

CLIMS: Cross Language Image Matching for Weakly Supervised Semantic SegmentationJinheng Xie[†], Xianxu Hou[†], Kai Ye, Linlin Shen^{*}

School of Computer Science & Software Engineering, Shenzhen University, China

Shenzhen Institute of Artificial Intelligence of Robotics of Society, Shenzhen, China

Guangdong Key Laboratory of Intelligent Information Processing, Shenzhen University, China

National Engineering Laboratory for Big Data System Computing Technology, Shenzhen University, China

{xiejinheng,yekai}2020@email.szu.edu.cn, hxianxu@gmail.com, llshen@szu.edu.cn

Abstract

It has been widely known that CAM (Class Activation Map) usually only activates discriminative object regions and falsely includes lots of object-related backgrounds. As only a fixed set of image-level object labels are available to the WSSS (weakly supervised semantic segmentation) model, it could be very difficult to suppress those diverse background regions consisting of open set objects. In this paper, we propose a novel Cross Language Image Matching (CLIMS) framework, based on the recently introduced Contrastive Language-Image Pre-training (CLIP) model, for WSSS. The core idea of our framework is to introduce natural language supervision to activate more complete object regions and suppress closely-related open background regions. In particular, we design object, background region and text label matching losses to guide the model to excite more reasonable object regions for CAM of each category. In addition, we design a co-occurring background suppression loss to prevent the model from activating closely-related background regions, with a predefined set of class-related background text descriptions. These designs enable the proposed CLIMS to generate a more complete and compact activation map for the target objects. Extensive experiments on PASCAL VOC2012 dataset show that our CLIMS significantly outperforms the previous state-of-the-art methods. Code will be available at <https://github.com/CVI-SZU/CLIMS>.

1. Introduction

Semantic segmentation attempts to assign each pixel in an image with a semantic label. Though fully supervised semantic segmentation has achieved remarkable success in recent years, pixel-level annotation is significantly time-

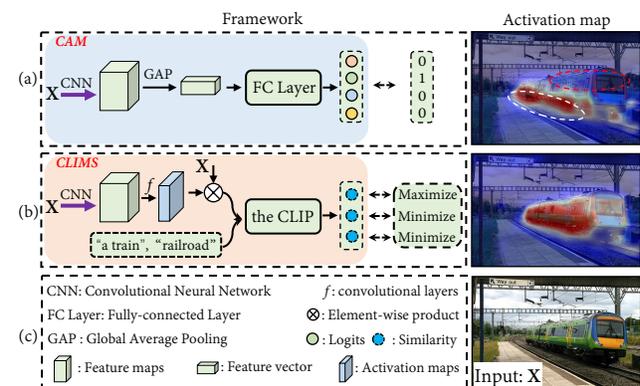


Figure 1. (a) Conventional CAM solution. (b) The proposed CLIMS. The problem of false-activation of irrelevant background, e.g., railroad and ground, and underestimation of object contents usually exist in conventional CAM method. To solve this problem, we propose a novel text-driven learning framework, CLIMS, which introduces natural language supervision, i.e., an open-world setting, for exploring complete object contents and excluding irrelevant background regions. Best viewed in color.

consuming and labor-intensive. Instead, weakly supervised semantic segmentation (WSSS) tries to mitigate this issue by relying solely on image-level [2, 15, 23, 24], bounding box-level [8, 22], point-level [3], or scribble-based supervision [20, 29]. This work aims to only use image-level labels in the learning of a semantic segmentation model.

Existing WSSS approaches typically follow a three-stage learning process. First, the image-level labels are used as supervision at the feature level to train a classification network to generate initial class activation maps (CAMs) (as shown in the left of Fig. 1(a)). Then the initial CAMs are refined as pseudo ground-truth masks using dense CRF [16], pixel affinity-based methods [1, 2], or additional saliency maps [12, 19, 21]. Finally, the refined pseudo ground-truth masks are used to further train a segmentation network. However, as only a fixed set of object categories is available during the first stage training, i.e., a close-world setting, class-related background pixels, e.g., railroad, also

*Corresponding Author

[†]Equal Contribution

contribute to the prediction of closely related objects, e.g., train. This results in unnecessary activation of background in the generation of initial CAMs, as shown in the right of Fig. 1(a). Besides, conventional CAM solution usually struggles in the underestimation of object contents. Both of them severely limit the quality of initial CAMs for the next two stages.

In this paper, we design a novel Cross Language Image Matching framework for WSSS, i.e., CLIMS, based on the power of recently introduced Contrastive Language-Image Pre-training (CLIP) [25] model, to address the aforementioned issues. The CLIP model is pretrained from scratch on a dataset of 400 million image-text pairs (automatically collected from the publicly available sources on the Internet), which enables CLIP to associate much wider visual concepts in the image with their text labels in an open-world setting, rather than a fixed set of predetermined object categories. Based on this, the proposed CLIMS has great potentials to generate a high-quality initial activation map for each object category without irrelevant background (as shown in the right of Fig. 1(c)).

In the left of Fig. 1(a), the conventional CAM method performs image-level supervision on the average feature after the global average pooling (GAP) layer. Given the trained model, the class activation maps (CAMs) can be extracted. However, the CLIP model cannot be directly used in this pipeline. Instead, as shown in Fig. 1(b), we replace the GAP and fully connected (FC) layer with convolutional layers to directly generate an activation map for each class under the supervision from CLIP model, where natural language can be used to guide the model for the activation maps generation.

Details of the proposed CLIMS are depicted in Fig. 2. It mainly consists of a backbone network and a text-driven evaluator including three CLIP-based loss functions, i.e., Object region and Text label Matching loss (\mathcal{L}_{OTM}), Background region and Text label Matching loss (\mathcal{L}_{BTM}), and Co-occurring Background Suppression loss (\mathcal{L}_{CBS}). The core idea is to complementarily learn the generation of initial CAMs \mathbf{P} through the supervision of the text-driven evaluator. First, given an image, the backbone network predicts the initial CAMs \mathbf{P} , which represents the probability of each pixel belonging to a category. \mathbf{P} and $(1-\mathbf{P})$ are then multiplied with the input image to mask out the object and background area, respectively, which serve as the input of the text-driven evaluator. As shown in Fig. 2(b), each mask-out area and its corresponding text category label are passed to the CLIP model to calculate their cosine similarity. The text labels for foreground objects, e.g., “train”, “cat” and “person”, etc, are defined according to the dataset. During training, \mathcal{L}_{OTM} aims to maximize the similarity between the foreground object area and the given text label, e.g., “a photo of train”. In this way, though the generated CAMs

