Learning From Pixel-Level Noisy Labels: A New Perspective for Weakly-Supervised Semantic Segmentation

Weikang Xiang, Afang Yang, Quan Zhou, *Senior Member, IEEE,* Xun Sun, Jie Cheng, and Huimin Lu, *Senior Member, IEEE*

***Abstract*—The recent advances of weakly-supervised semantic segmentation (WSSS) have witnessed remarkable progress using image-level labels. However, many existing approaches always suffer from inaccurate class activation maps (CAMs), which arise from translating category information into object localization. This paper approaches WSSS from a novel perspective, treating incomplete and incorrect activations as pixel-level noisy labels that can be effectively calibrated within a developed contrastive learning framework. Unlike prior efforts that mainly relied on data augmentation to generate supervised signals, we create positive and negative training pairs by jointly capturing pixel-level context in both embedding and semantic spaces. To accomplish this, we employ shrinking and expansion operations to produce pixel-level pseudo-labels that supervise the contrastive learning process. Specifically, during the shrinking operation, a series of robust seeds are produced under the constraints of feature and semantic consistency. In contrast, the expansion operation designs a distance-constrained feature similarity module to propagate categorical semantics based on the principle that pixels belonging to the same object should exhibit similar feature embeddings and close spatial locations. This approach helps the classifier in identifying more under-activated pixels while suppressing incorrectly activated ones. In addition to enhancing the accuracy of CAMs, our method allows for the utilization of more positive training pairs to learn robust feature representations, thus ultimately improving noisy label calibration. We conducted exhaustive experiments on the PASCAL VOC 2012, MS COCO 2014, and Cityscapes datasets. The extensive experimental results demonstrate the effectiveness of our method, achieving 79.5%, 51.9%, and 53.4% mIoU on three datasets, respectively.**

Manuscript received XXXX XX, 2025; revised XXXX XX, 2025; accepted XXXX XX, 2026. This work was jointly supported in part by the National Natural Science Foundation of China under Grants 62476139 and 62576090, Fundamental Research Funds for the Central Universities under Grant 4008002403, Key Project of the Jiangsu Provincial Basic Research Program under Grant SBK20251000047, Major Innovation Platform Plan of Jiangsu Provincial Department of Science and Technology under Grant BM2024020, and the National Natural Science Foundation of Jiangsu Province under Grant BK2024023.

Corresponding author: Quan Zhou.

Weikang Xiang, Afang Yang, and Quan Zhou are with Jiangsu Key Laboratory of Intelligent Information Processing and Communication Technology, Nanjing University of Posts and Telecommunications, Nanjing Jiangsu, P. R. China. Quan Zhou is also with Advanced Ocean Institute of Southeast University, Nantong Jiangsu, P. R. China. (e-mail: {1022010310, 1025010319, quan.zhou}@njupt.edu.cn)

Xun Sun is with the Institute of Guizhou Aerospace Measuring and Testing Technology, Guiyang Guizhou, P. R. China. (e-mail: sunxunup@alumni.sjtu.edu.cn)

Jie Cheng is with Huawei Technologies Co., Ltd. Shenzhen Guangdong, P. R. China. (e-mail: chengjie8@huawei.com)

Huimin Lu is with School of Automation, Southeast University, Nanjing, P. R. China, and with Advanced Ocean Institute of Southeast University, Nantong Jiangsu, P. R. China. (e-mail: dr.huimin.lu@ieee.org)

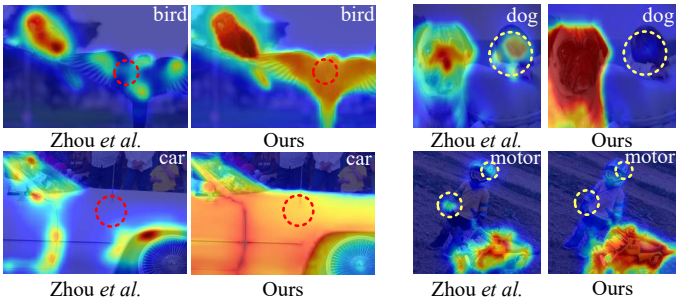


Fig. 1. Visual examples of CAMs using [14] and our method on PASCAL VOC dataset [15]. Areas with a deeper red indicate higher activation levels, while those in blue represent lower activations. For clarity, the categorical tags that require to be activated have also been superimposed in the top-right corner of each example. Compared with Zhou *et al.* [14], the CAMs generated by our approach are more accurate, as they correctly highlight nearly the entire objects (indicated by red dashed circles) and effectively suppress co-occurring background regions (denoted by yellow dashed circles). (Best viewed in color.)

***Index Terms*—Weakly-supervised semantic segmentation, noisy labels, contrastive learning, pixel-level pseudo-labels.**

I. Introduction

RECENTLY, semantic segmentation [1], [2] has made significant progress with the rapid advancements in deep learning. However, the fully supervised learning paradigm requires massive manual labeling, which is both labor-intensive and time-consuming, especially when annotating pixel-level ground truth for training semantic segmentation models. To mitigate this limitation, numerous efforts have been proposed to develop semantic segmentation under weaker forms of supervision, such as points [3], [4], scribbles [5], [6], bounding boxes [7], [8], and image-level labels [9]–[13]. Due to the least annotation efforts, using only image-level labels has become the dominant strategy in the weakly-supervised setting for semantic segmentation (WSSS), where our work also aligns with this paradigm.

When image-level labels are the sole source of supervision, existing advanced methods for WSSS typically involve two main steps. Initially, a classifier network is trained to generate class activation maps (CAMs) [14], which serve as pseudo ground truths for training subsequent pixel-level segmentation networks. However, as illustrated in Fig. 1, it is widely recognized that the trained classifier tends to activate the most discriminative parts rather than entire object regions [16], [17]. Relying on such pixel-wise inaccurate CAMs as

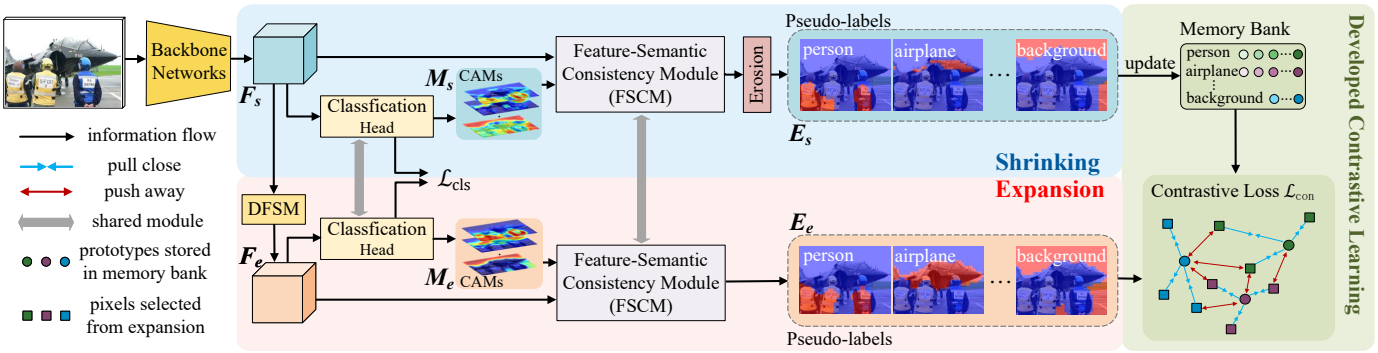


Fig. 2. Overall pipeline of our method. After input images are encoded into feature embeddings using backbone networks, the shrinking and expansion operations are employed to create robust and reliable pseudo-label masks. Concretely, the shrinking operation aims to filter out robust seeds (indicated by the orange region in shrinking operation) from noisy CAMs by assigning them very high confident scores. These strong seeds are stored as category-specific prototypes (represented by circles with different colors) in a memory bank, which can be iteratively updated throughout the training process. Conversely, the expansion operation seeks to identify more reliable pixels (indicated by the orange region in expansion operation) with the assistance of DFSM. These newly discovered category-specific pixels (denoted by squares with different colors), together with prototypes stored in memory bank, are used to construct positive and negative training pairs for training a developed contrastive loss, where the pixel-level noisy labels can be effectively calibrated. (Best viewed in color)

supervisory signals can adversely affect the training of the follow-up segmentation networks [10], [18], [19]. In response to this challenge, various efforts have been made to produce more accurate CAMs, which can be broadly categorized into four approaches: 1) *adversarial region erasing* [20]–[23] that gradually erases highly activated areas to uncover additional object regions; 2) *contextual formulation* [9], [24], [25] that thoroughly investigates or decomposes object-related context within or across images; 3) *affinity expansion* [26]–[28] that propagates pixel information through feature similarities to recalibrate incorrectly activated seeds; and 4) *foundation models* [10], [12], [29]–[31] that transfer object-aware knowledge learned from visual-prompt pairs to mitigate inaccurate CAMs. In spite of achieving impressive results, these advanced approaches inherently suffer from certain limitations. Concretely, as additional object regions are progressively discovered, the first category is prone to over-activation [13], [32], resulting in the highlighting of non-target regions. In contrast, the second category may incorrectly activate background areas due to the inherent context of category co-occurrence [9], [25]. The third category improves the quality of CAMs but at the cost of introducing additional and complex affinity networks [26], [27], while the last category requires extra prompt supervision rather than relying solely on image-level labels [12], [31], [33].

Unlike these advanced approaches [9], [10], [13], [28], this paper tackles the issue of CAM calibration from a novel perspective: we treat estimated incomplete and incorrect activations as pixel-wise noisy labels, framing WSSS as a noisy label correction problem. In spite of extensive literature that focuses on learning from noisy labels in image classification [34]–[36], there are very few efforts [10], [18], [37]–[39] addressing WSSS from the standpoint of noisy label learning. Due to the memory effect in deep neural networks (DNNs) [34], which eventually tend to fit noisy labels during training, ADELE [37] adaptively corrects noisy labels in the early stages of learning. Guided by object boundaries, BECO [38] rectifies noise in pseudo-labels using a co-training scheme. URN [39] mitigates noisy supervision by estimating uncertainty in pseudo-label masks. S2C [10] directly transfers the knowledge of SAM [40] to enhance the quality of CAMs. In contrast to these methods, we calibrate pixel-level noise labels within a developed contrastive learning framework. Differing from prior approaches that primarily rely on data augmentation for supervision in WSSS [16], [41], we construct positive and negative training pairs for each category by jointly capturing pixel-level context in both embedding and semantic space. To achieve this, as shown in Fig. 2, we employ shrinking and expansion operations to generate robust pseudo-labels used to supervise contrastive loss. Taking feature and semantic consistency into account, a series of reliable seeds are produced through the shrinking operation in pseudo-labels, ready to update the memory bank for each category. Conversely, a distance-constrained feature similarity module (DFSM) is designed in the expansion operation to convey categorical semantics according to a common principle: pixels with similar feature embeddings and closer spatial distances are more likely to belong to the same object. Two operations complement each other, enabling contrastive loss to progressively correct noisy labels throughout the training phase. It is worthy that, thanks to the shrinking and expansion operations, our developed contrastive loss is capable of exploring multiple positive training pairs, rather than merely relying on single one adopted in data augmentation [42]–[46], which is too weak to learn robust feature representations. This approach allows us to fully leverage the potential of noisy CAMs to refine incomplete activations while calibrating incorrect ones.

In general, our method offers several key advantages. Firstly, the proposed contrastive learning framework facilitates robust representation learning to combat pixel-level noisy activations. By capturing pixel-level context in both embedding and semantic spaces, our method identifies reliable pixels and less discriminative ones using CAMs shrinking and expansion, respectively, thereby enabling the creation of multiple positive training pairs used in a developed contrastive loss functions. Secondly, unlike previous affinity expansion methods that rely on graph models [28] or auxiliary cross-task learning [26] to propagate features, our shrinking and expansion operations are both highly effective and simple to implement. Lastly, our

method presents an elegant framework that is conveniently scalable and generalizes well, allowing for seamless integration with different backbones for WSSS, including ResNet families [10], [32], [47] and Transformers [30], [48]. The most closely related work to our approach is [49], which addresses the segmentation problem under semi-supervised settings where a small amount of fully annotated pixel-level ground truths are available. In contrast, our work tackles a more challenging scenario in which only image-level labels are provided for training segmentation networks. In nutshell, the major contributions of our paper are three-fold:

- We propose a parameter-efficient and backbone-agnostic framework to address pixel-level noisy labels in WSSS using a contrastive learning paradigm. By leveraging the shrinking and expansion operations on CAMs, we can gradually eliminate pixel-level noisy labels, ultimately providing reliable supervision for segmentation tasks.
- Instead of relying on data augmentation, which typically limits the utilization of only one positive training pair in traditional contrastive loss, our method enables to explore multiple positive training pairs. This approach enhances inter-class distinction and intra-class compactness, benefiting for learning more robust feature representations.
- We evaluated our approach on three widely used semantic segmentation datasets: PASCAL VOC 2012 [15], MS COCO 2014 [50], and Cityscapes [51]. The experimental results demonstrate that our method outperforms recent state-of-the-arts, achieving segmentation mIoU scores of 79.5%, 51.9%, and 53.4% on three datasets, respectively.

The remainder of this paper is organized as follows. After a brief introduction of related work in Section II, we elaborate on the details of our method in Section III. Experimental results and ablation studies are given in Section IV, and Section V provides concluding remarks and future work.

II. Related work

In this section, we briefly review recent methods from two perspectives: WSSS, and learning from noisy labels.

