Class-Specific Affinity based Weakly Supervised Semantic Segmentation with Neutral Region Exploration

Keke Chen*, Patrick P. K. Chan*, Tianyi Xiang*, Natasha Kees*, Daniel S. Yeung†
*School of Computer Science and Engineering, South China University of Technology, Guangzhou, China
†Hong Kong
Email: kkechen@qq.com, patrickchan@ieee.org, xty435768@gmail.com, natasha.kees@yahoo.com, danyeung@ieee.org

2021 International Joint Conference on Neural Networks (IJCNN) | 978-1-6654-3900-8/21/$31.00 ©2021 IEEE | DOI: 10.1109/IJCNN52387.2021.9534197

***Abstract*—Image-level weakly supervised semantic segmentation (WSSS) reduces the cost of semantic segmentation significantly as only category labels are required. In the WSSS pipeline, the initially obtained seed areas are imprecise and should be refined. AffinityNet is a widely used seed area refinement method which learns pixel-level affinity between coordinates to adjust the object regions. However, the capabilities of AffinityNet are limited due to the complex simultaneous exploration of the affinities of all object classes, as the classes have different characteristics. This paper proposes a class-specific AffinityNet (CSANet) in which the affinities of each class are learnt separately by the class-specific affinity layers. Moreover, as the learning task is simple because the affinity of only one class is focused, the information contained by the uncertain neutral regions is extracted to provide additional information in the training of the class-specific affinity layers. The empirical results demonstrate the effectiveness of our method, which captures more precise boundaries and significantly improves segmentation results on the PASCAL VOC 2012 segmentation benchmark.**

I. Introduction

The rapid development of deep convolutional neural networks has driven extensive advances in computer vision [1], [2]. However, training a fully supervised model is expensive and time-consuming since a large amount of annotated data is needed. This problem gets worse in semantic segmentation as pixel-level annotation is required [3]. To ease the annotation burden, weakly supervised semantic segmentation (WSSS) methods [4] which use low-level annotations like bounding boxes [5], [6], scribbles [7], [8] and image-level class labels [9], [10] as supervision have been widely studied in recent years. In this work, the methods using image-level labels as supervision, which are the most economical to obtain, are investigated.

To bridge the gap between image-level annotation and pixel-level prediction, a multi-class classification network is first trained in the WSSS pipeline [11]. Seed areas are identified by class activation maps. One problem in traditional WSSS methods is that the seed areas generated by the classification network are usually biased towards the most discriminative object parts and fail to cover an object entirely. The seed areas are then revised by a refinement method [9], [12] according to the semantic consistency of object classes and local information like textures and colors in order to provide high quality Pseudo-Masks for training the segmentation network [13] that is used in inference.

Fig. 1: Example of the preprocessed seed areas and confident regions of an image in PASCAL VOC 2012 dataset. The confident regions are colored and the neutral region is in white.

AffinityNet [9], a widely used seed area refinement method, is trained by using pixel-level affinities containing a pair of coordinates selected from the confident areas identified by class activation maps (*i.e.*either foreground or background). The label of an affinity, which indicates whether the two coordinates belong to the same category, is used as supervision in AffinityNet training. The transition probability matrix, calculated according to the output feature map of AffinityNet, helps to propagate the activation scores of class activation maps to the nearby areas by the random walk method.

Although AffinityNet achieves promising results on expanding the seed areas [14], [15], learning the affinities of all categories and the background together limits its performance since the variance of coordinates in each class and characteristics between classes may vary considerably. The background class may suffer even worse accuracy due to high variations as it consists of any non-foreground regions. The class-agnostic transition probability matrix may not properly represent the affinities of all classes in AffinityNet. Moreover, neutral coordinates provide useful information to separate the foreground and background since they are usually located on the boundary. However, the training of AffinityNet only relies on confident coordinates and the rest of the coordinates are ignored, which may cause information loss. Fig. 1 visualizes the preprocessed seed areas and confident regions of an image in PASCAL VOC 2012 dataset [16]. Most boundaries are located in the unconfident regions displayed in white.

By observing that different object classes have their own

distinct properties, the class-specific AffinityNet (CSANet) is devised in this paper. CSANet contains class-specific affinity layers in order to learn class-specific affinities for each class separately. The network architecture of CSANet and the Activation Matrix refinement step are illustrated in Fig. 2. Learning the affinities of each class separately not only reduces the complexity of the learning task but also makes the representation on the intra-class and inter-class characteristics more precisely. As a result, neutral coordinates ignored in the training of AffinityNet can be utilized in our proposed model. As the affinity of a class considered in our model is clearer and simpler, the uncertain information can be explored. To be specific, the affinity of a background coordinate and a neutral coordinate with higher than the adaptive threshold will be enhanced in learning.

Extensive experiments have been conducted on the PASCAL VOC 2012 dataset to examine the effectiveness of our method. The Pseudo-Masks and final segmentation masks generated by our method combined with the seed area localization methods including CAM [11] and SEAM [14] are evaluated. The state-of-the-art methods are also considered for comparison. The empirical evaluation confirms that CSANet successfully improves the performance of its baseline method, *i.e.*AffinityNet. The contribution of our work can be summarized as follows:

- We extend AffinityNet with a novel class-specific training scheme, which effectively captures the semantic information of each object class by a class-specific affinity layer individually.
- Uncertain regions indicated by class activation maps can be explored due to the simplicity of the class-specific affinity tasks. A more precise object boundary is revealed through the exploration in the neutral region. Over expansion on an object region is suppressed.
- The quality of the generated Pseudo-Masks for most categories and the final segmentation masks on PASCAL VOC 2012 are significantly improved by CSANet compared to the baseline, AffinityNet.

