USAGE: A Unified Seed Area Generation Paradigm for Weakly Supervised Semantic Segmentation

Zelin Peng[1], Guanchun Wang[2], Lingxi Xie[3], Dongsheng Jiang[3], Wei Shen[1(✉)], and Qi Tian[3]
[1]MoE Key Lab of Artificial Intelligence, AI Institute, Shanghai Jiao Tong University
[2]School of Artificial Intelligence, Xidian University [3]Huawei Inc.
{zelin.peng, wei.shen}@sjtu.edu.cn; gwang_2@stu.xidian.edu.cn;
198808xc@gmail.com; dongsheng_jiang@outlook.com; tian.qi1@huawei.com

Abstract

*Seed area generation is usually the starting point of weakly supervised semantic segmentation (WSSS). Computing the Class Activation Map (CAM) from a multi-label classification network is the de facto paradigm for seed area generation, but CAMs generated from Convolutional Neural Networks (CNNs) and Transformers are prone to be under- and over-activated, respectively, which makes the strategies to refine CAMs for CNNs usually inappropriate for Transformers, and vice versa. In this paper, we propose a **U**nified optimization paradigm for **S**eed **A**rea **GE**neration (USAGE) for both types of networks, in which the objective function to be optimized consists of two terms: One is a generation loss, which controls the shape of seed areas by a temperature parameter following a deterministic principle for different types of networks; The other is a regularization loss, which ensures the consistency between the seed areas that are generated by self-adaptive network adjustment from different views, to overturn false activation in seed areas. Experimental results show that USAGE consistently improves seed area generation for both CNNs and Transformers by large margins, e.g., outperforming state-of-the-art methods by a mIoU of 4.1% on PASCAL VOC. Moreover, based on the USAGE-generated seed areas on Transformers, we achieve state-of-the-art WSSS results on both PASCAL VOC and MS COCO.*

1. Introduction

The goal of weakly supervised semantic segmentation (WSSS) is to train a semantic segmentation model under weak supervision, *i.e.*, image-level labels, so that the burden of relying on pixel-level labels is largely reduced. Among various types of weak supervision, we focus on studying WSSS under image-level labels [12, 20, 17, 18, 10, 37], which is considered one of the most challenging scenarios and has attracted increasing attention.

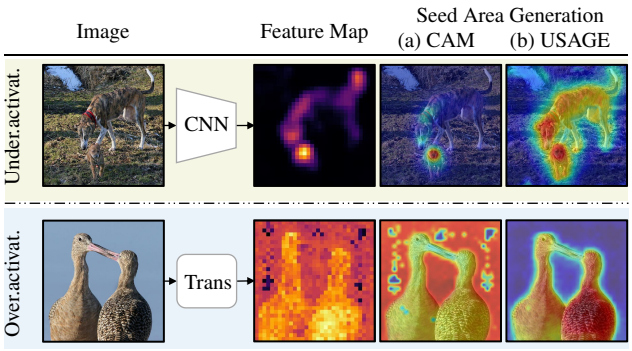


Figure 1. Seed area visualization. (a) CAM-based seed areas generated from a CNN (ResNet38 [48]) and a Transformer (DeiT-S [42]). (b) USAGE-based seed areas (ours) generated from the CNN and the Transformer.

Seed area generation is usually the first step of WSSS, which produces an initial pseudo mask based on the image-level labels on each training image. This is often achieved by first optimizing a mapping between the dense feature map of the image and the image-level labels (*i.e.,* training a deep neural network for multi-class classification), and then inferring the contribution of each location on the feature map to the classification result *w.r.t.* each specific class.

The Class Activation Map (CAM) [28] has been the *de facto* paradigm for seed area generation. However, it is known that, based on the CAM, seed areas generated from Convolutional Neural Networks (CNNs) are prone to be under-activated [3]. In addition, when the backbone is changed from CNNs to Transformers (which have been adopted in WSSS very recently and report state-of-the-art performance [51]), the CAM is inversely prone to result in over-activated seed areas [35], as shown in Fig. 1. Consequently, it is difficult to design a seed area generation paradigm that is appropriate for both CNN-based

✉Corresponding Author.

and Transformer-based WSSS. For example, the widely used CAM refinement strategy-"seed and expand" [20] has achieved success in CNN-based WSSS, but it is easy to deteriorate the over-activation of seed areas in Transformer-based WSSS.

In this paper, we propose a **U**nified optimization paradigm for **S**eed **A**rea **GE**neration (USAGE) for both types of networks, in which the objective function to be optimized consists of two terms: a generation loss and a regularization loss. The generation loss measures the fitness between the contribution of a seed area and the classification result and controls the shape of the seed area by introducing a temperature parameter based on a deterministic principle. This means how to set the temperature parameter for different types of networks is deterministic, *i.e.*, set a small/large temperature for Transformers/CNNs to sharpen/smooth the seed area.

The regularization loss is designed to ensure the consistency between the seed areas that are generated from different views, which can overturn false activation in seed areas. Previous WSSS methods [45, 19] mainly adopt geometric transformations to generate different views to instantiate such a regularization loss, which has achieved a great success on CNN-based WSSS, but this view generation strategy might not be appropriate for Transformer-based WSSS, since Transformers are insensitive to various geometric transformations [34]. Besides, geometric transformations, *e.g.,* multi-crop, may generate views that only contain the background region, which may make the consistency optimization process ill-posed. To alleviate the issue, we propose a self-adaptive network adjustment strategy to generate different views. In particular, we obtain different views by making adjustments to the architecture of the classification network, where the magnitudes of adjustments are determined by the learning status of the network. This strategy is shown to be effective for both types of networks in rectifying erroneously activated seed areas.

Extensive experiments show that USAGE can significantly improve the quality of the seed areas for both CNNs and Transformers. Furthermore, under the paradigm of USAGE, our instantiation for transformers, *i.e.*, applying sharpening and regularization on the seed areas, leads to new state-of-the-art WSSS results on both the PASCAL VOC [11] and MS COCO [29] dataset.

2. Related Work

WSSS methods can be categorized into two types: step-wise [51, 5, 53, 23] and end-to-end [36, 2]. Since step-wise methods generally achieve better performance than end-to-end methods, we focus on the former in this paper. Most step-wise WSSS methods follow such a sequence of steps: seed area generation, pseudo mask generation, and segmentation model training. Seed area generation is the first step for WSSS which provides initial cues to generate pseudo masks for further segmentation network training. The CAM [28] is a widely used technique to generate seed area. However, CAM-based seed areas generated from CNNs and Transformers are prone to be under- and over-activated, respectively.