can gradually approach the target object in the image, the completeness of the regions is not guaranteed. For example, the image still looks like a bird even when only head of the bird is visible. Thus, we propose \mathcal{L}_{BTM} to minimize the similarity between mask-out foreground area $\mathbf{X} \cdot (1-\mathbf{P})$ and the same object category text label as \mathcal{L}_{OTM} . This excludes object regions out of $(1-\mathbf{P})$ and recovers more probable object contents in \mathbf{P} . In addition, to constrain the size of activated regions, we design a regularization term to ensure the compactness of \mathbf{P} . However, when the object area is activated, the background closely related to the object, e.g., train and railroad, boat and river, etc, will usually be activated as well, as no pixel-level labels are available. To address this issue, we additionally define a set of class-related background text labels, such as “railroad” (background of the train) and “river” (background of the boat), etc. Based on these text labels and the CLIP model, we design \mathcal{L}_{CBS} to minimize the similarity between the mask-out object area $\mathbf{X} \cdot \mathbf{P}$ and these co-occurring background text labels. This enables CLIMS to exclude the irrelevant class-related background, e.g., railroad, out of the initial CAMs \mathbf{P} . These CLIP-based loss functions work complementarily to refine the generated initial CAMs by the exploration of more complete object areas and exclusion of irrelevant background pixels.

Collectively, the main contributions of this paper can be summarized as:

- We propose a text-driven learning framework, CLIMS, to introduce image-text matching model based supervision, i.e., an open-world setting, for WSSS.
- We design three CLIP-based loss functions and an area regularization. The object, background region and matching losses ensure the correctness and completeness of initial CAMs. The co-occurring background suppression loss can further substantially mitigate the influence of class-related background. The area regularization can constrain the size of activated regions.
- Extensive experiments on PASCAL VOC2012 dataset show that the proposed CLIMS significantly outperforms the previous state-of-the-art methods.

2. Related Work

In this section, we will first review existing weakly supervised semantic segmentation methods according to the three-stage learning process of WSSS: initial CAMs generation, CAMs refinement techniques, and segmentation network training. In addition, we will briefly discuss the CLIP model, which serves as the motivation for this work.

Weakly supervised semantic segmentation. The pipeline of the conventional CAM [38] solution has been used in the majority of previous WSSS works. Hou *et*

al. [11] propose two self-erasing strategies for focusing attention only on the reliable regions, generating complete initial CAMs. Chang *et al.* [4] propose to investigate object sub-categories to mine more object parts and then improve the completeness of initial CAMs. Sun *et al.* [28] incorporate two neural co-attentions into the classifier for discovering shared or unshared semantics in a pair of training images. This helps to extract more complete initial CAMs from the classifier. Jungbeom *et al.* [18] propose an anti-adversarial manner to discover more regions of the target object in the activation map. Ahn and Kwak [2] design a deep neural network, called AffinityNet, to predict semantic affinity between a pair of adjacent image coordinates. This semantic affinity is then applied to refine the generated initial CAMs as pseudo ground-truth masks. Previous works [12, 19, 21, 32] instead use additional saliency maps from a fully supervised saliency detector to refine the generated initial CAMs. The series of DeepLab [5, 6] models are typically used to train a semantic segmentation network with the pseudo ground-truth masks.

Contrastive Language-Image Pre-training (CLIP). The contrastive language-image pretraining (CLIP) [25] shows great success and potential on many vision tasks in a zero-shot setting. The CLIP model consists of an image encoder and a text encoder. Given a batch of image and text pairs, the CLIP model learns the embedding to measure the similarity between image and text. The CLIP model is trained on a large dataset of 400 million image-text pairs, the set of object categories CLIP can recognize is much bigger and diverse than a fixed set of object categories in a small dataset, e.g., PASCAL VOC2012 [9]. The image-text pairs are automatically collected from the Internet, without the involvement of manual efforts.

3. Methodology

In this section, we will first review the conventional CAM pipeline and its limitations. The proposed learning framework, i.e., CLIMS, is then introduced. Finally, we will detail three CLIP-based loss functions: object region and text label matching loss, background region and text label matching loss and co-occurring background suppression loss, and an area regularization.

3.1. Revisiting the Conventional CAM

The majority of previous WSSS works follow the CAM pipeline to generate an initial activation map for each category in the image. Given an input image \mathbf{X} and corresponding image-level label $y \in \mathbb{R}^{1 \times K}$, a backbone network will first embed \mathbf{X} to high-level feature maps $\mathbf{Z} \in \mathbb{R}^{C \times H \times W}$, where K denotes the number of classes, C and $H \times W$ denote the number of channels and spatial dimension, respectively. A global average pooling (GAP) layer and a 1×1 convolution layer with learnable matrix $\mathbf{W} \in \mathbb{R}^{C \times K}$ are

then applied on \mathbf{Z} to produce prediction logits $\hat{y} \in \mathbb{R}^{1 \times K}$. During training, the sigmoid cross entropy loss is calculated as follow:

$$\mathcal{L}(\hat{y}, y) = - \sum_{k=1}^K y_k \cdot \log \sigma(\hat{y}_k) + (1 - y_k) \cdot \log(1 - \sigma(\hat{y}_k)), \quad (1)$$

where σ is the sigmoid activation function.

Given the trained backbone network, \mathbf{W} is directly applied on \mathbf{Z} to generate initial CAMs $\mathbf{P} \in \mathbb{R}^{K \times H \times W}$:

$$\mathbf{P}_k(h, w) = \mathbf{W}_k^\top \mathbf{Z}(h, w), \quad (2)$$

where $\mathbf{Z}(h, w)$ denotes the representation vector located on (h, w) . The corresponding weight vector and activation map for a given object category is \mathbf{W}_k and \mathbf{P}_k . Due to the limited supervision, conventional CAM, despite being simple and effective, may struggle with the only activation of discriminative object parts and the unnecessary activation of closely related backgrounds.