A. WSSS using Image-level Labels

The most typical category is *adversarial region erasing* [11], [13], [20]–[23] that drives the classier to pay attention to different parts of an object by hiding the most activated regions step-by-step. As a pioneer work, AE-PSL [22] introduces an iterative framework, where images with erased highly activated regions are fed into next iteration. Another excellent work is [20], which explores the potential of the classifier to mine out additional regions. ECSNet [21] constrains the relationships between CAMs during the erasing process. Chen *et al.* [11] employ a cross-image erasing strategy to alleviate over-expansion issue of CAMs by transferring object prior knowledge. In [23], reliable regions are produced by erasing previous expanded ones. The most recent work is [13], which gradually learns a faithful mask to correct CAM confusions.

Another group of methods to improve the quality of CAM is *context modeling* [24]–[28]. For example, CDA [52] decouples object context in images for WSSS. PLDA [53] captures the context between discriminative and non-discriminative regions by addressing their distribution discrepancies. WS-FCN [25] integrates global context and local content to refine CAMs. SeCo [9] learns object co-occurrence to produce more accurate pseudo-label masks. Moreover, coarse CAMs can be calibrated using affinity matrices that encode the pixel-wise context [26]–[28]. Rather than focusing on object relations within images, some methods emphasize capturing cross-image context. In [24], Wang *et al.* investigates cross-image semantics for WSSS using object co-attention. MCIS [54] utilizes a pair of images to extract inter-image context. MBCC [55] designs a memory bank to save cross-image context, which can be used again to produce class-specific masks for current images.

Instead of focusing on the refinement of CAMs and pseudo-labels, an alternative method formulates WSSS from the perspective of contrastive learning [41]–[46]. For instance, Du *et al.* [41] improves the quality of CAMs using pixel-to-prototype contrast learning. Ke *et al.* [42] improve WSSS through pixel-to-segment contrast, assuming that the segments are predefined. Duan *et al.* [43] explores multi-label prototypes using cross-class and cross-image prototype contrastive learning. In [44], intra-class variations are captured through context-aware prototypes. RCA [45] mines dataset-level context from region-based semantic contrast. Different from these methods that produce CAMs for each semantic category, C2AM [46] generates class-agnostic CAMs for foreground objects and background elements using cross-image contrast.

Most recently, visual-language *foundation models*, such as CLIP [56] and SAM [40], have attracted great attention for WSSS [10], [29]–[31]. With the supervision of text prompts, CLIMS [29] is the first work to leverage CLIP [56] to activate more comprehensive object regions while suppressing closely related backgrounds. CLIP-ES [30] employs the softmax function in CLIP to compute GradCAM, providing reliable pseudo-labels to train subsequent segmentation networks. S2C [10] transfers object knowledge from SAM [40] to the classifier, ultimately enhancing the quality of CAMs. In contrast to these multi-stage methods, WeCLIP [31] and ExCEL [12] adopt a single-stage pipeline, where [31] investigates the potential of the CLIP backbone to extract robust semantic features, while [12] designs a dense patch-text alignment paradigm to explore CLIP knowledge. A comprehensive review on the approaches to WSSS using foundation models can be found in [19].

In contrast to these advanced approaches, our method formulates WSSS from the perspective of *noisy label learning*, where the incomplete and incorrect activations are considered as pixel-level noise. By leveraging the shrinking and expansion operations, these noisy labels can be progressively calibrated to produce more reliable pseudo-labels for segmentation.

B. Learning from noisy labels

Learning from noisy labels has been extensively studied in the field of image classification, which can be roughly divided into four groups: (1) robust network architecture [57]–[60], (2) robust regularization [61]–[66], (3) robust loss functions [67]–[69], and (4) sample selection [70]–[72]. The first group focuses on developing various robust network

architectures [58]–[60] that can effectively model the noise transition matrix associated with a noisy dataset. However, these approaches often struggle when faced with a high noise ratio. To mitigate this issue, the second group turns to robust regularization strategies, such as data augmentation [61], [62], robust early learning [37], [63], [64], and Mixup [65], [66]. In terms of designing robust loss functions, the third group like loss correction [67]–[69] and loss reweighting [73], [74] dynamically adjusts the weights assigned to different samples based on their confidence levels. An alternative approach is to use contrastive learning [75]–[78], which maximizes inter-class disparities while minimizing intra-class distances. Since clean and wrong labels are not available in advance, multi-view data augmentation is frequently employed in contrastive learning. Nonetheless, creating a reliable metric to identify noisy samples remains a challenge, potentially leading to an accumulation of errors from incorrect selections. To minimize these false corrections, recent studies have adopted sample selection techniques [70]–[72] to isolate label-corrected examples from label-noisy ones. In particular, multi-round learning [79] iteratively refines the selected examples, while co-training methods [80]–[82] utilize multiple networks to collaborate effectively with each other.

In contrast to image classification tasks, there are very few studies [10], [18], [37]–[39] that specifically address pixel-level noisy labels for WSSS. This field initially emerged from medical image analysis [83], [84]. LVCNet [84] incorporates domain-specific prior knowledge to improve robustness. COPLE-Net [83] inherits U-Net architecture and employs dice loss to segment medical images. By separating the early learning phase and the memorization phase, ADELE [37] is the first work against image-level noisy labels. DuPL [18] presents a dual-path network, in which each sub-network contributes to progressively correcting noisy labels. BECO [38] introduces a co-training paradigm that designs two robust loss functions to separate the reliable and unreliable components of pixel-level pseudo-labels. Based on the observation that uncertain pixels are closely related to the response scale, URN [39] estimates pixel uncertainty across different views to mitigate the impact of pixel-level noisy labels.

Unlike these impressive methods, we develop a contrastive learning framework to address pixel-level noisy labels in WSSS. Our method does not rely on multi-view data augmentation, which typically uses only a single positive pair [41], [42], [45], [46] in loss function. Instead, it allows for the use of multiple positive training pairs to learn more robust feature representations. Moreover, rather than focusing on building robust segmentation networks [37], [38] under the supervision of pixel-level noisy labels, we concentrate on creating more accurate and reliable pseudo-label masks for training subsequent segmentation networks.

III. Our Method

The whole pipeline of our method is shown in Fig. 2, which consists of three main components: (1) *shrinking operation* that filters out reliable seed pixels from noisy CAMs; (2) *expansion operation* that seeks more potential object regions; and

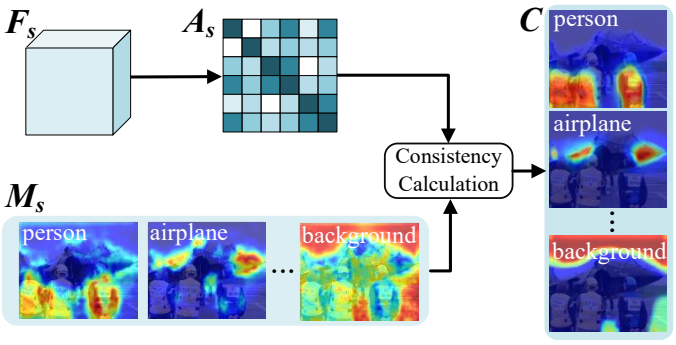


Fig. 3. The detailed architecture of FSCM. The heat maps of $\boldsymbol{M}_s$ represent the activated degrees of the pixels, where deeper red color denotes higher activated level, and deeper blue color indicates lower activated level. For clarity, the categorical labels have also been superimposed in the top-left corner of each image. (Best viewed in color)

(3) *developed contrastive learning* that progressively calibrates noisy labels at the pixel level in CAMs. Immediately below, we introduce each component in detail.

A. Shrinking Operation

Identifying reliable pixels or regions in CAMs is crucial for WSSS [21], [22], [27], [28]. Some previous approaches [21]–[23] implicitly capture contextual information between initialized and erased CAMs, yet they often ignore semantic context to produce reliable regions. Lots of alternative efforts have been proposed to find reliable pixels by exploring pixel-level interactions [27], [28]. These methods, however, typically require learning auxiliary affinity attention sub-networks that involve large amount of computations. In contrast, this section describes a simple shrinking operation that leverages pixel-level context in both embedding and semantic space to create reliable seed regions from noisy CAMs.

The detailed structure of the shrinking operation is shown in the blue area of Fig. 2. Let $\boldsymbol{F}_s \in \mathbb{R}^{C\times H\times W}$ be encoded feature embeddings produced by a pre-trained backbone network (e.g., ResNet families [24], [25], [47] or Transformers [16], [85], [86]), where H, W, and C are height, width, and number of feature channels of $\boldsymbol{F}_s$, respectively. Following [14], [32], the first step in the shrinking operation is to project $\boldsymbol{F}_s$ into noisy CAMs $\boldsymbol{M}_s \in \mathbb{R}^{(K+1)\times H\times W}$ for K foreground objects and an extra background region through a classification head. Subsequently, the produced $\boldsymbol{M}_s$, together with encoded feature embeddings $\boldsymbol{F}_s$, pass through a feature-semantic consistency module (FSCM), where pixel-wise feature and semantic context are fully considered. Finally, the reliable and consistent pseudo-labels $\boldsymbol{E}_s \in \mathbb{R}^{(K+1)\times H\times W}$ are generated by applying an erosion operation to eliminate boundary noise and tiny foreground regions. The selected reliable pixels are then employed to produce robust prototypes, stored in a memory bank for developed contrastive learning. In the following, we elaborate on the details of FSCM, as the core component in our shrinking operation.

1) FSCM: As illustrated in Fig. 3, the goal of FSCM is to encourage pixels activated with the same category ought to have similar feature representations. To this end, $\boldsymbol{F}_s$ is first flattened into a 2-D sequence $\boldsymbol{X} \in \mathbb{R}^{HW\times C}$, facilitating the

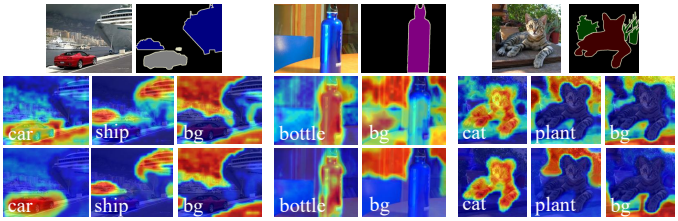


Fig. 4. Visual comparison of CAMs before and after applying FSCM. From top to bottom are input images and the associated ground-truths, noisy CAMs $\boldsymbol{M}_s$, and reliable CAMs $\boldsymbol{C}$, where areas with a deeper red indicate higher activation levels, while those in blue represent lower activations. For clarity, the categorical labels have also been superimposed in the bottom-left corner of each images. (Best viewed in color)

computations of following symmetric similarity matrix $\boldsymbol{A}_s \in \mathbb{R}^{HW\times HW}$:

$$\boldsymbol{A}_s = \sigma(\boldsymbol{X}\boldsymbol{X}^\top) \tag{1}$$

where $\sigma(\cdot)$ stands for softmax function. After that, the similarity matrix $\boldsymbol{A}_s$ is reshaped into a 3-D tensor $\boldsymbol{T} \in \mathbb{R}^{HW\times H\times W}$, where the i^{th} channel $\boldsymbol{T}_i \in \mathbb{R}^{H\times W}$ indicates the similarity scores of all feature positions with respect to pixel i.

Next, to effectively capture pixel-level context in both feature and semantic space, a consistency score $\boldsymbol{c}^k(i)$ for pixel i in k^{th} category is calculated using an element-wise product "$\circ$" from the semantic guidance of $\boldsymbol{M}_s$ and the feature similarities encoded by $\boldsymbol{T}_i$:

$$\boldsymbol{c}^k(i) = \frac{\sum_{j=1}^{HW} \boldsymbol{M}_s^k(j) \circ \boldsymbol{T}_i(j)}{\sum_{j=1}^{HW} [\boldsymbol{M}_s^k(j) + \boldsymbol{T}_i(j) - \boldsymbol{M}_s^k(j) \circ \boldsymbol{T}_i(j)]} \tag{2}$$

where $\boldsymbol{M}_s^k \in \mathbb{R}^{H\times W}$ refers to the k^{th} channel of $\boldsymbol{M}_s$, and the index j denotes all possible positions in both $\boldsymbol{M}_s^k$ and $\boldsymbol{T}_i$. Collecting $\boldsymbol{c}^k(i)$ for all positions i and all categories k forms more reliable CAMs $\boldsymbol{C} \in \mathbb{R}^{(K+1)\times H\times W}$.

Unlike self-attention [27], [28] that requires feature projection to encode global context, the calculation of $\boldsymbol{A}_s$ in Eq. (1) is parameter-free, facilitating to save both model size and computational costs. Moreover, since no parameters are learned in FSCM, the consistency score defined in Eq. (2) encourages the encoder to learn more robust feature representations, which not only highlights the shared components between $\boldsymbol{M}_s^k$ and $\boldsymbol{T}_i$, but also constrains them to be as consistent as possible, thereby ensuring the reliability of the selected seed pixels.

To further demonstrate the effectiveness of FSCM, Fig. 4 illustrates some visual examples that compare $\boldsymbol{M}_s$ and $\boldsymbol{C}$. It is evident that the activated area of noisy $\boldsymbol{M}_s$ shrinks to more reliable $\boldsymbol{C}$, concentrating more on foreground object instances. In particular, nearly all seed pixels are located within object regions, such as "ship", "bottle", and "cat" in these examples.