Seed Area Generation from CNNs. Prior to the widespread usage of CAMs, earlier methods [32, 31] explored using multiple instance learning (MIL) for seed area generation. Pinheiro *et al.* [32] utilized LogSumExp-aggregation [4] to aggregate pixel-level predictions in the output layer, generating image-level scores. Due to the limited quality in seed area generation, MIL-based methods became inactive after the popularization of CAM. Adversarial erasing [46, 16] is a well-known CAM [28] expanding strategy that erases the most discriminative region in a CAM to enforce the classification network to activate on other areas. Wei *et al.* [47] proposed to enlarge seed areas by an ensemble of the CAMs computed using multiple dilated convolutional blocks with different dilation rates. Lee *et al.* [25] applied an anti-adversarial manner to perturb images along gradients *w.r.t.* the classification loss. Different from the above methods that mitigate under-activation by expanding seed areas via iteratively assembling, some other methods [45, 53] attempted to solve this problem by maintaining consistency between seed areas from different views, jointly optimized with the classification loss. The most representative method is [45], which generated views by different geometric transformations. However, the view consistency strategy under different geometric transformations might not be appropriate for Transformers, as Transformers are insensitive to various geometric transformations. Unlike these methods, the view consistency strategy in our USAGE is based on network adjustment, which is appropriate to both CNNs and Transformers.

Seed Area Generation from Transformers. Following the research line of Transformers, very recently, Xu *et al.* [51] proposed a multi-class token transformer to generate the class-to-patch attention map as the seed area. Although the class-to-patch attention map provides an upgraded version of the CAM [28] (more concrete details are discussed in Sec. 4.3), it also suffers from over-activation.

Our USAGE can address both the over-activation issue for Transformers and the under-activation issue for CNNs. We also show that, both CAM-based and MIL-based seed area generation methods are special cases of USAGE (Sec. 4.3).

3. Revisiting CAM in Seed Area Generation

The pipeline of the mainstream WSSS methods consists of three sequential steps: 1) Generating seed areas for training images from a multi-label classification network; 2) Refining seed areas to be pseudo masks via affinity propa-

Methods	mIoU (%)↑	FPR (%)↓	FNR (%)↓
CNN. CAM	47.8	22.6	30.1
CNN. USAGE w/o REG	49.6	22.9	27.2
CNN. USAGE	57.7	20.4	22.6
Trans. CAM	52.3	28.4	19.1
Trans. USAGE w/o REG	64.0	20.3	15.5
Trans. USAGE	**67.7**	**17.8**	**14.7**

Table 1. Analysis of different seed areas computed from the CNN and the Transformer on the PASCAL VOC 2012 *train* set [11].

gation, *e.g.*, PSA [1]; 3) Training a segmentation network based on pseudo masks which can be used to perform segmentation on a test image. We mainly focus on the first step, *i.e.*, seed area generation, since the quality of seed areas can directly influence succeeding steps.

We provide a critical review of CAM [28] in seed area generation for WSSS regarding to different types of network backbones, *i.e.*, CNN and Transformer and reveal that how it suffers from problematic activations.

3.1. Preliminary Background

We begin by introducing the learning paradigm of seed area generation. Formally, let $\mathcal{C}$ be a pre-defined category label set, given an input image $\mathbf{I}$ with its image-level label $\mathbf{y} \in \{0,1\}^{|\mathcal{C}|}$, existing methods first train a neural network $\mathcal{H} = \mathcal{F} \circ \mathcal{G}$, where a feature extractor $\mathcal{F}$ maps $\mathbf{I}$ to a dense feature map $\mathbf{A} = \mathcal{F}(\mathbf{I}) \in \mathbb{R}^{W \times H \times D}$, and a classifier $\mathcal{G}$, parameterized with $\mathbf{w}$, maps $\mathbf{A}$ to a score vector over pre-defined categories $\mathbf{s} = \mathcal{G}(\mathbf{A}, \mathbf{w}) \in \mathbb{R}^{|\mathcal{C}|}$. $\mathcal{F}$ can be any type of neural networks, *e.g.,* CNN or Transformer. Accordingly, the objective function of seed area generation is formulated as:

$$\begin{aligned}(\hat{\mathbf{A}}, \hat{\mathbf{w}}) &= \underset{\mathbf{A},\mathbf{w}}{\operatorname{argmin}}\, \mathcal{L}(\mathbf{y}, \mathbf{s}) \\ &= \underset{\mathbf{A},\mathbf{w}}{\operatorname{argmin}}\, \frac{1}{|\mathcal{C}|} \sum_{c} \mathcal{L}^{c}_{\mathrm{CE}}\left(y^c, \sigma\left(s^c\right)\right),\end{aligned} \tag{1}$$

which defines a cross entropy (CE) loss function for the mapping between features and image-level labels. s^c and y^c denote the score and image-level label of class c, respectively, and σ is the sigmoid function. During inference, the seed area $\mathbf{M}^c$ for each class c is obtained as:

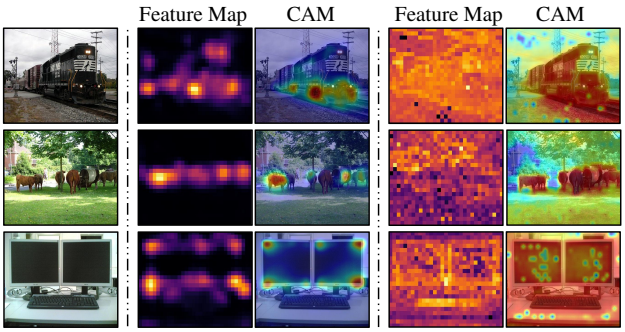
$$M^c_{ij} = \sum_{d=1}^{D} \hat{w}^c_d \hat{A}^d_{ij}, \tag{2}$$

where $(\hat{A})^d_{i,j}$ is the d-th dimension feature at position (i,j), $\hat{w}^c_d$ is the weight of the classifier corresponding to class c for feature dimension d, and M^c_{ij} is the value of the seed area $\mathbf{M}^c$ at position (i,j).

CAM [28] is the *de facto* paradigm for seed area generation, which adopts a global average pooling (GAP) and a fully-connected layer as the classifier. The mapping of the classifier $\mathcal{G}(\cdot,\cdot)$ is specified as:

$$\begin{aligned}s^c &= \frac{1}{N} \sum_{d=1}^{D} w^c_d \sum_{i=1}^{W} \sum_{j=1}^{H} A^d_{ij} \\ &= \frac{1}{N} \sum_{i=1}^{W} \sum_{j=1}^{H} \sum_{d=1}^{D} w^c_d A^d_{ij}.\end{aligned} \tag{3}$$

where N is the spatial size (= width W × height H) of the feature map $\mathbf{A}$.



(a) Image (b) CNN (c) Transformer

Figure 2. Comparison of CAMs and feature maps generated from CNNs and Transformers. (a) Input Image. (b) Feature maps and CAMs computed from a CNN. (c) Feature maps and CAMs computed from a Transformer. The above feature maps are averaged along the channel dimension.

