Learning Effectively from Noisy Supervision for Weakly Supervised Semantic Segmentation

Wenbin Xie
xiewb18@mails.tsinghua.edu.cn
QiaoQiao Wei
wqq18@mails.tsinghua.edu.cn
Zheng Li
zhengli-17@mails.tsinghua.edu.cn
Hui Zhang
huizhang@tsinghua.edu.cn

School of Software,
BNRist,
Tsinghua University
Beijing, China

Abstract

Semantic segmentation based on deep learning has undergone tremendous progress in recent years. However, it continues to depend heavily on massive densely annotated data. In this paper, we propose a novel framework for weakly supervised semantic segmentation (WSSS) using bounding boxes to alleviate the need for pixel-wise annotations. We argue that the most important problem of WSSS should be learning effectively from noisy supervision. Therefore, we present a constrained foreground segmentation network (CFS) to generate high-quality dense annotations from noisy proposals. The network converts the segmentation task from multi-class classification to two-class classification and removes most of irrelevant regions, making the task easier to optimize. Besides, we introduce a loss-guided self-attention (LGSA) module to encourage self-correction among intra-class pixels. Instead of allowing global information exchanges in existing Non-local modules, our module imposes loss constraints on the information exchanges between different categories and learns a more reasonable affinity matrix that can be used for further random walk. Experiments indicate that our LGSA module has better performance and interpretability than Non-local modules even with noisy supervision. We obtain state-of-the-art results on the Pascal VOC 2012 *validation* set by combining the two novel components.

1 Introduction

Deep learning has achieved great success on computer vision, but the success depends heavily on a massive amount of labeled data. It's laborious to collect data annotations, especially for semantic segmentation where pixel-level annotations are usually necessary. Therefore, weakly supervised semantic segmentation (WSSS), which uses cheaper annotations instead of dense annotations for semantic segmentation, has drawn great attention recently. The cost of data labeling can be reduced considerably by using cheaper annotations, *e.g.*, image-level

labels [13, 17], bounding boxes [8, 23], scribbles [25] and clicks [3]. In this paper, we focus on WSSS based on bounding box annotations.

Most WSSS methods use off-the-shelf approaches, such as CAM [30], GrabCut [22], DenseCRF [16] and MCG [20], to generate pixel-level proposals. The dense proposals tend to supervise semantic segmentation networks directly or after refinement. However, the generated proposals are inevitably noisy, because the off-the-shelf approaches either have limited localization ability or use only low-level features, such as color and spatial information. The learning of semantic segmentation networks is hard because of the noisy supervision. Therefore, we argue that the most important problem of WSSS should be learning effectively from noisy supervision. The problem can be further divided into two parts: getting more accurate proposals from noisy ones and making semantic segmentation networks robust to noisy supervision. Hence, we present a constrained foreground segmentation (CFS) network to generate high-quality proposals, and a loss-guided self-attention (LGSA) module to encourage semantic segmentation networks to self-correct, making the networks robust to noisy supervision.

The basic idea of CFS is that given a massive amount of data, a light weight network should identify the common features of the data and reduce noise before over-fitting happens. The idea has been applied in many previous works [8, 15]. However, they apply the denoising idea in the training process of semantic segmentation and use complex networks to fit dirty supervision in all pixels and all classes directly, making the networks hard to optimize. Therefore, they have to use recursive training (*i.e.*, using network outputs in the last epoch as supervision in the next epoch directly or indirectly) and graph models frequently, which are either complex or time-consuming. The recursive training is likely to be trapped into poor local optimum [8]. CFS uses an independent light weight network for noise reduction and simplifies the task from difficult semantic segmentation to simple class-agnostic foreground segmentation. We segment foreground within bounding boxes. The location and class label of the foreground object are also known. Thus, we call it *Constrained Foreground Segmentation*.

CFS brings several benefits. First, the search area is narrowed down because only a small region instead of the full image needs to be handled. Second, most of the irrelevant information is discarded. Each bounding box includes exactly one foreground object and the multi-scale problem is solved because of the tightness of bounding boxes, Third, the target problem is simplified because binary classification is less complicated than multi-class classification. Based on the above benefits, CFS uses less resources and is much easier to optimize than fitting the whole image with a complex semantic segmentation network.