II. Related Work on Weakly Supervised Semantic Segmentation

The weakly supervised semantic segmentation (WSSS) methods generally follow the three-step pipeline [10], including seed area identification, seed area refinement, and segmentation model training. The first two steps, which play an important role, are introduced.

A. Seed Area Identification

Most of the advanced seed area identification methods are based on Class Activation Map [11] (CAM) which compute the contribution of each pixel on the classification task. As the most discriminative object parts usually dominate the decisions, the response of CAM on an object usually tends to be sparse and incomplete.

Several recent CAM-based works have been devised to reveal more complete object regions in order to generate better initial seed areas. Some methods locate more object areas by discarding partial information during training, for example, hidden units selected stochastically from the feature map are dropped out in FickleNet [17], and the classifier is retrained after discarding the most discriminative regions in each epoch in [18]. Another type of seed area identification methods consider extra constraints, including a sub-category classification task by SCE [15], and the spatial transformation equivariance characteristic by SSENet [19] and SEAM [14], in classifier training. In our study, the standard CAM and SEAM are considered as the seed area identification method.

B. Seed Area Refinement

The identified seed areas cannot provide sufficient information for segmentation network training since they are usually sparse and cannot cover objects completely. Therefore, the seed areas should be refined to generate pixel-level labels for training.

The classic Seed, Expand and Constraint method [4] uses the Global Weighted Rank Pooling as the expansion algorithm which takes the advantages of both Global Average and Global Maximum Pooling. The boundary is constrained with respect to the low-level image information by applying dense Conditional Random Field (dCRF) [20]. Other methods expand the seed areas according to a criterion. MDC [21] adopts different dilation rates in multiple convolutional layer branches, DSRG [12] integrates the seed region growth algorithm with a deep segmentation network, and OAA [22] relies on the accumulated pixel scores along the training process of a seed identification method. Another type of method applies an end-to-end framework for the training of seed area refinement and segmentation models in order to reduce the time complexity of the WSSS framework. The reliable regions of seed areas are directly served as ground-truth labels for the parallel segmentation branch in RDM [25]. ICD [26] is another end-to-end method that focuses on the boundaries for each class.

AffinityNet [9] is the first method that considers pixel-level affinities in the WSSS framework. AffinityNet learns pixel-level affinities from CAMs in order to propagate seed areas to the nearby coordinates with high affinity. IRNet [27], which is an extension of AffinityNet, detects object boundaries to estimate individual instance regions roughly based on pairwise semantic affinities and displacement vector fields. A cross-image affinity module is proposed by CIAN [28] to capture the cross-image relationship from different images containing objects in the same category.

III. Method

A. Overview

This section discusses the proposed class-specific affinity network (CSANet) in detail. In contrast with AffinityNet, CSANet learns the affinities of each category separately since the variance of pixels in each class and characteristics between classes may be different. The network architecture of CSANet including the backbone network and class-specific affinity

layers is introduced in Section III-B. The diagram of our network is presented in Fig. 2.

Before extracting affinities, the confident regions are defined according to the values of class activation maps in Section III-C. The affinities are then extracted to supervise the learning of class-specific affinity layers (Section III-D). In addition to the confident regions, affinities of coordinates in the neutral regions, which provide useful information to separate the foreground and background, are also considered in CSANet. The objective function of class-specific affinity layers is discussed in Section III-E. Finally, the Pseudo-Masks are obtained according to the Random Walk method (Section III-F).

B. Network Architecture

1) Backbone Network: Given an input image I, the feature maps from multiple levels of the backbone network are concatenated by applying 1×1 convolution filters to provide both the high-level semantics and low-level details of the input image. We define the concatenated feature maps as $\phi(I, \theta) \in \mathbb{R}^{k \times h \times w}$, where ϕ represents the CNN-based backbone network and θ denotes its parameters. k, h and w represent the channels, height and width of the concatenated feature map respectively. The feature maps extracted by the backbone network are shared by all the class-specific affinity layers introduced below.

2) Class-Specific Affinity Layers: Different from AffinityNet, which learns the affinities of all classes together, we argue that learning a distinct affinity representation from each class could capture more detailed class-specific affinity information. As shown in Fig. 2, our CSANet contains novel class-specific affinity layers represented by the filters $\{F_c\}_{c \in \mathcal{C}_{fg}}$ aiming to learn the pixel-level affinities for each foreground category c, where $\mathcal{C}_{fg}$ is the set of foreground classes. The affinity of c is produced according to the shared backbone network and also its own class-specific affinity filters, which is defined as:

$$f_c = F_c * \phi(I, \theta), \tag{1}$$

where $*$ denotes the convolution operations of F_c.

The affinity matrix A_c is a $M \times M$ matrix representing the affinity of class c for all coordinate pairs in I, where $M = h \times w$. An affinity of the i-th and j-th coordinates is quantified by the L_1 distance as follows:

$$A_c^{ij} = exp\{-\|f_c(x_i, y_i) - f_c(x_j, y_j)\|_1\} \tag{2}$$

where A_c^{ij} denotes the value located at the i-th row and j-th column of A_c, $f_c(x_i, y_i)$ and $f_c(x_j, y_j)$ represents the i-th and j-th feature vectors of f_c.