3.2. Under-activation and Over-activation

To determine whether a seed area is over- or under-activated, we first give some metrics derived from the classical definition of false positive and false negative rates.

$$\mathrm{FPR} = \mathrm{FP}/(\mathrm{T{+}FP}), \mathrm{FNR} = \mathrm{FN}/(\mathrm{P{+}FN}), \tag{4}$$

where the terms FPR and FNR correspond to false positive rate and false negative rate *w.r.t.* a target class, respectively. T and P refer to true predictions and positive predictions for the target class, respectively. Meanwhile, FP and FN denote false positive predictions and false negative predictions for that class, respectively. Intuitively, a higher FNR value implies that more foreground regions are not activated, indicating under-activation. Conversely, a higher FPR value indicates more background regions are mistakenly activated as seed areas, signifying over-activation. Besides, the mean intersection-over-union (mIoU) is used as the typical metric for segmentation.

Table 1 shows quantitative results with three metrics. Compared to the CAM-based seed area generated from the CNN, the CAM-based seed area generated from the Transformer improves mIoU by 3.5%, which reveals the Transformer is a more powerful alternative network for

WSSS. Besides, our proposed two metrics verify that the Transformer can help the CAM-based seed area to alleviate under-activation by 11.0% FNR, compared with the CNN. However, the CAM-based seed area generated from the Transformer also suffered from over-activation, which increases FPR by 5.8%. Furthermore, we show some qualitative results of CAMs in Fig. 2. It reveals that CAMs generated from CNNs and Transformers are under- and over-activated, respectively.

3.3. Observation and Analysis

We can observe that the feature characteristics of different networks would directly influence the shapes of CAM-based seed areas (over-activated or under-activated) by comparing the visualizations in both Fig. 1 and Fig. 2. Specifically, the feature maps generated from the Transformer are more likely to involve more global context information. Meanwhile, the irrelevant information would not be suppressed from the following classifier, since the GAP follows the principle that each location on a feature map is capable of contributing to the classification results equally. Therefore, during inference, the seed area is unable to distinguish irrelevant context and the region of a target class, leading to the over-activation problem. In contrast, the CNN features capture local context, leading to under-activation.

According to these experimental results, we observe that the core issue of the current seed area generation paradigm is that feature characteristics directly influence the shapes of seed areas. Thus, if we can design a mechanism to prevent the negative effects of the feature characteristics from both CNNs and Transformers, it would be capable of controlling the activation of seed areas in a unified paradigm.

4. Unified Seed Area Generation

In this section, we introduce the proposed **U**nified optimization paradigm for **S**eed **A**rea **GE**neration (USAGE). USAGE is motivated by previous analysis, in which the objective function consists of two terms: a generation loss $\mathcal{L}_{\text{GEN}}$ to make seed areas adapt to different types of networks and a regularization loss $\mathcal{L}_{\text{REG}}$ to overturn falsely activation seed areas, as shown in Fig. 3. The core of the generation loss is to introduce a *spatial activation distribution* $\boldsymbol{\alpha} \in \mathbb{R}^{W\times H\times|\mathcal{C}|}$ to realize the shape of seed areas, which explicitly indicate the influence at each spatial location on the feature map contributed to the final classification result. By controlling the $\boldsymbol{\alpha}$ via smoothing or sharpening, the seed areas can be adaptive to the feature characteristics of different networks. Moreover, a regularization loss $\mathcal{L}_{\text{REG}}(\boldsymbol{\alpha})$ is employed to regularize the seed areas.

The objective function of USAGE is formulated as:

$$\begin{aligned}(\hat{\mathbf{A}}, \hat{\mathbf{w}}) &= \underset{\mathbf{A},\mathbf{w}}{\operatorname{argmin}}\, \mathcal{L}_{\text{GEN}}(\mathbf{y}, \tilde{\mathbf{s}}) + \lambda \mathcal{L}_{\text{REG}}(\boldsymbol{\alpha}) \\ &= \underset{\mathbf{A},\mathbf{w}}{\operatorname{argmin}}\, \frac{1}{|\mathcal{C}|}\sum_{c} \mathcal{L}^{c}_{\text{CE}}(y^c, \sigma(\tilde{s}^c)) + \lambda \mathcal{L}_{\text{REG}}(\boldsymbol{\alpha}),\end{aligned} \tag{5}$$

where λ is a coefficient and

$$\tilde{s}^c = \frac{\sum_{i=1}^{W}\sum_{j=1}^{H}\alpha^c_{ij}\sum_{d=1}^{D} w^c_d A^d_{ij}}{\sum_{i=1}^{W}\sum_{j=1}^{H}\alpha^c_{ij}}. \tag{6}$$

Intuitively, the activation value $\sum_{d=1}^{D} w^c_d A^d_{ij}$ is an off-the-shelf prior for indicating the distribution of each location on the feature map during training. For simplicity, we employ it to compose the spatial activation distribution α^c_{ij}. Moreover, we argue that α^c_{ij} should also be normalized along with the category channel since one location should only contribute to one class. In light of these clues, the spatial activation distribution $\boldsymbol{\alpha}$ is obtained by applying a softmax function to the activation value $\sum_{c=1}^{D} w^c_d A^d_{ij}$:

$$\alpha^c_{ij} = \frac{\exp\left(\sum_{d=1}^{D} w^c_d A^d_{ij}\right)}{\sum_{c=1}^{|\mathcal{C}|+1}\exp\left(\sum_{d=1}^{D} w^c_d A^d_{ij}\right)}. \tag{7}$$

We add an additional background channel with a constant value for rationality since not all locations are covered by foreground classes.

4.1. Activation Shape Controlling

To cope with different types of networks, we integrate the temperature scaling [13] to control the spatial activation distribution. To this end, a simple modification to the mapping function (Eq. 6) is made as follows:

$$\tilde{s}^c = \frac{\sum_{i=1}^{W}\sum_{j=1}^{H}(\alpha^c_{ij})^{1/\tau_1}\sum_{d=1}^{D} w^c_d A^d_{ij}}{\sum_{i=1}^{W}\sum_{j=1}^{H}(\alpha^c_{ij})^{1/\tau_1}}, \tag{8}$$

where $\tau_1 \in \mathbb{R}_+$ denotes a temperature parameter, it can control spatial activation distribution to be sharpened or smoothed, *e.g.,* a smaller value of τ_1 encourages the network to focus on a portion of locations with higher contribution and alleviate over-activation for Transformers. In contrast, when τ_1 is nearly infinite, it will force the network to learn from features at each location, which is beneficial for CNNs.