CFS can generate more high-quality proposals as supervision of semantic segmentation networks. However, the supervision is still noisy inevitably. The semantic segmentation network should have self-correction ability and noisy robustness. We therefore design a self-attention mechanism LGSA to capture long-range dependencies and reduce noise. Similar to the Non-local module [26], LGSA calculates an affinity matrix to reweight feature maps. The reweighting is similar to Non-local means [5] which is effective for image denoising.

The reweighting in the Non-local module is completely unconstrained and may cause unexpected global information exchanges. In contrast, LGSA is guided by a novel loss which encourages information to exchange only among intra-class pixels. We prove that the loss tries to maximize inter-class distances and minimize intra-class distances. LGSA also supports reweighting segments in addition to feature maps, because the module can generate a more reasonable affinity matrix under the guidance. Therefore, we use random work [1, 2] to further reweight the segments and get more accurate results. Experiments show

that LGSA encourages inter-class competition and intra-class self-correction. Compared to the Non-local module, LGSA has better performance and interpretability even with noisy supervision.

Our contributions can be summarized as follows. First, we introduce a constrained foreground segmentation network to generate high-quality dense annotations from noisy proposals by simplifying multi-class semantic segmentation to class-agnostic foreground segmentation. Second, we propose a loss-guided self-attention module that constrains the information exchanges among different kinds of pixels. The module encourages self-correction among intra-class pixels and generates a reasonable affinity matrix to reweight both feature maps and segments. Third, we evaluate the proposed method on the Pascal VOC 2012 benchmark and achieve state-of-the-art performance.

2 Related Works

Semantic segmentation, *i.e.*, assigning a class label to each pixels of an image, is an important computer vision task. The state-of-the-art methods [2, 7, 18, 29] usually use fully convolution neural network to predict dense segmentations and need expensive pixel-level annotations. Therefore, WSSS using cheaper annotations draws a lot of attention recently. The most related works to this paper are the ones using bounding box annotations. The methods [8, 15, 19, 23] usually use GrabCut [22], DenseCRF [16] or MCG [20] to generate proposals. WSSL [19] applies deep learning first for WSSS using bounding boxes. The method uses an expectation-maximization algorithm to learn from noisy supervision. BoxSup [8] introduces an iterative strategy that generates multiple proposals and trains the network alternatively to refine the estimated proposals. SDI [15] designs a novel strategy to combine the proposals generated by GrabCut [22] and MCG [20]. Song *et al*. [23] employ filling rates to reduce the effects of incorrectly labeled data. Different from the methods that update the supervision recursively [8, 15] or use graph models frequently [15], we employ an end-to-end constrained foreground segmentation network to generate high quality proposals directly.

Semantic segmentation also benefits from attention mechanism. Song *et al*. [23] use soft attention to generate a global spatial mask and reduce the weights of irrelevant regions. In the paper, self-attention is applied instead. The Non-local module [26] introduces a novel self-attention which uses deep features to calculate an affinity matrix and then uses the matrix to reweight features. Based on the Non-local module, DANet [10] and CCNet [14] are proposed for fully supervised semantic segmentation. DANet [10] applies attention mechanism in both spatial and channel dimensions and CCNet [14] harvests non-local contextual information in a criss-cross way. Yao *et al*. [27] apply the non-local module to generate proposals for WSSS using image-level labels. Our self-attention module LGSA is similar to [26] and [27], but we design a loss to guide the information exchanges among pixels of different categories. Under the guidance, LGSA generates a more reasonable affinity matrix which can be used to reweight segments in addition to features.

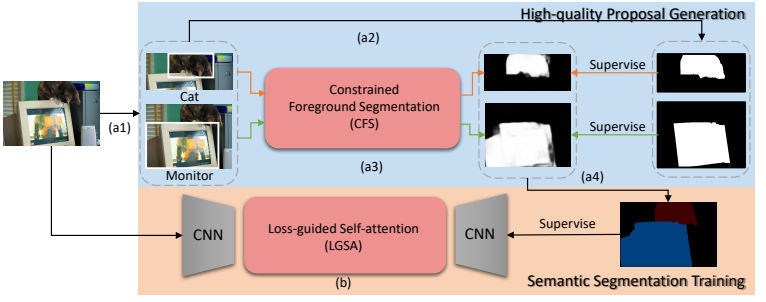


Figure 1: Overview of the proposed method. We expand the bounding boxes to obtain the inputs of the CFS (a1). GrabCut is applied for the coarse proposal generation (a2). CFS (a3) and the full-resolution proposal generation (a4) are employed successively to achieve high-quality proposals. Finally the LGSA-embedded semantic segmentation training is performed (b).

