Forget and Diversify: Regularized Refinement for Weakly Supervised Object Detection

Jeany Son[1], Daniel Kim[1,2], Solae Lee[2], Suha Kwak[2], Minsu Cho[2], and Bohyung Han[1(✉)]

[1] Computer Vision Lab., ASRI, Seoul National University, Seoul, Korea
{jeany,bhhan}@snu.ac.kr
[2] Computer Vision Lab., POSTECH, Pohang, Korea
{daniel.kim,solae,suha.kwak,mscho}@postech.ac.kr

Abstract. We study weakly supervised learning for object detectors, where training images have image-level class labels only. This problem is often addressed by multiple instance learning, where pseudo-labels of proposals are constructed from image-level weak labels and detectors are learned from the potentially noisy labels. Since existing methods train models in a discriminative manner, they typically suffer from collapsing into salient parts and also fail in localizing multiple instances within an image. To alleviate such limitations, we propose simple yet effective regularization techniques, weight reinitialization and labeling perturbations, which prevent overfitting to noisy labels by forgetting biased weights. We also introduce a graph-based mode-seeking technique that identifies multiple object instances in a principled way. The combination of the two proposed techniques reduces overfitting observed frequently in weakly supervised setting, and greatly improves object localization performance in standard benchmarks.

Keywords: Weakly supervised learning · Object detection · Regularization

1 Introduction

Object detection algorithms recently demonstrate remarkable performance thanks to advances of deep neural network technologies [5,12,13,24,27,28,30] and well-established datasets provided with bounding box annotations [11,20,23]. Despite their great success, many object detection algorithms still suffer from a critical limitation caused by lack of training examples with proper annotations. In particular, due to substantial cost for bounding box labeling and inherent skewness of training data distributions, existing datasets for object detection are often insufficient in their quantity and diversity for majority of classes. This fact incurs overfitting to datasets and damages generalization performance of models.

C. V. Jawahar et al. (Eds.): ACCV 2018, LNCS 11364, pp. 632–648, 2019.
https://doi.org/10.1007/978-3-030-20870-7_39

Weakly supervised object detection (WSOD) has been studied as a solution to the above issues [1,2,4,21,22,39]. The primary goal of this task is to train object detectors using image-level class labels only. The limitations of the standard object detection algorithms can be alleviated by weakly supervised approaches because image-level class labels are readily available in several existing large-scale datasets for image classification, *e.g.* ImageNet [6], or easily obtainable due to their low annotation cost. However, learning object detectors based only on image-level class labels is challenging because the labels indicate presence or absence of each object class without localization information of objects.

Many recent weakly supervised object detection algorithms rely heavily on weakly supervised deep detection network (WSDDN) [2]. This approach identifies relevant bounding boxes to individual classes by applying softmax operations to score matrices across object proposals and candidate class labels. The performance of this method has been improved by adding a few refinement layers [39]. However, WSDDN and its extensions have the following critical limitations. First, as in many other weakly supervised object detection techniques, noisy annotations estimated by object detectors based on weak labels may make models converge to bad local optima in training. Second, due to characteristics of softmax functions, the method is prone to identify only a single target class and object instance in an input image. Consequently, they are not effective to handle images with multiple objects corresponding to diverse class labels.

To alleviate the limitations, we propose simple yet effective multi-round regularization techniques for handling noisy labels, and introduce a graph-based labeling method for mining multiple instances in the same class. Specifically, we integrate refinement layers into the WSDDN architecture and perform multiple rounds of training with randomly reinitialized weights of the refinement layers. This regularization technique prevents the deep neural network from overfitting by forgetting biased weights. Also, a mode-seeking algorithm is performed on a graph of object proposals to identify multiple target instances in a principled way, where the graph is constructed to diversify pseudo-labels by perturbing a threshold to connect vertices corresponding to proposals. The combination of the multi-round regularization and the graph-based labeling improves object detection accuracy substantially in the standard weakly supervised setting for object detection. Our main contributions are summarized as follows:

- We introduce simple multi-round regularization techniques for weakly supervised object detection, which are based on refinement layer reinitializations and labeling perturbations, to tackle overfitting issues caused by falling into bad local optima.
- We propose a mode-seeking technique for labeling candidate bounding boxes, where a graph structure of object proposals is constructed based on their class scores and spatial proximities. This method is helpful to identify multiple object instances of a class in a principled way.

