Computer Vision and Image Understanding 239 (2024) 103886

Contents lists available at ScienceDirect

Computer Vision and Image Understanding

journal homepage: www.elsevier.com/locate/cviu

Multi-layered self-attention mechanism for weakly supervised semantic segmentation

Avinash Yaganapu, Mingon Kang *

Department of Computer Science, University of Nevada Las Vegas, 4505 S. Maryland Pkwy. Las Vegas, NV 89154-4022, United States of America

ARTICLE INFO

Communicated by Ioannis Stamos

MSC:
68U10
68T45

Keywords:
Weakly supervised semantic segmentation
Segmentation
Self-attention
Image-level labels

ABSTRACT

Weakly Supervised Semantic Segmentation (WSSS) provides efficient solutions for semantic image segmentation using image-level annotations. WSSS requires no pixel-level labeling that Fully Supervised Semantic Segmentation (FSSS) does, which is time-consuming and label-intensive. Most WSSS approaches have leveraged Class Activation Maps (CAM) or Self-Attention (SA) to generate pseudo pixel-level annotations to perform semantic segmentation tasks coupled with fully supervised approaches (e.g., Fully Convolutional Network). However, those approaches often provides incomplete supervision that mainly includes discriminative regions from the last convolutional layer. They may fail to capture regions of low- or intermediate-level features that may not be present in the last convolutional layer. To address the issue, we proposed a novel Multi-layered Self-Attention (Multi-SA) method that applies a self-attention module to multiple convolutional layers, and then stack feature maps from the self-attention layers to generate pseudo pixel-level annotations. We demonstrated that integrated feature maps from multiple self-attention layers produce higher coverage in semantic segmentation than using only the last convolutional layer through intensive experiments using standard benchmark datasets.

1. Introduction

Semantic segmentation is one of the fundamental computer vision tasks to predict pixel-level classification labels on images. Semantic segmentation partitions an image into multiple image segments and assigns corresponding class labels to pixels. Semantic segmentation is mainly applied across various multimedia, including visual recognition in images and videos (Zhu et al., Jun 2019; Garcia-Garcia et al., 2018; Zhao et al., 2018). Applications of semantic segmentation include autonomous driving, robotic navigation, and Aerial image analysis (Alonso et al., 2020; Liu et al., 2020; Wurm et al., 2019). It is also widely used in medicine fields, such as biomedical imaging, cancer detection, accurate diagnosis and surgical procedures (Wu et al., 2020; Taghanaki et al., 2019; Müller and Kramer, 2021).

Fully Supervised Semantic Segmentation (FSSS) has been considered as a conventional solution for semantic segmentation (Pinheiro and Collobert, 2014). FSSS trains models with pixel-level annotations in a supervised manner and outputs a segmentation mask. Fully Convolutional Networks (FCN) (Shelhamer et al., 2017) and U-Net (Ronneberger et al., 2015) are the most representative methods in FSSS. FCN and U-Net are comprised encoder–decoder layers. Encoder layers extract semantic and contextual information as a downsampling path, whereas decoder layers localize objects as an upsampling path. Furthermore, Deep Convolutional Neural Networks (DCNN) extract semantic-aware features from deep layers, but loose spatial details due to pooling and stride convolution operations (Xia et al., Dec 2013; Qiao et al., 2017). DeepLabV3 explicitly controls the density of extracted features using dilated convolutions and Conditional Random Field (CRF) for the restoration of spatial information (Chen et al., 2017). However, FSSS requires a large number of pixel-level annotation, which is time-consuming and labor-intensive.

Weakly Supervised Semantic Segmentation (WSSS) has proposed efficient and promising solutions in semantic segmentation using image-level labels (e.g., weakly supervised data) (Ahn and Kwak, 2018; Kolesnikov and Lampert, 2016; Oh et al., 2017; Oquab et al., 2014; Pinheiro and Collobert, 2015). In WSSS, image-level labels indicate only presence of objects on images without the information of object location. WSSS identifies class-specific regions with weakly supervised data by excluding potential false positive pixels, intrinsically generated by the weak supervision, in an image. Typically, Class Activation Maps (CAM) have been widely used to identify class-specific regions in weakly supervised settings. CAM computes class-specific scores on each pixel and segment them into pseudo pixel-level annotations. The generated pseudo pixel-level annotations are used as supervision for training a fully supervised segmentation network.

Most modern WSSS methods use the two approaches to improve CAM-based pseudo pixel-level annotations: (1) Seeded Region Growing

* Corresponding author.
E-mail address: mingon.kang@unlv.edu (M. Kang).

https://doi.org/10.1016/j.cviu.2023.103886
Received 9 March 2023; Received in revised form 20 September 2023; Accepted 15 November 2023
Available online 22 November 2023

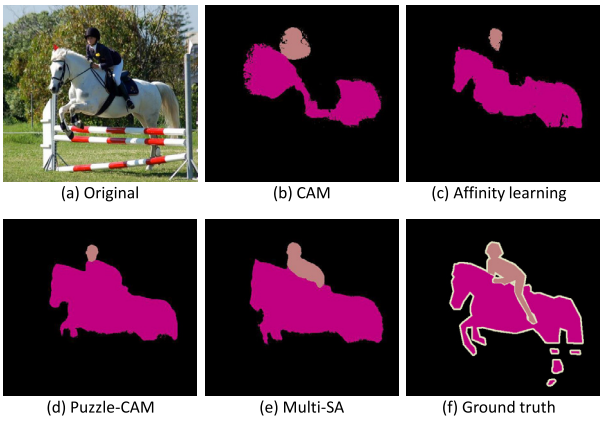

Fig. 1. Semantic segmentation of our proposed Multi-SA. (a) An original image; Segmentation results with (b) conventional CAM, (c) conventional CAM with affinity learning (e.g., AffinityNet), (d) Puzzle-CAM, and (e) Multi-SA; (f) ground truth.

(SRG) and (2) affinity learning coupled with CAM. SRG initially uses CAM to identify class-specific regions as initial seeds and then grows the seeds towards homogeneous regions (Kolesnikov and Lampert, 2016; Adams and Bischof, 1994). Deep Seeded Region Growing (DSRG) employs saliency maps to classify foreground from background (Huang et al., 2018). Then, DSRG stacks the seeds from both foreground and background into a single channel segmentation network and thereby grows the seeds to adjacent pixels by following a region similarity criteria. Mining Common Object Features (MCOF) localize objects from superpixels and CAM (Wang et al., Jun 2018). Then, MCOF trains a segmentation network with object regions as supervision to generate pseudo pixel-level annotations.

Anti-adversarially CAM (AdvCAM) generates attribute maps that progressively identify regions of a target object by using an anti-adversarial technique.

AdvCAM allows non-discriminative regions to be involved in subsequent classifications (Lee et al., 2022). Uncertainty estimation via Response scaling for Noise mitigation (URN) reduces noisy pixels by rescaling pseudo masks multiple times (Li et al., 2022b).