2) Pseudo-labels Generation: Given reliable $\boldsymbol{C}$, the final step of the shrink operation is to produce pseudo-label masks $\boldsymbol{E}_s \in \mathbb{R}^{(K+1)\times H\times W}$. Following [20]–[23], those pixels, whose confidence level are lower than a pre-defined threshold $\eta \in [0, 1]$, are first treated as "unknown", then $\boldsymbol{E}_s$ is generated by assigning the highest probability scores in $\boldsymbol{C}$ for each pixel:

$$\boldsymbol{E}_s^k(i) = \begin{cases} 1, & \text{if } \boldsymbol{C}^k(i) \geq \eta \;\;\&\&\;\; k = \arg\max_k \boldsymbol{C}^k(i) \\ 0, & \text{else} \end{cases} \tag{3}$$

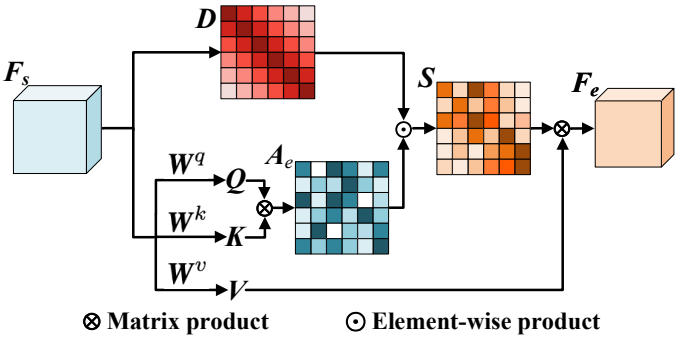


Fig. 5. The detailed architecture of DFSM. (Best viewed in color)

where $\boldsymbol{E}_s^k(i)$ denotes the k^{th} channel for pixel i in $\boldsymbol{E}_s$. Finally, an erosion operation is applied to further shrink $\boldsymbol{E}_s$, ensuring the produced pseudo-label masks are fully reliable.

B. Expansion Operation

Expanding CAMs is widely used in WSSS to discover more potential object regions [11], [13], [23]. Most previous efforts iteratively accumulate multiple CAMs through region erasing [20]–[22]. These methods, however, need caution to avoid over-estimated activations. An alternative approach [23] seeks under-estimated activations by learning an offset network using deformable convolution. In contrast, we design a DFSM to expand CAMs based on following fundamental principles: pixels with similar feature embeddings and closer spatial distances are more likely to belong to the same object.

The detailed structure of expansion operation is depicted in the orange area of Fig. 2. Apart from DFSM, it has nearly the duplicated architecture compared with shrinking operation, where the classification head and FSCM are shared to produce pseudo-label masks $\boldsymbol{E}_e \in \mathbb{R}^{(K+1)\times H\times W}$. Note FSCM is also adopted in expanding operation, as we expect the expanded pixels to be reliable as well, benefiting to provide robust supervision for forthcoming contrastive learning. In the following, we introduce the details of DFSM.

1) DFSM: Fig. 5 presents the detailed structure of DFSM, which is mainly inspired by affinity attention [27], [28]. However, besides taking feature similarities into account, DFSM also considers location constraint, which is often ignored by [27], [28]. Concretely, the feature similarities are encoded by a spatial attention map $\boldsymbol{A}_e \in \mathbb{R}^{HW\times HW}$, where the pixel-wise interactions are fully explored. In addition, DFSM designs an extra location map $\boldsymbol{D} \in \mathbb{R}^{HW\times HW}$, which restricts positional relationships among different pixels.

Given the encoded feature $\boldsymbol{F}_s$, we first flatten it into a 2-D sequence $\boldsymbol{X} \in \mathbb{R}^{HW\times C}$, then three linear transformations $\{\boldsymbol{W}^k, \boldsymbol{W}^q, \boldsymbol{W}^v\} \in \mathbb{R}^{C\times C}$ are learned to project input sequence $\boldsymbol{X}$ into key, query, and value $\{\boldsymbol{K}, \boldsymbol{Q}, \boldsymbol{V}\} \in \mathbb{R}^{HW\times C}$:

$$\boldsymbol{K} = \boldsymbol{X}\boldsymbol{W}^k, \qquad \boldsymbol{Q} = \boldsymbol{X}\boldsymbol{W}^q, \qquad \boldsymbol{V} = \boldsymbol{X}\boldsymbol{W}^v \tag{4}$$

After that, the pixel-wise mutual interactions are calculated using a matrix product between $\boldsymbol{K}$ and $\boldsymbol{Q}$, producing feature similarities map $\boldsymbol{A}_e$:

$$\boldsymbol{A}_e = \sigma(\frac{\boldsymbol{Q}\boldsymbol{K}^\top}{\sqrt{d_k}}) \tag{5}$$

where $\sigma(\cdot)$ stands for softmax function and d_k is a scale factor determined by the feature dimension of $\boldsymbol{K}$.

On the other hand, let (x_i, y_i) and (x_j, y_j) be position coordinates for pixel i and pixel j, respectively, then their absolute spatial distance $d(i,j)$ is calculated using the Euclidean distance:

$$d(i,j) = \sqrt{(x_i - x_j)^2 + (y_i - y_j)^2} \tag{6}$$

However, d_{ij} changes along with the variance of the feature resolutions. We thus recompute it with respect to the diagonal length of feature resolution, leading to a normalized relative distance $\tilde{d}(i,j)$:

$$\tilde{d}(i,j) = \frac{d(i,j)}{\sqrt{W^2 + H^2}} \tag{7}$$

Intuitively, if two pixels have long distance, they are not likely to belong to same object, and vice versa. As a result, we define $\boldsymbol{D}(i,j)$ as an exponential inverse function of $\tilde{d}(i,j)$:

$$\boldsymbol{D}(i,j) = \exp\left(-\tilde{d}(i,j)\right) \tag{8}$$

By collecting $\boldsymbol{D}(i,j)$ for all pairs of positions (i,j), we form a symmetric location map $\boldsymbol{D}$. Notably, calculating $\boldsymbol{D}$ is essentially different from positional encoding in self-attention [27], [28] that addresses the inherent limitation of disorder property of tokenlized embeddings. Instead, the purpose of $\boldsymbol{D}$ is to ensure that tokens with close spatial positions should have similar feature representations, and vice versa. Moreover, positional encoding adopts a sinusoidal fashion, while we employ normalized Euclidean distance.

Finally, as our DFSM considers both feature similarities and location constraint, a feature weight matrix $\boldsymbol{S} \in \mathbb{R}^{HW \times HW}$ is defined using an element-wise product between $\boldsymbol{A}_e$ and $\boldsymbol{D}$:

$$\boldsymbol{S} = \boldsymbol{A}_e \circ \boldsymbol{D} \tag{9}$$

However, as each element of $\boldsymbol{D}$ is not restricted to a certain range, directly integrating it with $\boldsymbol{A}_e$ may be inappropriate. As the value of each element in $\boldsymbol{A}_e$ is at the range of $[0,1]$, the location map $\boldsymbol{D}$ has to be normalized into the same value range of $\boldsymbol{A}_e$:

$$\tilde{\boldsymbol{D}} = \frac{\boldsymbol{D} - \boldsymbol{D}_{min}}{\boldsymbol{D}_{max} - \boldsymbol{D}_{min}} \tag{10}$$

where $\boldsymbol{D}_{min}$ and $\boldsymbol{D}_{max}$ are the minimum and maximum value of $\boldsymbol{D}$, respectively. Then, Eq. (9) can be rewritten as:

$$\boldsymbol{S} = \boldsymbol{A}_e \circ \tilde{\boldsymbol{D}} \tag{11}$$

After that, the output feature $\boldsymbol{F}_o \in \mathbb{R}^{HW \times C}$ is calculated using a matrix product between weight matrix $\boldsymbol{S}$ and value $\boldsymbol{V}$:

$$\boldsymbol{F}_o = \boldsymbol{S}\boldsymbol{V} \tag{12}$$

which is reshaped to a 3-D feature tensor $\boldsymbol{F}_e \in \mathbb{R}^{C \times H \times W}$ with equal dimension with respect to input feature $\boldsymbol{F}_s$ that is ready for following CAM $\boldsymbol{M}_e$ creation, as shown in Fig. 2.

Guided from the supervision of image classification, DFSM conveys categorical semantics within the constraint of feature similarities and relative distance, helping the classifier identify more under-activated pixels. Fig. 6 shows some visual examples of $\boldsymbol{M}_s$ and $\boldsymbol{M}_e$, where $\boldsymbol{M}_e$ highlights more under-activated regions, resulting in expansions into less discriminative parts, such as "horse", "train" and "cat" in these examples.

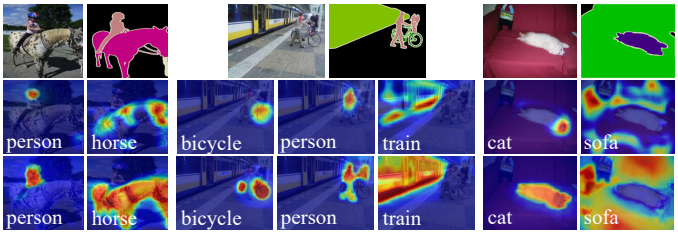


Fig. 6. Visual comparison of CAMs before and after applying DFSM. From top to bottom are input images and the associated ground-truths, noisy CAMs $\boldsymbol{M}_s$, and expanded CAMs $\boldsymbol{M}_e$, where areas with a deeper red indicate higher activation levels, while those in blue represent lower activations. For clarity, the categorical labels have been also superimposed in the bottom-left corner of each images. (Best viewed in color)

C. Developed Contrastive Learning

Contrastive learning is commonly-used in prototype learning for WSSS [32], [41], [43]–[45]. As pixel-wise supervision is not available, previous methods [41], [46] typically employ linear or nonlinear transformations to produce multi-view augmented images. However, since pseudo-labels have been generated in $\boldsymbol{E}_s$ and $\boldsymbol{E}_e$, respectively, we are able to train contrastive loss under the supervision of $\boldsymbol{E}_s$ and $\boldsymbol{E}_e$, without relying on data augmentation.

As shown in the green area of Fig. 2, our developed contrastive learning framework consists of two major components: a memory bank that stores the prototypes for each semantic category, and developed contrastive loss that corrects noisy pixel-level labels in CAM. Concretely, after receiving $\boldsymbol{E}_s$ in the shrink operation, a memory bank $\boldsymbol{B} \in \mathbb{R}^{C \times N \times (K+1)}$ is constructed and iteratively updated throughout the entire training phase, where N is the number of representative prototypes for each category. On the other hand, let $\boldsymbol{F} \in \mathbb{R}^{C \times M_k \times (K+1)}$ be selected feature set by masking $\boldsymbol{E}_e$ onto feature maps $\boldsymbol{F}_s$, where M_k is the number of selected features for k^{th} category. By pairing each selected feature with prototypes, the positive and negative training pairs are constructed, which can be utilized to supervise our proposed contrastive loss, thereby fostering robust feature representations that progressively eliminate noise in the CAMs. Immediately below, we elaborate on the details of memory bank construction and the definition of our developed contrastive loss.

1) Memory Bank Initialization and Update: This section introduces a non-parametric and dynamic memory bank designed to store category-specific prototype features. At the beginning of training phase, a series of feature embeddings of a specific category are selected by masking $\boldsymbol{E}_s$ onto feature maps $\boldsymbol{F}_s$. Then, the associated prototypes are initialized using K-means algorithm [87], where each cluster center corresponds to a prototype. During each training iteration, let $\boldsymbol{f}^k$ be a newly selected feature for k^{th} category. We first determine $\boldsymbol{f}^k$ belongs to which prototype based on their distance in feature space:

$$i = \arg\min_i ||\boldsymbol{f}^k - \boldsymbol{b}_i^k||_2, \quad \forall i \tag{13}$$

where $||\cdot||_2$ represents ℓ_2 norm. Thereafter, inspired by [45], the prototype $\boldsymbol{b}_i^k$ is updated by smoothly integrated with $\boldsymbol{f}^k$ using a momentum-based update strategy:

$$\boldsymbol{b}_i^k \leftarrow \gamma \boldsymbol{b}_i^k + (1-\gamma)\boldsymbol{f}^k \tag{14}$$

where γ is a non-negative momentum parameter at the range of [0, 1]. This momentum-based strategy ensures a smooth temporal evolution of the prototype features, allowing the preservation of historical information while incorporating new observations. Throughout the training process, the updated prototypes can capture a more accurate and comprehensive diversity of each category, thereby enhancing the robustness and consistency of the learned representations.

2) Developed Contrastive Loss: Since the pseudo-labels are available in the memory bank $\boldsymbol{B}$ and selected features $\boldsymbol{F}$, our method indeed falls into the regime of supervised contrastive learning. The first step is to build training pairs based on $\boldsymbol{B}$ and $\boldsymbol{F}$. Let $\boldsymbol{b}_j^k \in \boldsymbol{B}$ be j^{th} prototype, also known as *anchor*, for k^{th} category, and $\boldsymbol{f}_i^l \in \boldsymbol{F}$ be i^{th} selected feature for l^{th} category, both of which is a C-dimensional vector. As a result, the negative and positive training pairs are constructed by simply judging whether their labels are consistent or not:

$$(\boldsymbol{z}_i, \boldsymbol{z}_j^+) \mid \boldsymbol{z}_i = \frac{\boldsymbol{f}_i^l}{||\boldsymbol{f}_i^l||_2},\ \boldsymbol{z}_j^+ = \frac{\boldsymbol{b}_j^k}{||\boldsymbol{b}_j^k||_2},\ k = l,\ \forall(i,j) \tag{15}$$

$$(\boldsymbol{z}_i, \boldsymbol{z}_m^-) \mid \boldsymbol{z}_i = \frac{\boldsymbol{f}_i^l}{||\boldsymbol{f}_i^l||_2},\ \boldsymbol{z}_m^- = \frac{\boldsymbol{b}_j^k}{||\boldsymbol{b}_j^k||_2},\ k \neq l,\ \forall(i,j) \tag{16}$$

where $||\cdot||_2$ denotes ℓ_2 norm. Since there are $K+1$ categories and each category has N prototypes, each selected feature $\boldsymbol{f}_i^l$ has a total of KN negative training pairs.