- We demonstrate that our approach improves performance significantly with respect to the state-of-the-art methods in weakly supervised object detection on the standard benchmarks such as PASCAL VOC 2007 and 2012.

This paper has the following organization. Section 2 discusses related work and Sect. 3 presents technical background of our problem. We describe the proposed regularization and label generation techniques in Sect. 4. Experimental results with internal and external comparative study are presented in Sect. 5, and we conclude this paper in Sect. 6.

2 Related Work

This section describes existing approaches about weakly supervised object detection and regularization of deep neural networks.

2.1 Weakly Supervised Object Detection

Weakly supervised object detection algorithms typically rely only on image-level class labels in text to find all the bounding boxes corresponding to objects in target classes. There have been a large volume of research in this interesting topic [1,2,4,18,21,22,26,39,46]. Most approaches in this line of research follow the idea of Multiple Instance Learning (MIL) [8]; a set of proposals from an image constructs a bag, and its label is determined by its image-level weak labels. During training, the approaches alternate selecting the most representative proposals in positive images and learning object detectors using tentatively estimated positive and negative instances. Since a list of true positive instances per image is latent, the optimization is inherently sensitive to initializations of individual examples and prone to fall into bad local optima consequently.

Most MIL-based approaches attempt to improve initialization [7,22,36,37,45] and enhance classifiers through optimization [1,2,4,33,35,39,41]. Li *et al.* [22] collect class specific object proposals and optimize the network progressively using confident object candidates. Self-taught learning approach [17] has been proposed to obtain high-quality proposals progressively. Diba *et al.* [7] introduce cascaded networks with multiple learning stages, which incorporate class specific proposal generation and MIL in an end-to-end pipeline. Multi-fold MIL method [4] splits training data into multiple subsets and learn models to escape from local optima and avoid overfitting. Wan *et al.* [41] perform clustering of object proposals based on their scores and overlap ratios, and minimize entropy of proposal scores in the same cluster, by which it improves localization accuracy and reduces localization randomness.

WSDDN [2] is probably the most popular MIL based end-to-end deep framework, where image-level classification loss is computed by a sum of proposal scores. This framework has been investigated further and a variety of extensions have been proposed recently [4,18,33,39,41,44,45]. Kantorov *et al.* [18] integrates semantic context information to identify tight bounding boxes corresponding objects of interest. Tang *et al.* [39] diffuses labels estimated by WSDDN

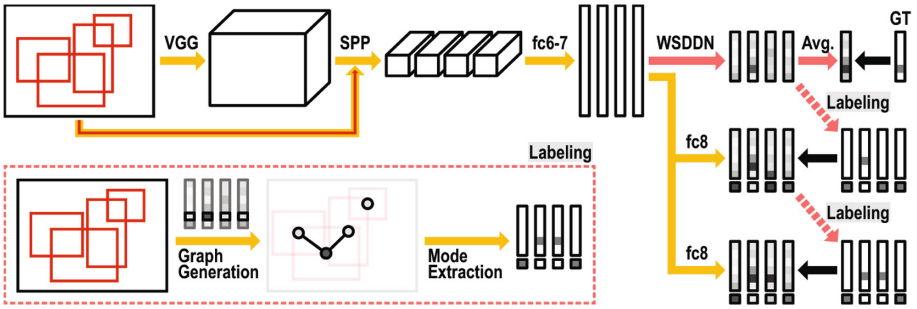

Fig. 1. The network architecture of the proposed approach. A feature of each proposal is extracted from a spatial pyramid pooling layer followed by two `fc` layers and then fed to WSDDN and multiple classifier refinements for training. Supervision for each refinement step is given by the predictions of the preceding step. Our graph-based labeling generates pseudo ground-truth labels for the proposals that are used to learn refinement layers.