C. Confident Region Identification

Seed areas, represented as the activation score matrix M_c for each class c, are identified by a seed area identification method. $M_c(x, y)$ denotes an activation score for each coordinate (x, y) representing the probability of (x, y) belonging to a foreground category c.

The same as AffinityNet, the activation score matrix M_{bg} for the background region is defined as:

$$M^{bg}(x, y) = \left(1 - \max_{c \in \mathcal{C}_{fg}} M_c^{fg}(x, y)\right)^{\alpha} \tag{3}$$

where $\alpha \in \mathcal{Z}^+$ is the hyper-parameter to control the values of $M^{bg}(x, y)$, *i.e.*$M^{bg}(x, y)$ decreases with the increase of α.

Concatenated M^c and M^{bg} are then refined by dCRF [20] to better refine the boundaries. The confident foreground and background regions are identified using different α values. The confident region for each foreground object class $\mathcal{R}_c$ contains a pixel with the activation score of c larger than other classes including the background, *i.e.*$\mathcal{R}_c = \{(x, y) | M_c(x, y) > M_{c'}(x, y), c' \neq c, c' \in C_{fg} \cup \{bg\}\}$. On the other hand, a confident background region is defined as $\mathcal{R}_{bg} = \{(x, y) | M_{bg}(x, y) > M_c(x, y), c \in C_{fg}\}$. It should be noted that α used in $\mathcal{R}_c$ is smaller than $\mathcal{R}_{bg}$. A smaller α causes larger activation scores for the background region which increases the difficulty of being a coordinate in $\mathcal{R}_c$. On the other hand, the difficulty of being a coordinate in $\mathcal{R}_{bg}$ also increases with the increase of α, which reduces the values of the scores of the background region. This mechanism guarantees the confidence of being a foreground class and background coordinate. The remaining coordinates are considered as neutral regions, $\mathcal{R}_n$.

D. Affinity Extraction

By considering reliability and time complexity, only the affinity of an adjacent pixel pair within distance γ is considered in the training of CSANet. The pairwise affinity set P is defined as follows:

$$P = \{(i, j) | d(i, j) < \gamma, i \neq j\}, \tag{4}$$

where $d(i, j)$ is the Euclidean distance of the i-th and j-th coordinates.

P can be further divided into two subsets (*i.e.*the positive and negative affinity sets, P^+ and P^-). P^+ contains the coordinate pairs belonging to the same class, and their affinity should be strengthened in training. On the other hand, the affinities of coordinate pairs in P^- should be weakened since the coordinates are not in the same class.

Four subsets of P for each c are used in the training: three subsets are from the confident region (*i.e.*P_c^+, $P_{c'}^+$, and P_c^-) and one subset from the neutral region (*i.e.*P_c^{n+}). These four subsets are described in the next sub-sections.

1) Confident Region: The class-specific affinity layer for the class c should return a strong affinity if both coordinates either belong to or do not belong to c. If one coordinate belongs to c and another one does not, their affinity should be weak. Therefore, three kinds of affinities chosen from the confident regions are considered for class c.

Two of them are the positive affinity, including P_c^+ containing the pair of both coordinates in R_c (Eq. (5)), and $P_{c'}^+$ containing the pair of both coordinates not in R_c (Eq. (6)).

$$P_c^+ = \{(i, j) | (x_i, y_i), (x_j, y_j) \in \mathcal{R}_c\}, \tag{5}$$

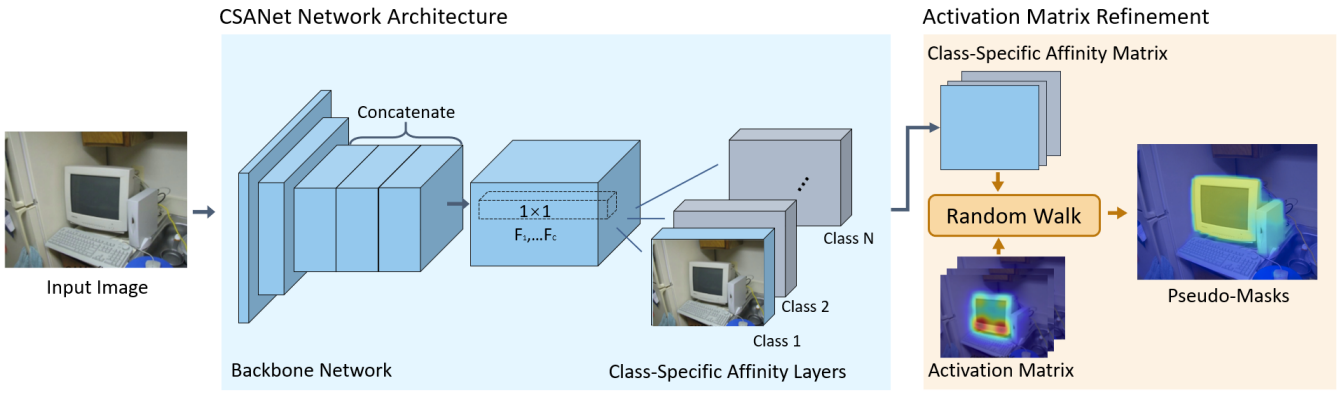

Fig. 2: Network architecture and Activation Matrix refinement of our proposed CSANet. Affinities of each class are learnt separately by each class-specific affinity layer. The Pseudo-Masks are generated according to the class-specific affinity matrix and class activation maps by the Random Work method.