3 Our Method

3.1 Overview

The procedure of our method is shown in Figure 1. We train a constrained foreground segmentation network to identify the foreground objects and obtain high-quality proposals. Then, the proposals serve as pixel-level supervision for training the semantic segmentation network and LGSA is inserted into the semantic segmentation network for further improvements.

3.2 High-quality Proposal Generation

We apply GrabCut to obtain coarse proposals for images within each bounding box, denoted as $M_{init} \in [0,1]$ (0 for background and 1 for foreground).

The proposals from GrabCut are noisy due to the lack of semantic information. We apply CFS to get high-quality proposals. The visible region to CFS is limited within the bounding boxes to alleviate the interference from irrelevant information. Concretely, for each bounding box, we expand it by increasing its height and width by 50%. The expanded region, whose annotations are determinate as background, helps to distinguish foreground objects. We then crop the region within the expanded bounding box from the original image, denoted as I_{exp}. Accordingly, the initial proposal M_{init} is expanded to M_{exp}:

$$M_{exp}(i',j') = \begin{cases} M_{init}(i,j), & \text{if } (i,j) \text{ in } \mathcal{B}, \\ 0, & \text{otherwise.} \end{cases} \tag{1}$$

where $\mathcal{B}$ is the original bounding box and (i,j) is the corresponding coordinates of pixel (i',j') in M_{init}. M_{exp} will serve as the ground truth of CFS. Similarly, we encode the position information of the original bounding box into B_{exp}:

$$B_{exp}(i',j') = \begin{cases} 1, & \text{if } (i,j) \text{ in } \mathcal{B}, \\ 0, & \text{otherwise.} \end{cases} \tag{2}$$

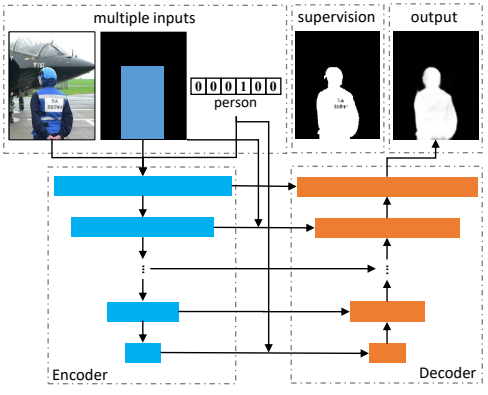


Figure 2: The structure of CFS network. The details can be found in the supplementary material

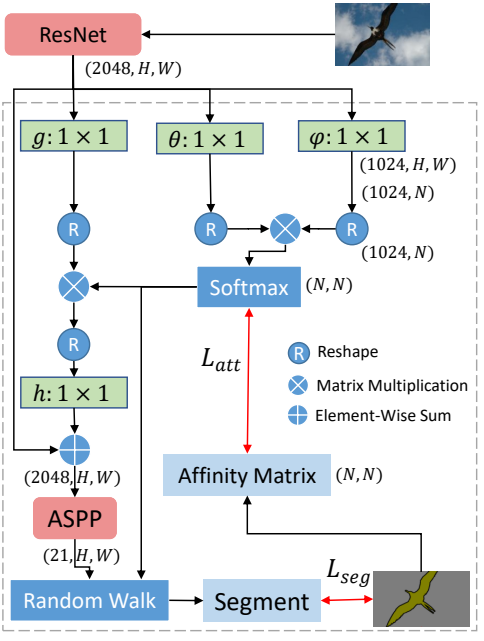


Figure 3: The structure of LGSA. ASPP is a moudle of DeepLab-v2[7].

B_{exp} and class label y serve as the inputs of CFS in the one-hot format. These prior knowledge benefits the network training effectively.