to highly-overlapped bounding boxes and learns object detectors end-to-end. Saliency-guided proposal selection has been proposed in [21] to generate reliable positive examples by drawing boxes enclosing areas with heavy class-specific attention, where classification and saliency losses of the proposals are jointly optimized to localize objects. Zhang *et al.* [45] generate diverse and reliable positive examples by merging boxes with detection scores from [39]. Zhang *et al.* [44] train a detector by feeding training examples in an increasing order of difficulty. Shen *et al.* [33] present a generative adversarial learning method to train a detector, which emulates a surrogate detector similar to WSDDN, using image-level annotations only.

2.2 Regularization of Deep Neural Networks

Regularization on deep neural networks is a crucial technique to avoid overfitting that results from overparametrized nature of networks. Even simple heuristics including early stopping, weight decay, and data augmentation turn out to be effective in practice. A class of well-known techniques is regularization by noise injection, where random noises are added to input images [29], ground-truth labels [43], or network weights [19,42] during training for better generalization. In particular, dropout [38] and dropconnect [42] employ binary random noise to hidden units or connections of neural networks, and learning with stochastic depth [14,15] can be interpreted as a regularization method by noise injection into model architecture. Recently, [25] discusses theoretical aspect of regularization-by-noise techniques, and [31] proposes a confidence calibration technique based on stochastic regularization. Unlike existing methods, the proposed multi-round regularization technique is specialized to the scenario of weakly supervised object detection.

3 Preliminaries

Our approach builds on WSDDN [2] and its refinement [39]. Figure 1 illustrates the network architecture and label generation algorithm of our approach. Given an image I and its binary label vector with C classes $\mathbf{y} = [y_1, \ldots, y_C]$, WSDDN learns objectness score $s_{c,r}$ for class c of proposal r through elementwise multiplication of classification confidence, $\boldsymbol{\psi}_{\text{cls}} \in \mathbb{R}^{C \times |\mathcal{R}|}$ and localization confidence, $\boldsymbol{\psi}_{\text{loc}} \in \mathbb{R}^{C \times |\mathcal{R}|}$. The value of an element corresponding to an (r, c) pair in the resulting matrix is given by

$$\begin{aligned} s_{c,r} &= \boldsymbol{\psi}_{\text{cls}}(c; r) \cdot \boldsymbol{\psi}_{\text{loc}}(r; c) \\ &= \frac{f_{\text{cls}}(c; r)}{\sum_{i=1}^{C} \exp(f_{\text{cls}}(i; r))} \cdot \frac{f_{\text{loc}}(r; c)}{\sum_{i=1}^{|\mathcal{R}|} \exp(f_{\text{loc}}(i; c))}, \end{aligned} \tag{1}$$

where $f_{\text{cls}}(c; r)$ denotes an activation of a class c given a proposal r in the network while $f_{\text{loc}}(r; c)$ is an activation of a proposal r given a class c. Image-level class score vector, $\boldsymbol{\phi} = \{\phi_1, \ldots, \phi_C\}$, is computed by a global sum pooling over all proposals, which is given by

$$\phi_c = \sum_{r=1}^{|\mathcal{R}|} s_{c,r}, \tag{2}$$

and the score is employed to compute a multi-class cross entropy loss $\mathcal{L}_{\text{wsddn}}$ as follows:

$$\mathcal{L}_{\text{wsddn}} = -\sum_{c=1}^{C} y_c \log \phi_c + (1 - y_c) \log (1 - \phi_c). \tag{3}$$