4.2. Activation Shape Regularization

Since the spatial activation distribution $\boldsymbol{\alpha}$ is highly correlated with learned features, it lacks the ability to rectify mistakes caused by feature representations *i.e.,* overturning incorrectly activated seed areas. To solve this problem, we

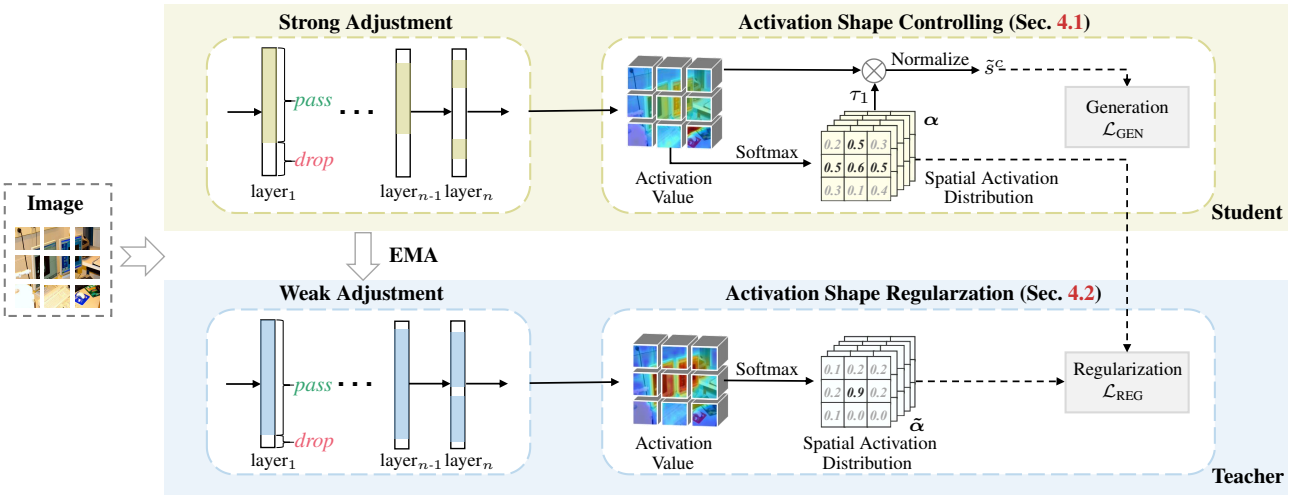


Figure 3. The overall training process of our unified seed area generation paradigm (USAGE). The objective function of USAGE includes two terms: The first term is a generation loss which optimizes a mapping function between features and image-level labels controlled by the spatial activation distribution. The second term is a regularization loss to ensure the consistency between spatial activation distributions from two views. The two views are computed from a teacher network and a student network, respectively. The teacher and student networks are obtained by making different adjustments to the architecture of a classification network.

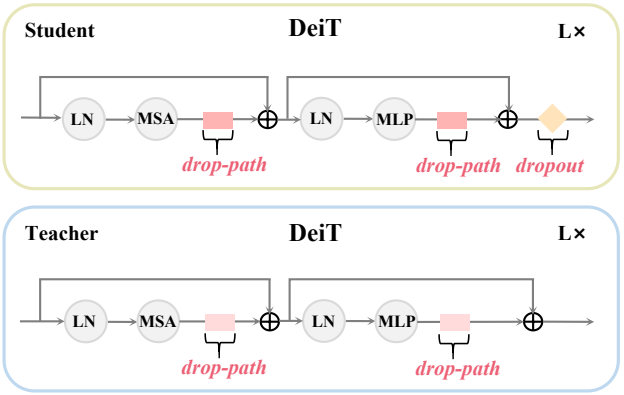


Figure 4. The diagram of how to make adjustment to the architecture of a classification network (DeiT [42]) to form the student network and the teacher network. Two adjustment operation are used, *i.e.,* dropout [38] and drop-path [21].

apply a regularization loss to regularize the spatial activation distribution to encourage semantic consistency between two views.

Self-adaptive Network Adjustment. The different views here are obtained by making different adjustments to the architecture of the classification network, where the adjustment is achieved by applying dropout [38] and drop-path [21] to the architecture (the detailed design is shown in Fig. 4 as an example). Specifically, we adopt a strong adjustment and a weak adjustment to the architecture, resulting in a student network and a teacher network, respectively. The weights of the teacher are updated from the student using Exponential Moving Average (EMA) [22]. However, maintaining a fixed adjustment gap between the teacher and student models may overlook the varying learning difficulties across different stages of training. To address this, we propose a self-adaptive network adjustment scheme that can dynamically increase the adjustment gap based on the current learning status of the models. We estimate the current learning status as the EMA of the classification score at each training time step, as inspired by recent work [44]. We use the cross entropy loss function to measure the consistency between the spatial activation distribution $\boldsymbol{\alpha}_{ij}$ from the student network and the spatial activation distribution $\tilde{\boldsymbol{\alpha}}_{ij}$ from the teacher network. The regularization term is specified as:

$$\mathcal{L}_{\text{REG}}\left(\boldsymbol{\alpha}\right) = -\tau_2^2 \frac{1}{N} \sum_{i=1}^{W} \sum_{j=1}^{H} (\tilde{\boldsymbol{\alpha}}_{ij})^{1/\tau_2} \log \left(\boldsymbol{\alpha}_{ij}\right)^{1/\tau_2}, \quad (9)$$

where $\tau_2 \in \mathbb{R}_+$ is a temperature parameter.

4.3. Relationship to Other Variants

In this subsection, we analyze the relationship between USAGE and other seed area generation approaches, including cross-view regularization by geometric transform, multiple instance learning and class-to-patch attention mapping. We show that all of them are special cases of USAGE and conduct throughout comparison with them in Sec. 5.3.

Cross-view Regularization by geometric Transform. Most previous WSSS methods [19, 45] cross-view regularization by geometric transform, *i.e.*, applying different manually pre-defined geometric transforms to input images

to construct multiple views. Their objective function for seed area generation has the same form as ours, *i.e.*, Eq. 5, but they constructed the mapping of the classifier $\mathcal{G}(\cdot,\cdot)$ by Eq. 3 rather than Eq. 8. Without the activation shape controlling ability provided by Eq. 8, their methods are easily influenced by the feature characteristics of different networks, and thus suffer from problematic activations. Besides, we also show that our instantiation for regularization, *i.e.*, cross-view regularization by architecture adjustment is more effective for seed area generation from Transformers (Sec. 5.3).

Multiple Instance Learning. In the early stage, *i.e.,* prior to the popularization of the CAM, pioneer WSSS methods used multiple instance learning (MIL) [30] to mine seed areas. For example, Pinheiro *et al.* [33] instantiated the mapping of the classifier $\mathcal{G}(\cdot,\cdot)$ as a MIL problem by leveraging Eq. 6, and computed the objective function following Eq. 1. However, since Eq. 6 lacks activation shape controlling as well as Eq. 1 lacks the regularization term, MIL-based methods are easily affected by different feature characteristics, which leads to problematic activations.

Class-to-patch Attention Mapping. As mentioned in Sec. 2, Xu *et al.* [51] proposed to generate seed areas from Transformers based on the class-to-patch attention map rather than the CAM [28]. They used Eq. 1 as the objective function for seed area generation and instantiated the mapping of the classifier $\mathcal{G}(\cdot,\cdot)$ by:

$$\hat{s}^c = \frac{\sum_{i=1}^{W}\sum_{j=1}^{H}\beta_{i,j}^c \sum_{d=1}^{D}\frac{1}{D}\mathcal{P}(A_{ij}^d)}{\sum_{i=1}^{W}\sum_{j=1}^{H}\beta_{i,j}^c}, \tag{10}$$

where

$$\beta_{i,j}^c = w_d^c A_{i,j}^d, \tag{11}$$

and $\mathcal{P}(\cdot)$ is a multilayer perceptron (MLP) layer. Eq. 10 shows that [51] directly used the activation value to indicate the influence at each spatial location rather than a normalized distribution. Since they neither explicitly controlled the shape of activation (Eq. 10) nor involved the regularization term (Eq. 1), the seed areas generated from Transformers are inevitably over-activated (see the second column of Fig. 5).