On the other hand, affinity learning-based approaches have improved CAM-based identification of class-specific regions by computing an affinity matrix that represents pairwise relationships between pixels (Maire et al., Jun 2016). AffinityNet creates a neighborhood graph based on initial CAM and computes similarity scores (i.e., affinity matrix) between pixels in the graph (Ahn and Kwak, 2018). Then, AffinityNet multiplies the affinity matrix with CAM to propagate activations in nearby semantically identical areas. Response Expansion by Transferring Semantic Affinity and Boundary (RETAB) learns both semantic affinities and CAM from images, and then splits CAM into boundary and non-boundary regions (Zhou et al., 2021). RETAB improves CAM using an affinity-based propagation along with random walk on boundary and non-boundary regions separately. Besides of the two approaches, Puzzle-CAM improves initial CAM by dividing an image into multiple patches and generating CAM on each patch (Jo and Yu, 2021). Then, Puzzle-CAM merges the multiple patches' CAM into a single CAM and trains a classifier with a reconstruction regularization loss to minimize the loss between CAM of the original image and the merged CAM.

Recently, self-attention mechanisms with CAM have shown significant improvement in WSSS by computing attentions to pixels belonging to objects in an image (Li et al., 2018; Wang et al., 2020; Liang et al., 2021). Self-attention mechanism produces self-attention feature maps from Convolutional Neural Networks (CNN), where the size of a self-attention feature map is identical to the input size. Then, CAM are generated from the self-attention feature maps, which are more informative than without self-attention. For instance, Self-supervised Equivariant Attention Mechanism (SEAM) integrates a self-attention mechanism into a pixel correlation module that captures contextual information, using equivariant cross regularization loss to generate a CAM with less over-activated and under-activated regions (Wang et al., 2020). Saliency Guided Self-attention Network (SGAN) propagates activations from discriminative regions to non-discriminative regions by performing element-by-element multiplication of saliency and spatial attention maps (Yao and Gong, 2020). SGAN captures contextual information from spatial and saliency attention maps to generate initial CAM.

However, the current approaches with self-attention on CAM have the two limitations: incomplete supervision and low resolution. In WSSS, pseudo pixel-level annotations often miss non-discriminative parts of objects. Most WSSS methods extract feature maps mainly from the last convolution layer for initial CAM and rescale them to the input size (Yao and Gong, 2020; Xiu et al., 2020). However, deeper convolutional layers with multiple pooling layers produce feature maps of lower resolution, limiting the ability of CAM to capture fine-grained details of objects. Additionally, deeper convolutional layers tend to include more discriminative features to improve predictive performance rather than segmentation. Thus, the final CAM are limited to capture coarse regions of a object.

We present a novel and effective approach in addressing the above WSSS problems with multi-layered self-attention mechanisms. Our method, Multi-layered Self-Attention (a.k.a. Multi-SA) includes a single network that extracts feature maps at multi-levels of intermediate convolutional layers as well as the last convolutional layer to accurately capture whole object for semantic segmentation. The CAM generated from multi-level feature maps contains various levels of object information, which produces the finest complete object masks from an image. Fig. 1 illustrates that pixel-level pseudo masks, by Multi-SA, localize most whole objects compared to traditional CAM, CAM with affinity learning, and Puzzle-CAM, which is one of the current best WSSS models using affinity learning coupled with CAM. In summary, our main contributions are:

- Multi-SA improves initial CAM with self-attention mechanisms on multiple convolutional layers by incorporating intermediate features in a network in weakly supervised semantic segmentation, and
- Multi-SA is more capable of capturing complex patterns of objects (e.g., person) through the multiple intermediate features than the other existing benchmark models.

The rest of the paper is organized as follows. Section 2 describes a typical WSSS framework and our proposed Multi-SA in detail. We demonstrate experimental results for the performance comparison between Multi-SA and current state-of-the-arts methods in Section 3.

2. Methods

2.1. Overview of the typical WSSS framework

Most WSSS follows a framework to create pseudo pixel-level annotations. The framework typically includes three steps: (1) generating initial CAM from a backbone model (e.g., ResNet (He et al., 2015)), (2) improvising the initial CAM using affinity learning to create pseudo pixel-level annotations, and (3) training a fully supervised segmentation network with the pseudo pixel-level annotations. First, weighted sum of feature maps (i.e. CAM) is generated from a pre-trained backbone model coupled with the Global Average Pooling (GAP) layer. CAM identify class-specific regions by up-sampling CAM to the input size. Second, the initial CAM are refined by affinity learning that learns pixel-level affinities in random walk to propagate activation scores of CAM (Lov´asz and Lov´asz, 1993). The improved CAM are then used to synthesize pseudo pixel-level annotations. Lastly, the generated pseudo pixel-level annotations are considered as supervision in a fully supervised semantic segmentation model (e.g., DeepLabV3). In this study, our proposed method focuses on the first step, which improves initial CAM, in the framework (see Fig. 2).

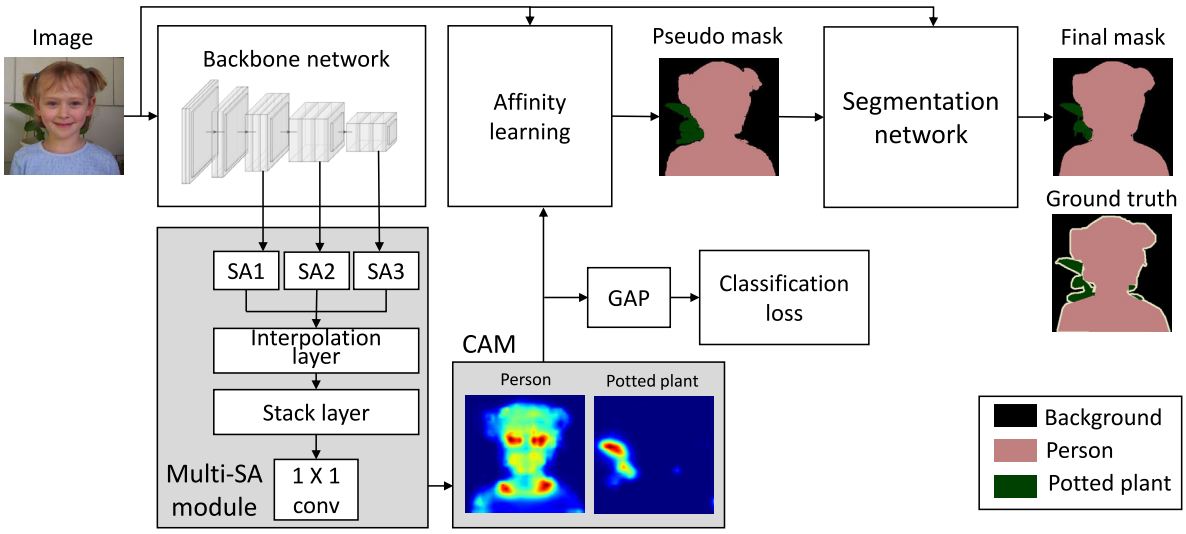

Fig. 2. The overall framework of proposed Multi-layered Self-Attention mechanism (Multi-SA).

2.2. Motivation

We hypothesize that self-attention mechanisms at multiple intermediate layers of a network may retrieve class-specific features that are not present in the last convolutional layer, and the combination of feature maps from intermediate layers and the final convolutional layers may provide richer features for semantic segmentation than with only the last convolutional layer. Self-attention improves convolutional feature maps by capturing dependency of context features with attention mechanisms. However, most current state-of-the-art methods applied self-attention mechanisms to the last convolutional layer only, which mainly covers discriminative regions of objects (Wang et al., 2020; Yao and Gong, 2020; Sun et al., 2020). In a network, the outputs of intermediate layers have larger resolutions than of the last convolution layer. The intermediate layers include features representing visual patterns of objects as well as discriminative features, whereas the final convolutional layer mainly includes discriminative features. For instance, VGG16, one of the most popular CNN models, comprises 13 convolutional and 5 pooling layers with an input image of size 224 × 224. The feature maps extracted from the 7th intermediate convolutional layer has a output size of 56 × 56, whereas the feature maps extracted from the last convolutional layer has a output size of 14 × 14. The feature maps of the 7th convolutional layer include visual patterns of objects, and the flattened feature maps in the last convolutional layer are introduced to dense layers. The coarse flattened feature maps are optimized for classification tasks.