To avoid converging discriminating parts of an object, additional refinement layers are added to WSDDN. The refinement layers are trained using pseudo ground-truth labels determined by proposal scores from preceding steps as illustrated with red dashed arrows in Fig. 1. The loss function for the k^{th} refinement step, where $k \in \{1, 2, \ldots, K\}$, is given by

$$\mathcal{L}^k_{\text{refine}} = -\frac{1}{|\mathcal{R}|} \sum_{r=1}^{|\mathcal{R}|} \sum_{c=1}^{C+1} w^k_r z^k_{c,r} \log s^k_{c,r}, \tag{4}$$

where $s^k_{c,r}$ and $z^k_{c,r}$ denote the output score and the pseudo-label of a proposal r in the k^{th} refinement for a class c, respectively, while w^k_r is the weight of the proposal, which is used to manage noisy supervision in the refinement layers and avoid unstable solution. Note that each class has a class index $c \in \{1, 2, \ldots, C+1\}$ in a fixed order and the last index $C + 1$ corresponds to background. The total loss of our overall network is obtained by combining those two losses as follows:

$$\mathcal{L} = \mathcal{L}_{\text{wsddn}} + \sum_{k=1}^{K} \mathcal{L}^k_{\text{refine}}. \tag{5}$$

Algorithm 1. Learning our WSOD network with multi-round regularization

Input: Number of training rounds T, number of refinement steps K.
for $i = 1$ to T **do**
  Initialize parameters of refinement layers randomly. (Sect. 4.1)
  Update θ_{IoU} for labeling perturbation. (Sect. 4.1)
  for each iteration **do**
    Build a proposal graph with θ_{IoU} in each refinement step of each image.
    Generate labels of individual labels. (Sect. 4.2)
    Train the network with K refinement steps using the loss function in Eq. (5).
  end for
end for

During inference, a final detection score for each proposal is computed by averaging softmax scores over all refinement classifiers.

4 Our Approach

The architecture introduced in Sect. 3 has two inherent issues. First, as the architecture is trained using pseudo-labels in refinement steps, the learning procedure is prone to fall in bad local optima. Second, due to the limitation of the labeling scheme during refinement steps, it identifies only a single object instance in an image even in the case multiple instances exist in the image.

To tackle these challenges, we propose *multi-round regularization* and *graph-based labeling* techniques in our weakly supervised object detection framework. Both components are useful to improve object detection performance. The overall learning procedure is outlined in Algorithm 1, and we discuss the details of each component in the rest of this section.

4.1 Multi-round Regularization of Refinement

Our weakly supervised object detection algorithm relies on MIL, where we obtain pseudo ground-truth labels for individual bounding boxes based on their prediction scores for training object detector. However, as expected, this strategy may incur a lot of label noises, which leads to increase of modeling bias and prediction error. To mitigate this limitation, we present multi-round regularization techniques that improve target object representation and avoid overfitting of our weakly supervised object detection network. Note that the multi-round regularization is specialized to weakly supervised learning because labels of all examples are dynamically determined in each stage depending on network parameters. We claim that multi-round regularization is useful to consider potential label noise and reduce training bias in weakly supervised setting.

Our multi-round regularization has two components, refinement layer reinitialization and label perturbation. The second component is related to graph-based label generation method. The multiple rounds of training with reinitialization and perturbation reduce the bias of the learned models affected by a fixed

but potentially erroneous labels and prevent the models from being converged to bad local optima.

Reinitialization of Refinement Layers. Our refinement network is composed of a single `fc` layer in each stage, and we simply reinitialize the parameters in the refinement layers of all three stages in each round of training. Since the last `fc` layers in the classifier refinements are trained using the labels predicted by the preceding stages, these layers may be biased by noisy labels. However, if the `fc` layers are reinitialized before starting the next round, we can diversify labels and avoid overfitting problem while feature extraction parts of the network learn better representations for target classes.

Labeling Perturbation for Refinements. The pseudo ground-truth labels of individual bounding boxes are determined by a graph-based labeling algorithm, which will be discussed in Sect. 4.2. Another regularization scheme for our weakly supervised object detection is to perturb the instance labels during our training procedure. This regularization method is based on a similar motivation to the reinitialization technique discussed above, where we aim to reduce bias of learned models originated from noisy labels. Instead of random perturbation, we adjust a parameter, which directly affects label assignment for each bounding box, the graph construction in each round of training and decide the label of each proposal using the graph-based labeling algorithm with the perturbed parameters. This label perturbation strategy increases diversity in the number of detected objects, and make our models optimized towards a new objective given by a different label set in each round.