$$P^+_{c'} = \{(i,j)|(x_i,y_i) \in \mathcal{R}_{c'}, (x_j,y_j) \in \mathcal{R}_{c''}, \\ c \neq c', c \neq c'', \quad c', c'' \in \mathcal{C}_{fg} \cup \{bg\}\}. \tag{6}$$

The size of P^- is usually smaller than P^+ since negative affinities are sampled around the boundaries while positive affinities are sampled within an object or the background. To alleviate this imbalance problem, the loss for each subset is calculated separately in the objective function.

$$P^-_c = \{(i,j)|(x_i,y_i) \in \mathcal{R}_c, (x_j,y_j) \in \mathcal{R}_{c'}, \\ c \neq c', c' \in \mathcal{C}_{fg} \cup \{bg\}\}, \tag{7}$$

2) Neutral Region Exploration: AffinityNet [9] discards neutral coordinates which may provide useful information for learning since they are usually located at the boundary. Fig. 1 illustrates an example in which most boundaries are located in the unconfident neutral regions. Since the class-specific affinities learnt in CSANet are more precise and simple, uncertain information provided by class activation maps can be carefully exploited in order to provide additional information for learning.

An pairwise affinity set P^{n+}_c in which the coordinate pairs containing neutral and background coordinates with the affinity lager than $t_c(i)$, defined as Eq. (9) is served as supplementary to the class-specific affinities learning in our model.

$$P^{n+}_c = \{(i,j)|i \in \mathcal{R}_n, j \in R_{bg}, A^{ij}_c > t_c(i)\}, \tag{8}$$

As affinity may vary in local regions, the adaptive threshold $t_c(i)$ calculated by the average affinity between a neutral coordinate and its nearby coordinates is adopted. A neutral coordinate is likely to be the background if its affinity with the confident background coordinate is larger than $t_c(i)$. It ensures the reliability of P^{n+}_c.

$$t_c(i) = \mathbb{E}_{k \in \{j|d(i,j)<\gamma, i \neq j\}}[A^{ij}_c], \tag{9}$$

A coordinate pair with an affinity larger than the adaptive threshold are served as a positive affinity. The background region can be explored more completely to determine a precise object boundary.

E. Model Learning and Optimization Loss

The following objective function is minimized for each image during training:

$$L = \sum_{c \in \mathcal{C}_{fg}} [\delta_c(L^+_c + L^+_{c'} + L^-_c + L^{n+}_c) + (1-\delta_c)(L^+_{c'})] \tag{10}$$

where $\delta_c = \{0,1\}$ indicates the existence of class c in an input image. If the image contains c, $\delta_c = 1$, otherwise, $\delta_c = 0$. The loss components are defined as the cross-entropy loss of the positive and negative pairwise affinities:

$$\begin{aligned} L^+_c &= -\frac{1}{|P^+_c|} \sum_{(i,j) \in P^+_c} log A^{ij}_c, \\ L^+_{c'} &= -\frac{1}{|P^+_{c'}|} \sum_{(i,j) \in P^+_{c'}} log A^{ij}_c, \\ L^{n+}_c &= -\frac{1}{|P^{n+}_c|} \sum_{(i,j) \in P^{n+}_c} log A^{ij}_c, \\ L^-_c &= -\frac{1}{|P^-_c|} \sum_{(i,j) \in P^-_c} log(1 - A^{ij}_c) \end{aligned} \tag{11}$$

where A^{ij}_c denotes the affinity matrix calculated from the output feature map defined as Eq. (2).

The positive affinity sets, *i.e.*L^+_c,$L^+_{c'}$ and L^{n+}_c, in the first component of Eq. (10) ensures the class-specific affinity layer of c outputs a strong affinity for both coordinates belonging to and not belonging to c, while L^-_c encourages a small affinity for one coordinate belonging to c and the other one does not.

As an image contains few object classes, most of the network cannot be trained if only the layers of presented classes are considered in each update. As a result, the learning is ineffective. Therefore, $L^+_{c'}$ is minimized for the class c which does not exist in an image (*i.e.*$\delta_c = 0$) to enforce the affinities of all coordinates in the confident regions output by the class-specific affinity layer of c is the similar.

F. Activation Matrix Refinement by Random Walk

By following [9], a transition probability matrix is calculated according to the class-specific affinity matrix for each c and refines the activation matrix by the random walk method [29]. The transition probability matrix is defined as:

$$T_c = \left(D^{-1} A_c^{\circ\beta}\right)^t, where\ D_{ii} = \sum_j (A_c^{\circ\beta})^{ij} \tag{12}$$

where $A_c^{\circ\beta}$ is the Hadamard power of A_c and β denotes its hyper-parameter. The insignificant affinities, the ones with a small value, will tend to 0 after taking the Hadamard power. The diagonal matrix D normalizes $A_c^{\circ\beta}$ by row. The normalized matrix is multiplied by t times to represent the probability that the length of a random walk is t.