2.3. Multi-layered self-attention mechanism

We propose a Multi-layered Self-Attention (Multi-SA) that improves pseudo pixel-level annotations in a weakly supervised semantic segmentation setting by integrating attention features of multiple convolutional layers. Multi-SA consists of the two components (see gray boxes in Fig. 2): (1) multiple self-attention layers in a Multi-SA module and (2) the integration of the multiple self-attention layers to improve initial CAM. The Multi-SA module includes attention features from multiple intermediate layers as well as the last convolution layer of a network (e.g., ResNet) with a self-attention mechanism that produces self-attention feature maps individually. Then, Multi-SA computes multiple self-attention maps and combine them with interpolation along channel dimension in the stack layer. The stack layer creates an initial CAM that accurately highlight class-specific regions of a whole object.

2.3.1. Multiple self-attention layers in a multi-SA module

Given training set of n number of samples, $\mathcal{D} = \{(\mathbf{X}_i, \mathbf{y}_i)\}_n$, where $\mathbf{X}_i$ is the ith input image, and $\mathbf{y}_i \in \{0,1\}^p$ is an one-hot encoding image label associated with the p number of semantic objects (i.e., ground truth). Let a multi-class classification network of a backbone model includes m number of convolutional layers. In Multi-SA, we consider k number of convolutional layers to integrate among the m layers in a network ($k \leq m$). The k number of convolutional layers can be any intermediate or last convolutional layers in a network. For instance, last k consecutive layers in AlexNet or last layers of k ResNet blocks in ResNet can be considered.

Feature maps from the lth convolution layer ($1 \leq l \leq k$) are denoted as $F_l \in \Re^{h_l \times w_l \times c_l}$, where h_l and w_l are the height and the width of the feature map respectively, and c_l is the number of feature maps in the convolutional layer. Three 1×1 convolutions are applied on F_l to create feature maps, f_1, f_2 and f_3 to calculate attentions on pixels towards the class objects (see Fig. 3). The feature maps of f_1, f_2 and f_3 are represented as:

$$f_r = W_r F_l^\top, \quad r = \{1,2,3\}, \tag{1}$$

where $W_r \in \Re^{c_l \times c_l}$ is a weight matrix of 1×1 convolutions. Then, dot product between f_1 and f_2 is computed to obtain a pixel-pairwise similarity score, S_l, as:

$$S_l = f_1^\top f_2. \tag{2}$$

Then, softmax is applied to S_l for generating an attention map. The attention map, A_l, contains attention scores of pairs of pixels. A high attention score indicates that two pixels belong to the same class. The attention map, A_l, is represented as:

$$A_l = Softmax(S_l), \tag{3}$$

where $A_l \in \Re^{(h_l \times w_l) \times (h_l \times w_l)}$. Self-attention feature maps, O_l, are obtained by multiplying A_l with f_3 for retaining the shape of the original input:

$$O_l = A_l f_3^\top, \tag{4}$$

where $O_l \in \Re^{h_l \times w_l \times c_l}$.

2.3.2. Integration of the multiple self-attention layers

Self-attention feature maps, O_l ($1 \leq l \leq k$), are obtained from the k number of self-attention layers. The self-attention feature maps are added with a adaptive trainable weight, β_l, on the original feature

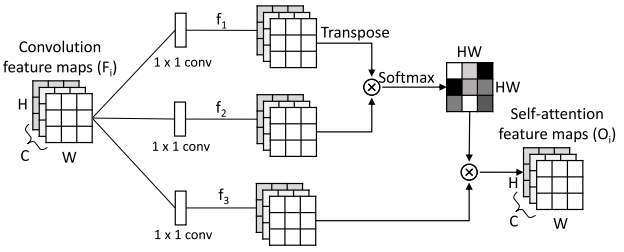

Fig. 3. The architecture of self-attention.

maps, F_l, to create enhanced feature maps. The enhanced feature maps V_i are represented as:

$$V_l = \beta_l O_l + F_l, \tag{5}$$

where $V_l \in \Re^{h_l \times w_l \times c_l}$.

Then, V_l $(2 \le l \le k)$ is resized to the same shape of V_1 by a bi-linear interpolation (ϕ) in the interpolation layer, since V_1 is the feature maps of the highest resolution. The bi-linear interpolated feature maps are stacked along the channel dimension to create a stacked features maps, Z, that are computed as:

$$Z = \begin{bmatrix} V_1 \\ \phi(V_2) \\ \vdots \\ \phi(V_k) \end{bmatrix}, \tag{6}$$

where $Z \in \Re^{h_1 \times w_1 \times c^*}$ and $c^* = \sum_{i=1}^{k} c_i$.

Then, the final feature maps, Θ, are computed through a 1×1 convolution layer on the stacked feature maps, Z, to capture relevant feature maps for localizing whole objects. The final features maps, Θ, are computed by:

$$\Theta = W Z^{\top}, \tag{7}$$

where $\Theta \in \Re^{h_1 \times w_1 \times c^*}$ and $W \in \Re^{c^* \times c^*}$ is the weight matrix of 1×1 convolution. Finally, Θ is followed by a GAP and classifier layers. The CAM extracted from Θ are used as an input for affinity learning to generate a pseudo segmentation mask.

2.4. CAM and loss function

In a WSSS framework (Fig. 2), CAM is utilized to extract class-specific features. After training Multi-SA, we extract CAM (M_i) for each class by projecting the weights of classifier on convolution feature maps (Θ) of Multi-SA:

$$M_i = W_i^{\top} \Theta, \qquad 1 \le i \le p \tag{8}$$

where W_i are classifier weights for the ith class. M_i is further normalized with a maximum value of M_i as:

$$M_i = M_i / max(M_i). \tag{9}$$

Thus, the value range of M_j is between zero and one. Background and foreground objects can be separated by the optimal threshold maximizing mIoU with validation data.

2.5. Loss function

We used a multi-label soft margin loss function, $\ell(\mathbf{X}, y)$, for a multi-label classification network. In the network setting of backbone model, the classification loss is computed as:

$$\ell(\mathbf{X}, y) = \frac{-1}{p-1} \sum_{i=1}^{p-1} y_i \log\left(\frac{1}{1+exp(-\mathbf{X}_i)}\right) + (1-y_i) \log\left(\frac{exp(-\mathbf{X}_i)}{1+exp(-\mathbf{X}_i)}\right). \tag{10}$$

The overall procedure of Multi-SA is explained in Algorithm 1.