4.2 Graph-Based Label Generation

Since images often include multiple instances of a class, the label generation method should be able to handle an arbitrary number of object instances conceptually. Hence, we propose a new labeling method based on mode-seeking on a graph structure, which is illustrated inside the red dashed box of Fig. 1. Our graph-based labeling technique facilitates to identify diverse positive proposals by building a graph structure of proposals based on their overlap relations and finding multiple modes with high classification scores. This graph-based labeling allows us to obtain accurate labels by diversifying annotations and improve quality of trained models. Note that Tang *et al.* [39] regard proposals that have large overlap ($\geq$0.5 in terms of IOU) with the top-scoring bounding box as positive instances while making the remaining ones negative; selecting only positive examples from a single mode inherently limits capability to handle multiple objects in an images.

Our graph-based label generation method first constructs a neighborhood graph of proposals for each object class in each image. In the graph of class c at refinement step k, denoted by $G_c^k = (\mathcal{V}_c^k, \mathcal{E}_c^k)$, a vertex corresponds to a proposal with a sufficiently large classification score given by the preceding step, and an

Algorithm 2. Graph-based mode-seeking algorithm

Input: Graph $G = (\mathcal{V}, \mathcal{E})$ and weight vector $\mathbf{w} \in \mathbb{R}^{|V|}$
Output: A detected mode set $\mathcal{M}$
$\mathbf{h} \leftarrow [1, 2, \ldots, |V|] \in \mathbb{R}^{|V|}$ /* $\mathbf{h}$ is a cluster indicator vector */
while until $\mathbf{h}$ converges **do**
  for $u \in \mathcal{V}$ **do**
    $h(u) \leftarrow \text{argmax}_{v \in \mathcal{N}_{h(u)}} w(v)$ /* medoid-shift */
  end for
end while
$\mathcal{M} \leftarrow$ a set of unique elements of $\mathbf{h}$

edge connects two of vertices if the proposals for the vertices have sufficiently large overlap to each other. Formally, the sets of vertices and edges are defined by

$$\begin{aligned} \mathcal{V}_c^k &= \{v | s_{c,v}^{k-1} > \theta_s, \quad v \in \mathcal{R}\} \\ \mathcal{E}_c^k &= \{(u, v) | \ \text{IoU}(u, v) > \theta_{\text{IoU}}, \quad u, v \in \mathcal{R}\}, \end{aligned} \tag{6}$$

where u and v denote object proposals, $s_{c,v}^{k-1}$ is a proposal score predicted in the preceding step, θ_s is a threshold for the score, $\text{IoU}(u, v)$ is intersection-over-union measure between proposals, and θ_{IoU} is an IoU threshold.

Then we perform a mode-seeking algorithm, medoid-shift [3,32], on this graph. The algorithm is useful in practice because it finds multiple reliable modes of data distribution and requires no manual initialization and terminating conditions. Specifically, we first compute the weight of each node $u \in \mathcal{V}$ of G by

$$w_c(u) = \sum_{v \in \mathcal{V}} s_{c,v} \delta(u, v). \tag{7}$$

where $\delta(\cdot, \cdot) = 1$ if there exists an edge $(u, v) \in \mathcal{E}$, and 0 otherwise. Then, medoid-shift algorithm is applied to the graph and identifies a set of modes, where each vertex is associated with one of the modes after convergence. Since such a mode-seeking algorithm often finds spurious modes, we adopt a mode filtering technique, which maintains only salient modes based on topological persistence of a graph [9]. The proposals corresponding to the modes obtained from mode-seeking and mode filtering procedures receive positive labels. The entire procedure of the proposed method is summarized in Algorithm 2.