5. Experiments

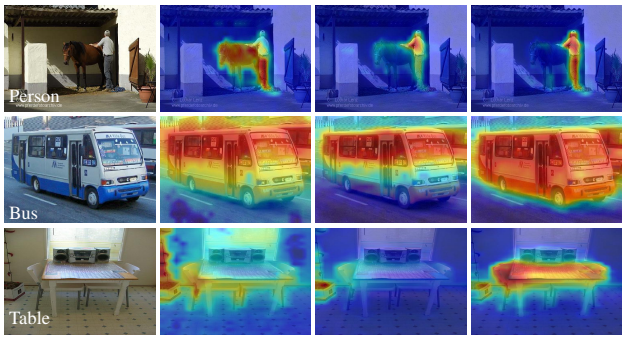
In this section, we first describe the experimental setup and implementation details (Sec. 5.1). Then, we compare our method with state-of-the-art weakly supervised semantic segmentation methods (Sec. 5.2). Finally, we conduct an ablation study to show the contribution of each component in our method (Sec. 5.3).

5.1. Experimental Settings

Datasets. We evaluate the proposed approach on two datasets, *i.e.*, PASCAL VOC 2012 [11] and MS COCO 2014 [29]. **PASCAL VOC** has three subsets, *i.e.*, training (train), validation (val) and test sets, each of which contains 1,464, 1,449, and 1,456 images, respectively. It has 20 object classes and one background class for the semantic segmentation task. Following [24, 53, 19, 52], an augmented set of 10,582 images, with additional data from [14], is used for training. **MS COCO** contains 80 object classes and one background class for semantic segmentation. Its training and validation sets contain 80K and 40K images, respectively.

Method	Seed	Mask
Chang *et al.* (CVPR20) [5]	50.9	63.4
SEAM (CVPR20) [45]	55.4	63.6
AdvCAM (CVPR21) [25]	55.6	68.0
CDA (ICCV21) [39]	58.4	66.4
Zhang *et al.*(ICCV21) [53]	57.4	67.8
RIB (NeurIPS21) [23]	56.5	68.6
SIPE (CVPR22) [7]	58.6	-
ReCAM (CVPR22) [8]	56.6	-
CLIMS (CVPR22) [49]	56.6	70.5
MCTformer (CVPR22) [51]	61.7	69.1
ViT-PCM (ECCV22) [35]	63.6	67.1
USAGE (Ours)	**67.7**	**72.8**

Table 2. Evaluation of the seed area (Seed) and the corresponding pseudo mask (Mask) refined by PSA [1] in terms of mIoU (%) on the PASCAL VOC 2012 [11] *train* set.



(a) Image (b) MCTformer (c) USAGE w/o REG (d) USAGE

Figure 5. Seed area visualization. (a) Input image. (b) Seed area generated in MCTformer [51]. (c) USAGE (Ours) without using activation shape regularization. (d) USAGE (Ours).

Evaluation metrics. Following prior works [51], we use the mIoU to evaluate the semantic segmentation performance on the *val* set, of the two benchmarks. We obtained the semantic segmentation results on the PASCAL VOC *test* set from the official evaluation server.

Implementation details. We perform our USAGE on two types of networks, *i.e.,* Transformer and CNN. For Transformer, following [51], we use the DeiT-S [42] pre-trained on ImageNet [9] as our network. We followed the data aug-

Method	Seg.Back.	Sup.	Val	Test
SeeNet (NeurIPS18) [16]	ResNet38	I+S	63.1	62.8
Sun *et al.* (ECCV20) [40]	ResNet101	I+S	66.2	66.9
EPS (CVPR21) [26]	ResNet101	I+S	70.9	70.8
AuxSegNet (ICCV21) [50]	ResNet38	I+S	69.0	68.6
ESOL (NeurIPS22) [27]	ResNet101	I+S	71.1	70.4
ReCAM (CVPR22) [25]	ResNet101	I+S	71.8	72.2
L2G (CVPR22) [19]	ResNet101	**I+S**	**72.1**	**71.7**
Araslanov *et al.* (CVPR20) [2]	ResNet38	I	62.7	64.3
SEAM (CVPR20) [45]	ResNet38	I	64.5	65.7
BES (ECCV20) [6]	ResNet101	I	65.7	66.6
CONTA (NeurIPS20) [52]	ResNet38	I	66.1	66.7
AdvCAM (CVPR21) [25]	ResNet101	I	68.1	68.0
ECS-Net (ICCV21) [41]	ResNet38	I	66.6	67.6
CDA (ICCV21) [39]	ResNet38	I	66.1	66.8
Zhang *et al.* (ICCV21) [53]	ResNet38	I	67.8	68.5
AdvCAM (CVPR21) [25]	ResNet101	I	68.1	68.0
RIB (NeurIPS21) [23]	ResNet38	I	68.3	68.6
SIPE (CVPR22) [7]	ResNet38	I	68.8	69.7
CLIMS (CVPR22) [49]	ResNet101	I	70.4	70.0
MCTformer (CVPR22) [51]	ResNet38	I	71.1*	71.6
USAGE (Ours)	ResNet38	**I**	**71.9**	**72.8**

Table 3. Performance comparison of WSSS methods in terms of mIoU (%) on the PASCAL VOC 2012 [11] *val* and *test* sets using different segmentation backbones. Seg.Back.: Network backbone for segmentation. Sup.: supervision. I: image-level labels. S: Saliency maps. *: Our reimplemented results using official code.

mentation and default training parameters provided in [42]. Besides, we also employ the patch-level pairwise affinity proposed by [51] to refine the seed area without additional computations. We set τ_1 to 1 and τ_2 to 0.1. We set the drop path rate γ_t in the teacher network to 0.05. The drop path rate γ_s and the drop rate δ_s in the student network are raised from 0.05 to 0.15 and from 0 to 0.01, respectively. The λ is 0.25 and the update rate for EMA is set to 0.99. For CNN, we followed the procedure of [1], including the use of ResNet38 [48]. We set τ_1 to 50 as a way of smoothing, and τ_2 also to 0.1. To report final semantic segmentation results, we follow prior works [1, 53, 51] to use ResNet38 [48] and ResNet101 [15] as the backbones for segmentation. Notably, since the seed area computed from the Transformer achieves better performance, we adopt DeiT-S [42] as our default network for seed area generation.