We design a CFS network similar to U-Net [21] with a ResNet-50 backbone (Figure 2 and more details in the supplementary material). The network takes I_{exp}, B_{exp} and y as inputs. The bounding box masks and class labels help distinguish foreground objects in shallow layers, but the information usually vanishes in deep layers. Therefore, following the idea of residual blocks [12], we add skip connections to feed B_{exp} and y to deep layers directly. The loss function of CFS is defined as

$$L_{CFS} = \mathrm{CE}(\hat{M}_{exp}, M_{exp}), \tag{3}$$

where $\hat{M}_{exp}$ is the output of CFS network and $\mathrm{CE}(\cdot)$ represents the pixel-wise cross-entropy loss.

Because only cross-entropy loss is used for training, the network output lacks precise boundaries. Therefore, we use GrabCut to refine them. Specifically, pixels with foreground probability higher than 60% are considered as *foreground*, and pixels with foreground probability lower than 40% are considered as *background*, and the rest are regarded as *possible foreground*. GrabCut is performed with the above trimap as a mask prior to obtain a precise proposal.

In the inference, we apply CFS for each bounding box respectively. The class label of the bounding box is also known. We then combine the results to a full-resolution multi-class proposal. If two boxes overlap, we process the larger one first because the smaller one is more likely to be in the front of foreground.

3.3 Semantic Segmentation Learning

We use a fully convolution network for semantic segmentation. CFS is effective in reducing noise and generating high-quality proposals but cannot eliminate all mistakes. Therefore, we design LGSA to achieve network self-correction and noise robustness. The structure of LGSA is shown in Figure 3.

Similar to the Non-local module, we use deep features to calculate an affinity matrix and then use the matrix to reweight the features. Given a feature $X \in R^{C\times H\times W}$, where C, H and W denote the channel, height and width of X respectively, we applies three different linear embedding functions (*e.g.*, 1×1 convolutions) θ, ϕ and g to obtain $\theta(X)$, $\phi(X)$ and $g(X)$. The embedded results with a shape of $C'\times H\times W$ are reshaped to $C'\times N$ where $N = H\times W$. The affinity matrix $W \in R^{N\times N}$ is produced by matrix multiplication and softmax:

$$W = \text{softmax}(\theta(X)^T \cdot \phi(X)). \tag{4}$$

Then we use the matrix to reweight the feature:

$$Z = h(W \cdot g(X)) + X, \tag{5}$$

where Z is the reweighted feature, h is another linear embedding function whose weight and bias are initialized to zero. We refer readers to [26] for more details about the basic self-attention module.

Up to now, the calculation of the affinity matrix is completely unconstrained which may cause unreasonable information exchanges. For example, suppose an image containing a *cat* and a *dog* is used. The information should not be exchanged between the two objects because it will make their features similar and thus confuse the network. To overcome the problem, the proposed LGSA constrains information exchanges among pixels of different categories by performing an loss function on the generated affinity matrix. We denote $W \in R^{N\times N}$ as the generated affinity matrix of LGSA and define the ground truth matrix W_{gt} as

$$W_{gt}(i,j) = \begin{cases} 0, & \text{if } M(i) \neq M(j), \\ 1, & \text{otherwise}, \end{cases} \tag{6}$$

where M is the generated full-resolution multi-class proposals in section 3.2. Then each $W_{gt}(i,j)$ is divided by $\sum_i W_{gt}(i,j)$ for normalization. The loss function of LGSA can be formulated as

$$L_{att} = \text{MSE}(W, W_{gt}), \tag{7}$$

where $\text{MSE}(\cdot)$ is mean squared error. With the loss, LGSA encourages inter-class competitions and intra-class self-correction. The weight between two pixels of different categories tends to be 0. Hence, the information exchanges are constrained between different categories and inter-class distances will tend to the maximum. The weight between two pixels of the same category tends to be the mean value of all the pixels of the category, leading to the minimum of intra-class distances.

In the Non-local module, reweighting is only performed to deep features. However, LGSA can also reweight the segments because LGSA generates a more reasonable affinity matrix under the loss guidance. We use random walk [1, 4] to reweight the segments as

$$\hat{M}_{rw} = W \cdot \hat{M}, \tag{8}$$

where $\hat{M}$ is the original network output and $\hat{M}_{rw}$ is the refined result.