Algorithm 1: An algorithm for Multi-SA

Data: Training data, $\mathcal{D}$, and k number of convolutional layers
Result: Multi-SA-based CAM, M_i
for $1 \le l \le k$ **do**
  $f_r \leftarrow W_r F_l^{\top}$ $(1 \le r \le 3)$;
  $S_l \leftarrow f_1^{\top} f_2$;
  $A_l \leftarrow Softmax(S_l)$;
  $O_l \leftarrow A_l f_3^{\top}$;
  $V_l \leftarrow \beta_l O_l + F_l$;
end
$Z = V_1$;
for $2 \le l \le k$ **do**
  Stack $\phi(V_l)$ on Z along channels dimension
end
$\Theta \leftarrow W Z^{\top}$;
for $1 \le i \le p$ **do**
  $M_i \leftarrow W_i^{\top} \Theta$;
end

3. Experiments

3.1. Dataset

We evaluated the performance of Multi-SA using the PASCAL VOC 2012 and MS COCO 2014 data, which are the most popular benchmark datasets in WSSS. PASCAL VOC 2012 consists of three datasets of 1,464 (PASCAL-I), 1,449 (PASCAL-II), and 1,456 (PASCAL-III) images, where PASCAL-I and PASCAL-II contain pixel-level annotations of 21 classes (20 foreground objects and background). PASCAL-III includes only images, where pixel-level annotations are not available in the public. The model performance is evaluated with PASCAL-III in the PASCAL VOC evaluation server. Note that PASCAL I–III refer to training, validation, and test data in the PASCAL VOC 2012, respectively. We also considered additional 10,582 images (a.k.a. SBD) from the Semantic Boundary Dataset (Hariharan et al., 2011), which is originally derived from the training data of PASCAL VOC 2012 but includes only image-level labels rather than pixel-level annotations. We randomly re-scaled SBD images in the range between 320 and 640, and then cropped to 512×512 pixel images for the input of the network.

The MS COCO 2014 dataset consists of 123,287 images, which comprises 82,783 training images and 40,504 validation images. MS COCO includes 81 classes (80 foreground objects and background). We excluded training samples that lack image-level labels in the experiment. The images were re-scaled to fit the networks. It is worth noting that PASCAL VOC has been considered as the primary benchmark dataset that includes meticulous pixel-level annotations for the WSSS problem. Whereas, the MS COCO dataset often includes small-scale partial objects, and its annotations are mainly limited to instance outlines. Thus, MS COCO is more suitable for object detection or instance segmentation, although it has also been used as supplementary benchmark data.

In the experiments with PASCAL VOC 2012, we trained our model using SBD images and generated initial CAM. We used only image-level labels without pixel-level annotations in a weakly supervised learning manner. PASCAL-I was with tuning and ablation study, and PASCAL-II and PASCAL-III were for the performance comparison with the current state-of-the-art methods. Similarly, we trained the model with MS COCO training images in the experiments with MS COCO 2014. Then, the MS COCO validation data was used for the comparison of mIoU with current state of the art methods.

3.2. Backbone models

For the experiments, we adopted ResNet (50 and 101 layers) and ResNeSt (101 layers) architectures as back-bone networks with pre-trained ImageNet weights. We integrated the Multi-SA module to the backbone network to extract the feature maps from the multiple layers,

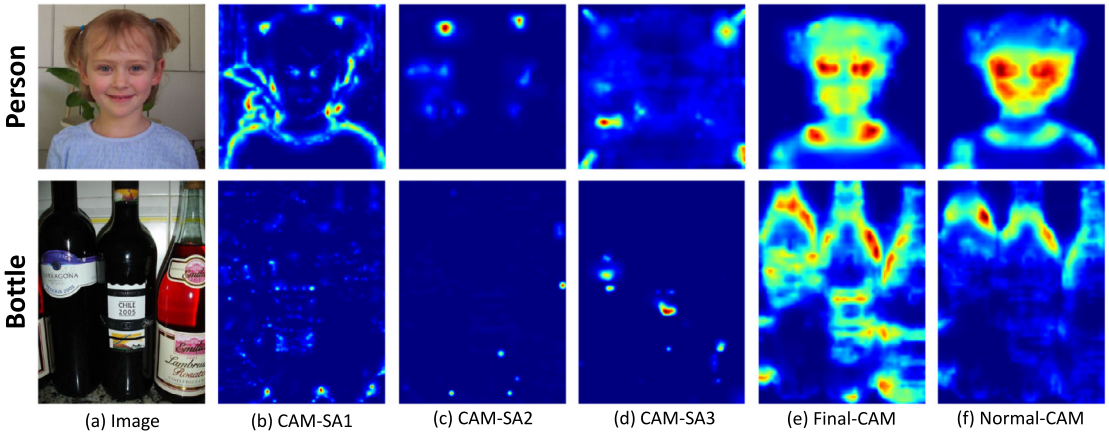

Fig. 4. Comparison of CAM at multiple levels of Multi-SA. (a) Original images of person and bottle, CAM extracted at the last convolutional layers of the last three consecutive blocks, (b) SA1, (c) SA2, (d) SA3, in ResNeSt-101 network, (e) the final CAM by Multi-SA network, and (f) conventional CAM extracted from the last convolutional layer of a traditional classification network without Multi-SA module.

Table 1
Comparison of multi-SA with various numbers of self-attention mechanisms using ResNet-50 as backbone network.

	CAM w/o SA	1SA	2SAs	3SAs
mIoU	46.30	47.60	47.01	47.93

and applied self-attention mechanisms to generate self-attention feature maps. Then, all self-attention feature maps are combined along the channel dimension. Specifically, ResNet consists of five blocks in the architecture, where each block includes multiple convolutional layers. We considered the last convolutional layers of the three blocks (i.e., block 3–5) for the multiple self-attention mechanisms and extracted self-attention feature maps. The initial CAM are generated from the last 1×1 convolutional layer in the Multi-SA module. PolyOptimizer loss function with momentum of 0.9 and batch size of 8 were used for training backbone network.

3.3. Tuning of Multi-SA

We empirically optimized the hyper-parameter of the number of self-attention mechanisms (i.e., k). We trained Multi-SA with SBD images varying a number of self-attention mechanisms, and computed mean Intersection over Union (mIoU) with PASCAL-I images. Initial CAM from the models were converted into segmentation masks by the optimal threshold that maximizes mIoU. We compared mIoU of Multi-SA with various self-attention mechanisms as well as with a conventional CAM without self-attention mechanism. The highest mIoU (47.93%) was shown with three self-attention mechanisms (see Table 1). The further experiments were conducted with the three self-attention mechanisms in the paper.

3.4. Segmentation with affinity learning

The initial CAM were improved with affinity learning to generate pseudo segmentation masks, which were used as a supervision for segmentation network. CAM were extracted from Multi-SA and improved with AffinityNet by learning semantic affinities between pair of pixels (affinity matrix) in an image. Initial CAM with SBD images were used to create affinity labels. AffinityNet was trained using affinity labels as supervision to generate a affinity matrix and multiplied it over the initial CAM. Then, it was diffused by Random Walk (RW) for improving the initial CAM. Since the resolution of CAM were smaller than the input image, they were resized with dense Conditional Random Field (dCRF).

The improved initial CAM were converted into pseudo pixel-level annotations by the optimal threshold that maximizes mIoU. We used DeepLabV3+ using the pseudo pixel-level annotations in an supervised manner for the final semantic segmentation.

3.5. Self-attention feature maps from multiple intermediate convolutional layers in Multi-SA

Multi-SA improves the WSSS task using the self-attention feature maps from multiple convolutional layers. We visualize the CAMs from multiple self-attention feature maps of Multi-SA with images of person and bottle (see Fig. 4). Self-attention feature maps at intermediate convolutional layers capture various levels of visual patterns of an object. Let the self-attention-based CAMs from the last convolutional layers of the last three blocks in ResNeSt-101 network be CAM-SA1, CAM-SA2, and CAM-SA3 in Fig. 4. CAM-SA1 produced the higher output resolution than CAM-SA2 and CAM-SA3. CAM-SA1 mainly captured the boundary of objects, whereas CAM-SA2 separated the foreground objects (person and bottle). CAM-SA3 highlighted discriminative pixels in an image, as it only contains the low spatial information from the last convolution layer of backbone network. The combination of the multiple self-attention-based CAM localize whole object areas and provide accurate boundaries of objects, than normal CAM.