3.2. Cross Language Image Matching Framework

Fig. 2 depicts an overview of CLIMS. The backbone network illustrated in Fig. 2(a) is similar to conventional CAM solution, except that the GAP layer is removed and a sigmoid function σ is directly applied after \mathbf{W} :

$$\mathbf{P}_k(h, w) = \sigma(\mathbf{W}_k^\top \mathbf{Z}(h, w)). \quad (3)$$

While conventional WSSS methods only use the supervision of a fixed set of predetermined object categories, we propose the text-driven evaluator based on the CLIP model to explore additional object categories in the dataset. As shown in Fig. 2(b), the text-driven evaluator consists of an image encoder $f_i(\cdot)$ and a text encoder $f_t(\cdot)$ from the CLIP model. To begin with, \mathbf{P}_k and $(1 - \mathbf{P}_k)$ are multiplied by \mathbf{X} to mask out the foreground object and background pixels, respectively. The results are then mapped to representation vectors \mathbf{v}_k^{io} and \mathbf{v}_k^{ib} by $f_i(\cdot)$:

$$\mathbf{v}_k^{io} = f_i(\mathbf{X} \cdot \mathbf{P}_k), \quad \mathbf{v}_k^{ib} = f_i(\mathbf{X} \cdot (1 - \mathbf{P}_k)). \quad (4)$$

Following CLIP [25], the corresponding object text prompt \mathbf{t}_k^o for $(\mathbf{X} \cdot \mathbf{P}_k)$ is represented as “a photo of { }”, e.g., “a photo of the train”. On the contrary, the corresponding class-related background text prompts $\mathbf{t}_{k,l}^b$ are manually pre-defined as a set of L co-occurring background closely related with the object of the k -th category. For example, the class-related background of boat (k -th object) is {“a photo of river”, “a photo of a lake”}. Then $\mathbf{t}_{k,0}^b = \{$ “a photo of river”}, $\mathbf{t}_{k,1}^b = \{$ “a photo of a lake”}, the text representations can be obtained as follow:

$$\mathbf{v}_k^{to} = f_t(\mathbf{t}_k^o), \quad \mathbf{v}_{k,l}^{tb} = f_t(\mathbf{t}_{k,l}^b), \quad (5)$$

where \mathbf{t}_k^o and $\mathbf{t}_{k,l}^b$ denote text label of the object and the l -th class-related co-occurring background for the specific class k , respectively.

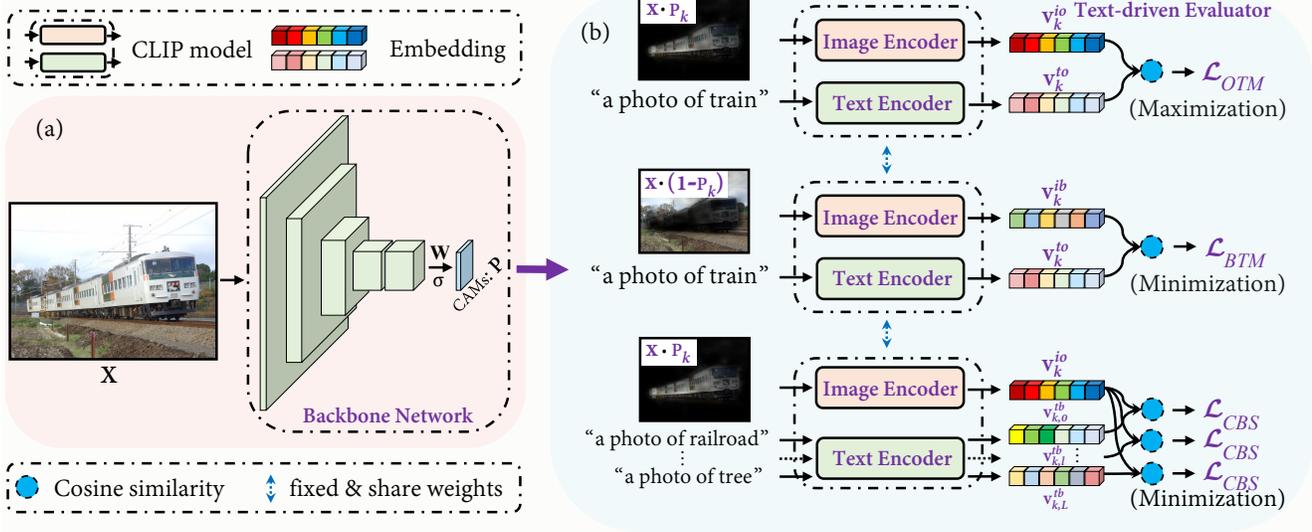


Figure 2. An overview of the proposed Cross Language Image Matching framework for WSSS, i.e., CLIMS. (a) The backbone network for predicting initial CAMs. σ denotes the sigmoid activation function. \mathbf{W} denotes the weight matrix of convolutional layers. (b) The text-driven evaluator. It consists of three CLIP-based loss functions, i.e., object region and text label matching loss \mathcal{L}_{OTM} , background region and text label matching loss \mathcal{L}_{BTM} , and co-occurring background suppression loss \mathcal{L}_{CBS} . Best viewed in color.

3.3. Object region and Text label Matching

Given the k -th foreground object representation \mathbf{v}_k^{io} and its corresponding text representation \mathbf{v}_k^{to} , we begin by calculating the cosine similarity between image and text representations and then maximize it using the proposed object region and text label matching loss \mathcal{L}_{OTM} :

$$\mathcal{L}_{OTM} = - \sum_{k=1}^K y_k \cdot \log(s_k^{oo}), \quad (6)$$

$$s_k^{oo} = \text{sim}(\mathbf{v}_k^{io}, \mathbf{v}_k^{to}), \quad (7)$$

s_k^{oo} indicates the cosine similarity between \mathbf{v}_k^{io} and \mathbf{v}_k^{to} . The generated initial CAMs will gradually approach the target object under the supervision of \mathcal{L}_{OTM} . However, the \mathcal{L}_{OTM} alone can not encourage the model to explore non-discriminative object regions and suppress the background regions activated in \mathbf{P}_k .

3.4. Background region and Text label Matching

To improve the completeness of activated object regions, we design the background region and text label matching loss \mathcal{L}_{BTM} to include more object contents. Given the background representation \mathbf{v}_k^{ib} and its corresponding text representation \mathbf{v}_k^{to} (note that, the text label for \mathcal{L}_{BTM} is the same with that for \mathcal{L}_{OTM}), the \mathcal{L}_{BTM} is calculated as follow:

$$\mathcal{L}_{BTM} = - \sum_{k=1}^K y_k \cdot \log(1 - s_k^{bo}), \quad (8)$$

$$s_k^{bo} = \text{sim}(\mathbf{v}_k^{ib}, \mathbf{v}_k^{to}), \quad (9)$$

where s_k^{bo} represents the cosine similarity between \mathbf{v}_k^{ib} and \mathbf{v}_k^{to} . When \mathcal{L}_{BTM} is minimized, fewer target object pixels are reserved in $\mathbf{X} \cdot (1 - \mathbf{P}_k)$ and more target object contents are recovered in $(\mathbf{X} \cdot \mathbf{P}_k)$. This ensures that more complete object contents are activated in \mathbf{P}_k .