We train the semantic segmentation network in an end-to-end manner. The loss for segmentation is

$$L_{seg} = \mathrm{CE}(\hat{M}_{rw}, M). \tag{9}$$

The total loss for the network is

$$L_{total} = L_{seg} + \alpha \cdot L_{att}, \tag{10}$$

where α is a hyper-parameter to balance the two losses.

4 Experiments

4.1 Experimental Setup

We conduct experiments on the Pascal VOC 2012 benchmark [9], which contains 1,461 images for training, 1,449 images for validation, and 1,459 images for testing. Following the setting of other similar works, we augment the training set to 10,582 images according to [11].

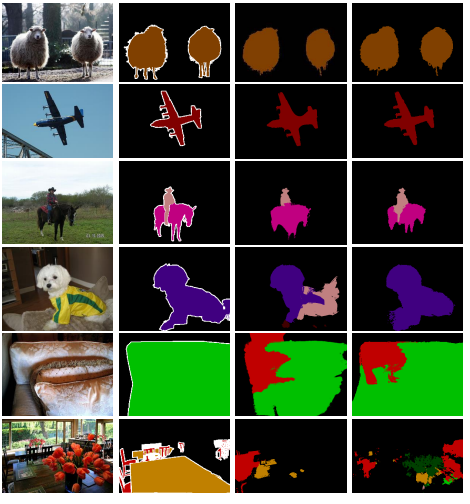
When training CFS, small bounding boxes, *i.e.*, less than 50 pixels in height or width, are filtered out. All boxes remain in the inference. The input size is 224×224. The optimizer is SGD with an initial learning rate of 0.005. A step learning rate scheduler is implemented where the learning rate is multiplied by 0.8 every 10 epochs. The total epoch numbers is 60.

For the semantic segmentation training, DeepLabv2 with a ResNet-101 backbone [7] pretrained on *ImageNet* is adopted. We follow the settings in [7] except that the initial learning rate is 0.00025 and the training epochs are 15. A warm-up strategy is employed where the learning rate in the first epoch is divided by 10. We find the strategy to be useful when the network includes self-attention modules. By default, we apply two cascaded LGSA modules for the balance of performance and efficiency, and insert them right after the ResNet backbone. α is set to 10,000 when using two or more LGSA modules and 1,000 when using one module. We only use the affinity matrix generated by the last LGSA module for random walk. In the inference, DenseCRF [16] is applied for post-processing.

All experiments are implemented using PyTorch and performed on four NVIDIA GTX 2080 Ti GPUs. We use mean Intersection-over-Union (mIoU) to evaluate semantic segmentation results.

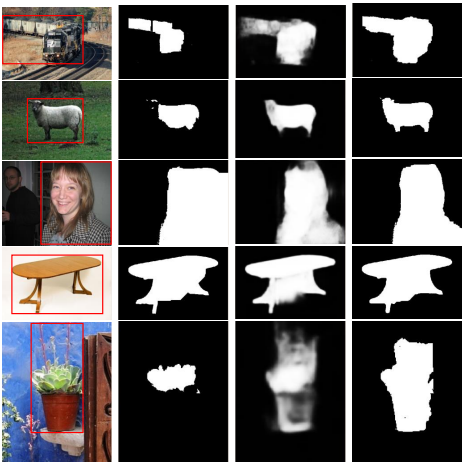
4.2 Comparisons with Other Methods

We compare our method with other WSSS methods on the Pascal VOC 2012 *validation* set. Table 1 shows that our method achieves better performance than other WSSS methods using bounding box annotations. Figure 4 shows qualitative segment results of our method compared with Song *et al*. [23]. The first three rows indicate our segments are more accurate, especially on the object boundaries. The last three rows come from the failure cases in their paper. The method of Song *et al*. is confused by some rare patterns, *e.g.*, *a dog with clothes*, whereas our method is more robust. Our method also fails to distinguish *chair* and *sofa* with similar appearances, but our method still outperforms than theirs. For instance, the incorrect *chair* regions are smaller, the boundaries of the *sofa* are more accurate, and the *potted plant* is recognized.



(a) (b) (c) (d)

Figure 4: Qualitative segmentation results on PASCAL VOC 2012 *val* set. (a)-origin image, (b)-ground truth, (c)-results of Song *et al*. [23], (d)-ours.