Multi-SA optimizes adaptive trainable weights (β_l in Eq. (5)) for the combination of multiple intermediate layers to generate accurate CAM. The adaptive integration with trainable weights reduces false positives and captures fine-grained object parts, comparing to the similar methods that use multiple intermediate layers for object localization, including LayerCAM and Shallow feature-aware Pseudo supervised Object Localization (SPOL) (Jiang et al., 2021; Wei et al., 2021). Those localization methods have limitations, such as image occlusion. Multi-SA overcomes the limitation by utilizing a self-attention mechanism on individual intermediate layer feature maps and integrate the most relevant ones with trainable adaptive weights. For the optimal model, we obtained the optimal betas of 0.034, 0.142, and −0.351 for CAM-SA1 to CAM-SA3, respectively. The negative beta value of CAM-SA3 in the final layer appears little contribution towards the self-attention mechanism on the feature maps, since the final layer in the backbone model is optimized for classification rather than segmentation. The positive beta values in CAM-SA1 and CAM-SA2 show that the intermediary layers emphasize self-attention and contribute more to segmentation as intermediate information of the objects. Additionally, Multi-SA enhances the features maps from intermediate layers preserving its original dimension. Moreover, Multi-SA assigns higher weights to the important features of objects and suppresses the impact of irrelevant background noise.

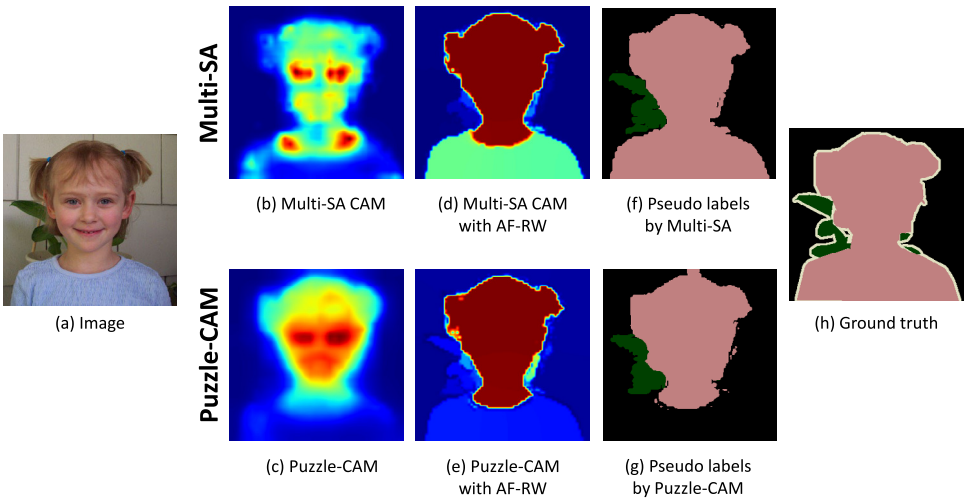

Fig. 5. The comparison of CAM and pseudo labels from different stages of WSSS. Top-row and bottom-row illustrates the results of Multi-SA and Puzzle-CAM. (a) Original Image. CAM generated by (b) Multi-SA and (c) Puzzle-CAM. (d) Multi-SA CAM and (e) Puzzle-CAM generated from the AffinityNet with RW (AF-RW) as post processing method. Pseudo labels generated by (f) Multi-SA and (g) Puzzle-CAM from CAMs of AF-RW without dCRF. (h) Ground truth.

3.6. Ablation studies

We conducted ablation study to verify the effectiveness of Multi-SA on mIoU. First, we trained a backbone network without the Multi-SA module (baseline) using SBD images and computed initial CAM. A mIoU with the initial CAM was calculated using PASCAL-I images. The baseline model achieved the mIoU of 50.64%. Second, we incorporated the Multi-SA module into the backbone network and trained it using SBD images to generate Multi-SA-based initial CAM. The mIoU on the PASCAL-I images was 51.49% with Multi-SA. The Multi-SA-based initial CAM was further significantly improved with AffinityNet and RW. The CAM was rescaled to original image size by using dCRF. We achieved the mIoU of 67.54% by using Multi-SA with RW, and 68.07% by using Multi-SA with both RW and dCRF (see Table 2).

The integration of Multi-SA with AffinityNet and RW resulted in a notable improvement of 16% in mIoU. This enhancement was achieved with Multi-SA's significantly improved initial CAM by incorporating high-resolution boundary information and discriminative features derived from the intermediate convolutional layers. We visualize how Multi-SA improves the WSSS task coupled with AffinityNet and RW by comparing with Puzzle-CAM. Fig. 5 shows the final CAMs that Puzzle-CAM and Multi-SA generated as well as the original image and the ground truth. It demonstrates that Multi-SA CAM shows the complete coverage of the person, and its object boundaries are closer to the ground truth segmentation mask compared to Puzzle-CAM. The object boundary information was mainly captured from the intermediate layers of the backbone network in Multi-SA, which was not captured in Puzzle-CAM, since they primarily focus only on the last convolutional layer of backbone network. The CAMs of Multi-SA and Puzzle-CAM were further enhanced by AffinityNet and RW by diffusing the activation scores along the semantic boundary of an object (see Figs. 5d & 5e). RW diffuses the highest activation scores to every object parts consistently in Multi-SA with the accurate boundary information, whereas Puzzle-CAM's highest activation scores were only limited to the discriminative regions. Pseudo labels were generated from the CAMs for both Multi-SA and Puzzle-CAM.

Furthermore, the Multi-SA CAM were slightly enhanced by dCRF. dCRF enforces label consistency between pixels by considering the neighboring features of the Multi-SA CAM to determine the final labels and improve CAM predictions. Multi-SA and the post-processing techniques led to a significant increase in mIoU, reaching 68.07% on the PASCAL-I dataset. The improved CAM were then converted into pseudo segmentation labels with a threshold and provided as fully-supervised data for segmentation network.

Table 2
Ablation study of Multi-SA using ResNeSt-101 as the backbone on PASCAL-I data. AF-RW: AffinityNet with random walk, dCRF: dense conditional random field.

CAM w/o SA	Multi-SA	AF-RW	dCRF	mIoU
✓				50.64
✓	✓			51.49
✓	✓	✓		67.54
✓	✓	✓	✓	68.07

3.7. Comparison with existing state-of-the-art methods

We compared the performance of the pseudo pixel-level annotations generated from Multi-SA and existing state-of-the-art methods in WSSS. In the experiments, we employed AffinityNet as a common network for affinity learning. The pseudo pixel-level annotations generated from AffinityNet were used as a supervision for segmentation network. DeepLab networks (e.g., DeepLabV1, DeepLabv2, DeepLabv3 and DeepLabv3+) were used as common segmentation networks to perform the final segmentation tasks. For the fair comparison, we excluded CLIP (Lin et al., Jun 2023), MARS (Jo et al., 2023), and SANCE (Li et al., 2022a) that incorporate extra supervision (e.g., text), although they are highly ranked in terms of performance. For instance, CLIP employed additional text data that specifies visual concepts of the classes as a supervision (Lin et al., Jun 2023). MARS utilizes an additional unsupervised semantic segmentation pre-trained model (USS) (Jo et al., 2023). SANCE used extra data of contour detection networks (Li et al., 2022a).