3.5. Co-occurring Background Suppression

The aforementioned two loss functions, however, only ensure \mathbf{P} to completely cover the target object, without taking into account the false-activation of co-occurring class-related background. The inclusion of co-occurring background may significantly degrade the quality of generated pseudo ground-truth masks. However, pixel-level labeling of these backgrounds is very time-consuming and labor expensive, and is usually not available in WSSS. As the set of background is much more diverse than that of foreground, lots of them might not have been seen by the classification network trained using ImageNet. However, it's much easier to use the pretrained CLIP to identify these backgrounds given the corresponding text descriptions. To solve this problem, we design the following co-occurring background suppression loss. Given the target object representation \mathbf{v}_k^{io} and its corresponding text representation of class-related background $\mathbf{v}_{k,l}^{tb}$, the loss is calculated as:

$$\mathcal{L}_{CBS} = - \sum_{k=1}^K \sum_{l=1}^L y_k \cdot \log(1 - s_{k,l}^{ob}), \quad (10)$$

$$s_{k,l}^{ob} = \text{sim}(\mathbf{v}_k^{io}, \mathbf{v}_{k,l}^{tb}), \quad (11)$$

where $s_{k,l}^{ob}$ indicates the cosine similarity between \mathbf{v}_k^{io} and $\mathbf{v}_{k,l}^{tb}$. During training, the backbone network will gradually suppress the false-activation of class-related background regions in \mathbf{P}_k for the minimization of \mathcal{L}_{CBS} .

3.6. Area Regularization

With only \mathcal{L}_{OTM} , \mathcal{L}_{BTM} , and \mathcal{L}_{CBS} , if both the irrelevant backgrounds and the target object are included in the activation map, the CLIP model can still correctly predict the target object. Therefore, we design a pixel-level area regularization term to constraint the size of activation maps to ensure that the irrelevant backgrounds are excluded in the activation map \mathbf{P}_k :

$$\mathcal{L}_{REG} = \frac{1}{K} \sum_{k=1}^K S_k, \quad \text{where} \quad S_k = \frac{1}{HW} \sum_{h=1}^H \sum_{w=1}^W \mathbf{P}_k(h, w). \quad (12)$$

3.7. Overall Training Objective

The overall training loss for the proposed text-driven learning framework CLIMS can be formulated as:

$$\mathcal{L} = \alpha \mathcal{L}_{OTM} + \beta \mathcal{L}_{BTM} + \gamma \mathcal{L}_{CBS} + \delta \mathcal{L}_{REG}, \quad (13)$$

where α , β , γ , and δ are the hyper-parameters weighting the four loss terms.

4. Experiments

4.1. Experimental Setup

Datasets and Evaluation Metric. PASCAL VOC2012 [9] is a popular semantic segmentation dataset with 20 object categories, which consists of 1,464 images for training, 1,449 images for validation, and 1,456 images for test. Following the common protocol in previous studies, we train the proposed CLIMS using an augmented training set of 10,582 images. The mean Intersection over Union (mIoU) is adopted as the evaluation metric for all experiments.

Implementation Details. The input image is randomly rescaled and then augmented by random cropping to 512×512 . Horizontal flipping is also used for augmenting training data. SGD is adopted as the default optimizer. The cosine annealing policy is applied to schedule learning rate. The default batch size is 16. As there are minor differences in describing classes between PASCAL VOC2012 and CLIP, we use the text label descriptions in the training set to finetune the CLIP model (both image and text encoder) for 20 epochs, with an initial learning rate of 0.00005 and a weight decay of 0.003. The CLIMS model is trained for 10 epochs, with an initial learning rate of 0.00025 and a weight decay of 0.0001. We follow [2] to adopt ResNet-50 [10] as backbone network for the generation of initial

Table 1. Comparison of the quality of initial CAMs and refined pseudo ground-truth masks using RW (PSA [2]) on PASCAL VOC2012. The mIoU values here are reported on the *train* set. Bac. denotes the backbone network for CAMs generation.

Method	Bac.	CAMs	+RW
PSA _{CVPR'2018} [2]	WR38	48.0	61.0
SC-CAM _{CVPR'2020} [4]	WR38	50.9	63.4
SEAM _{CVPR'2020} [31]	WR38	55.4	63.6
PuzzleCAM _{ICIP'2021} [13]	R50	51.5	64.7
VWE _{IJCAI'2021} [26]	R50	52.9	-
AdvCAM _{CVPR'2021} [18]	R50	55.6	68.0
CLIMS (Ours)	R50	56.6	70.5

CAMs. All models are implemented in PyTorch and trained on NVIDIA A100 GPU with 40 GB memory.

Refinement of initial CAMs. Because the initial CAMs only coarsely cover the target object, refinement techniques such as PSA [2] and IRNet [1] are commonly used to improve the quality of initial CAMs before using them as pseudo ground-truth masks. To make a fair comparison, we follow SEAM [31], PuzzleCAM [13], and AdvCAM [18] to adopt PSA [2] for initial CAM refinement.

Table 2. Evaluation results on PASCAL VOC2012 *val* and *test* sets. The best results are in **bold**. Sup. denotes the weak supervision type. \mathcal{F} denotes full supervision. \mathcal{S} denotes saliency map supervision. \mathcal{I} denotes image-level supervision. Seg. denotes the segmentation network. Bac. denotes the backbone network for CAMs generation. V1: DeepLabV1. V2: DeepLabV2. V16: VGG-16 [27]. R50: ResNet-50 [10]. WR38: WideResNet38 [33]. ‡: Segmentation network pretrained using MS COCO dataset.

Sup.	Method	Seg.	Bac.	<i>val</i>	<i>test</i>	
Full supervision.						
\mathcal{F}	DeepLabV1 _{ICLR'15} [5]	-	-	75.5	-	
	DeepLabV2 _{TPAMI'18} [6]	-	-	77.6	79.7	
	WideResNet38 _{PR'19} [33]	-	-	80.8	82.5	
Image-level supervision + Saliency maps.						
$\mathcal{I} + \mathcal{S}$	OAA+ _{ICCV'19} [12]	V1 [‡]	-	65.2	66.4	
	MCIS _{ECCV'20} [28]	V1 [‡]	V16	66.2	66.9	
	LIID _{TPAMI'21} [21]	V2	R50	66.5	67.5	
	NSROM _{CVPR'21} [35]	V2 [‡]	V16	68.3	68.5	
	DRS _{AAAI'21} [14]	V2 [‡]	V16	70.4	70.7	
	EPS _{CVPR'21} [19]	V2 [‡]	WR38	70.9	70.8	
	EDAM _{CVPR'21} [32]	V2 [‡]	WR38	70.9	70.6	
	AuxSegNet _{ICCV'21} [34]	WR38	-	69.0	68.6	
	Image-level supervision only.					
	\mathcal{I}	IAL _{ICCV'20} [30]	V2	-	64.3	65.4
SEAM _{CVPR'20} [31]		V3	WR38	64.5	65.7	
BES _{ECCV'20} [7]		V2	R50	65.7	66.6	
SC-CAM _{CVPR'20} [4]		V2 [‡]	WR38	66.1	65.9	
CONTA _{NeurIPS'20} [37]		V3	WR38	66.1	66.7	
A ² GNN _{TPAMI'21} [36]		V2	WR38	66.8	67.4	
VWE _{IJCAI'2021} [26]		V2 [‡]	R50	67.2	67.3	
AdvCAM _{CVPR'21} [18]		V2	R50	68.1	68.0	
Kweon <i>et al.</i> _{ICCV'21} [17]		WR38	WR38	68.4	68.2	
CLIMS (Ours)		V2	R50	69.3	68.7	
CLIMS (Ours)	V2 [‡]	R50	70.4	70.0		