After finding the modes, the rest of proposals r for a class c are given a pseudo-label $z_{c,r}$ as follows:

$$z_{c,r} = \begin{cases} 1 & \text{if IoU}(m, r) > 0.5, \ m \in \mathcal{M}_c \text{ and } y_c = 1 \\ 0 & \text{otherwise} \end{cases}, \tag{8}$$

where $\mathcal{M}_c$ is a set of detected modes for class c and y_c denotes image-level binary class label for class c. In other words, proposals sufficiently overlapped with any of detected modes are labeled to be positive and the rest are given negative labels

in a similar way to OICR [39]. This labeling method is employed to compute a loss in Eq. (4) for all refinement steps.

5 Experiments

This section presents performance of our regularization algorithm with graph-based labeling. We also compare the proposed approach with the state-of-the-art methods and show results from ablation study of our technique.

5.1 Implementation Details

We use VGG_M and VGG16 [34] networks pretrained on ImageNet [6] classification task to obtain image representation. To compute feature descriptors of all proposals at once, the last max-pooling layer is replaced by a spatial pyramid pooling (SPP) layer as in Fast-RCNN [12]. For training, we employ the standard stochastic gradient descent (SGD) method with batch size 2. The model is trained with 50K iterations in each round, where the learning rate of the first 40K iterations is set to 0.001 and then decreased to 0.0001 for the last 10K iterations. Initial momentum and weight decay are set to 0.9 and 0.0005, respectively. Every image is rescaled to the sizes that the length of the shorter side becomes one of {480, 576, 688, 864, 1200} while we preserve aspect ratios. Approximately 2,000 object proposals are generated for each image by applying selective search algorithm [40] in fast mode. We set the score threshold θ_s to the half of the maximum proposal scores for each class c. Our algorithm runs 4 rounds of iterative training procedure with parameter reinitialization while the number of refinement steps is set to 3, *i.e.*, $K = 3$. Our experiments run on a NVIDIA GTX Titan Xp GPU and the implementation is based on the Caffe [16] framework.

5.2 Datasets and Evaluation Metrics

We evaluate our method on PASCAL VOC 2007 [11] and 2012 [10] datasets, which consist of a total of 9,963 and 22,531 images from 20 object classes. We train our model on train+validation splits of PASCAL VOC 2007 and 2012 datasets, consisting of 5,011 and 11,540 images, respectively. Since our approach lies on weakly supervised setting, only image-level annotations for class labels are used for training. For testing, we utilize 4,952 and 10,991 test images from PASCAL VOC 2007 and 2012 datasets, respectively. All ablation studies are performed on PASCAL VOC 2007 dataset.

Our quantitative evaluation metric is the mean of Average Precisions (APs) over classes. The number of true positives is the count of object proposals that have more than 0.5 IoU overlap with ground-truths. We also measure Correct localization (CorLoc) to evaluate localization accuracy of our model on the training set. The final inference is given by averaging scores from all the refinement steps. Before evaluating and measuring AP and CorLoc scores, non-maximum suppression is applied to positive examples with 0.3 IoU threshold.

Table 1. Comparison between network refinement with and without layer reinitialization. We test VGG_M and VGG16 networks with several different numbers of refinement layers on VOC 2007 test set. We report accuracy in terms of mAP (%). RL means refinement layer in the table.

Methods (round/iterations)	(a) With layer reinitialization		(b) Without layer reinitialization	
	(R1/50k)	(R2/50k)	(–/50k)	(–/100k)
Ours-1RL-VGG_M	35.6	36.2	35.6	35.6
Ours-2RL-VGG_M	36.3	37.6	36.3	34.8
Ours-3RL-VGG_M	38.0	39.2	38.0	38.7
Ours-1RL-VGG16	36.2	40.2	36.2	37.4
Ours-2RL-VGG16	41.6	43.9	41.6	42.1
Ours-3RL-VGG16	42.6	44.6	42.6	42.2

Table 2. Comparison of two labeling methods after training for 50k iterations on VOC 2007 test set: (a) labeling example by propagating positive labels based on overlaps from the bounding box with the maximum classification score, and (b) labeling with multi-modal score distribution given by mode-seeking technique on a graph structure of proposals.