5.2. Comparisons with the State-of-the-Arts

PASCAL VOC. We follow [1, 45, 51] to apply PSA [1] on the proposed seed areas (seed) to generate pseudo masks (mask) on the train set. As shown in Table 2, the proposed USAGE performs better than existing works by large margins in terms of seed area generation and pseudo mask generation, *e.g.,* improving previous SOTA [35] by 4.1%. Table 3 shows that the proposed USAGE achieves segmentation results (mIoUs) of 71.9% and 72.8% on the val and test sets, respectively. The proposed USAGE performs significantly better than all the existing methods using only image-level labels. We also note that, our results achieve similar performance with most WSSS methods which leverage saliency maps. We also show some qualitative results of the seed areas in Fig. 5 and segmentation results in Fig. 6. This indicates our USAGE is effective to generate high-quality results. Even if the context is ambiguous, our method still performs well.

Method	Seg.Back.	Sup.	Val
EPS (CVPR21) [26]	ResNet101	I+S	35.7
AuxSegNet (ICCV21) [50]	ResNet38	I+S	33.9
ESOL (NeurIPS22) [27]	ResNet101	I+S	42.6
L2G (CVPR22) [19]	ResNet101	**I+S**	**44.2**
Wang *et al.* (IJCV20) [43]	VGG16	I	27.7
SEAM (CVPR20) [45]	ResNet38	I	31.9
CONTA (NeurIPS20) [52]	ResNet38	I	32.8
CDA (ICCV21) [39]	ResNet38	I	33.2
SIPE (CVPR22) [7]	ResNet101	I	40.6
MCTformer (CVPR22) [51]	ResNet38	I	42.0
USAGE (Ours)	ResNet38	I	42.7
USAGE (Ours)	ResNet101	**I**	**44.3**

Table 4. Performance comparison of WSSS methods in terms of mIoU (%) on the MS COCO [29] *val* set.

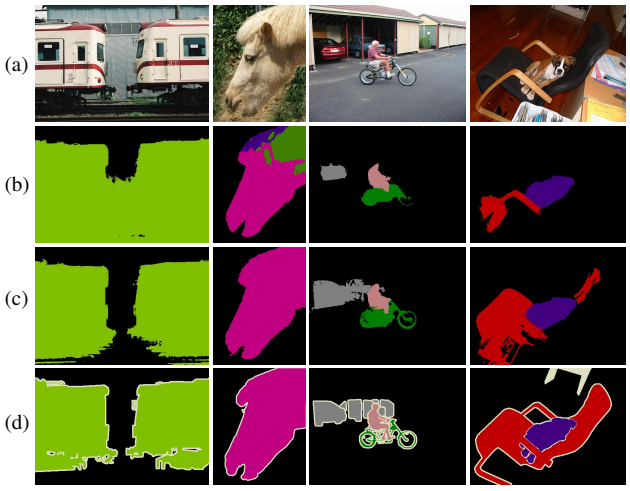


Figure 6. Qualitative segmentation results on the PASCAL VOC 2012 [11] *val* set. (a) Input image. (b) MCTformer [51]. (c) USAGE (Ours). (d) Ground-truth.

MS COCO. Table 4 shows the comparison results of our method against several existing methods on the MS COCO 2014 dataset [29]. For example, by adopting the ResNet101

Temperature τ_1	0.5	1	2	25	50	75
Trans.USAGE	*sharpening*					
mIoU (%)	67.2	**67.7**	67.1	52.3	51.4	50.3
CNN.USAGE				*smoothing*		
mIoU (%)	48.9	49.6	50.5	57.5	**57.7**	56.7

Table 5. Discussion of the temperature parameter τ_1 following the deterministic principle, *i.e.,* smoothing for CNNs or sharpening for Transformers.

as the segmentation network backbone, our approach surpasses previous methods using only image-level labels by 2.3%. In particular, our method can even achieve comparable results compared to the methods using additional saliency maps. We also offer a superior performance using the ResNet38 as the segmentation network backbone, achieving 42.7% mIoU. More qualitative results are shown in the supplementary material.

5.3. Ablation Study

Ablation of main components. Here, we do an ablation study to show the benefit brought by each component of our proposed USAGE, *i.e.,* activation shape controlling and activation shape regularization. As shown in Table 1, by introducing the activation shape controlling on the spatial activation distribution $\boldsymbol{\alpha}$, our method can obtain much better mIoUs for the seed area computed from the CNN and the Transformer (49.6% and 64.0%). Meanwhile, the activation shape controlling yields great performance improvements on the other two metrics, FPR and FNR, by 3.9% and 3.3% on average. This reveals that the activation shape controlling alleviates problematic activations caused by feature characteristics. Furthermore, the seed area results are further boosted to 57.7% and 67.7%, respectively, after integrating the activation shape regularization. As shown in the Fig. 5, we observe that the seed areas appear more complete (the first two rows) and accurate (the last row), which verifies the ability of the activation shape regularization in overturning misled seed areas.

Discussion of temperature parameter. We conduct experiments of the hyper-parameter temperature τ_1, as shown in Table 5. The results show that controlling temperature τ_1 to adapt to different types of networks is easy, as it follows a deterministic principle, *i.e.,* smoothing for CNNs or sharpening for Transformers, and the performance is robust across different temperature values.

Rates of drop path and dropout for our network adjustment. We use drop path and dropout to realize network adjustment, where whether an adjustment is strong or weak is determined by the rates of both drop path and dropout. High and low rates lead to strong and weak adjustments, respectively. If the difference between the rates for the teacher and that for the student increases, then their adjustment gap becomes larger. The "weak/strong adjustment" encourages a model to produce consistent predictions for the same pixel under different views, leading to robust localization. As shown in Table. 6, our USAGE demonstrates robustness *w.r.t.* these hyper-parameters. γ_t: drop path rate in the teacher model (default setting). γ_s: drop path rate in the student model. δ_s: drop rate in the student model.

Discussion of Self-adaptive strategy for our network adjustment. In order to control the growing gap between the student and teacher's network adjustment, we introduce a self-adaptive strategy. By adaptively increasing the adjustment gap, the student will efficiently learn to distinguish and preserve all features that are pertinent to the foreground objects. We demonstrate the effectiveness of the self-adaptive strategy by comparing it with two commonly used functions, namely "Fixed", maintaining the adjustment gap at a fixed value, and "Linear", an increase in the adjustment gap at a fixed value in a linear fashion with training progresses. As shown in Table 7, our proposed strategy achieves better performance with "Fixed" and "Linear" strategies, which indicates that the self-adaptive strategy is essential for the proposed network adjustment framework.