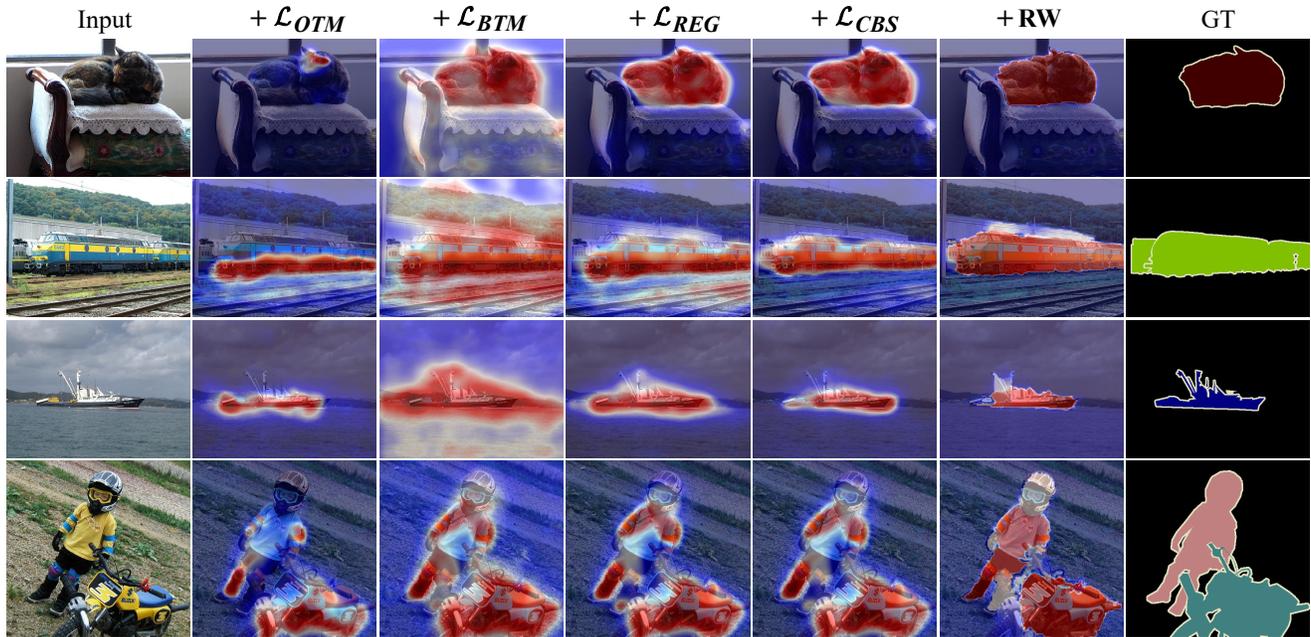


Figure 3. Initial CAMs generated by the proposed CLIMS using different combinations of loss functions. Input images are shown in column 1. Columns 2 to 5 present the generated CAMs using \mathcal{L}_{OTM} , $\mathcal{L}_{OTM} + \mathcal{L}_{BTM}$, $\mathcal{L}_{OTM} + \mathcal{L}_{BTM} + \mathcal{L}_{REG}$, and $\mathcal{L}_{OTM} + \mathcal{L}_{BTM} + \mathcal{L}_{REG} + \mathcal{L}_{CBS}$, respectively. **RW** denotes the refinement of PSA [2]. Best viewed in color.

Segmentation Network. Given pseudo ground-truth masks, we follow VWE [26], SC-CAM [4] and Adv-CAM [18] to adopt DeepLabV2 with ResNet-101 [10] as the segmentation network. For experiments on PASCAL VOC2012 dataset, we follow the default setting of deeplab-pytorch toolkit[†] to train DeepLabV2 with weights pretrained using MS COCO dataset. Besides, we also follow the setting of AdvCAM [18] to train DeepLabV2 with weights pretrained using ImageNet-1K dataset.

4.2. Results on PASCAL VOC2012 Dataset

Quality of Initial CAMs and Pseudo Labels. The quality of generated initial CAMs and refined pseudo ground-truth masks on the PASCAL VOC2012 *train* set is compared in Table 1. **+RW** denotes that the initial CAMs are refined by PSA [2]. As shown in the table, the initial CAMs generated by our CLIMS reach 56.6% mIoU, significantly outperforming the state-of-the-art methods, e.g., PuzzleCAM and AdvCAM. This mostly attributes to the designs of three CLIP-based loss functions. When compared to SEAM (WideResNet38 [33] is adopted as the backbone network), our CLIMS (the smaller ResNet-50 is adopted as backbone) achieves better results. Besides, our pseudo labels refined with PSA achieve 70.0% mIoU, which is 5.3% and 2.0% higher than that of PuzzleCAM and AdvCAM, respectively.

In Fig. 4, we visualize the initial CAMs and compare them with the results of conventional CAM [38] and recent

[†]<https://github.com/kazuto1011/deeplab-pytorch>

Table 3. Comparison of the quality of initial CAMs on PASCAL VOC2012 dataset using different combinations of loss functions. The mIoU(%) results are reported on the *train* set. \mathcal{L}_{CLS} indicates that we use a pretrained classifier, i.e., VGG-16, to replace our text-driven evaluator for comparison.

\mathcal{L}_{OTM}	\mathcal{L}_{BTM}	\mathcal{L}_{REG}	\mathcal{L}_{CBS}	\mathcal{L}_{CLS}	mIoU(%)
				✓	28.6
✓					37.2
✓	✓				41.3
✓	✓	✓			53.1
✓	✓		✓		45.4
✓	✓	✓	✓		56.6

method Adv-CAM [18]. It is observed that the proposed CLIMS typically activates more complete object contents and less class-related background regions. Specifically, as shown in the first two and last two columns, CAM and Adv-CAM may underestimate the region of boat, person, and cat or falsely activate the region of river. Compared with them, initial CAMs generated by CLIMS are more complete and compact, including more reasonable object regions for next stage of refinement. In the third and sixth columns, lots of false activations of class-related background, i.e., railroad, are available in the results of CAM and Adv-CAM. With the supervision provided by \mathcal{L}_{CBS} , our CLIMS can efficiently reduce the false activation of co-occurring background regions and yield complete and compact CAMs.