Recall that the objective of our developed contrastive loss is to enhance the similarities between the features in $\boldsymbol{B}$ and $\boldsymbol{F}$ with the same class, while minimizing the similarities to the features that have different semantic labels, therefore, the contrastive loss $\mathcal{L}_{con}^k$ for k^{th} category is defined as:

$$\mathcal{L}_{con}^k = -\frac{1}{N}\sum_{j=1}^{N}\log\frac{\sum_{i=1}^{M_k}\exp\{\boldsymbol{z}_i\cdot\boldsymbol{z}_j^+/\tau\}}{\sum_{i=1}^{M_k}\exp\{\boldsymbol{z}_i\cdot\boldsymbol{z}_j^+/\tau\}+\sum_{i=1}^{M_k}\sum_{m=1}^{KN}\exp\{\boldsymbol{z}_i\cdot\boldsymbol{z}_m^-/\tau\}} \tag{17}$$

where "$\cdot$" denotes inner product, and τ is a non-negative scalar temperature parameter. Finally, the total contrastive loss for all categories is equally combined as:

$$\mathcal{L}_{con} = \frac{1}{K+1}\sum_{k=1}^{K+1}\mathcal{L}_{con}^k \tag{18}$$

In contrast to conventional contrastive loss [42], [45], [46] that solely depends on one positive training pair, the loss defined in Eq. (17) allows us to explore multiple positive training pairs as well as a large number of negative training pairs, both of which are beneficial for learning more robust feature representations. Moreover, previous contrastive learning methods [17], [45] are prone to the problem of weak and noisy pseudo-labels, posing great challenges in learning robust representations. However, our training algorithm corrects the pixel-level noisy label in a recursive manner: contrastive loss first enhances feature representations for shrinking and expansion operations to produce high-quality CAMs; inversely, the generated CAMs provide more reliable training pairs used to supervise our contrastive loss.

3) Total Loss: As shown in Fig. 2, there are two losses used to supervised our WSSS method. The first is our developed contrastive loss $\mathcal{L}_{con}$ defined in Eq. (18), while the second is classification loss $\mathcal{L}_{cls}$, which is defined using cross-entropy loss [20]–[23]. Thus, the total loss is a weighted sum defined between $\mathcal{L}_{cls}$ and $\mathcal{L}_{con}$:

$$\mathcal{L}_{total} = \mathcal{L}_{cls} + \lambda\mathcal{L}_{con} \tag{19}$$

whereλ is a non-negative parameter that controls the individual contribution of $\mathcal{L}_{cls}$ and $\mathcal{L}_{con}$.

IV. Experiments

To evaluate our method, we conducted exhaustive experiments on three semantic segmentation datasets: PASCAL VOC 2012 [15], MS COCO 2014 [50], and Cityscapes [51], where our approach and most representative state-of-the-art WSSS networks are compared in terms of segmentation accuracy. In addition, a series of ablation studies were carried on to reveal the potential impact of various components, and gain a deeper understanding of the underlying behavior of our method.

A. Dataset and Evaluation Metric

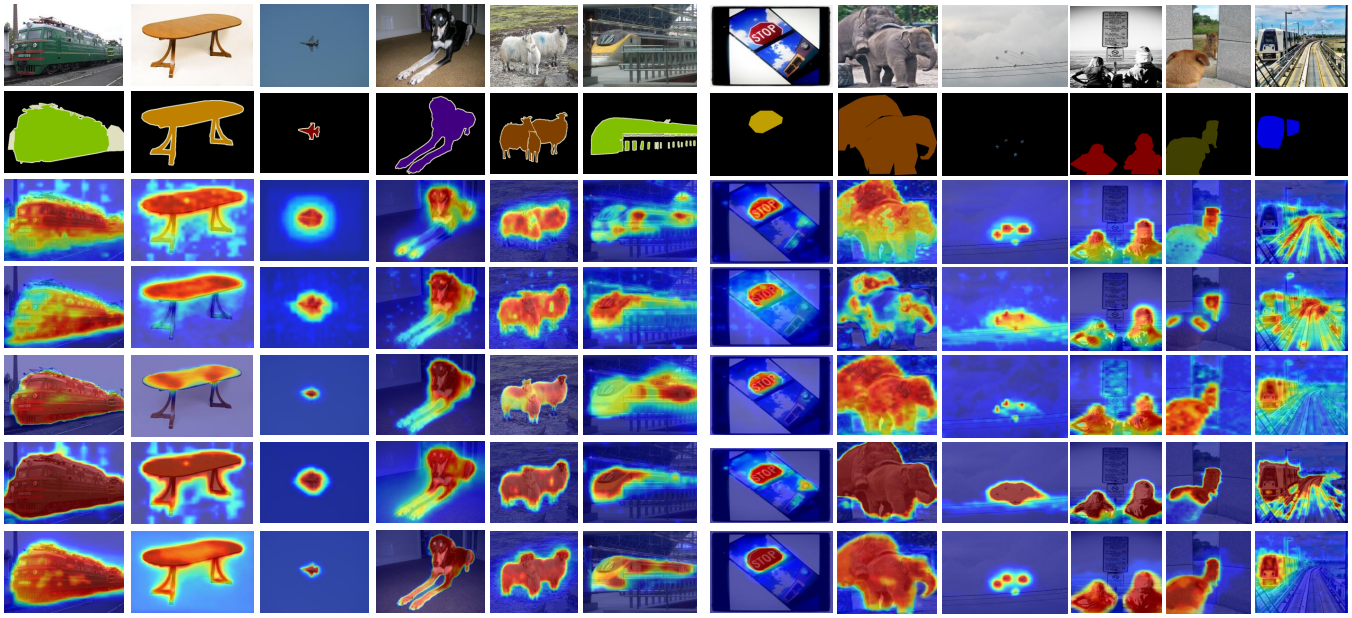
1) PASCAL VOC 2012: As a widely-used benchmark for WSSS, PASCAL VOC 2012 dataset [15] consists of a total of 4,369 images, with 1,464 images in the training set, 1,449 images in the validation set, and 1,456 images in the test set. The dataset provides pixel-level annotations for 20 foreground classes and 1 background class. In line with standard practices [17], [32], we enhance the training set by incorporating additional annotated images from the SBD [88] dataset, resulting in a total of 10,582 images in the augmented training set, while keeping the validation and test sets unchanged.

2) MS COCO 2014: MS COCO 2014 [50] is a more challenging dataset that involves more object categories and training images. It contains 80 foreground classes and 1 background class, including approximately 80K training images and 40K validation images.

3) Cityscapes: Unlike PASCAL VOC [15] and MS COCO [50], Cityscapes [51] is a small-scale dataset, which is split to 2,975/500/1,525 for training, validation, and testing, respectively. It contains a total of 30 classes, and only 19 classes are used for public assessment.

Note although pixel-level ground truths are both available in two datasets, our method only utilizes image-level labels from the training set, and its performance is evaluated on the validation and test sets.

4) Evaluation Metric: Following [20]–[22], we employed the popular mean intersection-over-union (mIoU) across all classes between the segmentation outputs and the pixel-wise ground-truths to assess the segmentation performance. Moreover, following [17], [32], [44], we also used mIoU to evaluate the quality of CAMs and pseudo-label masks. To ensure a fair comparison, we submit the semantic segmentation outputs of PASCAL VOC test set to the official evaluation server. On the other hand, the widely used floating point operations (FLOPs) and model size (parameters) are used to measure computational complexity of our method.



(a) PASCAL VOC 2012 (b) MS COCO 2014

Fig. 7. Visual comparison of CAMs on (a) PASCAL VOC 2012 and (b) MS COCO 2014 datasets. From top to bottom are original images, corresponding ground truth, CAMs produced from SIPE [32], CLIP-ES [30], POT [33], S2C [10], and our method. Compared to these state-of-the-art baselines, our approach obtains more complete, consistent, and accurate CAMs while effectively suppressing background activated noise. In the last column of two datasets, we also exhibit a failure example, where we still achieve better results compared to selected baselines. (Best viewed in color)

B. Implementation Details

1) Training Settings: To evaluate the scalability and generalization of our method, we employ ResNet families [10], [32], [47] and Transformers [30], [48], both pre-trained on ImageNet [89] with output stride of 16, as our backbone networks. For first two datasets, the input images are resized to a fixed resolution of 512×512, and the data augmentation strategy follows the setup proposed by [17], [32], including random flipping, scaling, and cropping. The model is trained using the stochastic gradient descent optimizer, with an initial learning rate of 1e-2 for the backbone and 1e-1 for the other components. The momentum and weight decay are set to 0.9 and 5e-4, respectively. Following [20]–[22], our network is trained for a total of 10 epochs, during which the first 2 epochs adopt a warm-up strategy that only applies cross-entropy loss, i.e., $\lambda = 0$ in Eq. (19). Particularly, we also trained S2C [10] for 10 epochs to ensure fairness. The learning rate is adjusted using a poly scheduler, with a power decay of 0.9. The batch size is set to 32 for the PASCAL VOC dataset and 8 for the MS COCO dataset, respectively. To ensure a diverse distribution and to prevent excessive computational overhead, the size of the per-class memory bank is set to 500. Following [17], [30], [32], the enhanced CAMs are used to produce pseudo-labels, which serve as pixel-level supervision for training the subsequent DeepLabV2 [90]. Consistent with [20]–[22], the segmentation results are ultimately refined using Dense CRF model. Other hyper-parameters are empirically determined as follows: confidence threshold $\eta = 0.8$ in Eq. (3); momentum weight $\gamma = 0.9$ in Eq. (14), temperature coefficient $\tau = 0.5$ in Eq. (17), and loss weight $\lambda = 0.1$ in Eq. (19), respectively. Our source code is publicly available online at https://github.com/njupt-quanzhou/LNL-for-WSSS.

2) Selected State-of-the-Art Baselines: To show the advantages of our method, we selected 29 state-of-the-art baselines for comparison, including both single-stage and multi-stage pipelines. The first group includes AFA [27], ToCo [85], TSCD [91], CRME [13], ExCEL [12], and WeCLIP [31]. On the other hand, the second one contains region erasing methods [14], [20], [21], context modeling approaches [41], [48], [52], [54], contrastive learning networks [32], [41]–[45], and foundation models [29], [30], [33], [92], [93]. Particularly, those WSSS approaches [10], [18], [37]–[39] addressing noisy label learning are also invited as baselines.

C. Comparisons With State-of-the-art WSSS Baselines

1) Performance of CAMs and Pseudo-Label Masks: We begin by evaluating the quality of various localization maps: CAMs, their enhanced versions after CRF refinement, and pseudo-label masks. Table I reports the comparison results between our method and state-of-the-art baselines on the PASCAL VOC 2012 training set. It reveals that our method consistently boosts performance by remarkable margins when combined with [10]. Moreover, when only image-level labels are available, our method significantly outperforms other baselines, achieving mIoU scores of 79.1%, 80.8%, and 84.3% in terms of CAMs, CRF, and Mask, respectively.

To further demonstrate the effectiveness of our method, Fig. 7 also exhibits the qualitative visual comparison of CAMs between several selected baselines [10], [30], [32], [33] and our approach on PASCAL VOC and MS COCO datasets. It demonstrates that whether involving single or multiple instances, our method is able to correctly and consistently activate entire object regions (e.g., “table”, “airplane”, “sheep”, “sign”, “elephant”, “person”, and “dog” in the examples

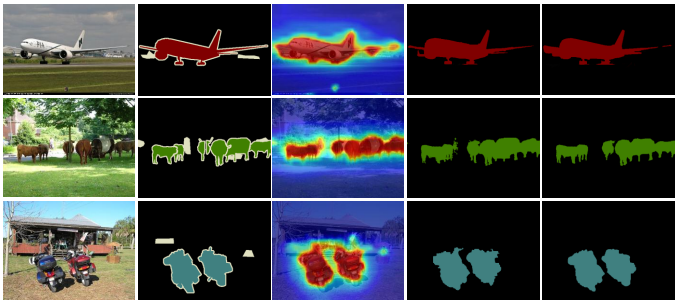


Fig. 8. Visual examples of localization maps on PASCAL VOC 2012 set. From left to right are input images, corresponding ground truth, CAMs, CRF refinement outputs, and the pseudo-label masks. (Best viewed in color)

TABLE I
LOCALIZATION MAPS AND PSEUDO-LABEL MASKS COMPARISON WITH THE STATE-OF-THE-ART APPROACHES IN TERMS OF MIOU (%) ON THE PASCAL VOC 2012 TRAIN SETS. "$\mathcal{I}$", "$\mathcal{S}$", AND "$\mathcal{L}$" STAND FOR SUPERVISION FROM IMAGE-LEVEL LABELS, SALIENCY MAPS, AND LANGUAGE PROMPTS, RESPECTIVELY. "-" DENOTES THE RESULTS ARE NOT REPORTED. THE BEST RESULTS ARE HIGHLIGHTED IN BOLD NUMBER.