Variants. The objective function of our USAGE (Eq. 5) includes a classification term and a regularization term. The first term aims to optimize a mapping function, which is realized by the activation shape controlling (CM) in our USAGE according to Eq. 8. As discussed in Sec. 4.3, the mapping function can also be realized by three variants, including CAM [28], class-to-patch attention mapping [51] and MIL-based method [33]. The second term is achieved by network architecture adjustment (AA), which also can be achieved by two variants, *i.e.,* geometric transform [45] and online attention accumulation [18]. As shown in Table 8, we conduct experiments to evaluate these variants, which reveals our terms are able to achieve superior results in terms of seed area generation from both CNNs and Transformers, and performs favourably against all variants. Based on the experiment results, we have the following observations: 1) Compared to CNN, we argue that Transformer is a more effective network for seed area generation (52.3% vs. 48.7%), especially when using MIL as a type of mapping (54.3% vs. 29.3%). 2) We replace our network adjustment with geometric transform [45] for regularization and its performance dramatically drops by 5.3%. We analyze that the reason might be the regularization by geometric transform barely overturn misled seed areas, since erroneous seed areas are also capable of keeping consistent between different transformations. Online attention accumulation [18] also shows inferior performance, with performance drops by 3.6% on average. 3) Even without using our mapping (*i.e.,* CM), our AA can also bring performance gains for seed area generation (56.5% vs. 52.3%), which reveals its ability to overturn misled seed areas. We show a

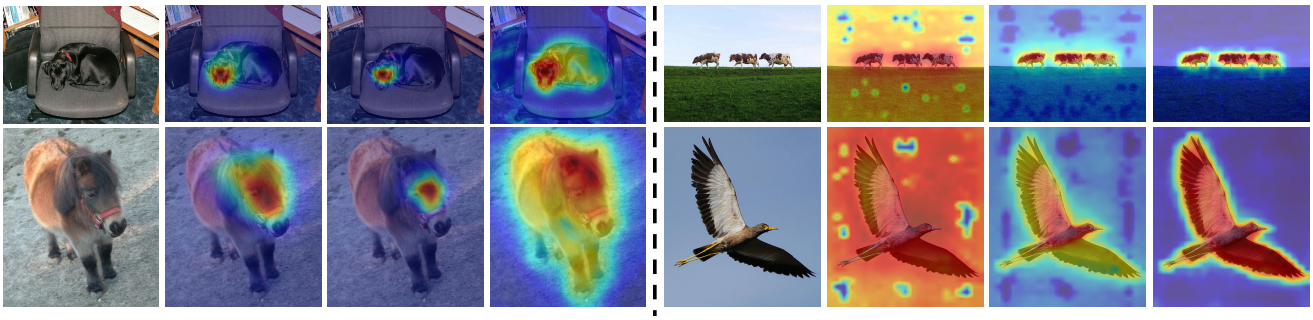


Figure 7. Seed area visualizations on the PASCAL VOC [11] *train* set. Left: Seed area generation from a CNN (ResNet38 [48]). Right: Seed area generation from a Transformer (DeiT-S [42]). MCT: Class-to-patch attention map [51]. MIL: a MIL-based method [33].

γ_t	0.05	0.05	0.05	0.05	0.05	0.05	0.05
γ_s	0.05	0.10	0.15	0.2	0.25	0.15	0.15
δ_s	0.01	0.01	0.01	0.01	0.01	0	0.02
mIoU (%)	66.4	67.2	**67.7**	67.3	66.9	67.0	66.8

Table 6. Ablation on different rates of drop path and dropout for our network adjustment, on the PASCAL VOC [11] *train* set. γ_t: drop path rate in the teacher model (default setting). γ_s: drop path rate in the student model. δ_s: drop rate in the student model.

Strategy	Fixed	Linear	Self-adaptive
		Trans.USAGE	
mIoU(%)	66.6	67.2 (+0.6)	**67.7 (+1.1)**
		CNN.USAGE	
mIoU(%)	55.5	56.9 (+1.4)	**57.7 (+2.2)**

Table 7. Discussion of the introduced self-adaptive strategy for our network adjustment, on the PASCAL VOC [11] *train* set. Here, we present three strategies that determine the change of the adjustment gap between weak and strong network adjustments. The term "Fixed" refers to a fixed value, "Linear" represents a linear function, and "Self-adaptive" indicates the self-adaptive strategy.

few qualitative results comparing seed areas among different variants in Fig. 7. More comparisons are provided in the supplementary material.

6. Conclusion

We developed a unified optimization paradigm for generating seed areas, called USAGE. The objective function of USAGE consists of two terms: a generation loss that enables seed areas to be adaptive to different types of networks via a simple temperature parameter, and a regularization loss ensures the consistency between the seed areas that are generated by self-adaptive network adjustment from different views. By incorporating these two terms, USAGE can solve problematic activations of seed areas for both CNNs and Transformers. Experimental results demonstrated that

Mapping				Regularization				mIoU (%)
CAM	MCT	MIL	CM	w/o	GT	OAA	AA	
				Seed area generation from CNN				
✓				✓				48.7
		✓		✓				29.3
✓						✓		53.9
✓					✓			55.4
			✓				✓	**57.7**
				Seed area generation from Transformer				
✓				✓				52.3
	✓							55.2
		✓		✓				54.3
			✓		✓			62.4
✓						✓		65.1
✓							✓	56.5
			✓				✓	**67.7**

Table 8. Comparison of the variants of the USAGE on the PASCAL VOC 2012 [11] *train* set. MCT: Class-to-patch attention map [51]. MIL: MIL-based method [33]. GT: cross-view regularization by geometric transform [45]. CM: Activation shape controlling based mapping function in our USAGE. OAA: cross-view regularization by online attention accumulation [18]. AA: Activation shape regularization by architecture adjustment in our USAGE. Mapping: The mapping function optimized by the first term in Eq. 5. Regularization: The second term in Eq. 5.

our USAGE could bring continuous performance gains on seed area generation and achieved state-of-the-art results on PASCAL VOC and COCO datasets.

Limitation. Throughout this paper, we investigated the step-wise pipeline of WSSS, while the end-to-end pipeline that often produces inferior results remains uncovered. In the future, we look forward to generalizing the idea of USAGE to improve the end-to-end WSSS algorithms.

Acknowledge This work was supported by NSFC 62176159, Natural Science Foundation of Shanghai 21ZR1432200, Shanghai Municipal Science and Technology Major Project 2021SHZDZX0102, and the Fundamental Research Funds for the Central Universities.

References

[1] Jiwoon Ahn and Suha Kwak. Learning pixel-level semantic affinity with image-level supervision for weakly supervised semantic segmentation. In *CVPR*, 2018. 3, 6, 7

[2] Nikita Araslanov and Stefan Roth. Single-stage semantic segmentation from image labels. In *CVPR*, 2020. 2, 7

[3] Wonho Bae, Junhyug Noh, and Gunhee Kim. Rethinking class activation mapping for weakly supervised object localization. In *ECCV*, 2020. 1

[4] Stephen Boyd, Stephen P Boyd, and Lieven Vandenberghe. *Convex optimization*. Cambridge university press, 2004. 2

[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. 2, 6

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

[7] Qi Chen, Lingxiao Yang, Jian-Huang Lai, and Xiaohua Xie. Self-supervised image-specific prototype exploration for weakly supervised semantic segmentation. In *CVPR*, 2022. 6, 7

[8] Zhaozheng Chen, Tan Wang, Xiongwei Wu, Xian-Sheng Hua, Hanwang Zhang, and Qianru Sun. Class re-activation maps for weakly-supervised semantic segmentation. In *CVPR*, 2022. 6