Segmentation Performance. To further validate the quality of the pseudo ground-truth masks generated by

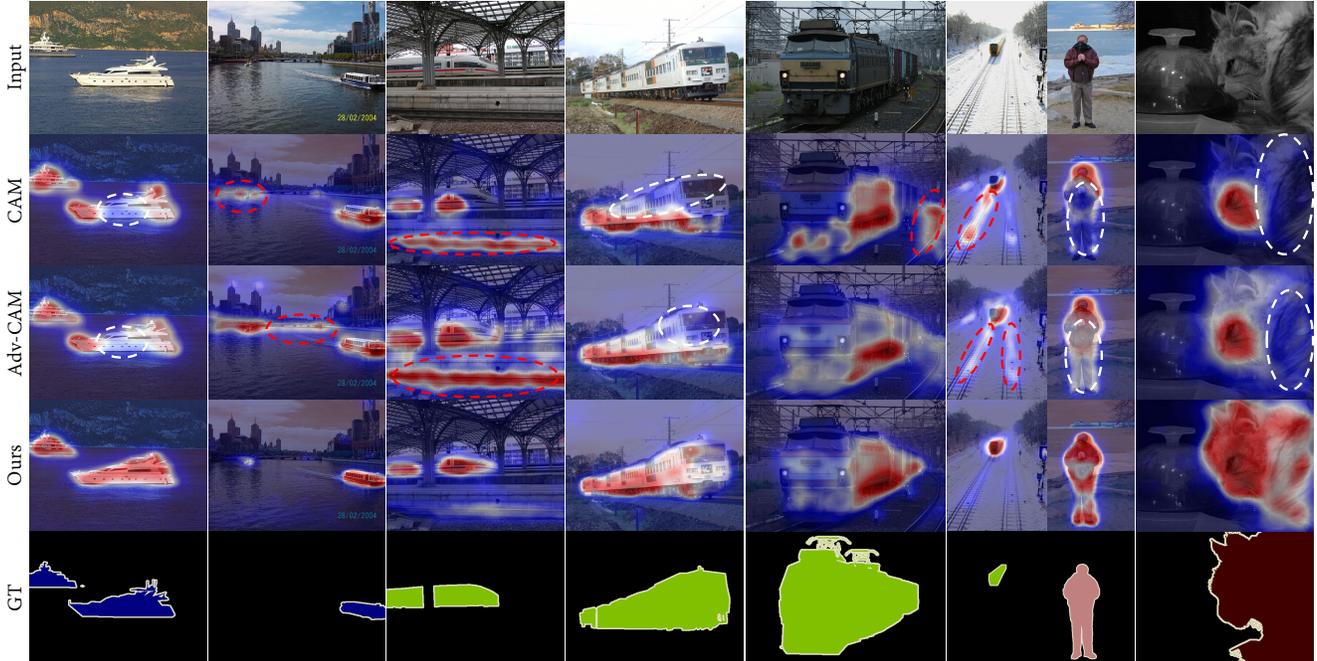


Figure 4. Visualization of the initial CAMs generated by CAM, Adv-CAM, and the proposed CLIMS. White dotted circles illustrate the missed object regions. Red dotted circles illustrate the false activation of class-related background regions, e.g., the river and railroad.

our method, we fully train a segmentation network, i.e., DeepLabV2, on the PASCAL VOC2012 dataset using the generated pseudo labels. The evaluation results are reported in Table 2. It is observed that, compared to the methods with only image-level supervision, our method achieves the best results. Specifically, DeepLabV2 trained using the mask generated by our CLIMS achieves 69.3% and 68.7% mIoU on *val* and *test* set, respectively. Compared with Adv-CAM, our method outperforms it by 1.2% and 0.7% mIoU on *val* and *test* sets, respectively. Compared to methods with additional saliency map (obtained from a fully supervised model), e.g., EPS [19] and EDAM [32], our method also achieve competitive performance.

Table 4. Evaluation results of two specific object categories, i.e., boat and train, on PASCAL VOC2012 dataset with/out the addition of \mathcal{L}_{CBS} . The results are reported on the *train* set.

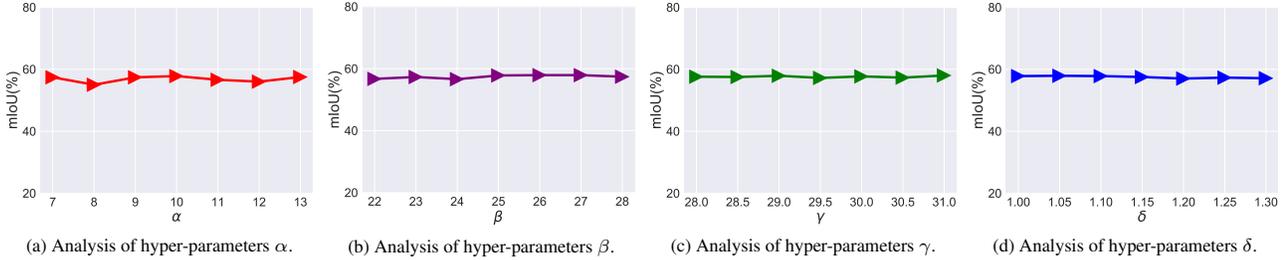
loss functions.	boat(%)	train(%)	mean(%)
$\mathcal{L}_{OTM} + \mathcal{L}_{BTM}$	7.1	30.7	18.9
$\mathcal{L}_{OTM} + \mathcal{L}_{BTM} + \mathcal{L}_{CBS}$	58.2 _{+51.1}	63.9 _{+33.2}	61.1 _{+42.2}

4.3. Ablation Studies

Impact of Loss Functions. The proposed CLIMS, as described in Sec. 3, is optimized by a combination of four loss functions, i.e., object region and text label matching loss \mathcal{L}_{OTM} , background region and text label matching loss \mathcal{L}_{BTM} , co-occurring background suppression loss \mathcal{L}_{CBS} , and area regularization \mathcal{L}_{REG} , which play different roles in guiding backbone network to generate initial

CAMs. Here, we perform ablation experiments on the PASCAL VOC2012 dataset to further validate the effectiveness of each loss function. The visual results are shown in Fig. 3 (note that, $+\mathcal{L}$ means that we successively add the loss function). As shown in the second column of Fig. 3, by only using \mathcal{L}_{OTM} , we observe that 1) only the discriminative object parts are activated in the CAMs; 2) class-related backgrounds, e.g., railroad in the second row and water in the third row, are also activated in the initial CAMs. As seen in the third column, an addition of \mathcal{L}_{BTM} significantly increases the size of activated regions, such that more complete object regions are activated. However, $\mathcal{L}_{OTM} + \mathcal{L}_{BTM}$ falsely activate the background regions. As shown in the fourth and fifth column in Fig. 3, the addition of \mathcal{L}_{REG} and \mathcal{L}_{CBS} helps address the aforementioned issues. We can observe that \mathcal{L}_{REG} efficiently constrains the size of activated regions and \mathcal{L}_{CBS} significantly excludes the class-related background, e.g., railroad, out of the CAMs. Furthermore, as shown in the sixth column, with the refinement of RW (PSA [2]), the pseudo masks are closely similar to the ground-truth.