(a) (b) (c) (d)

Figure 5: Qualitative results of CFS. (a)-origin images, (b)-GrabCut results, (c)-CFS results, (d)-refined CFS results.

Table 1: Comparisons with other WSSS methods. Ann.-annotations, F-Full supervision, I-image-level label, S-scribble, B-bounding box.

Method	Backbone	Ann.	val(mIoU)
Deeplab v1 [6]	VGG-16	F	67.6
Deeplab v2 [7]	ResNet-101	F	76.2
FickleNet [17]	ResNet-101	I	64.9
RDM [28]	ResNet-101	I	66.3
Kernel Cut [24]	ResNet-101	S	75.0
BPG [25]	ResNet-101	S	76.0
BoxSup [8]	VGG-16	B	62.0
WSSL [19]	VGG-16	B	60.6
SDI [15]	ResNet-101	B	69.4
Song *et al*. [23]	ResNet-101	B	70.2
Ours	ResNet-101	B	**72.4**

Table 2: Comparisons with other supervisions using official DeepLab-v2 in Pascal VOC 2012 *val* set.

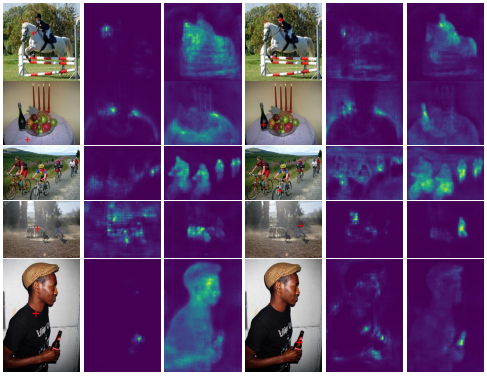
Supervision	mIoU
Bounding box	55.4
GrabCut	66.6
CFS	**70.8**

Table 3: Comparisons with the Non-local module.

	Block Number		
	1	2	3
Non-local	71.3	71.5	71.5
Ours	**71.9**	**72.4**	**72.6**

Table 4: Per-class results in Pascal VOC 2012 *val set (mIoU in %).*

Methods	bkg	plane	bike	bird	boat	bottle	bus	car	cat	chair	cow	table	dog	horse	moto	person	plant	sheep	sofa	train	tv	**mean**
DeepLab$_{grabcut}$	90.7	71.1	31.0	81.6	56.6	69.2	83.3	74.1	83.2	32.8	77.9	55.7	77.7	70.0	65.0	71.2	49.4	71.4	45.1	73.5	70.1	66.6
Ours	**92.3**	69.3	**31.1**	**84.6**	**71.0**	**76.2**	**90.7**	**81.8**	**87.3**	**39.6**	**84.5**	**57.5**	**83.1**	**74.0**	**73.3**	**76.4**	**60.6**	**80.8**	**57.5**	**80.4**	69.2	**72.4**



(a) Image (b) Non-local (c) LGSA (d) Image (e) Non-local (f) LGSA

Figure 6: Visualizations of Non-local and LGSA. The query point is marked as a red '+' and the corresponding weights are sampled from the affinity matrix and shown as a heatmap.

Table 4 shows per-class results of our method and DeepLab-v2 supervised by GrabCut proposals. Our method outperforms GrabCut in most situations.

4.3 Ablation Study

Effectiveness of CFS. We train the official DeepLab-v2 model with different supervisions, including bounding boxes, GrabCut proposals and high-quality proposals generated by CFS, to verify the effectiveness of CFS. The models are evaluated on Pascal VOC 2012 *val* set and the result is in Table 2. Our CFS is more outstanding than GrabCut, indicating that CFS can generate more high-quality dense annotations from noisy proposals. Figure 5 shows that GrabCut can locate the foreground object but shows performance degradation when either the foreground colors are close to background or the textures are complex, *e.g.*, the *person* and the *potted plant*. Our CFS outperforms in those situations because CFS tries to find common semantic patterns of the data and therefore generates more high-quality proposals.