For class c, the product of the transition probability matrix T_c and activation matrix M_c are used to diffuse the activation scores. The revised activation matrix M_c^* is defined as follows:

$$vec(M_c^*) = T_c \cdot vec(M_c), \tag{13}$$

where vec( ) represents the vectorization of a matrix.

IV. Experiment

A. Implementation Details

1) Dataset and Evaluation: The proposed approach is evaluated on the PASCAL VOC 2012 segmentation benchmark dataset [16] which contains 1,464 images, 1,449 images and 1,456 images in the training, validation and test sets respectively. By following common practices, an additional 10,582 augmented training images [30] are applied to train the network. Only image-level category labels are considered in training. The segmentation performance is evaluated by the standard mean Intersection over Union (mIoU). Pseudo-Masks are evaluated in the training set since it is used as the supervision for training the segmentation network. The validation set and testing sets are adopted to evaluate the final segmentation masks.

2) Seed Areas Identification Method: The seed area identification method includes the standard Class Activation Maps (*CAM*) [11] and the advanced Self-supervised Equivariant Attention Mechanism (*SEAM*) [14] are adopted for seed area identification. A classification network is trained and the activation matrix is obtained by quantifying the contribution of each pixels to the classification decision in CAM [11]. SEAM [14] focuses that the network outputs the same CAMs on the same images with different spatial transformation by minimizing the equivariant regularization losses. Moreover, the pixel correlation module (PCM) is proposed to refine CAM according to context and appearance information. It should be noted that performance boost is expected with more advanced seed area identification methods.

3) Baseline Model: AffinityNet [9] is denoted as *baseline* in the experimental analysis.

TABLE I: Quality of seed areas (Seed) and Pseudo-Masks (Mask) between *baseline* and *CSANet* with and without NRE in terms of mIoU (%) on the training set of PASCAL VOC 2012 dataset.

	Model		Single-Class		Multi-Class	
			Seed	**Mask**	**Seed**	**Mask**
w/o NRE	*CAM*	*baseline*	47.72	61.01	44.92	56.98
		CSANet		61.56$_{+0.55}$		**57.76**$_{+0.78}$
	SEAM	*baseline*	55.41	63.40	50.42	55.71
		CSANet		**63.92**$_{+0.52}$		56.85$_{+1.14}$
w/ NRE	*CAM*	*baseline*	47.72	61.33	44.92	56.93
		CSANet		61.90$_{+0.57}$		**58.53**$_{+1.55}$
	SEAM	*baseline*	55.41	63.85	50.42	56.01
		CSANet		**64.72**$_{+0.87}$		57.40$_{+1.39}$

4) Parameters Settings: ResNet38 [31] pre-trained on ImageNet [32] is used as the backbone network. The dilated convolution is applied to the last three Resnet blocks and all fully connected layers in the original network are removed. A block in Resnet is a set of residual units with the same output size. The dilated rate is set as 2 and 4 for the third last and the last two layers respectively. These settings are the same as [9].

During training, horizontal flipping, random cropping and color jittering [33] are used to augment the training data. The default parameters of dCRF [20] are used according to its source codes. α in Eq. (3) is set to 4 and 32 for *CAM* to identify the confident foreground and background regions respectively, while 4 and 24 are used for *SEAM* by following its original settings. γ in Eq. (4) and β in Eq. (13) are 5 and 8. t in the random walk process is set to 256.

5) Reproducibility: The experiments are carried out on 2 NVIDIA GTX 1080 Ti GPUs and implemented with PyTorch. Codes are available at https://github.com/kkechen/CSANet.

B. Ablation Study

The effectiveness of the main components of the proposed methods, including *CSANet* and Neutral Region Exploration (NRE), is verified in the circumstances with different complexity levels in this ablation study. The training set of the PASCAL VOC 2012 segmentation benchmark dataset is separated into single and multi-class images which contain one and more than one object classes respectively.

The quality of the Pseudo-Masks generated by *baseline* and *CSANet* without and with NRE on the overall training set and its subset of multi-class images are shown in Table I. The quality of seed areas of both *CAM* and *SEAM* are improved significantly after being refined by *baseline* and our *CSANet* in any case, and our method always achieves better quality than *baseline*.

Dealing with multi-class images is a more difficult task compared to the single one, *i.e.*the mIoU values on multi-class images are lower. *CAM* performs better on the multi-class images while *SEAM* provides better results on the overall training set. The improvement of *CSANet* is more significant on the multi-class images compared to the overall training set