Table 3 shows quantitative comparisons between different combinations of loss functions. As can be seen, CLIMS only obtains 37.2% mIoU on the *train* set when only \mathcal{L}_{OTM} is used. An addition of \mathcal{L}_{BTM} improves mIoU from 37.2% to 41.3%. By ensuring the compactness of initial CAMs, the inclusion of \mathcal{L}_{REG} improves the mIoU from 41.3% to 53.1%. As expected, \mathcal{L}_{CBS} can efficiently remove the class-related background regions from the generated CAMs



(a) Analysis of hyper-parameters α . (b) Analysis of hyper-parameters β . (c) Analysis of hyper-parameters γ . (d) Analysis of hyper-parameters δ . Figure 5. Sensitivity analyses of hyper-parameters α , β , γ , and δ . The mIoU values here are reported on PASCAL VOC2012 *train* set.

and improve the mIoU by 3.5% on the *train* set. To validate the effectiveness of CLIP model, we use a pretrained classifier, i.e., VGG-16, to replace text-driven evaluator and compare their performances. It means that only mask-out area of image is fed into the classifier for evaluation. The result is illustrated by \mathcal{L}_{CLS} . As shown in Table 3, \mathcal{L}_{CLS} only achieves 28.6% mIoU, which is much lower than \mathcal{L}_{OTM} (37.2%). This suggest that, without other constraints, supervision from CLIP leads the backbone network to generate better CAMs than a pretrained classifier.

Class-related Background. In our experiments, we only pre-define a set of class-related backgrounds for some object categories, such as train and boat. For example, the background sets of train and boat are {"railroad", "railway", "tree"} and {"river", "sea", "lake"}, respectively. To further demonstrate the effectiveness of co-occurring background suppression loss \mathcal{L}_{CBS} , we report the IoU of boat and train in Table 4 with/out \mathcal{L}_{CBS} . As can be seen, an addition of \mathcal{L}_{CBS} can significantly improve the IoU of both boat and train, from 7.1% to 58.2% and 30.7% to 63.9%, respectively. The results suggest that CLIP can effectively identify those diverse background categories. Based on the embeddings of background regions and text descriptions, \mathcal{L}_{CBS} can effectively exclude these co-concurring backgrounds from the activated regions of foreground objects.

Sensitivity Analyses. There are four hyper-parameters in Eq. 13. The sensitivity analyses of these four parameters are performed on PASCAL VOC2012 *train* set and the evaluation results are presented in Fig. 5. It is observed that the performances of our approach are stable with the variation of α (from 7 to 13), β (from 22 to 28), γ (from 28 to 31), and δ (from 1 to 1.3), i.e., our approach is not sensitive to hyper-parameters. In our experiments, the default value of α , β , γ , and δ are 10, 25, 29.5, and 1.15, respectively.

Feature Representation Analysis. Fig. 6 depicts the relationship between learned features and the weight vector \mathbf{W}_k . The demo image used to extract representation features is shown in the left of Fig. 6. To investigate the quality of learned features, we sample some features from the pixels belonging to the class of train and background and compute the similarity between weight vector \mathbf{W}_k and them. The results are shown in the right of Fig. 6. The sample features are denoted in the x-axis, and the weight

vector of a class is denoted in the y-axis. It is observed that the extracted features of the blue stars (index from 10 to 19) are highly related to the weight vector \mathbf{W}_k of train and are unrelated to the other classes. This suggests that the features of train are clustered with \mathbf{W}_k of train and distribute far away from other classes. The sample regions from the class-related background are represented by yellow stars (index from 0 to 9). Their learned features have a weak relationship with the weight vector of train. This ensures that the class-related background will not be activated in the activation map generation for the class of train.

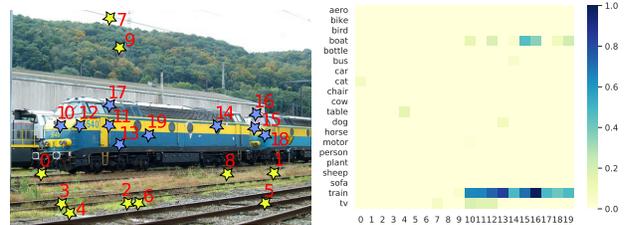


Figure 6. Left: the sample image. Yellow and blue stars in the image denote the sample regions. Right: the similarity matrix. The x-axis denotes the feature of sample region and the y-axis denotes the weight vector \mathbf{W}_k of each class. The (i, j) -th element means the cosine similarity between i -th class and j -th region in the image. Note that, the calculated cosine similarities are truncated and normalized into $[0, 1]$. Best viewed in color.

5. Conclusion

This paper proposes a novel Cross Language Image Matching framework, i.e., CLIMS, to introduce natural language supervision for WSSS. The designed four loss functions can efficiently handle the problem of underestimation of complete object contents and unnecessary activation of closely-related background regions. Extensive experiments conducted on the PASCAL VOC2012 dataset validate the effectiveness of CLIMS. The experimental results reveal that our method generates more complete and compact initial CAMs and refined pseudo ground-truth masks than baseline and state-of-the-art methods.

Acknowledgments

The work was supported by the National Natural Science Foundation of China under grant 91959108.