Method	Year	Sup.	CAMs	CRF	Mask
CSENet [20]	ICCV21	$\mathcal{I}$	56.0	62.8	-
ECSNet [21]	ICCV21	$\mathcal{I}$	56.6	58.6	-
CDA [52]	ICCV21	$\mathcal{I}$	58.4	-	66.4
SIPE [32]	CVPR22	$\mathcal{I}$	58.6	64.7	68.0
AFA [27]	CVPR22	$\mathcal{I}$	55.4	-	68.7
P2PC [41]	CVPR22	$\mathcal{I}$	61.5	64.0	69.2
CLIMS [29]	CVPR22	$\mathcal{I}+\mathcal{L}$	56.6	62.4	70.5
MCT [48]	CVPR22	$\mathcal{I}$	61.7	68.5	72.1
RCA [45]	CVPR22	$\mathcal{I}+\mathcal{S}$	-	-	73.2
BECO [38]	CVPR23	$\mathcal{I}$	65.5	-	70.9
ToCo [85]	CVPR23	$\mathcal{I}$	71.6	-	72.2
CLIP-ES [30]	CVPR23	$\mathcal{I}+\mathcal{L}$	68.2	72.0	75.0
DuPL [18]	CVPR24	$\mathcal{I}$	-	-	75.1
CPAL [44]	CVPR24	$\mathcal{I}+\mathcal{L}$	71.9	-	75.8
PSDPM [92]	CVPR24	$\mathcal{I}+\mathcal{L}$	-	-	77.3
MuP-VSS [43]	CVPR25	$\mathcal{I}$	71.7	72.6	74.1
POT [33]	CVPR25	$\mathcal{I}+\mathcal{L}$	75.0	-	79.3
S2C [10]	CVPR24	$\mathcal{I}$	76.7	79.3	82.5
+Ours	-	$\mathcal{I}$	**79.1**(↑2.4)	**80.8**(↑1.5)	**84.3**(↑1.8)

of two datasets), while suppressing incorrect activations in background regions, which is attributed to our capability of eliminating pixel-level noisy labels. In the final column of two datasets, we also show two failure examples, where the region of "train" is not entirely and consistently highlighted, probably due to its significant appearance diversity and extremely clutter background. Even so, compared with other baselines, we still effectively restrain background noisy activations. Fig. 8 also illustrates some visual qualitative results of CAMs, CRF refinement outputs, and final pseudo-label masks, where object boundaries and shapes are well delineated.

2) Segmentation Performance on PASCAL VOC 2012: Table II reports the segmentation results compared with single-stage and multi-stage baselines on PASCAL VOC 2012 val and test sets. Whether a ResNet [10], [47] or Transformer [30], [48] backbone is adopted, our method is able to consistently improve performance. In particular, integrating our method with S2C [10] leads to the highest mIoU scores of 80.1% and 79.5% on val and test set, respectively. We also discover that the largest mIoU gains (e.g., 3.3% and 3.8%) are obtained when equipping with CLIP-ES [30], probably because of using

TABLE II
SEGMENTATION RESULTS COMPARISON WITH THE STATE-OF-THE-ART APPROACHES IN TERMS OF MIOU (%) ON PASCAL VOC 2012 VAL AND TEST SETS. "$\mathcal{I}$", "$\mathcal{S}$", AND "$\mathcal{L}$" STAND FOR SUPERVISION FROM IMAGE-LEVEL LABELS, SALIENCY MAPS, AND LANGUAGE PROMPTS. THE BASELINES THAT ADDRESS NOISY LABEL LEARNING ARE HIGHLIGHTED IN GRAY COLOR. THE BEST RESULTS ARE HIGHLIGHTED IN BOLD NUMBER.

Method	Year	Backbone	Sup.	Val	Test
Single-stage WSSS:					
AFA [27]	CVPR22	MiT-B1	$\mathcal{I}$	66.0	66.3
TSCD [91]	AAAI23	MiT-B1	$\mathcal{I}$	67.3	67.5
ToCo [85]	CVPR23	ViT-B	$\mathcal{I}$	71.1	72.2
DuPL [18]	CVPR24	ViT-B	$\mathcal{I}$	73.3	72.8
WeCLIP [31]	CVPR24	ViT-B	$\mathcal{I}+\mathcal{L}$	76.4	77.2
CRME [13]	CVPR25	ViT-B	$\mathcal{I}$	75.5	75.9
ExCEL [12]	CVPR25	ViT-B	$\mathcal{I}+\mathcal{L}$	78.4	78.5
Multi-stage WSSS:					
MCIS [54]	ECCV20	R101	$\mathcal{I}+\mathcal{S}$	66.2	66.9
CDA [52]	ICCV21	WR38	$\mathcal{I}$	66.1	66.8
ECSNet [21]	ICCV21	WR38	$\mathcal{I}$	66.6	67.6
CSENet [20]	ICCV21	WR38	$\mathcal{I}$	68.4	68.2
PSCL [42]	ICLR21	R101	$\mathcal{I}$	69.5	71.6
P2PC [41]	CVPR22	WR38	$\mathcal{I}$	67.7	67.4
CLIMS [29]	CVPR22	R50	$\mathcal{I}+\mathcal{L}$	69.3	68.7
ADELE [37]	CVPR22	WR38	$\mathcal{I}$	69.3	68.8
URN [39]	AAAI22	R101	$\mathcal{I}$	69.5	69.7
SIPE [32]	CVPR22	R101	$\mathcal{I}$	68.8	69.7
RCA [45]	CVPR22	WR38	$\mathcal{I}+\mathcal{S}$	71.1	71.6
BECO [38]	CVPR23	R50	$\mathcal{I}$	72.1	71.8
CPAL [44]	CVPR24	R101	$\mathcal{I}+\mathcal{L}$	74.5	74.7
PSDPM [92]	CVPR24	R101	$\mathcal{I}+\mathcal{L}$	74.1	74.9
MuP-VSS [43]	CVPR25	WR38	$\mathcal{I}$	73.6	74.7
POT [33]	CVPR25	R50	$\mathcal{I}+\mathcal{L}$	76.1	76.9
BAS [47]	IJCV24	R50	$\mathcal{I}$	69.6	69.9
+Ours	-	R50	$\mathcal{I}$	72.5(↑2.9)	72.1(↑2.2)
CLIP-ES [30]	CVPR23	ViT-B	$\mathcal{I}+\mathcal{L}$	71.1	71.4
+Ours	-	ViT-B	$\mathcal{I}+\mathcal{L}$	74.4(↑3.3)	74.2(↑3.8)
MCT [48]	CVPR22	DeiT-S	$\mathcal{I}$	71.9	71.6
+Ours	-	DeiT-S	$\mathcal{I}$	75.1(↑3.2)	75.3(↑3.7)
S2C [10]	CVPR24	WR38	$\mathcal{I}$	78.8	78.1
+Ours	-	WR38	$\mathcal{I}$	**80.1**(↑1.3)	**79.5**(↑1.4)

text prompts as additional supervision. Another interesting phenomenon is that adopting Transformer backbones (e.g., DeiT-S and ViT-B) always results in larger improvements than ResNet families (e.g., R50 and WR38), averagely enhancing 1.2% and 1.9% mIoU scores. This indicates that incorporating more powerful pre-trained backbones will lead to more accurate segmentation results. In particular, we also highlight the baselines that address noisy label learning in gray. Our method significantly improves these approaches, yielding average gains of 7.5% and 7.3% on val and test set, respectively. Fig. 9 shows qualitative comparison results with several state-of-the-art methods [10], [30], [33]. As can be seen, compared with the selected baselines, our approach yields more consistent segmentation predictions with precisely delineated object shapes and boundaries (e.g., "TV", "bottle", "person", "horse", "sofa", and "dog"), which are closer to the ground truths.

3) Segmentation Performance on MS COCO 2014: In this section, we carry out experiments on more challenging MS COCO 2014 dataset to further demonstrate the effectiveness of our method. The results are reported in Table III. Consistent with Table II, our method achieves remarkable improvement of mIoU scores (e.g., 4.0%, 4.8%, 5.0%, and 1.5% when

Fig. 9. Visual examples of semantic segmentation results on PASCAL VOC 2012 val set. From top to bottom are input image, corresponding ground truth, and segmentation results from CLIP-ES [30], S2C [10], POT [33], and our approach, respectively. (Best viewed in color)

TABLE III
SEGMENTATION RESULTS COMPARISON WITH THE STATE-OF-THE-ART APPROACHES IN TERMS OF MIOU (%) ON MS COCO 2014 VAL SET. "$\mathcal{I}$", "$\mathcal{S}$", AND "$\mathcal{L}$" STAND FOR SUPERVISION FROM IMAGE-LEVEL LABELS, SALIENCY MAPS, AND LANGUAGE PROMPTS. THE BASELINES THAT ADDRESS NOISY LABEL LEARNING ARE HIGHLIGHTED IN GRAY COLOR. THE BEST RESULTS ARE HIGHLIGHTED IN BOLD NUMBER.

Method	Year	Backbone	Sup.	Val
Single-stage WSSS:				
AFA [27]	CVPR22	MiT-B1	$\mathcal{I}$	38.9
TSCD [91]	AAAI23	MiT-B1	$\mathcal{I}$	40.1
ToCo [85]	CVPR23	ViT-B	$\mathcal{I}$	42.3
DuPL [18]	CVPR24	ViT-B	$\mathcal{I}$	44.6
WeCLIP [31]	CVPR24	ViT-B	$\mathcal{I}+\mathcal{L}$	47.1
CRME [13]	CVPR25	ViT-B	$\mathcal{I}$	47.2
ExCEL [12]	CVPR25	ViT-B	$\mathcal{I}+\mathcal{L}$	50.3
Multi-stage WSSS:				
CDA [52]	ICCV21	WR38	$\mathcal{I}$	33.2
CSENet [20]	ICCV21	WR38	$\mathcal{I}$	36.4
RCA [45]	CVPR22	VGG16	$\mathcal{I}+\mathcal{S}$	26.7
URN [39]	AAAI22	R101	$\mathcal{I}$	40.7
MCT [48]	CVPR22	DeiT-S	$\mathcal{I}$	42.0
BECO [38]	CVPR23	R50	$\mathcal{I}$	45.1
CPAL [44]	CVPR24	R101	$\mathcal{I}+\mathcal{L}$	46.8
PSDPM [92]	CVPR24	R101	$\mathcal{I}+\mathcal{L}$	47.2
MuP-VSS [43]	CVPR25	WR38	$\mathcal{I}$	46.6
POT [33]	CVPR25	R50	$\mathcal{I}+\mathcal{L}$	47.9
SIPE [32]	CVPR22	R101	$\mathcal{I}$	40.6
+Ours	-	R101	$\mathcal{I}$	44.6(↑4.0)
MCT [48]	CVPR22	DeiT-S	$\mathcal{I}$	42.0
+Ours	-	DeiT-S	$\mathcal{I}$	46.8(↑4.8)
CLIP-ES [30]	CVPR23	ViT-B	$\mathcal{I}+\mathcal{L}$	45.4
+Ours	-	ViT-B	$\mathcal{I}+\mathcal{L}$	50.4(↑5.0)
S2C [10]	CVPR24	WR38	$\mathcal{I}$	50.4
+Ours	-	WR38	$\mathcal{I}$	**51.9**(↑1.5)

combined with SIPE [32], MCT [48], CLIP-ES [30], and S2C [10], respectively. In particular, when SIPE [32] is integrated, we achieve comparable results with respect to [30], [38]. However, when the backbone is replaced with more powerful DeiT-S [48] and ViT-B [30], our approach obtains higher

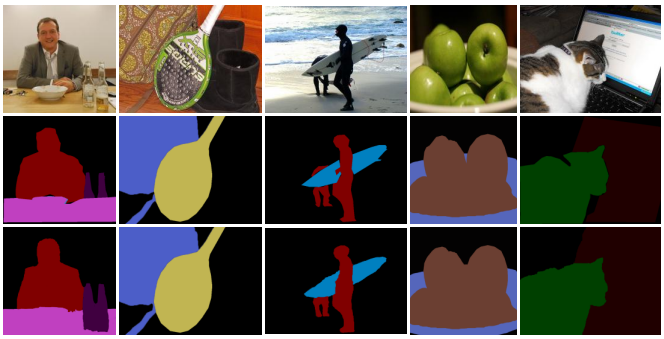


Fig. 10. Visual examples of semantic segmentation results on MS COCO 2014 val set. From top to bottom are original images, corresponding ground truth, and our segmentation outputs. (Best viewed in color)

incremental enhancement than combined with [32], yielding 0.8% and 1.0% mIoU improvement. It is also discovered that our method outperforms all baselines that address noisy label learning, which is once again consistent with Table II. More specifically, it improves mIoU scores by 11.2%, 7.3%, 6.8%, and 1.5% compared to URN [39], DuPL [18], BECO [38], and S2C [10], respectively. Fig. 10 also illustrates several visual examples of segmentation outputs on MS COCO 2014 dataset. It is evident that our approach effectively handles challenging cases involving object occlusion (e.g., "surfboard"), complex-shaped objects (e.g., "person"), and multiple object instances (e.g., "apple"). It is worth to note that in the first example, our method successfully segments the entire region of "bottle", even though the lower part of pixel-wise annotations are not correctly available in ground truth.

4) Segmentation Performance on Cityscapes: This section assesses the effectiveness of our method on small-scale dataset Cityscapes [51]. Following [93], the input images are randomly cropped into 512×1024 for fair comparison. Consistent with the training protocol used for the PASCAL VOC and

TABLE IV
SEGMENTATION RESULTS COMPARISON WITH THE STATE-OF-THE-ART APPROACHES IN TERMS OF MIOU (%) ON CITYSCAPES VAL AND TEST SETS. "$\mathcal{I}$" AND "$\mathcal{L}$" STAND FOR SUPERVISION FROM IMAGE-LEVEL LABELS AND LANGUAGE PROMPTS. THE BEST RESULTS ARE HIGHLIGHTED IN BOLD NUMBER.