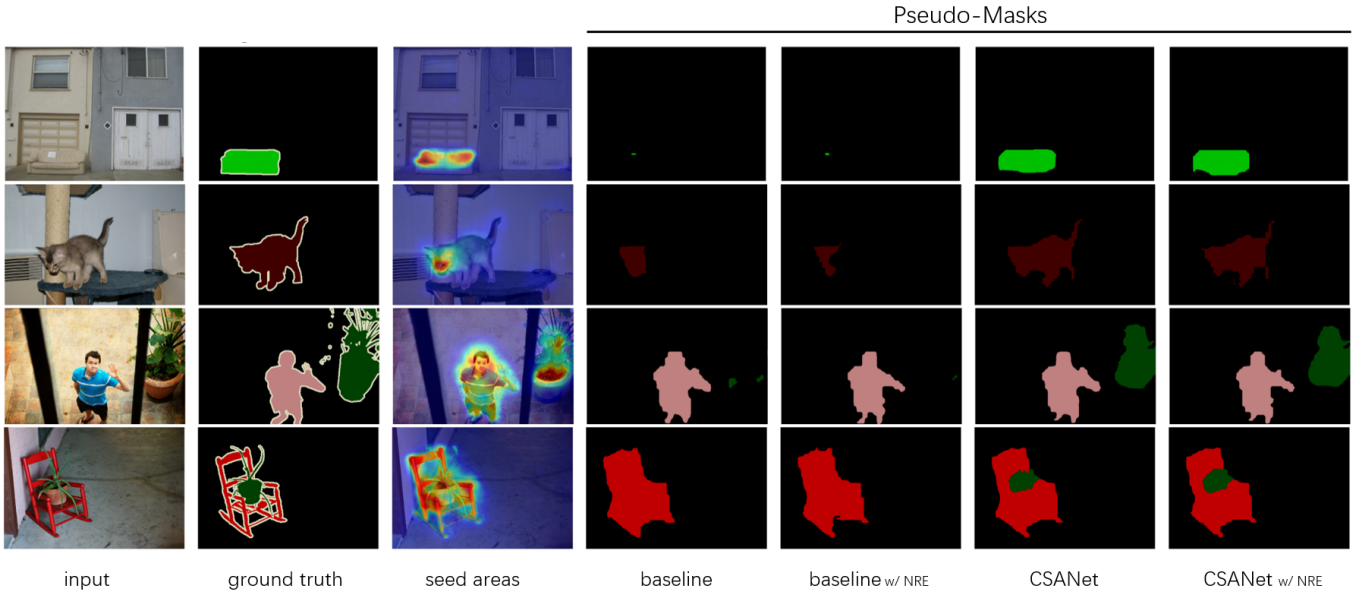

Fig. 3: Visualization samples of Pseudo-Masks generated by baseline AffinityNet and our CSANet with and without the NRE with SEAM as the seed area identification method on PASCAL VOC 2012 training set.

TABLE II: Category performance comparisons of Pseudo-Masks with SEAM as the seed area identification methods on PASCAL VOC 2012 training set.

Score	bg	plane	bike	bird	boat	bottle	bus	car	cat	chair	cow	table	dog	horse	motor	person	plant	sheep	sofa	train	tv	mIoU
baseline	86.1	60.6	37.7	78.5	35.2	56.2	71.7	67.8	79.5	29.3	78.1	56.0	79.5	79.5	75.0	72.2	40.8	83.6	57.7	59.3	47.1	63.4
CSANet	86.0	58.2	38.8	74.3	33.0	60.2	**73.5**	69.4	78.6	30.1	80.1	**57.9**	78.8	80.7	76.2	71.9	**43.3**	83.5	**59.0**	59.9	**48.9**	63.9
$baseline_{w/\ NRE}$	**86.6**	61.2	37.5	**79.4**	**38.2**	56.8	71.5	67.9	79.7	29.5	79.4	55.3	**79.9**	81.3	75.2	72.9	40.7	**84.1**	57.5	59.9	46.4	63.9
$CSANet_{w/\ NRE}$	**86.6**	**61.5**	**40.5**	77.9	35.1	**60.6**	72.7	**69.9**	**79.8**	**30.4**	**80.5**	57.6	79.1	**81.8**	**77.6**	**73.2**	43.1	84.0	58.1	**60.6**	48.7	**64.7**

when using both *CAM* and *SEAM*. For example, the mIoU of *CSANet* is 0.52% and 1.14% higher than *baseline* on the overall training set and multi-class images when using *SEAM*. These results demonstrate the affinities of classes is too complex to be represented by one network. The class-specific affinity layers are able to learn the relation of each class individually.

The averaged mIoU of *CSANet* is 0.80% improved by using NRE while *baseline* is 0.45% when using *SEAM*. The uncertain information, *i.e.*the neutral regions, contributes more to *CSANet* since it considers the affinities of each class separately, which is less complicated than *baseline*. These confirm that the neutral regions are useful in learning affinities and our method utilizes the information provided by not only confident but also neutral regions.

As the quality of Pseudo-Masks with *SEAM* as the seed area identification method is greatly better than *CAM*, *SEAM* is adopted for the rest of the experimental discussion on *CSANet* and *basline*.

C. Visualization and Category-Level Comparison

The visualization samples of seed areas identified by *SEAM* and its corresponding Pseudo-Masks generated by *baseline* and *CSANet* are first investigated and shown in Fig. 3. It should be noted that the illustrated Pseudo-Masks are the output of the random walk method and are not post-processed by dCRF. The visualization results demonstrate that the class-specific affinities of our method diffuse the activation scores properly, *i.e.*the object regions are identified more completely and precisely compared to *baseline*.

The mIoU values of Pseudo-Masks in each category with *SEAM* are shown in Table II. *CSANet* improves the quality of Pseudo-Masks compared to *baseline* in 13 out of 21 categories (including the background). The results also confirm that learning with the affinity of the neutral regions works well in most categories. The mIoU values of 16 categories of $CSANet_{w/\ NRE}$ are higher than *CSANet*, while $baseline_{w/\ NRE}$ has 15 categories with higher mIoU than *baseline*. After combining CSANet with NER, the overall mIoU increases by 1.3% and the category performance is improved in 18 out of 21 categories compared to the baseline, which demonstrates that $CSANet_{w/\ NRE}$ learns the boundaries of all objects evenly.