Table 3 shows the mIoU of the 21 classes for the proposed method (Multi-SA) on the PASCAL-II images, comparing to the state-of-the-art methods. Multi-SA produced the highest overall mIoU of 69.7%, showing significant improvement on the classes of aeroplane, bike, boat, cow, table, person, sheep, and sofa. It is worth noting that Multi-SA achieved the highest mIoU of 79.6% in the class of *person* in

Table 3
Category performance comparisons on PASCAL-II data with image-level supervision.

Model	bkg	aero	bike	bird	boat	bottle	bus	car	cat	chair	cow	table	dog	horse	mbk	person	plant	sheep	sofa	train	tv	mIoU
AffinityNet (CVPR'18)	88.2	68.2	30.6	81.1	49.6	61.0	77.8	66.1	75.1	29.0	66.0	40.2	80.4	62.0	70.4	73.7	42.5	70.7	42.6	68.1	51.6	61.7
SEAM (CVPR'20)	88.8	68.5	33.3	85.7	40.4	67.3	78.9	76.3	81.9	29.1	75.5	48.1	79.9	73.8	71.4	75.2	48.9	79.8	40.9	58.2	53.0	64.5
FickleNet (CVPR'19)	89.5	76.6	32.6	74.6	51.5	71.1	83.4	74.4	83.6	24.1	73.4	47.4	78.2	74.0	68.8	73.2	47.8	79.9	37.0	57.3	64.6	64.9
SC-CAM (CVPR'20)	88.8	51.6	30.3	82.9	53.0	75.8	88.6	74.8	86.6	32.4	79.9	53.8	82.3	78.5	70.4	71.2	40.2	78.3	42.9	66.8	58.8	66.1
Puzzle-CAM (ICIP'21) (ResNeSt-101)	88.3	82.0	35.0	87.9	63.5	75.4	87.3	77.4	93.2	28.4	86.6	28.4	89.2	82.8	78.5	32.2	57.0	84.5	37.9	69.0	41.4	66.9
AdvCAM (PAMI'22)	90.0	79.8	34.1	82.6	63.3	70.5	89.4	76.0	87.3	31.4	81.3	33.1	82.5	80.8	74.0	72.9	50.3	82.3	42.2	74.1	52.9	68.1
Multi-SA (ResNeSt-101)	89.6	**83.1**	**35.5**	84.2	**69.0**	59.1	86.1	80.9	88.4	30.0	**87.1**	**58.5**	80.9	82.1	**79.2**	**79.6**	54.7	**86.0**	**53.6**	55.8	41.0	**69.7**

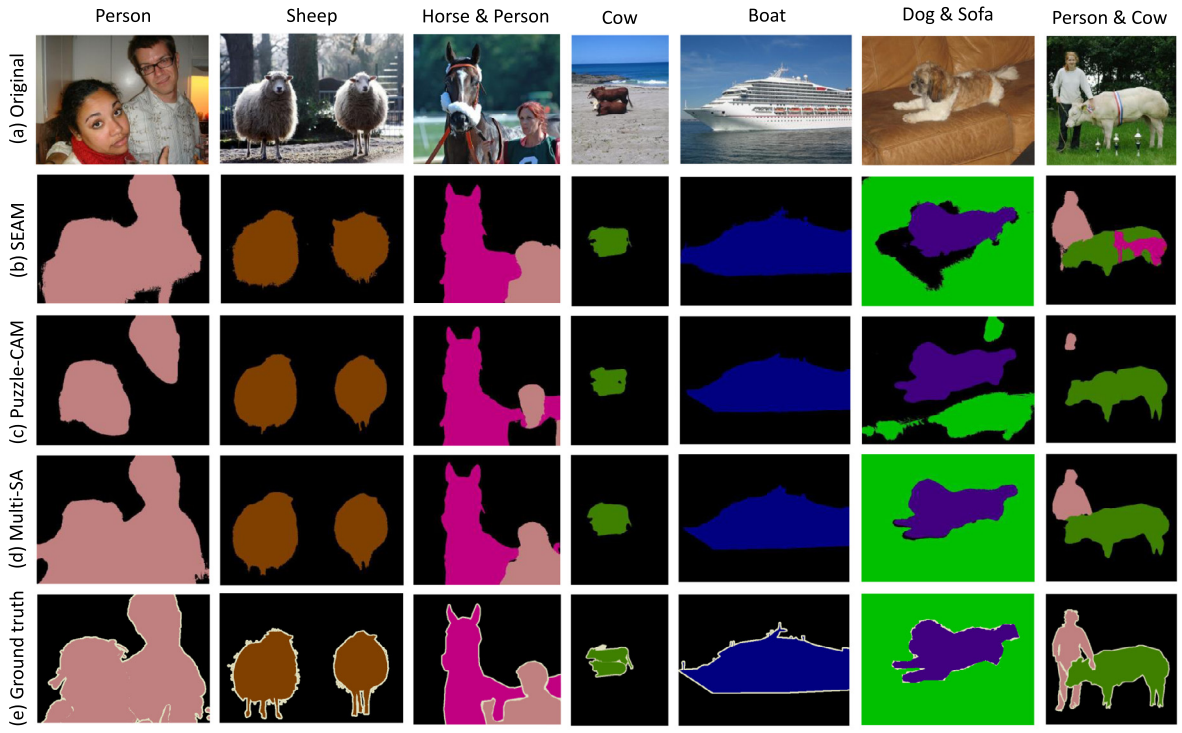

Fig. 6. Comparison of segmentation results on the PASCAL VOC 2012 val set. (a) Original images, Segmentation results of (b) SEAM with ResNet-38, (c) Puzzle-CAM with ResNeSt-101, and (d) Multi-SA, and (e) ground truth.

the PASCAL VOC dataset, which improved 4.1% mIoU comparing to the second best. This significant improvement broadens the scope of Multi-SA's potential uses in a number of applications, including face recognition, human behavior research, and augmented reality, where segmentation of person holds substantially importance.

We noticed that Puzzle-CAM's mIoU results are inconsistent across majority (person) and minority (plant) classes of PASCAL VOC data (see Table 3). On the other hand, Multi-SA produced consistent results across all classes of PASCAL VOC data and achieved higher mIoU overall.

Furthermore, we compared the performance with a number of state-of-the-art methods on PASCAL-II and PASCAL-III images. Multi-SA showed the highest mIoU of 69.7% and 70.1% on PASCAL-II (validation) and PASCAL-III (test), respectively (see Table 4).

Additionally, we evaluated the performance of Multi-SA by using the MS COCO dataset (see Table 4). Multi-SA showed competitive performance to the benchmark models, obtaining the total mIoU of 40.9% with MS COCO.

We illustrated pixel-level annotation results of Multi-SA comparing to SEAM and Puzzle-CAM as well as ground truth on several images of PASCAL-II (Fig. 6). The segmentation results of Multi-SA localized objects of interest on the images with single and multi-labels. Multi-SA detected smooth and accurate boundaries of the objects that were close to ground truth. Compared with SEAM and Puzzle-CAM, Multi-SA localized both small and large objects with highest coverage area.