[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. 6

[10] Ye Du, Zehua Fu, Qingjie Liu, and Yunhong Wang. Weakly supervised semantic segmentation by pixel-to-prototype contrast. In *CVPR*, 2022. 1

[11] Mark Everingham, Luc Van Gool, Christopher KI Williams, John Winn, and Andrew Zisserman. The pascal visual object classes (voc) challenge. *IJCV*, 88(2):303–338, 2010. 2, 3, 6, 7, 9

[12] Junsong Fan, Zhaoxiang Zhang, Chunfeng Song, and Tieniu Tan. Learning integral objects with intra-class discriminator for weakly-supervised semantic segmentation. In *CVPR*, 2020. 1

[13] Chuan Guo, Geoff Pleiss, Yu Sun, and Kilian Q Weinberger. On calibration of modern neural networks. In *ICML*, 2017. 4

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

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

[16] Qibin Hou, PengTao Jiang, Yunchao Wei, and Ming-Ming Cheng. Self-erasing network for integral object attention. In *NeurIPS*, 2018. 2, 7

[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. 1

[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, 8, 9

[19] Peng-Tao Jiang, Yuqi Yang, Qibin Hou, and Yunchao Wei. L2g: A simple local-to-global knowledge transfer framework for weakly supervised semantic segmentation. In *CVPR*, 2022. 2, 5, 6, 7

[20] Alexander Kolesnikov and Christoph H Lampert. Seed, expand and constrain: Three principles for weakly-supervised image segmentation. In *ECCV*, 2016. 1, 2

[21] Gustav Larsson, Michael Maire, and Gregory Shakhnarovich. Fractalnet: Ultra-deep neural networks without residuals. *arXiv preprint arXiv:1605.07648*, 2016. 5

[22] AJ Lawrance and PAW Lewis. An exponential moving-average sequence and point process (ema1). *Journal of Applied Probability*, 14(1):98–113, 1977. 5

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

[24] Jungbeom Lee, Eunji Kim, Sungmin Lee, Jangho Lee, and Sungroh Yoon. Ficklenet: Weakly and semi-supervised segmentation using stochastic inference. In *CVPR*, 2019. 6

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

[26] 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. 7

[27] Jinlong Li, Zequn Jie, Xu Wang, Xiaolin Wei, and Lin Ma. Expansion and shrinkage of localization for weakly-supervised semantic segmentation. *NeurIPS*, 2022. 7

[28] 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, 3, 6, 8

[29] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. Microsoft coco: Common objects in context. In *ECCV*, 2014. 2, 6, 7

[30] Oded Maron and Tomás Lozano-Pérez. A framework for multiple-instance learning. *NIPS*, 1997. 6

[31] Deepak Pathak, Evan Shelhamer, Jonathan Long, and Trevor Darrell. Fully convolutional multi-class multiple instance learning. *arXiv preprint arXiv:1412.7144*, 2014. 2

[32] Pedro O Pinheiro and Ronan Collobert. From image-level to pixel-level labeling with convolutional networks. In *CVPR*, 2015. 2

[33] Pedro O Pinheiro and Ronan Collobert. From image-level to pixel-level labeling with convolutional networks. In *CVPR*, 2015. 6, 8, 9

[34] Yao Qin, Chiyuan Zhang, Ting Chen, Balaji Lakshminarayanan, Alex Beutel, and Xuezhi Wang. Understanding and improving robustness of vision transformers through patch-based negative augmentation. In *Advances in Neural Information Processing Systems*, 2021. 2

[35] Simone Rossetti, Damiano Zappia, Marta Sanzari, Marco Schaerf, and Fiora Pirri. Max pooling with vision transformers reconciles class and shape in weakly supervised semantic segmentation. In *ECCV*, 2022. 1, 6, 7

[36] Lixiang Ru, Yibing Zhan, Baosheng Yu, and Bo Du. Learning affinity from attention: End-to-end weakly-supervised semantic segmentation with transformers. In *CVPR*, 2022. 2

[37] Krishna Kumar Singh and Yong Jae Lee. Hide-and-seek: Forcing a network to be meticulous for weakly-supervised object and action localization. In *ICCV*, 2017. 1

[38] Nitish Srivastava, Geoffrey Hinton, Alex Krizhevsky, Ilya Sutskever, and Ruslan Salakhutdinov. Dropout: a simple way to prevent neural networks from overfitting. *The journal of machine learning research*, 15(1):1929–1958, 2014. 5

[39] Yukun Su, Ruizhou Sun, Guosheng Lin, and Qingyao Wu. Context decoupling augmentation for weakly supervised semantic segmentation. In *ICCV*, 2021. 6, 7

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

[41] Kunyang Sun, Haoqing Shi, Zhengming Zhang, and Yongming Huang. Ecs-net: Improving weakly supervised semantic segmentation by using connections between class activation maps. In *ICCV*, 2021. 7

[42] Hugo Touvron, Matthieu Cord, Matthijs Douze, Francisco Massa, Alexandre Sablayrolles, and Hervé Jégou. Training data-efficient image transformers & distillation through attention. In *ICML*, 2021. 1, 5, 6, 7, 9

[43] Xiang Wang, Sifei Liu, Huimin Ma, and Ming-Hsuan Yang. Weakly-supervised semantic segmentation by iterative affinity learning. *IJCV*, 128(6):1736–1749, 2020. 7

[44] Yidong Wang, Hao Chen, Qiang Heng, Wenxin Hou, Marios Savvides, Takahiro Shinozaki, Bhiksha Raj, Zhen Wu, and Jindong Wang. Freematch: Self-adaptive thresholding for semi-supervised learning. *arXiv preprint arXiv:2205.07246*, 2022. 5

[45] Yude Wang, Jie Zhang, Meina Kan, Shiguang Shan, and Xilin Chen. Self-supervised equivariant attention mechanism for weakly supervised semantic segmentation. In *CVPR*, 2020. 2, 5, 6, 7, 8, 9

[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 *CVPR*, 2017. 2

[47] 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 *CVPR*, 2018. 2

[48] 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. 1, 7, 9

[49] Jinheng Xie, Xianxu Hou, Kai Ye, and Linlin Shen. Clims: Cross language image matching for weakly supervised semantic segmentation. In *CVPR*, 2022. 6, 7

[50] Lian Xu, Wanli Ouyang, Mohammed Bennamoun, Farid Boussaid, Ferdous Sohel, and Dan Xu. Leveraging auxiliary tasks with affinity learning for weakly supervised semantic segmentation. In *ICCV*, 2021. 7

[51] Lian Xu, Wanli Ouyang, Mohammed Bennamoun, Farid Boussaid, and Dan Xu. Multi-class token transformer for weakly supervised semantic segmentation. In *CVPR*, 2022. 1, 2, 6, 7, 8, 9

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

[53] Fei Zhang, Chaochen Gu, Chenyue Zhang, and Yuchao Dai. Complementary patch for weakly supervised semantic segmentation. In *ICCV*, 2021. 2, 6, 7