D. State-of-the-Art Method Comparison

To further compare to the state-of-the-art methods, *SEAM* and DeepLab-v2 [13] with ResNet101 as backbone network are adopted as the seed area identification method and segmentation model respectively for *CSANet*. The state-of-the-art WSSS methods adopting ResNet-based Deeplab as the segmentation network with image-level annotation listed in

TABLE III: Performance of state-of-the-art methods of weakly supervised semantic segmentation using ResNet-based Deeplab as the segmentation network in terms of mIoU (%). Val and test denote the validation and testing sets of the PASCAL VOC 2012 dataset. The supervision includes image-level labels (I) and the additional off-the-shelf saliency map (S).

Methods	Backbone	Supervision	val	test
MCOF [24]	ResNet-101	I+S	60.3	61.2
DSRG [12]	ResNet-101	I+S	61.4	63.2
AffinityNet [9]	ResNet-38	I	61.7	63.7
FickleNet [17]	ResNet-101	I+S	64.9	65.3
SSENet [19]	ResNet-38	I	63.3	64.9
ICD [26]	ResNet-101	I	64.1	64.3
OOA [22]	ResNet-101	I+S	63.9	65.6
CIAN [28]	ResNet-101	I	64.1	64.7
IRNet [27]	ResNet-50	I	63.5	64.8
SEAM [14]	ResNet-38	I	64.5	65.7
SCE [15]	ResNet-101	I	66.1	65.9
SEAM*	ResNet-101	I	64.4	64.9
CSANet$_{w/\ NRE}$	ResNet-101	I	65.4	65.8

$*$ Re-implemented result with Deeplab-V2 (ResNet-101) as the segmentation model.

Table III are considered in this comparison. As DeepLab-v2 with ResNet101 is used in most of state-of-the-art methods as the backbone network, SEAM is re-implemented by using ResNet101, denoted by *SEAM**, for a fair comparison.

CSANet with Neutral Region Exploration significantly promotes the final segmentation masks to 65.4% and 65.8% on the validation set and testing set on PASCAL VOC 2012 dataset respectively which are the highest values among all methods except *SCE*. As *SCE* [15] is an advanced seed area identification method using *AffinityNet* as the refinement method, better performance of our method collaborating with *SCE* is expected since *CSANet* performs much better than *AffinityNet*. Our method is 1.0% and 0.9% higher than *SEAM* with Resnet101. Although Resnet38 has a higher generalization ability than Resnet101, the performance of *CSANet* is still better than *SEAM* with Resnet38. Performance boost of *CSANet* is expected when a more advanced seed area identification method or *CONTA* [10], which is a general framework to improve the performance of any WSSS method, is applied.

V. Conclusion

This study proposes a seed area refinement method for the WSSS framework. The proposed CSANet contains class-specific affinity layers to learn affinities of each class separately due to different characteristics of categories and the background. Focusing on the affinity of one class makes the learning simple, which allows the exploration on uncertain coordinates, providing useful information of boundaries. The affinities between the neutral and confident background coordinates are also considered in the training of class-specific affinity layers. The experimental results on PASCAL VOC 2012 dataset confirm that the model learning on neutral regions suppresses the over-expansion of a class region. Moreover, CSANet effectively improves the quality of the Pseudo and final segmentation masks from its baseline, *i.e.*AffinityNet, and achieves satisfying performance compared to the state-of-the-art methods.

Acknowledgments

This paper is supported by the Natural Science Foundation of Guangdong Province, China (No. 2018A030313203).

References

[1] Krizhevsky, Alex, Ilya Sutskever, and Geoffrey E. Hinton. "ImageNet classification with deep convolutional neural networks," in *Communications of The ACM*, 60(6): 84-90, 2017.

[2] J. Long, E. Shelhamer, T. Darrell, "Fully convolutional networks for semantic segmentation," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 3431-3440, 2015.

[3] T. Lin, M. Maire, S.J. Belongie, L.D. Bourdev, R.B. Girshick, J. Hays, P. Perona, D. Ramanan, P. Dollár, C.L. Zitnick, "Microsoft COCO: common objects in context," in *European Conference on Computer Vision (ECCV)*, pp. 740-755, 2014.

[4] A. Kolesnikov, C.H. Lampert, "Seed, expand and constrain: three principles for weakly-supervised image segmentation," in *European Conference on Computer Vision (ECCV)*, pp. 695-711, 2016.

[5] J. Dai, K. He, J. Sun, "BoxSup: Exploiting bounding boxes to supervise convolutional networks for semantic segmentation," in *IEEE International Conference on Computer Vision (ICCV)*, pp. 1635-1643, 2015.

[6] A. Khoreva, R. Benenson, J. Hosang, M. Hein, B. Schiele, "Simple does it: weakly supervised instance and semantic segmentation," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 876-885, 2017.

[7] D. Lin, J. Dai, J. Jia, K. He, J. Sun, "ScribbleSup: Scribble-supervised convolutional networks for semantic segmentation," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 3159-3167, 2016.

[8] P. Vernaza, M. Chandraker, "Learning random-walk label propagation for weakly-supervised semantic segmentation," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 7158-7166, 2017.