Method	Year	Backbone	Sup.	Val	Test
CAM [14]	CVPR16	ViT-B	$\mathcal{I}$	33.0	33.2
CLIMS [29]	CVPR22	R50	$\mathcal{I}+\mathcal{L}$	18.1	18.0
CLIP-ES [30]	CVPR23	R101	$\mathcal{I}+\mathcal{L}$	35.4	35.0
CARB [93]	AAAI24	R50	$\mathcal{I}+\mathcal{L}$	52.1	51.8
+Ours	-	R50	$\mathcal{I}+\mathcal{L}$	**53.8**(↑1.7)	**53.4**(↑1.6)

TABLE V
ABLATION STUDIES FOR OBJECT BOUNDARY ACCURACY IN TERMS OF F-SCORE (%) ON PASCAL VOC 2012 AND MS COCO 2014 DATASETS. "BW" STANDS FOR BOUNDARY WITH (PX). "-/-" INDICATES THE RESULTS ON TWO DATASETS, RESPECTIVELY.

BW	1	2	3
PSDPM [92]	76.8/47.1	79.9/50.1	84.4/54.7
MuP-VSS [43]	76.2/46.4	79.2/49.4	83.0/53.6
POT [33]	78.3/47.7	81.5/50.9	85.2/55.1
S2C [10]	80.1/49.3	83.8/52.2	88.0/56.8
S2C+Ours	81.5/50.6	84.1/53.8	88.6/57.3

MS COCO, we first initialized the backbone network with pre-trained weights from [93], and then fine-tune it using our proposed method. The results are reported in Table IV. Consistent with the results reported in Table II and Table III, when combined with CARB [93], our approach still achieves remarkable improvements, yielding 53.8% and 53.4% mIoU scores in val and test set, respectively.

5) Performance for Object Boundary Accuracy: Although the segmentation results in Fig. 9 and Fig. 10 have shown that object boundaries and shapes are well delineated, this section still evaluates the object boundary accuracy in terms of F-score at different thresholds [94], [95]. Ideally, we hope our method works well in the strictest regime (e.g., smallest boundary width), where the estimated boundaries are expected to exactly match the ground truth. As a result, we first perform boundary detection on segmentation outputs and corresponding ground truth, then the experiments are conducted by reducing boundary width step-by-step. The results are reported in Table V. It is discovered that, compared with selected baselines [10], [33], [43], [92], our approach averagely improves 3.4 and 2.8 F-score, respectively. Particularly, our method achieves 3.7 and 3.0 F-score improvement in the strictest regime (width = 1px).

6) Comparison of Implementing Efficiency: This section compares the implementing efficiency between our method and several selected state-of-the-art baselines [10], [30], [32], [48]. The results are reported in Table VI. It is observed that when our system is integrated with the baseline backbones, the model size and GFLOPs are consistently reduced, while inference speed is uniformly improved. In particular, when combined with S2C [10], the model size and GFLOPs are approximately decreased by 81.3% and 94.3%, respectively, only achieving 123.2M model parameters with 156.8 GFLOPs. This advantage mainly stems from the exclusion of a frozen SAM [40] foundation model that dominates both model size and computational costs in [10]. Notably, the SAM [40] model

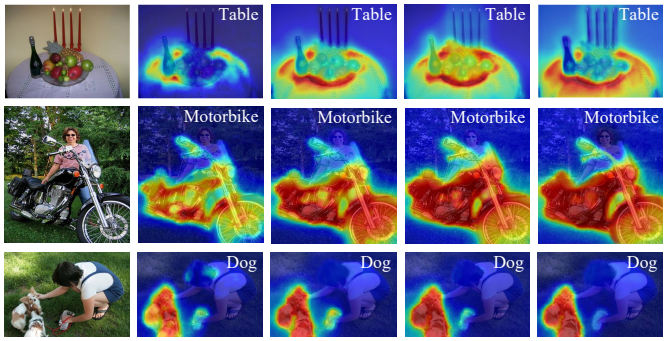


Fig. 11. Visual examples of the improvement in CAMs using different components on PASCAL VOC 2012 dataset. For clarity, the associated semantic labels are superimposed on the top-right of each results. From left to right are input images, CAMs produced from baselines, baselines+DFSM, baselines+DFSM+Erosion, and entire our method. (Best viewed in color)

TABLE VI
COMPARISON OF IMPLEMENTING EFFICIENCY IN TERMS OF MODEL SIZE, FLOPS, AND FPS.

Method	Params(M) ↓	FLOPs(G) ↓	FPS ↑
SIPE [32]	63.7	56.8	25.0
+Ours	57.1(↓10.4%)	46.2(↓18.7%)	38.2(↑52.8%)
MCT [48]	85.7	38.8	39.2
+Ours	54.6(↓36.3%)	32.3(↓16.8%)	42.2(↑7.7%)
CLIP-ES [30]	135.6	102.4	16.4
+Ours	91.4(↓32.6%)	84.6(↓17.4%)	21.3(↑29.9%)
S2C [10]	657.1	2733.6	12.1
+Ours	123.2(↓81.3%)	156.8(↓94.3%)	14.9(↑23.1%)

TABLE VII
ABLATION STUDIES FOR THE CONTRIBUTION OF EACH COMPONENT IN TERMS OF CAM MIOU (%), MODEL SIZE, AND GFLOPS ON PASCAL VOC 2012 DATASET.

Baseline	DFSM		Erosion	FSCM	Params(M)	FLOPs(G)	mIoU
	$\tilde{\boldsymbol{D}}$	$\boldsymbol{A}_e$					
✓					44.54	39.78	58.6
✓	✓				44.54	39.78	59.3
✓		✓			57.12	44.08	59.8
✓	✓	✓			57.12	44.08	60.3
✓	✓	✓	✓		57.12	44.08	60.7
✓	✓	✓	✓	✓	57.12	46.22	62.1

is no longer required during the inference phase, resulting in a slight improvement in running speed (14.9FPS vs 12.1FPS for the baseline) of our approach.

D. Ablation Studies

To better understand the underlying behavior of our method, this section reports the results of a series of ablation studies.

1) Ablation Studies of Different Components: This section evaluates the contribution of each component to the improvement of CAMs. We start from establishing a baseline based on [32], then DFSM, Erosion, and FSCM are sequentially introduced. To further assess the individual contributions of feature similarity maps $\boldsymbol{A}_e$ and normalized location maps $\tilde{\boldsymbol{D}}$ within DFSM, we exclude one component while remaining the rest. Note when only DFSM is introduced, we produce pseudo-label masks $\boldsymbol{E}_s$ and $\boldsymbol{E}_e$ from $\boldsymbol{M}_s$ and $\boldsymbol{M}_e$ according to

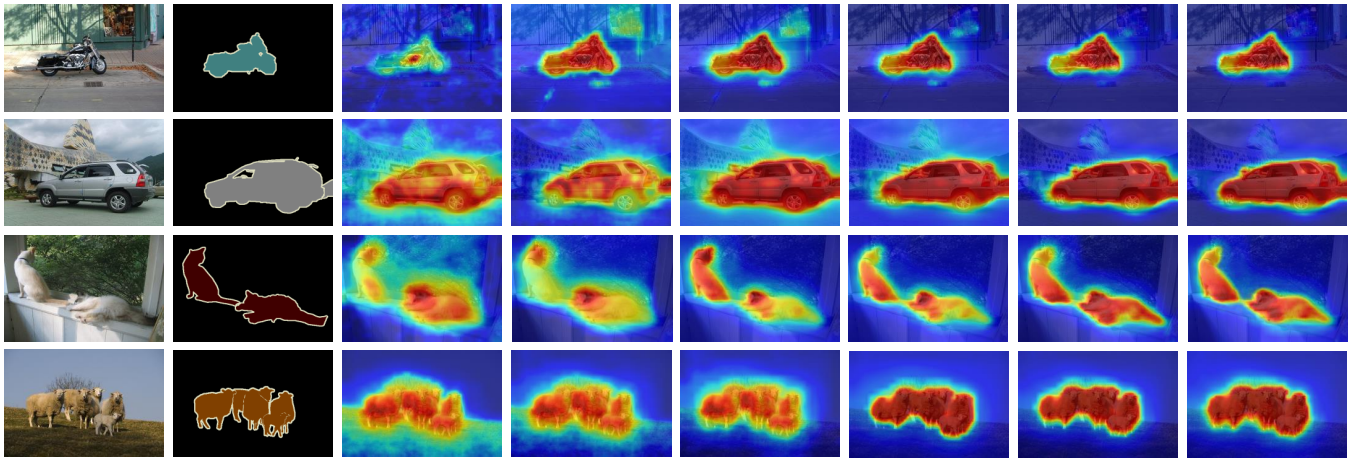


Fig. 12. Visual examples of the improvement in CAMs at different iterations throughout the training phase on the PASCAL VOC dataset. From left to right are original images, ground truth, and produced CAMs in different iterations, ranged from 500 to 3000 with updated step 500. (Best viewed in color)

TABLE VIII
ABLATION STUDIES FOR DIFFERENT CONTRASTIVE LEARNING STRATEGIES IN TERMS OF CAM MIOU (%) ON PASCAL VOC 2012 AND MS COCO 2014 DATASET.

Method / Dataset	Pixel-to-prototype [41]	Pixel-to-segment [45]	Ours
PASCAL VOC 2012 [15]	59.7	60.8	62.1
MS COCO 2014 [50]	35.5	36.4	38.1

TABLE IX
ABLATION STUDIES FOR ITERATIVE TRAINING IN TERMS OF CAM MIOU (%) ON PASCAL VOC 2012 AND MS COCO 2014 DATASET.

Epochs / Dataset	2	4	6	8	10
PASCAL VOC 2012 [15]	43.9	54.7	58.4	61.3	62.1
MS COCO 2014 [50]	27.5	32.8	35.7	37.1	38.1

III-A2, respectively. Except the baseline that directly creates CAMs using classification head, other ablation studies utilize developed contrastive loss to produce CAMs. The results are reported in Table VII. In general, performance improves progressively with the sequential introduction of each component. When only the baseline is employed, we achieve the poorest performance of 58.6 % mIoU score. The introduction of DFSM results in a significant improvement of 1.7% mIoU. More specifically, $\boldsymbol{A}_e$ and $\tilde{\boldsymbol{D}}$ contribute an increase of 1.2% and 0.7% mIoU, respectively, highlighting the essential role of feature affinity in calibrating noisy CAMs. The simple erosion operation leads to a slight improvement of 0.4% in performance. Finally, by introducing FSCM, our contrastive learning strategy enhances the performance to 62.1% mIoU. Table VII also reports the computational complexity for each individual component in terms of model size and GFLOPs. The entire method only requires 57.12M model size and 46.22GFLOPs. Note only the linear transformation in computing $\boldsymbol{A}_e$ requires parameters, while other components are parameter-free. Among all components, the backbone network (baseline) dominates the model size and computational costs, demonstrating that our method is parameter-efficient and easy to implement. In Fig. 11, we also display some visual results. As more components are progressively introduced, the accuracy of the CAMs continues to increase, consistenting with the results reported in Table VII.

2) Ablation Studies for Developed Contrastive Learning: As multiple positive training pairs are considered in Eq. (17), it is essential to compare our approach with existing pixel-to-prototype [41] and pixel-to-segment [45] contrastive methods, both of which rely solely on a single positive training pair in their loss functions. Concretely, we vary only the contrastive loss while keeping the rest of our system fixed. In the first setting, we directly substitute Eq. (17) with the traditional loss function of [41]. In the second setting, following [45], an average pooling is applied to the masked feature set $\boldsymbol{F}$ to produce segment-based prototypes, which are then used in subsequent pixel-to-segment contrast. The results are presented in Table VIII. It is clear that our contrastive learning strategy outperforms conventional approaches, achieving average improvements of 1.9% and 2.2% mIoU on two datasets.

3) Ablation Studies for Iterative Training: To present how our method refines incomplete activations while calibrating incorrect ones, this section evaluates the improvement of CAMs along with the increase of training epochs. Table IX reports the quantitative results on PASCAL VOC and MS COCO datasets. It is observed that the mIoU scores gradually improve throughout the training phase. Fig. 12 provides visual examples at different training iterations on PASCAL VOC dataset. The qualitative results demonstrate that regardless of whether the input images contain single (first two examples) or multiple object instances (last two examples), our method effectively eliminates noisy activations in the background while highlighting the entire object regions step-by-step.

E. Analysis of Parameter Settings

1) Effect of memory bank size: The size of memory bank determines how many prototypes are enough to fight against noisy labels, significantly influencing the trade-off between the representation capability, occupied storage space, and computational efficiency in our contrastive learning paradigm. We

TABLE X
PARAMETER ANALYSIS FOR MEMORY BANK SIZE N IN TERMS OF CAM MIOU (%), MEMORY, AND GFLOPS ON PASCAL VOC 2012 SET.

N	100	300	500	700	900
Memory (M)	4.1	12.3	20.5	28.7	36.9
FLOPs (G)	8.59	25.64	42.91	64.36	81.38
mIoU (%)	61.2	61.8	62.1	62.2	62.3

TABLE XI
PARAMETER ANALYSIS FOR CONFIDENCE THRESHOLD η AND LOSS WEIGHT λ IN TERMS OF CAM MIOU (%) ON PASCAL VOC 2012 SET.

