Jungbeom Lee, Eunji Kim, Sungmin Lee, Jangho Lee, and Sungroh Yoon. Ficklenet: Weakly and semi-supervised semantic image segmentation using stochastic inference. In *Proc. CVPR*, 2019. 2, 3, 4, 5, 6

[26] Sungmin Lee, Jangho Lee, Jungbeom Lee, Chul-Kee Park, and Sungroh Yoon. Robust tumor localization with pyramid grad-cam. *CoRR*, 2018. 2

[27] Kunpeng Li, Ziyan Wu, Kuan-Chuan Peng, Jan Ernst, and Yun Fu. Tell me where to look: Guided attention inference network. In *Proc. CVPR*, 2018. 3, 4

[28] Kunpeng Li, Yulun Zhang, Kai Li, Yuanyuan Li, and Yun Fu. Attention bridging network for knowledge transfer. In *Proc. ICCV*, 2019. 5, 6

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

[30] 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 *Proc. ECCV*, 2014. 5, 8

[31] Jiang-Jiang Liu, Qibin Hou, Ming-Ming Cheng, Jiashi Feng, and Jianmin Jiang. A simple pooling-based design for real-time salient object detection. In *Proc. CVPR*, 2019. 4, 5

[32] Jonathan Long, Evan Shelhamer, and Trevor Darrell. Fully convolutional networks for semantic segmentation. In *Proc. CVPR*, 2015. 1

[33] Jiasen Lu, Jianwei Yang, Dhruv Batra, and Devi Parikh. Hierarchical question-image co-attention for visual question answering. In *Proc. NeurIPS*, pages 289–297, 2016. 3

[34] George Papandreou, Liang-Chieh Chen, Kevin P Murphy, and Alan L Yuille. Weakly-and semi-supervised learning of a deep convolutional network for semantic image segmentation. In *Proc. ICCV*, 2015. 1

[35] Pedro O Pinheiro and Ronan Collobert. From image-level to pixel-level labeling with convolutional networks. In *Proc. CVPR*, 2015. 1, 6

[36] Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, et al. Imagenet large scale visual recognition challenge. *Int J Comput Vis*, 115(3):211–252, 2015. 6

[37] Tong Shen, Guosheng Lin, Chunhua Shen, and Ian Reid. Bootstrapping the performance of webly supervised semantic segmentation. In *Proc. CVPR*, 2018. 5

[38] T. Shen, G. Lin, C. Shen, and I. Reid. Bootstrapping the performance of webly supervised semantic segmentation. In *cvpr*, pages 1363–1371, 2018. 6

[39] Wataru Shimoda and Keiji Yanai. Self-supervised difference detection for weakly-supervised semantic segmentation. In *Proc. ICCV*, 2019. 6

[40] Guolei Sun, Wenguan Wang, Jifeng Dai, and Luc Van Gool. Mining cross-image semantics for weakly supervised semantic segmentation. In *Proc. ECCV*, 2020. 1, 2, 3, 4, 5, 6, 7

[41] Pavel Tokmakov, Karteek Alahari, and Cordelia Schmid. Weakly-supervised semantic segmentation using motion cues. In *Proc. ECCV*, 2016. 6

[42] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *Proc. NeurIPS*, 2017. 3, 4

[43] Xiaolong Wang, Ross B. Girshick, Abhinav Gupta, and Kaiming He. Non-local neural networks. *Proc. CVPR*, pages 7794–7803, 2017. 3

[44] Xiang Wang, Shaodi You, Xi Li, and Huimin Ma. Weakly-supervised semantic segmentation by iteratively mining common object features. In *Proc. CVPR*, 2018. 2, 3, 4, 5, 6

[45] Yude Wang, Jie Zhang, Meina Kan, Shiguang Shan, and Xilin Chen. Self-supervised equivariant attention mechanism for weakly supervised semantic segmentation. In *Proc. CVPR*, 2020. 1, 2, 6

[46] 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 *Proc. CVPR*, 2017. 1, 2, 3, 4, 6

[47] 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. *pami*, 39(11):2314–2320, 2016. 6

[48] Yunchao Wei, Huaxin Xiao, Honghui Shi, Zequn Jie, Jiashi Feng, and Thomas S. Huang. Revisiting dilated convolution: A simple approach for weakly- and semi-supervised semantic segmentation. In *Proc. CVPR*, 2018. 2, 3, 4, 5, 6

[49] Zifeng Wu, Chunhua Shen, and Anton Van Den Hengel. Wider or deeper: Revisiting the resnet model for visual recognition. *Pattern Recognition*, 90:119–133, 2019. 5

[50] Zhou Yu, Jun Yu, Jianping Fan, and Dacheng Tao. Multi-modal factorized bilinear pooling with co-attention learning for visual question answering. In *Proc. ICCV*, pages 1821–1830, 2017. 3

[51] Yu Zeng, Yunzhi Zhuge, Huchuan Lu, and Lihe Zhang. Joint learning of saliency detection and weakly supervised semantic segmentation. In *Proc. ICCV*, 2019. 6

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

[53] Zhen Zhu, Mengde Xu, Song Bai, Tengteng Huang, and Xiang Bai. Asymmetric non-local neural networks for semantic segmentation. In *Proc. ICCV*, pages 593–602, 2019. 3