References

- [1] Jiwoon Ahn, Sunghyun Cho, and Suha Kwak. Weakly supervised learning of instance segmentation with inter-pixel relations. In *CVPR*, pages 2209–2218, 2019. 1, 5
- [2] Jiwoon Ahn and Suha Kwak. Learning pixel-level semantic affinity with image-level supervision for weakly supervised semantic segmentation. In *CVPR*, pages 4981–4990, 2018. 1, 3, 5, 6, 7
- [3] Amy L. Bearman, Olga Russakovsky, Vittorio Ferrari, and Fei-Fei Li. What’s the point: Semantic segmentation with point supervision. In *ECCV*, pages 549–565, 2016. 1
- [4] Yu-Ting Chang, Qiaosong Wang, Wei-Chih Hung, Robinson Piramuthu, Yi-Hsuan ai, and Ming-Hsuan Yang. Weakly-supervised semantic segmentation via sub-category exploration. In *CVPR*, pages 8988–8997, 2020. 3, 5, 6
- [5] Liang-Chieh Chen, George Papandreou, Iasonas Kokkinos, Kevin Murphy, and Alan L. Yuille. Semantic image segmentation with deep convolutional nets and fully connected crfs. In *ICLR*, 2015. 3, 5
- [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. *TPAMI*, 40:834–848, 2018. 3, 5
- [7] Liyi Chen, Weiwei Wu, Chenchen Fu, Xiao Han, and Yuntao Zhang. Weakly supervised semantic segmentation with boundary exploration. In *ECCV*, pages 347–362, 2020. 5
- [8] Jifeng Dai, Kaiming He, and Jian Sun. Boxesup: Exploiting bounding boxes to supervise convolutional networks for semantic segmentation. In *ICCV*, pages 1635–1643, 2015. 1
- [9] M. Everingham, L. Van Gool, C. K. I. Williams, J. Winn, and A. Zisserman. The PASCAL Visual Object Classes Challenge 2012 (VOC2012) Results. <http://www.pascal-network.org/challenges/VOC/voc2012/workshop/index.html>. 3, 5
- [10] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *CVPR*, pages 770–778, 2016. 5, 6
- [11] Qibin Hou, Peng-Tao Jiang, Yunchao Wei, and Ming-Ming Cheng. Self-erasing network for integral object attention. In *NeurIPS*, pages 547–557, 2018. 3
- [12] 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*, pages 2070–2079, 2019. 1, 3, 5
- [13] Sanghyun Jo and In-Jae Yu. Puzzle-cam: Improved localization via matching partial and full features. In *2021 IEEE International Conference on Image Processing (ICIP)*, pages 639–643, 2021. 5
- [14] Beomyoung Kim, Sangeun Han, and Junmo Kim. Discriminative region suppression for weakly-supervised semantic segmentation. In *AAAI*, pages 1754–1761, 2021. 5
- [15] Alexander Kolesnikov and Christoph H. Lampert. Seed, expand and constrain: Three principles for weakly-supervised image segmentation. In *ECCV*, pages 695–711, 2016. 1
- [16] Philipp Krähenbühl and Vladlen Koltun. Efficient inference in fully connected crfs with gaussian edge potentials. In *NeurIPS*, pages 109–117, 2011. 1
- [17] Hyeokjun Kweon, Sung-Hoon Yoon, Hyeonseong Kim, Daehee Park, and Kuk-Jin Yoon. Unlocking the potential of ordinary classifier: Class-specific adversarial erasing framework for weakly supervised semantic segmentation. In *ICCV*, pages 6994–7003, 2021. 5
- [18] Jungbeom Lee, Eunji Kim, and Sungroh Yoon. Anti-adversarially manipulated attributions for weakly and semi-supervised semantic segmentation. In *CVPR*, pages 4071–4080, 2021. 3, 5, 6
- [19] 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*, pages 5495–5505, 2021. 1, 3, 5, 7
- [20] Di Lin, Jifeng Dai, Jiaya Jia, Kaiming He, and Jian Sun. Scribblesup: Scribble-supervised convolutional networks for semantic segmentation. In *CVPR*, pages 3159–3167, 2016. 1
- [21] Yun Liu, Yu-Huan Wu, Pei-Song Wen, Yu-Jun Shi, Yu Qiu, and Ming-Ming Cheng. Leveraging instance-, image- and dataset-level information for weakly supervised instance segmentation. *TPAMI*, 2020. 1, 3, 5
- [22] 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 *ICCV*, pages 1742–1750, 2015. 1
- [23] Deepak Pathak, Philipp Krähenbühl, and Trevor Darrell. Constrained convolutional neural networks for weakly supervised segmentation. In *ICCV*, pages 1796–1804, 2015. 1
- [24] Pedro H. O. Pinheiro and Ronan Collobert. From image-level to pixel-level labeling with convolutional networks. In *CVPR*, pages 1713–1721, 2015. 1
- [25] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen Krueger, and Ilya Sutskever. Learning transferable visual models from natural language supervision. In *ICML*, pages 8748–8763, 2021. 2, 3
- [26] Lixiang Ru, Bo Du, and Chen Wu. Learning visual words for weakly-supervised semantic segmentation. In *IJCAI*, 2021. 5, 6
- [27] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. In *ICLR*, 2015. 5
- [28] Guolei Sun, Wenguan Wang, Jifeng Dai, and Luc Van Gool. Mining cross-image semantics for weakly supervised semantic segmentation. In *ECCV*, pages 347–365, 2020. 3, 5
- [29] Paul Vernaza and Manmohan Chandraker. Learning random-walk label propagation for weakly-supervised semantic segmentation. In *CVPR*, pages 7158–7166, 2017. 1
- [30] Xiang Wang, Sifei Liu, Huimin Ma, and Ming-Hsuan Yang. Weakly-supervised semantic segmentation by iterative affinity learning. *IJCV*, 128:1736–1749, 2020. 5

- [31] Yude Wang, Jie Zhang, Meina Kan, Shiguang Shan, and Xilin Chen. Self-supervised equivariant attention mechanism for weakly supervised semantic segmentation. In *CVPR*, pages 12275–12284, 2020. 5
- [32] Tong Wu, Junshi Huang, Guangyu Gao, Xiaoming Wei, Xiaolin Wei, Xuan Luo, and Chi Harold Liu. Embedded discriminative attention mechanism for weakly supervised semantic segmentation. In *CVPR*, pages 16765–16774, 2021. 3, 5, 7
- [33] Zifeng Wu, Chunhua Shen, and Anton van den Hengel. Wider or deeper: Revisiting the resnet model for visual recognition. *PR*, 90:119–133, 2019. 5, 6
- [34] 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*, pages 6984–6993, 2021. 5
- [35] Yazhou Yao, Tao Chen, Guo-Sen Xie, Chuanyi Zhang, Fumin Shen, Qi Wu, Zhenmin Tang, and Jian Zhang. Non-salient region object mining for weakly supervised semantic segmentation. In *CVPR*, pages 2623–2632, 2021. 5
- [36] Bingfeng Zhang, Jimin Xiao, Jianbo Jiao, Yunchao Wei, and Yao Zhao. Affinity attention graph neural network for weakly supervised semantic segmentation. *TPAMI*, 2021. 5
- [37] Dong Zhang, Hanwang Zhang, Jinhui Tang, Xian-Sheng Hua, and Qianru Sun. Causal intervention for weakly-supervised semantic segmentation. In *NeurIPS*, 2020. 5
- [38] Bolei Zhou, Aditya Khosla, Agata Lapedriza, Aude Oliva, and Antonio Torralba. Learning deep features for discriminative localization. In *CVPR*, pages 2921–2929, 2016. 2, 6