λ \ η	0.5	0.6	0.7	0.8	0.9
0.05	56.1	59.8	60.9	61.7	61.4
0.10	54.8	60.4	61.3	62.1	61.8
0.15	52.9	59.9	61.4	61.9	61.6
0.20	51.5	59.6	61.1	61.5	61.3

TABLE XII
PARAMETER ANALYSIS FOR MOMENTUM WEIGHT γ IN TERMS OF CAM MIOU (%) ON PASCAL VOC 2012 SET.

γ	0.1	0.2	0.3	0.4	0.5	0.6	0.7	0.8	0.9	1
mIoU	60.1	60.4	60.2	60.5	60.9	61.1	61.4	61.8	62.1	58.7

thus evaluate the performance variance along with the changes of prototype number N, ranged from 100 to 900 with updated step 200. The results are reported in Table X. It reveals that smaller N leads to insufficient ability to represent the diversity of each category. In contrast, a larger N offers only slight improvements yet requires significantly huge computational budgets. The best trade-off is achieved when $N = 500$, thus chosen as default setting in our memory bank $\boldsymbol{B}$.

2) Effect of Confidence Threshold and Loss Weight: Among all hyper-parameters, confidence threshold η and loss weight λ are essential to the final performance. This section evaluates the effect by jointly tuning two parameters together, where η varies in the range of [0.5, 0.9] in step 0.1, while λ changes in the range of [0.05, 0.20] in step 0.05. The results are reported in Table XI. When η is too small, the performance drops drastically, as the pseudo-label masks become unreliable. It is also observed that the mIoU scores peak at $\eta = 0.8$ and $\lambda = 0.1$, which are opt to default settings in our method.

3) Effect of Momentum Weight: Momentum weight γ affects the update of prototypes that controls the representation capability of memory bank, thus playing a significant role in our contrastive learning. As a result, we conduct experiments by changing γ and report the results in Table XII. Note $\gamma = 1$ denotes there is no evolution of prototypes, and the memory bank is only determined by initialization, thus leading to the drastic drop of mIoU scores. In addition, we also observe that the performance slightly fluctuate along with the change of γ, and the highest mIoU score is obtained when $\gamma = 0.9$.

V. CONCLUSION REMARKS AND FUTURE WORK

In this paper, we formulate WSSS as a noisy label correction problem, where incomplete and incorrect activations are considered as pixel-level noisy labels. By applying expansion and shrinking operations, robust and reliable pseudo-labels are produced for constructing positive and negative training pairs. Under the supervision of these training pairs, we propose a developed contrastive learning strategy that utilizes multiple positive pairs to learn more robust feature representations, thereby facilitating the calibration of noisy labels in CAMs step-by-step. Extensive experiments demonstrate the effectiveness of our method on PASCAL VOC 2012, MS COCO 2014, and Cityscapes datasets. One limitation of our method is the inability to effectively handle substantial appearance variations and extremely clutter backgrounds, as the failure cases shown in Fig. 7. In the future, we aim to leverage foundation models (e.g., CLIP [56] and SAM [40]) to deal with this challenge. In addition, we would like to expand our work to a broader spectrum of weakly-supervised scenarios, such as video-based segmentation [5] and crowd density estimation [96].

REFERENCES

[1] E. Shelhamer, J. Long, and T. Darrell, "Fully convolutional networks for semantic segmentation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 39, no. 4, pp. 640–651, 2017.

[2] Y. Zhang, M.-H. Guo, M. Wang *et al.*, "Exploring regional clues in clip for zero-shot semantic segmentation," in *Proc. CVPR*, 2024, pp. 3270–3280.

[3] H. Chen, J. Wang, H. C. Chen *et al.*, "Seminar learning for click-level weakly supervised semantic segmentation," in *Proc. ICCV*, 2021, pp. 6920–6929.

[4] A. Bearman, O. Russakovsky, V. Ferrari *et al.*, "What's the point: semantic segmentation with point supervision," in *Proc. ECCV*, 2016, pp. 549–565.

[5] P. Huang, J. Han, N. Liu *et al.*, "Scribble-supervised video object segmentation," *IEEE/CAA J. Autom. Sinica*, vol. 9, no. 2, pp. 339–8096, 2022.

[6] Z. Pan, H. Sun, P. Jiang *et al.*, "Cc4s: encouraging certainty and consistency in scribble-supervised semantic segmentation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 46, no. 12, pp. 8918–8935, 2024.

[7] J. Dai, K. He, and J. Sun, "Boxsup: exploiting bounding boxes to supervise convolutional networks for semantic segmentation," in *Proc. ICCV*, 2015, pp. 1635–1643.

[8] Y. Oh, B. Kim, and B. Ham, "Background-aware pooling and noise-aware loss for weakly-supervised semantic segmentation," in *Proc. CVPR*, 2021, pp. 6913–6922.

[9] Z. Yang, K. Fu, M. Duan *et al.*, "Separate and conquer: decoupling co-occurrence via decomposition and representation for weakly supervised semantic segmentation," in *Proc. CVPR*, 2024, pp. 3606–3615.

[10] H. Kweon and K.-J. Yoon, "From sam to cams: exploring segment anything model for weakly supervised semantic segmentation," in *Proc. CVPR*, 2024, pp. 19 499–19 509.

[11] T. Chen, X. Jiang, G. Pei *et al.*, "Knowledge transfer with simulated inter-image erasing for weakly supervised semantic segmentation," in *Proc. ECCV*, 2024, pp. 441–458.

[12] Z. Yang, Y. Meng, K. Fu *et al.*, "Exploring clip's dense knowledge for weakly supervised semantic segmentation," in *Proc. CVPR*, 2025, pp. 20 223–20 232.

[13] X. Xu, P. Zhang, W. Huang *et al.*, "Weakly supervised semantic segmentation via progressive confidence region expansion," in *Proc. CVPR*, 2025, pp. 9829–9838.

[14] B. Zhou, A. Khosla, A. Lapedriza *et al.*, "Learning deep features for discriminative localization," in *Proc. CVPR*, 2016, pp. 2921–2929.

[15] M. Everingham, S. A. Eslami, L. Van Gool *et al.*, "The pascal visual object classes challenge: a retrospective," *Int. J. Comput. Vis.*, vol. 111, pp. 98–136, 2015.

[16] K. Cheng, J. Tang, H. Gu *et al.*, "Cross-block sparse class token contrast for weakly supervised semantic segmentation," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 34, no. 12, pp. 13 004–13 015, 2024.

[17] J. Wang, T. Dai, X. Zhao *et al.*, "Class activation map calibration for weakly supervised semantic segmentation," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 34, no. 11, pp. 11 668–11 681, 2024.

[18] Y. Wu, X. Ye, K. Yang *et al.*, "Dupl: dual student with trustworthy progressive learning for robust weakly supervised semantic segmentation," in *Proc. CVPR*, 2024, pp. 3534–3543.

[19] Z. Chen and Q. Sun, "Weakly-supervised semantic segmentation with image-level labels: from traditional models to foundation models," *ACM Computing Surveys*, vol. 57, pp. 1–29, 2023.

[20] H. Kweon, S.-H. Yoon, H. Kim *et al.*, "Unlocking the potential of ordinary classifier: class-specific adversarial erasing framework for weakly supervised semantic segmentation," in *Proc. ICCV*, 2021, pp. 6994–7003.

[21] K. Sun, H. Shi, Z. Zhang *et al.*, "Ecs-net: improving weakly supervised semantic segmentation by using connections between class activation maps," in *Proc. ICCV*, 2021, pp. 7283–7292.

[22] Y. Wei, J. Feng, X. Liang *et al.*, "Object region mining with adversarial erasing: a simple classification to semantic segmentation approach," in *Proc. CVPR*, 2017, pp. 1568–1576.

[23] J. Li, Z. Jie, X. Wang *et al.*, "Expansion and shrinkage of localization for weakly-supervised semantic segmentation," in *Proc. NeurIPS*, 2022, pp. 16 037–16 051.

[24] W. Wang, G. Sun, and L. Van Gool, "Looking beyond single images for weakly supervised semantic segmentation learning," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 46, no. 3, pp. 1635–1649, 2024.

[25] C. Wang, D. Zhang, L. Zhang *et al.*, "Coupling global context and local contents for weakly-supervised semantic segmentation," *IEEE Trans. Neural Networks Learn. Syst.*, vol. 35, no. 10, pp. 13 483–13 495, 2024.

[26] L. Xu, M. Bennamoun, F. Boussaid *et al.*, "Auxiliary tasks enhanced dual-affinity learning for weakly supervised semantic segmentation," *IEEE Trans. Neural Networks Learn. Syst.*, vol. 36, no. 3, pp. 5082–5096, 2025.

[27] L. Ru, Y. Zhan, B. Yu *et al.*, "Learning affinity from attention: end-to-end weakly-supervised semantic segmentation with transformers," in *Proc. CVPR*, 2022, pp. 16 846–16 855.

[28] B. Zhang, J. Xiao, J. Jiao *et al.*, "Affinity attention graph neural network for weakly supervised semantic segmentation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 44, no. 11, pp. 8082–8096, 2022.

[29] J. Xie, X. Hou, K. Ye *et al.*, "Clims: cross language image matching for weakly supervised semantic segmentation," in *Proc. CVPR*, 2022, pp. 4483–4492.

[30] Y. Lin, M. Chen, W. Wang *et al.*, "Clip is also an efficient segmenter: a text-driven approach for weakly supervised semantic segmentation," in *Proc. CVPR*, 2023, pp. 15 305–15 314.

[31] B. Zhang, S. Yu, Y. Wei *et al.*, "Frozen clip: a strong backbone for weakly supervised semantic segmentation," in *Proc. CVPR*, 2024, pp. 3796–3806.

[32] Q. Chen, L. Yang, J.-H. Lai *et al.*, "Self-supervised image-specific prototype exploration for weakly supervised semantic segmentation," in *Proc. CVPR*, 2022, pp. 4288–4298.

[33] J. Wang, T. Dai, B. Zhang *et al.*, "Pot: prototypical optimal transport for weakly supervised semantic segmentation," in *Proc. CVPR*, 2025, pp. 15 055–15 064.

[34] S. Liu, N.-W. Jonathan, R. Narges *et al.*, "Early-learning regularization prevents memorization of noisy labels," in *Proc. NeurIPS*, 2020, pp. 20 331–20 342.

[35] Z. Yu, Q. Xu, Y. Jiang *et al.*, "Enhancing sample utilization in noise-robust deep metric learning with subgroup-based positive-pair selection," *IEEE Trans. Image Process.*, vol. 33, pp. 6083–6097, 2024.

[36] C. Liang, Z. Yang, L. Zhu *et al.*, "Co-learning meets stitch-up for noisy multi-label visual recognition," *IEEE Trans. Image Process.*, vol. 32, pp. 2508–2519, 2023.

[37] S. Liu, K. Liu, W. Zhu *et al.*, "Adaptive early-learning correction for segmentation from noisy annotations," in *Proc. CVPR*, 2022, pp. 2606–2616.

[38] S. Rong, B. Tu, Z. Wang *et al.*, "Boundary-enhanced co-training for weakly supervised semantic segmentation," in *Proc. CVPR*, 2023, pp. 19 574–19 584.

[39] Y. Li, Y. Duan, Z. Kuang *et al.*, "Uncertainty estimation via response scaling for pseudo-mask noise mitigation in weakly-supervised semantic segmentation," in *Proc. AAAI*, 2022, pp. 1447–1455.

[40] A. Kirillov, E. Mintun, N. Ravi *et al.*, "Segment anything," in *Proc. ICCV*, 2023, pp. 4015–4026.

[41] Y. Du, Z. Fu, Q. Liu *et al.*, "Weakly supervised semantic segmentation by pixel-to-prototype contrast," in *Proc. CVPR*, 2022, pp. 4320–4329.

[42] T.-W. Ke, J.-J. Hwang, and S. X. Yu, "Universal weakly supervised segmentation by pixel-to-segment contrastive learning," in *Proc. ICLR*, 2021, pp. 1–12.

[43] S. Duan, X. Yang, and N. Wnag, "Multi-label prototype visual spatial search for weakly supervised semantic segmentation," in *Proc. CVPR*, 2025, pp. 30 241–30 250.

[44] F. Tang, Z. Xu, Z. Qu *et al.*, "Hunting attributes: context prototype-aware learning for weakly supervised semantic segmentation," in *Proc. CVPR*, 2024, pp. 3324–3334.

[45] T. Zhou, M. Zhang, F. Zhao *et al.*, "Regional semantic contrast and aggregation for weakly supervised semantic segmentation," in *Proc. CVPR*, 2022, pp. 4299–4309.

[46] J. Xie, J. Xiang, J. Chen *et al.*, "C2am: contrastive learning of class-agnostic activation map for weakly supervised object localization and semantic segmentation," in *Proc. CVPR*, 2022, pp. 989–998.

[47] W. Zhai, P. Wu, K. Zhu *et al.*, "Background activation suppression for weakly supervised object localization and semantic segmentation," *Int. J. Comput. Vis.*, vol. 132, no. 3, pp. 750–775, 2024.

[48] L. Xu, W. Ouyang, M. Bennamoun *et al.*, "Multi-class token transformer for weakly supervised semantic segmentation," in *Proc. CVPR*, 2022, pp. 4310–4319.

[49] R. Yi, Y. Huang, Q. Guan *et al.*, "Learning from pixel-level label noise: a new perspective for semi-supervised semantic segmentation," *IEEE Trans. Image Process.*, vol. 31, pp. 623–635, 2022.

[50] T.-Y. Lin, M. Maire, S. Belongie *et al.*, "Microsoft coco: common objects in context," in *Proc. ECCV*, 2014, pp. 740–755.