4. Conclusion

In this paper, we propose the Multi-SA module to capture whole object in an image using only image-level labels in an weakly supervised setting. Our method Multi-SA employs multiple self-attention mechanism, which improves initial CAM by integrating intermediate features that represents visual pattern of an object. The generated CAM are further refined with affinity learning, which generates pseudo pixel-level annotations. The segmentation network is trained with the supervision of pseudo pixel-level annotations. The Multi-SA method can

Table 4
Comparison of Multi-SA and existing state-of-the-art methods on the PASCAL-II, PASCAL-III and COCO 2014 images with only image-level labels as a supervision.

Methods	Backbone	PASCAL-II (Validation)	PASCAL-III (Test)	COCO 2014 (Validation)
AffinityNet (CVPR'18)	ResNet-38	61.7	63.7	-
IRNet (CVPR'19)	ResNet-50	63.5	64.8	-
ICD (CVPR'20)	ResNet-101	64.1	64.3	-
SEAM (CVPR'20)	ResNet-38	64.5	65.7	31.9
SC-CAM (CVPR'20)	ResNet-101	66.1	65.9	-
Puzzle-CAM (ICIP'21)	ResNeSt-101	66.9	67.7	-
AdvCAM (PAMI'22)	ResNet-101	68.1	68.0	-
URN (AAAI'22)	ResNet-101	69.5	69.7	40.7
Ours (Multi-SA)	ResNeSt-101	**69.7**	**70.1**	40.9

be applied to any convolutional neural network architecture, such as VGG, inception networks, and general adversarial networks.

CRediT authorship contribution statement

Avinash Yaganapu: Methodology, Writing – original draft, Writing – review & editing, Visualization, Experiment. **Mingon Kang:** Conceptualization, Methodology, Writing – original draft, Writing – review & editing, Supervision, Project administration, Funding acquisition.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Data availability

Data will be made available on request.

Acknowledgments

We acknowledge the support from National Science Foundation Major Research Instrumentation (NSF MRI), United States of America (Grant#:2117941) and Nevada NASA, United States of America.

References

Adams, R., Bischof, L., 1994. Seeded region growing. IEEE Trans. Pattern Anal. Mach. Intell. 16 (6), 641–647. http://dx.doi.org/10.1109/34.295913, URL: https://ieeexplore.ieee.org/document/295913.

Ahn, J., Kwak, S., 2018. Learning pixel-level semantic affinity with image-level supervision for weakly supervised semantic segmentation. In: Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition. IEEE Computer Society, pp. 4981–4990. http://dx.doi.org/10.1109/CVPR.2018.00523, URL: https://arxiv.org/abs/1803.10464v2.

Alonso, I., Riazuelo, L., Murillo, A.C., 2020. MiniNet: An efficient semantic segmentation ConvNet for real-time robotic applications. IEEE Trans. Robot. 36 (4), 1340–1347. http://dx.doi.org/10.1109/TRO.2020.2974099, URL: https://ieeexplore.ieee.org/document/9023474.

Chen, L.C., Papandreou, G., Schroff, F., Adam, H., 2017. Rethinking atrous convolution for semantic image segmentation. URL: https://arxiv.org/abs/1706.05587.

Garcia-Garcia, A., Orts-Escolano, S., Oprea, S., Villena-Martinez, V., Martinez-Gonzalez, P., Garcia-Rodriguez, J., 2018. A survey on deep learning techniques for image and video semantic segmentation. Appl. Soft Comput. 70, 41–65. http://dx.doi.org/10.1016/j.asoc.2018.05.018, URL: https://www.sciencedirect.com/science/article/pii/S1568494618302813.

Hariharan, B., Arbeláez, P., Bourdev, L., Maji, S., Malik, J., 2011. Semantic contours from inverse detectors. In: Proceedings of the IEEE International Conference on Computer Vision. pp. 991–998. http://dx.doi.org/10.1109/ICCV.2011.6126343.

He, K., Zhang, X., Ren, S., Sun, J., 2015. Deep residual learning for image recognition. In: 2016 IEEE Conference on Computer Vision and Pattern Recognition. CVPR, pp. 770–778, URL: https://api.semanticscholar.org/CorpusID:206594692.

Huang, Z., Wang, X., Wang, J., Liu, W., Wang, J., 2018. Weakly-supervised semantic segmentation network with Deep Seeded Region growing. In: Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition. IEEE Computer Society, pp. 7014–7023. http://dx.doi.org/10.1109/CVPR.2018.00733.

Jiang, P.T., Zhang, C.B., Hou, Q., Cheng, M.-M., Wei, Y., 2021. LayerCAM: Exploring hierarchical class activation maps for localization. IEEE Trans. Image Process.

Jo, S., Yu, I.J., 2021. Puzzle-CAM: Improved localization via matching partial and full features. In: Proceedings of the IEEE International Conference on Information Processing. Institute of Electrical and Electronics Engineers (IEEE), pp. 639–643. http://dx.doi.org/10.1109/ICIP42928.2021.9506058.

Jo, S., Yu, I.J., Kim, K., 2023. MARS: Model-agnostic biased object removal without additional supervision for weakly-supervised semantic segmentation. http://dx.doi.org/10.48550/arxiv.2304.09913, URL: https://arxiv.org/abs/2304.09913.

Kolesnikov, A., Lampert, C.H., 2016. Seed, expand and constrain: Three principles for weakly-supervised image segmentation. In: European Conference on Computer Vision. ECCV, Springer.

Lee, J., Kim, E., Mok, J., Yoon, S., 2022. Anti-adversarially manipulated attributions for weakly supervised semantic segmentation and object localization. IEEE Trans. Pattern Anal. Mach. Intell. PP, 1. http://dx.doi.org/10.1109/TPAMI.2022.3166916, URL: https://ieeexplore.ieee.org/document/9756329.

Li, Y., Duan, Y., Kuang, Z., Chen, Y., Zhang, W., Li, X., 2022b. Uncertainty estimation via response scaling for pseudo-mask noise mitigation in weakly-supervised semantic segmentation. In: Proceedings of the ... AAAI Conference on Artificial Intelligence, Vol. 36, No. 2. pp. 1447–1455. http://dx.doi.org/10.1609/aaai.v36i2.20034.

Li, J., Fan, J., Zhang, Z., 2022a. Towards noiseless object contours for weakly supervised semantic segmentation. In: 2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition. CVPR, pp. 16835–16844. http://dx.doi.org/10.1109/CVPR52688.2022.01635.

Li, K., Wu, Z., Peng, K.C., Ernst, J., Fu, Y., 2018. Tell me where to look: Guided attention inference network. In: Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition. IEEE Computer Society, pp. 9215–9223. http://dx.doi.org/10.1109/CVPR.2018.00960.

Liang, Y., Li, M., Jiang, C., 2021. Generating self-attention activation maps for visual interpretations of convolutional neural networks. Neurocomputing http://dx.doi.org/10.1016/J.NEUCOM.2021.11.084.

Lin, Y., Chen, M., Wang, W., Wu, B., Li, K., Lin, B., Liu, H., He, X., Jun 2023. CLIP is Also an Efficient Segmenter: A Text-Driven Approach for Weakly Supervised Semantic Segmentation. IEEE, pp. 15305–15314. http://dx.doi.org/10.1109/CVPR52729.2023.01469, URL: https://ieeexplore.ieee.org/document/10203854.