Effectiveness of LGSA. We compare LGSA with the Non-local module under different self-attention block number settings. Table 3 shows that the performance of LGSA surpasses the Non-local module's. Our LGSA also gives better interpretability as shown in Figure 6 (More visualizations are shown in the supplementary material). The affinity matrix generated by the Non-local module focuses on fragmentary regions, while LGSA's affinity matrix is more reasonable and focuses on all pixels of the same class. The main reason is that the Non-local module captures dependencies without guidance and is sensitive to certain small regions. LGSA guides the information exchanges to restrain unreasonable exchanges among different categories and enhance the correspondence of intra-class pixels.

Influence of α and random walk. We test the influence of α to further verify the proposed LGSA. Table 5 shows that when α is 0, *i.e.*, no loss guidance, the performance is 71.5,

Table 5: The influence of α and random walk (RW).

	α				
	0	5000	10000	15000	20000
Without RW	71.5	71.7	71.8	71.8	71.7
With RW	71.5	**72.1**	**72.4**	**72.3**	**72.2**

similar to Non-local modules. As α is increased, the network benefits from the loss guidance. However, a too large α may be harmful to segmentation learning. Table 5 also shows the influence of random walk, *i.e.*, reweighting segments. When α is 0, random walk has almost no influence to the performance because the affinity map is unreasonable. As the increment of α, random walk increases the performance significantly, indicating that the affinity map from LGSA is more reasonable under the loss guidance.

5 Conclusions

In this paper, we propose CFS and LGSA to learn effectively from noisy supervision for WSSS using bounding box annotations and achieve state-of-the-art results. CFS generates high-quality proposals by considerably simplifying the segmentation problem. LGSA generates a reasonable affinity matrix to reweight deep features and segments under a novel loss guidance, and obtains better performance and interpretability than the Non-local module.

Our current approach has some limitations. For example, the thresholds utilized for generating a trimap in the proposal refinement are fixed and LGSA has $O(n^2)$ time and space complexity. In the future, we will explore ways to adjust the thresholds adaptively and reduce the computing complexity of LGSA.

Acknowledgements This work was supported by Tsinghua-Kuaishou Institute of Future Media Data. We appreciate Dr. Wen Zheng and Huijuan Huang for their suggestion and help.

References

[1] Jiwoon Ahn and Suha Kwak. Learning pixel-level semantic affinity with image-level supervision for weakly supervised semantic segmentation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4981–4990, 2018.

[2] Vijay Badrinarayanan, Alex Kendall, and Roberto Cipolla. Segnet: A deep convolutional encoder-decoder architecture for image segmentation. *IEEE transactions on pattern analysis and machine intelligence*, 39(12):2481–2495, 2017.

[3] Amy Bearman, Olga Russakovsky, Vittorio Ferrari, and Li Fei-Fei. What's the point: Semantic segmentation with point supervision. In *European conference on computer vision*, pages 549–565. Springer, 2016.

[4] Gedas Bertasius, Lorenzo Torresani, Stella X Yu, and Jianbo Shi. Convolutional random walk networks for semantic image segmentation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 858–866, 2017.

[5] Antoni Buades, Bartomeu Coll, and J-M Morel. A non-local algorithm for image denoising. In *2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05)*, volume 2, pages 60–65. IEEE, 2005.

[6] Liang-Chieh Chen, George Papandreou, Iasonas Kokkinos, Kevin Murphy, and Alan L Yuille. Semantic image segmentation with deep convolutional nets and fully connected crfs. *arXiv preprint arXiv:1412.7062*, 2014.

[7] Liang-Chieh Chen, George Papandreou, Iasonas Kokkinos, Kevin Murphy, and Alan L Yuille. Deeplab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected crfs. *IEEE transactions on pattern analysis and machine intelligence*, 40(4):834–848, 2017.

[8] Jifeng Dai, Kaiming He, and Jian Sun. Boxsup: Exploiting bounding boxes to supervise convolutional networks for semantic segmentation. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 1635–1643, 2015.

[9] Mark Everingham, Luc Van Gool, Christopher KI Williams, John Winn, and Andrew Zisserman. The pascal visual object classes (voc) challenge. *International journal of computer vision*, 88(2):303–338, 2010.

[10] Jun Fu, Jing Liu, Haijie Tian, Yong Li, Yongjun Bao, Zhiwei Fang, and Hanqing Lu. Dual attention network for scene segmentation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3146–3154, 2019.