[51] M. Cordts, M. Omran, S. Ramos *et al.*, "The cityscapes dataset for semantic urban scene understanding," in *Proc. CVPR*, 2016, pp. 3213–3223.

[52] Y. Su, R. Sun, G. Lin *et al.*, "Context decoupling augmentation for weakly supervised semantic segmentation," in *Proc. ICCV*, 2021, pp. 7004–7014.

[53] Y. Du, Z. Fu, and Q. Liu, "Pixel-level domain adaptation: a new perspective for enhancing weakly supervised semantic segmentation," *IEEE Trans. Image Process.*, vol. 33, pp. 4654–4669, 2024.

[54] G. Sun, W. Wang, J. Dai *et al.*, "Mining cross-image semantics for weakly supervised semantic segmentation," in *Proc. ECCV*, 2020, pp. 347–365.

[55] J. Fan and Z. Zhang, "Memory-based cross-image contexts for weakly supervised semantic segmentation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 45, no. 5, pp. 6006–6020, 2023.

[56] A. Radford, J. W. Kim, C. Hallacy *et al.*, "Learning transferable visual models from natural language supervision," in *Proc. ICML*, 2021, pp. 8748–8763.

[57] B. Han, J. Yao, G. Niu *et al.*, "Masking: a new perspective of noisy supervision," in *Proc. NeurIPS*, 2018, p. 5841–5851.

[58] X. Chen and A. Gupta, "Webly supervised learning of convolutional networks," in *Proc. ICCV*, 2015, p. 1431–1439.

[59] T. Xiao, T. Xia, Y. Yang *et al.*, "Learning from massive noisy labeled data for image classification," in *Proc. CVPR*, 2015, pp. 2691–2699.

[60] D. Cheng, T. Liu, Y. Ning *et al.*, "Instance-dependent label-noise learning with manifold-regularized transition matrix estimation," in *Proc. CVPR*, 2023, pp. 16 609–16 618.

[61] H. Cheng, Z. Zhu, X. Li *et al.*, "Learning with instance-dependent label noise: A sample sieve approach," in *Proc. ICLR*, 2021.

[62] S. Kim, D. Lee, S. Kang *et al.*, "Learning discriminative dynamics with label corruption for noisy label detection," in *Proc. CVPR*, 2024, pp. 22 477–22 487.

[63] X. Xia, T. Liu, B. Han *et al.*, "Robust early-learning: hindering the memorization of noisy labels," in *Proc. ICLR*, 2020.

[64] Y. Bai, E. Yang, B. Han *et al.*, "Understanding and improving early stopping for learning with noisy labels," in *Proc. NeurIPS*, 2021, pp. 24 392–24 403.

[65] H. Zhang, M. Cisse, Y. N. Dauphin *et al.*, "Mixup: beyond empirical risk minimization," in *Proc. ICLR*, 2018, pp. 1–13.

[66] S. Yuan, X. Li, Y. Miao *et al.*, "Combating noisy labels by alleviating the memorization of dnns to noisy labels," *IEEE Trans. Multimedia*, 2024.

[67] X. Ma, H. Huang, Y. Wang *et al.*, "Normalized loss functions for deep learning with noisy labels," in *Proc. ICML*, 2020, pp. 6543–6553.

[68] Y. Wang, X. Ma, Z. Chen *et al.*, "Symmetric cross entropy for robust learning with noisy labels," in *Proc. ICCV*, 2019, pp. 322–330.

[69] Z. Zhang and M. Sabuncu, "Generalized cross entropy loss for training deep neural networks with noisy labels," in *Proc. NeurIPS*, 2018, pp. 8792–8802.

[70] N.-r. Kim, J.-S. Lee, and J.-H. Lee, "Learning with structural labels for learning with noisy labels," in *Proc. CVPR*, 2024, pp. 27 600–27 610.

[71] Y. Li, H. Han, S. Shan *et al.*, "Disc: learning from noisy labels via dynamic instance-specific selection and correction," in *Proc. CVPR*, 2023, pp. 24 070–24 079.

[72] X. Xia, B. Han, Y. Zhan *et al.*, "Combating noisy labels with sample selection by mining high-discrepancy examples," in *Proc. ICCV*, 2023, pp. 1833–1843.

[73] T. Liu and D. Tao, "Classification with noisy labels by importance reweighting," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 38, no. 3, pp. 447–461, 2015.
[74] F. Ma, Y. Wu, X. Yu *et al.*, "Learning with noisy labels via self-reweighting from class centroids," *IEEE Trans. Neural Networks Learn. Syst.*, vol. 33, no. 11, pp. 447–461, 2022.
[75] L. Yi, S. Liu, Q. She *et al.*, "On learning contrastive representations for learning with noisy labels," in *Proc. CVPR*, 2022, pp. 16 682–16 691.
[76] C.-Y. Chuang, R. D. Hjelm, X. Wang *et al.*, "Robust contrastive learning against noisy views," in *Proc. CVPR*, 2022, pp. 16 670–16 681.
[77] Z. Huang, J. Zhang, and H. Shan, "Twin contrastive learning with noisy labels," in *Proc. CVPR*, 2023, pp. 11 661–11 670.
[78] S. Li, X. Xia, S. Ge *et al.*, "Selective-supervised contrastive learning with noisy labels," in *Proc. CVPR*, 2022, pp. 316–325.
[79] Y. Wang, W. Liu, X. Ma *et al.*, "Iterative learning with open-set noisy labels," in *Proc. CVPR*, 2018, pp. 8688–8696.
[80] L. Jiang, Z. Zhou, T. Leung *et al.*, "Mentornet: learning data-driven curriculum for very deep neural networks on corrupted labels," in *Proc. ICML*, 2018, pp. 2304–2313.
[81] B. Han, Q. Yao, X. Yu *et al.*, "Co-teaching: robust training of deep neural networks with extremely noisy labels," in *Proc. NeurIPS*, 2018, pp. 8536–8546.
[82] X. Yu, B. Han, J. Yao *et al.*, "How does disagreement help generalization against label corruption?" in *Proc. ICML*, 2019, pp. 7164–7173.
[83] G. Wang, X. Liu, C. Li *et al.*, "A noise-robust framework for automatic segmentation of covid-19 pneumonia lesions from ct images," *IEEE Trans. Med. Imaging*, vol. 39, no. 8, pp. 2653–2663, 2020.
[84] Y. Shu, X. Wu, and W. Li, "Lvc-net: medical image segmentation with noisy label based on local visual cues," in *Proc. MICCAI*, 2019, pp. 558–566.
[85] L. Ru, H. Zheng, Y. Zhan *et al.*, "Token contrast for weakly-supervised semantic segmentation," in *Proc. CVPR*, 2023, pp. 3093–3102.
[86] Z. Peng, G. Wang, L. Xie *et al.*, "Usage: a unified seed area generation paradigm for weakly supervised semantic segmentation," in *Proc. ICCV*, 2023, pp. 624–634.
[87] K. Tapas, M. M. Daid, S. N. Nathan *et al.*, "An efficient k-means clustering algorithm: analysis and implementation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 24, no. 7, pp. 881–892, 2002.
[88] B. Hariharan, P. Arbeláez, L. Bourdev *et al.*, "Semantic contours from inverse detectors," in *Proc. ICCV*, 2011, pp. 991–998.
[89] J. Deng, W. Dong, R. Socher *et al.*, "Imagenet: a large-scale hierarchical image databasei," in *Proc. CVPR*, 2009, pp. 248–255.
[90] L.-C. Chen, G. Papandreou, I. Kokkinos *et al.*, "Deeplab: semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected crfs," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 40, no. 4, pp. 834–848, 2017.
[91] R. Xu, C. Wang, J. Sun *et al.*, "Self correspondence distillation for end-to-end weakly-supervised semantic segmentation," in *Proc. AAAI*, vol. 37, no. 3, 2023, pp. 3045–3053.
[92] X. Zhao, Z. Yang, T. Dai *et al.*, "Psdpm: prototype-based secondary discriminative pixels mining for weakly supervised semantic segmentation," in *Proc. CVPR*, 2024, pp. 3437–3446.
[93] K. Dongseob, L. Seungho, C. Junsuk *et al.*, "Weakly supervised semantic segmentation for driving scenes," in *Proc. AAAI*, 2024, pp. 2741–2749.
[94] T. Towaki, A. David, J. Varun *et al.*, "Gated-scnn: gated shape cnns for semantic segmentation," in *Proc. ICCV*, 2019, pp. 5229–5238.
[95] Y. Yuhui, X. Jingyi, C. Xilin *et al.*, "Segfix: model-agnostic boundary refinement for segmentation," in *Proc. ECCV*, 2020, pp. 489–506.
[96] Y. Li, R. Jia, Y. Hu *et al.*, "A weakly-supervised crowd density estimation method based on two-stage linear feature calibration," *IEEE/CAA J. Autom. Sinica*, vol. 11, no. 4, pp. 965–981, 2024.

Xiangwei Kang received the B.S. degree in communication engineering from Hangzhou Dianzi University, Hangzhou, P. R. China, in 2022. He is currently pursuing the M.S. degree in signal and information processing with Nanjing University of Posts and Telecommunications, Nanjing, P. R. China. His research interests include weakly supervised semantic segmentation and image understanding.

Afang Yang received the B.S. degree in communication engineering from Nanjing University of Posts and Telecommunications, Nanjing, P. R. China, in 2025. She is currently pursuing the M.S. degree in information and communication engineering with Nanjing University of Posts and Telecommunications, Nanjing, P. R. China. Her research interests include deep learning and computer vision.

Quan Zhou (Senior Member, IEEE) received the B.S. degree in electronics and information engineering from China University of Geosciences, Wuhan, P. R. China, in 2002, and the M.S. and Ph.D. degrees in electronics and information engineering from Huazhong University of Science and Technology, Wuhan, P. R. China, in 2006 and 2013, respectively. He was a Visiting Scholar with Temple University, Philadelphia, PA, USA, from 2019 to 2020. He is currently a Full Professor with Nanjing University of Posts and Telecommunications, Nanjing, P. R. China. He has authored or coauthored more than 100 related academic articles, including IEEE TRANSACTIONS ON IMAGE PROCESSING, IEEE TRANSACTIONS ON INTELLIGENT TRANSPORTATION SYSTEMS, IEEE TRANSACTIONS ON MULTIMEDIA, IEEE TRANSACTIONS ON MEDICAL IMAGING, IEEE TRANSACTIONS ON NEURAL NETWORKS AND LEARNING SYSTEMS, and *Pattern Recognition*. His research interests include deep learning, pattern recognition, and computer vision.

Dr. Zhou served as the Area Chairs for the IEEE ICME2019, ICME2026 and PRCV2022-26 and the Leading Guest Editor for IEEE TRANSACTIONS ON MULTIMEDIA, IEEE TRANSACTIONS ON FUZZY SYSTEMS, *Computers and Electrical Engineering,* and *Multimedia Tools and Applications.* Now he is Editorial Board Member of journals, such as *Pattern Recognition* and *Journal of The Franklin Institute*. He received the 2024 Most Influence Paper Award of IEEE ICIP and the Best student Paper Award of 2017 IEEE/SPIE ISAIR.

Xun Sun received the Ph.D. degree in electronic science and technology from Shanghai Jiao Tong University in 2019. From 2019 to 2020, he was a R&D Engineer in Intel semiconductor (Dalian) Co., Ltd. Since 2021, he has been an academic leader in Institute of Guizhou Aerospace Measuring and Testing Technology, which belongs to China Aerospace Science and Industry Corporation. He is the author of 9 articles, and 5 inventions. His research interests include advanced pressure sensor, image classification, data acquisition, model development and new testing technologies based on AI technology.

Jie Chen received the Ph.D. degree from the Huazhong University of Science and Technology, Wuhan, P. R. China, in 2011. He worked as a Post-Doctoral Researcher with Prince Edward Island University, Charlottetown, PE, Canada, from 2011 to 2015. He is currently a Senior Staff Engineer with the ICT BG, Huawei Technologies Company Ltd., Shenzhen, P. R. China. His research focuses on artificial intelligence, machine learning and computing infrastructure, including large language model, AI for science, and intelligence computing.

Huimin Lu (Senior Member, IEEE) received the Ph.D. degree in electrical engineering from the Kyushu Institute of Technology, Kitakyushu, Japan, in 2014. From 2013 to 2016, he was a JSPS Research Fellow with the Kyushu Institute of Technology. From 2016 to 2024, he was an Associate Professor with the Kyushu Institute of Technology and an Excellent Young Researcher of Ministry of Education, Culture, Sports, Science and Technology. He is currently a Professor with Southeast University, Nanjing, P. R. China. His research interests include artificial intelligence, computer vision, and robotics. He served as the Editor-in-Chief of *Computers & Electrical Engineering*. Now he is the Editor-in-Chief of *Cognitive Robotics Journal*, and the Editor of *Wireless Networks, Applied Soft Computing*, IEEE TRANSACTIONS ON ARTIFICIAL INTELLIGENCE, *IEICE Transactions on Information and Systems*, and *Pattern Recognition*. He is the Guest Editor for many journals, such as IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS FOR VIDEO TECHNOLOGY, IEEE TRANSACTIONS ON NETWORK SCIENCE AND ENGINEERING, *ACM Transactions on Internet Technology*, IEEE/CAA JOURNAL OF AUTOMATICA SINICA, IEEE INTERNET OF THINGS JOURNAL, and IEEE TRANSACTIONS ON INTELLIGENT TRANSPORTATION SYSTEMS. He is the Chair of IEEE Computer Society Technical Committee on Big Data.