Liu, X., Han, Y., Bai, S., Ge, Y., Wang, T., Han, X., Li, S., You, J., Lu, J., 2020. Importance-aware semantic segmentation in self-driving with discrete wasserstein training. In: AAAI 2020 - 34th AAAI Conference on Artificial Intelligence. AAAI Press, pp. 11629–11636. http://dx.doi.org/10.48550/arxiv.2010.12440, URL: https://arxiv.org/abs/2010.12440v1.

Lov´asz, L., Lov´asz, L., 1993. Random walks on graphs: A survey. Bolyai Soc. Math. Stud. 2, 1–46.

Maire, M., Narihira, T., Yu, S.X., Jun 2016. Affinity CNN: Learning pixel-centric pairwise relations for figure/ground embedding. pp. 174–182. http://dx.doi.org/10.1109/CVPR.2016.26, URL: https://ieeexplore.ieee.org/document/7780395.

Müller, D., Kramer, F., 2021. Miscnn: a framework for medical image segmentation with convolutional neural networks and deep learning. BMC Med. Imag. 21, 1–11. http://dx.doi.org/10.1186/S12880-020-00543-7/FIGURES/5, URL: https://bmcmedimaging.biomedcentral.com/articles/10.1186/s12880-020-00543-7.

Oh, S.J., Benenson, R., Khoreva, A., Akata, Z., Fritz, M., Schiele, B., 2017. Exploiting saliency for object segmentation from image level labels. In: Proceedings - 30th IEEE Conference on Computer Vision and Pattern Recognition. CVPR 2017, Vol. 2017-January, Institute of Electrical and Electronics Engineers Inc., pp. 5038–5047. http://dx.doi.org/10.1109/CVPR.2017.535, URL: https://arxiv.org/abs/1701.08261v2.

Oquab, M., Bottou, L., Laptev, I., Sivic, J., 2014. Learning and transferring mid-level image representations using convolutional neural networks. In: Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition. IEEE Computer Society, pp. 1717–1724. http://dx.doi.org/10.1109/CVPR.2014.222.

Pinheiro, P., Collobert, R., 2014. Recurrent convolutional neural networks for scene labeling. In: Xing, E.P., Jebara, T. (Eds.), Proceedings of the 31st International Conference on Machine Learning. In: Proceedings of Machine Learning Research, vol. 32, PMLR, Bejing, China, pp. 82–90, URL: https://proceedings.mlr.press/v32/pinheiro14.html.

Pinheiro, P.O., Collobert, R., 2015. From image-level to pixel-level labeling with convolutional networks. In: Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition. 07-12-June-2015, IEEE Computer Society, pp. 1713–1721. http://dx.doi.org/10.1109/CVPR.2015.7298780, URL: https://arxiv.org/abs/1411.6228v3.

Qiao, K., Chen, J., Wang, L., Zeng, L., Yan, B., 2017. A top-down manner-based DCNN architecture for semantic image segmentation. PLoS One 12 (3), e0174508. http://dx.doi.org/10.1371/journal.pone.0174508, URL: https://www.ncbi.nlm.nih.gov/pubmed/28339486.

Ronneberger, O., Fischer, P., Brox, T., 2015. U-net: Convolutional networks for biomedical image segmentation. In: Medical Image Computing and Computer-Assisted Intervention – MICCAI 2015, Springer International Publishing, Cham, pp. 234–241. http://dx.doi.org/10.1007/978-3-319-24574-4_28, URL: http://link.springer.com/10.1007/978-3-319-24574-4_28.

Shelhamer, E., Long, J., Darrell, T., 2017. Fully convolutional networks for semantic segmentation. IEEE Trans. Pattern Anal. Mach. Intell. 39 (4), 640–651. http://dx.doi.org/10.1109/TPAMI.2016.2572683, URL: https://ieeexplore.ieee.org/document/7478072.

Sun, G., Wang, W., Dai, J., Van Gool, L., 2020. Mining cross-image semantics for weakly supervised semantic segmentation. In: Computer Vision – ECCV 2020. Springer International Publishing, Cham, pp. 347–365.

Taghanaki, S.A., Abhishek, K., Cohen, J.P., Cohen-Adad, J., Hamarneh, G., 2019. Deep semantic segmentation of natural and medical images: A review. Artif. Intell. Rev. 54, 137–178. http://dx.doi.org/10.48550/arxiv.1910.07655, URL: https://arxiv.org/abs/1910.07655v3.

Wang, X., You, S., Li, X., Ma, H., Jun 2018. Weakly-supervised semantic segmentation by iteratively mining common object features. In: Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition. IEEE, pp. 1354–1362. http://dx.doi.org/10.1109/CVPR.2018.00147, URL: https://ieeexplore.ieee.org/document/8578245.

Wang, Y., Zhang, J., Kan, M., Shan, S., Chen, X., 2020. Self-supervised equivariant attention mechanism for weakly supervised semantic segmentation. In: Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition. IEEE Computer Society, pp. 12272–12281. http://dx.doi.org/10.1109/CVPR42600.2020.01229, URL: https://arxiv.org/abs/2004.04581v1.

Wei, J., Wang, Q., Li, Z., Wang, S., Zhou, S.K., Cui, S., 2021. Shallow feature matters for weakly supervised object localization. CVPR, pp. 5989–5997.

Wu, Y., Lin, L., Wang, J., Wu, S., 2020. Application of semantic segmentation based on convolutional neural network in medical images. J. Biomed. Eng. 37, 533–540. http://dx.doi.org/10.7507/1001-5515.201906067, URL: https://pubmed.ncbi.nlm.nih.gov/32597097/.

Wurm, M., Stark, T., Zhu, X.X., Weigand, M., Taubenböck, H., 2019. Semantic segmentation of slums in satellite images using transfer learning on fully convolutional neural networks. ISPRS J. Photogramm. Remote Sens. 150, 59–69. http://dx.doi.org/10.1016/J.ISPRSJPRS.2019.02.006.

Xia, W., Domokos, C., Dong, J., Cheong, L.F., Yan, S., Dec 2013. Semantic segmentation without annotating segments. In: IEEE/CVF Conference on Computer Vision and Pattern Recognition.

Xiu, J., Yang, Z., Liu, C., 2020. Dual path attention net for remote sensing semantic image segmentation. ISPRS Int. J. Geo-Inf. 9 (10), 571. http://dx.doi.org/10.3390/ijgi9100571, URL: https://search.proquest.com/docview/2548583044.

Yao, Q., Gong, X., 2020. Saliency guided self-attention network for weakly and semi-supervised semantic segmentation. IEEE Access 8, 14413–14423. http://dx.doi.org/10.1109/ACCESS.2020.2966647.

Zhao, H., Qi, X., Shen, X., Shi, J., Jia, J., 2018. Icnet for real-time semantic segmentation on high-resolution images. In: Computer Vision – ECCV 2018, Springer International Publishing, Cham, pp. 418–434. http://dx.doi.org/10.1007/978-3-030-01219-9_25, URL: http://link.springer.com/10.1007/978-3-030-01219-9_25.

Zhou, S., Niu, L., Si, J., Qian, C., Zhang, L., 2021. Weak-shot semantic segmentation by transferring semantic affinity and boundary. URL: http://arxiv.org/abs/2110.01519.

Zhu, Y., Sapra, K., Reda, F.A., Shih, K.J., Newsam, S., Tao, A., Catanzaro, B., Jun 2019. Improving semantic segmentation via video propagation and label relaxation. In: IEEE/CVF Conference on Computer Vision and Pattern Recognition.