[11] Bharath Hariharan, Pablo Arbeláez, Lubomir Bourdev, Subhransu Maji, and Jitendra Malik. Semantic contours from inverse detectors. In *2011 International Conference on Computer Vision*, pages 991–998. IEEE, 2011.

[12] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016.

[13] Zilong Huang, Xinggang Wang, Jiasi Wang, Wenyu Liu, and Jingdong Wang. Weakly-supervised semantic segmentation network with deep seeded region growing. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 7014–7023, 2018.

[14] Zilong Huang, Xinggang Wang, Lichao Huang, Chang Huang, Yunchao Wei, and Wenyu Liu. Ccnet: Criss-cross attention for semantic segmentation. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 603–612, 2019.

[15] Anna Khoreva, Rodrigo Benenson, Jan Hosang, Matthias Hein, and Bernt Schiele. Simple does it: Weakly supervised instance and semantic segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 876–885, 2017.

[16] Philipp Krähenbühl and Vladlen Koltun. Efficient inference in fully connected crfs with gaussian edge potentials. In *Advances in neural information processing systems*, pages 109–117, 2011.

[17] Jungbeom Lee, Eunji Kim, Sungmin Lee, Jangho Lee, and Sungroh Yoon. Ficklenet: Weakly and semi-supervised semantic image segmentation using stochastic inference. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 5267–5276, 2019.

[18] Jonathan Long, Evan Shelhamer, and Trevor Darrell. Fully convolutional networks for semantic segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3431–3440, 2015.

[19] George Papandreou, Liang-Chieh Chen, Kevin P Murphy, and Alan L Yuille. Weakly- and semi-supervised learning of a deep convolutional network for semantic image segmentation. In *Proceedings of the IEEE international conference on computer vision*, pages 1742–1750, 2015.

[20] Jordi Pont-Tuset, Pablo Arbelaez, Jonathan T Barron, Ferran Marques, and Jitendra Malik. Multiscale combinatorial grouping for image segmentation and object proposal generation. *IEEE transactions on pattern analysis and machine intelligence*, 39(1): 128–140, 2016.

[21] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. In *International Conference on Medical image computing and computer-assisted intervention*, pages 234–241. Springer, 2015.

[22] Carsten Rother, Vladimir Kolmogorov, and Andrew Blake. Grabcut: Interactive foreground extraction using iterated graph cuts. In *ACM transactions on graphics (TOG)*, volume 23, pages 309–314. ACM, 2004.

[23] Chunfeng Song, Yan Huang, Wanli Ouyang, and Liang Wang. Box-driven class-wise region masking and filling rate guided loss for weakly supervised semantic segmentation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3136–3145, 2019.

[24] Meng Tang, Federico Perazzi, Abdelaziz Djelouah, Ismail Ben Ayed, Christopher Schroers, and Yuri Boykov. On regularized losses for weakly-supervised cnn segmentation. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 507–522, 2018.

[25] Bin Wang, Guojun Qi, Sheng Tang, Tianzhu Zhang, Yunchao Wei, Linghui Li, and Yongdong Zhang. Boundary perception guidance: a scribble-supervised semantic segmentation approach. In *Proceedings of the 28th International Joint Conference on Artificial Intelligence*, pages 3663–3669. AAAI Press, 2019.

[26] Xiaolong Wang, Ross Girshick, Abhinav Gupta, and Kaiming He. Non-local neural networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 7794–7803, 2018.

[27] Qi Yao and Xiaojin Gong. Saliency guided self-attention network for weakly-supervised semantic segmentation. *arXiv preprint arXiv:1910.05475*, 2019.

[28] Bingfeng Zhang, Jimin Xiao, Yunchao Wei, Mingjie Sun, and Kaizhu Huang. Reliability does matter: An end-to-end weakly supervised semantic segmentation approach. *arXiv preprint arXiv:1911.08039*, 2019.

[29] Hengshuang Zhao, Jianping Shi, Xiaojuan Qi, Xiaogang Wang, and Jiaya Jia. Pyramid scene parsing network. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2881–2890, 2017.

[30] Bolei Zhou, Aditya Khosla, Agata Lapedriza, Aude Oliva, and Antonio Torralba. Learning deep features for discriminative localization. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2921–2929, 2016.