[9] J. Ahn, S. Kwak. "Learning pixel-level semantic affinity with image-level supervision for weakly supervised semantic segmentation," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 4981-4990, 2018.

[10] D. Zhang, H. Zhang, J. Tang, X. Hua, Q. Sun, "Causal intervention for weakly-supervised semantic segmentation," in *Conference on Neural Information Processing Systems (NIPS)*, 2020.

[11] B. Zhou, A. Khosla, A. Lapedriza, A. Oliva, A. Torralba, "Learning deep features for discriminative localization," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 2921-2929, 2016.

[12] Z. Huang, X. Wang, J. Wang, W. Liu, J. Wang, "Weakly-supervised semantic segmentation network with deep seeded region growing," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 7014-7023, 2018.

[13] L. Chen, G. Papandreou, I. Kokkinos, K. Murphy, A.L. Yuille, "Semantic image segmentation with deep convolutional nets and fully connected CRFs," in *arXiv preprint arXiv:1412.7062*, 2014.

[14] Y. Wang, J. Zhang, M. Kan, S. Shan, X. Chen, "Self-supervised equivariant attention mechanism for weakly supervised semantic segmentation," in *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 12275-12284, 2020.

[15] Y. Chang, Q. Wang, W. Hung, R. Piramuthu, Y. Tsai, M. Yang, "Weakly-supervised semantic segmentation via sub-category exploration," *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 876-885, 2020.

[16] M. Everingham, L.V. Gool, Christopher K.I. Williams, et al. "The PASCAL visual object classes (VOC) challenge," in *International Journal of Computer Vision*, 88(2), pp. 303-338, 2010.

[17] J. Lee, E. Kim, S. Lee, J. Lee, S. Yoon, "FickleNet: Weakly and semi-supervised semantic image segmentation using stochastic inference," in *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 5267-5276, 2019.

[18] Y. Wei, J. Feng, X. Liang, et al., "Object region mining with adversarial erasing: a simple classification to semantic segmentation approach," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 1568-1576, 2017.

[19] Y. Wang, J. Zhang, M. Kan, S. Shan, X. Chen, "Self-supervised scale equivariant network for weakly supervised semantic segmentation," in *arXiv preprint. arXiv:1909.03714*, 2019.
[20] P. Krhenbühl, V. Koltun, "Efficient inference in fully connected crfs with gaussian edge potentials," in *Advances in Neural Information Processing Systems*, pp. 109-117, 2011.
[21] Yunchao Wei, Huaxin Xiao, Honghui Shi, Zequn Jie, Jiashi Feng, and Thomas S. Huang, "Revisiting dilated convolution: A simple approach for weakly- and semi-supervised semantic segmentation," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 7268C 7277, 2018.
[22] Peng-Tao Jiang, Qibin Hou, Yang Cao, Ming-Ming Cheng, Yunchao Wei, and Hong-Kai Xiong, "Integral object mining via online attention accumulation," in *Proceedings ofthe IEEE International Conference on Computer Vision*, pp. 2070C2079, 2019.
[23] J. Fu, J. Liu, H. Tian, et al., "Dual attention network for scene segmentation," in *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 3146-3154, 2019.
[24] X. Wang, S. You, X. Li, H. Ma, "Weakly-supervised semantic segmentation by iteratively mining common object features," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 1354-1362, 2018.
[25] B. Zhang, J. Xiao, Y. Wei, M. Sun, K. Huang, "Reliability does matter: an end-to-end weakly supervised semantic segmentation approach," in *AAAI Conference on Artificial Intelligence*, 34(7), pp. 12765-12772, 2020.
[26] J. Fan, Z. Zhang, C. Song, T. Tan, "Learning integral objects with intra-class discriminator for weakly-supervised semantic segmentation," in *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 4283-4292, 2020.
[27] J. Ahn, S. Cho, S. Kwak, "Weakly supervised learning of instance segmentation with inter-pixel relations," in *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 2209-2218, 2019.
[28] J. Fan, Z. Zhang, T. Tan, C. Song, J. Xiao, "CIAN: Cross-image affinity net for weakly supervised semantic segmentation," in *AAAI Conference on Artificial Intelligence*, pp. 10762-10769, 2020.
[29] L. Lovász, "Random walks on graphs: a survey," in *Combinatorics, Paul erdos is eighty*, 2(1), pp. 1-46, 1993.
[30] M. Everingham, G.L. Van, C.K. Williams, J. Winn, A. Zisserman, "The pascal visual object classes (VOC) challenge," in *International Journal of Computer Vision*, 88(2), pp. 303-338, 2010.
[31] Z. Wu, C. Shen, A. Van Den Hengel, "Wider or deeper: revisiting the resnet model for visual recognition," in *Pattern Recognition, 90*, pp. 119-133, 2019.
[32] Russakovsky, O., Deng, J., Su, H., Krause, J., Satheesh, S., Ma, S., et al, "Imagenet large scale visual recognition challenge," in *International Journal of Computer Vision*, 115(3), pp. 211-252, 2015.
[33] K. Alex, S. Ilya, G. Hinton, "ImageNet classification with deep convolutional neural networks," in *International Conference on Neural Information Processing Systems*. pp. 1097-1105, 2012.
[34] S.J. Oh, R. Benenson, A. Khoreva, et al., "Exploiting saliency for object segmentation from image level labels," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 5038-5047, 2017.