Methods	Base network	RL	mAP
(a) Maximum GT	VGG_M	1	33.5
		2	36.0
		3	36.4
	VGG16	1	35.8
		2	39.1
		3	41.8
(b) Graph-based GT (ours)	VGG_M	1	35.6
		2	36.3
		3	38.0
	VGG16	1	36.2
		2	41.6
		3	42.6

5.3 Ablation Study

Impact of Refinement Layer Reinitialization. We first validate the effectiveness of our refinement layer reinitialization scheme on PASCAL VOC 2007 test set. For the purpose, we compare mAPs of two models—with and without reinitialization of the `fc` layers for refinement—after training for the same number of iterations altogether in both cases, 50k and 100k. Both VGG_M and VGG16 networks are employed as backbone CNNs for this experiment. Table 1 summarizes the results. The performance of the models with reinitialization is

Table 3. Results with different IoU thresholds in Eq. (6) for graph construction. Note that the labels obtained from the graph are integrated into the refinement layer reinitialization. Evaluation is performed with VGG16 network on VOC 2007 test set. We report accuracy in terms of mAP (%).

Methods	θ_{IoU}	Round1	Round2	Round3	Round4	Round5
Ours-1RL	0.1	36.2	40.2	40.8	41.4	41.6
Ours-2RL		41.6	43.9	43.6	43.8	43.1
Ours-3RL		42.6	44.6	44.4	43.4	43.1
Ours-1RL	0.5	36.0	39.5	40.2	40.8	40.5
Ours-2RL		38.0	40.5	41.9	42.5	41.9
Ours-3RL		40.3	41.5	41.6	41.2	41.2
Ours-1RL	0.1 (Round 1, 2)	36.2	40.2	40.4	41.1	40.1
Ours-2RL	0.5 (Round 3, 4, 5)	41.6	43.9	44.5	44.5	43.7
Ours-3RL		42.6	44.6	45.5	45.4	44.0

improved significantly in the second round while the ones without layer reinitialization generally have marginal gains in the second half of the 100k iterations.

Impact of Graph-Based Label Generation. Table 2 illustrates results from two different methods for generation of pseudo ground-truths. The one identifies positive examples from only a single mode corresponding to the bounding box with a maximum score (maximum GT) and the other extract them from multi-modal score distribution over bounding boxes given by the mode-seeking algorithm via medoid-shift on a graph structure (graph-based GT). For this experiment, IoU threshold θ_{IoU} for graph construction is set to 0.1. After one round of training, our graph-based mode-seeking technique outperforms the naïve single GT method on both VGG_M and VGG16 networks consistently.

Impact of Labeling Perturbations. We also investigate influence of label perturbation by varying IoU threshold for edge connectivity of proposal graph. As mentioned in Sect. 4.1, definition of spatial adjacency between vertices affects pseudo ground-truth construction and final label estimation. We test with two IoU thresholds, 0.1 and 0.5. Table 3 presents the results with VGG16 network for the several tested options. The proposed labeling perturbation method works well in general, especially with more refinement steps. Also, when we use a small threshold value at the early stage of training and then increase its value later, detection accuracies are improved compared to the cases with fixed thresholds. It is probably because this strategy is effective to reject noisy examples quickly in the early stages and maintain multiple positive instances in the later ones.

Table 4. AP (%) of all compared algorithms on VOC 2007 test set. Asterisk (*) denotes the method that uses an external detector such as Fast-RCNN or SSD within its framework.

Method	aero	bike	bird	boat	bottle	bus	car	cat	chair	cow	
WSDDN-VGG16 [2]	39.4	50.1	31.5	16.3	12.6	64.5	42.8	42.6	10.1	35.7	
WSDDN+context [18]	57.1	52.0	31.5	7.6	11.5	55.0	53.1	34.1	1.7	33.1	
OICR-VGG16 [39]	58.0	62.4	31.1	19.4	13.0	65.1	62.2	28.4	24.8	44.7	
SelfTaught-VGG16 [17]	52.2	47.1	35.0	26.7	15.4	61.3	66.0	54.3	3.0	53.6	
WCCN-VGG16 [7]	49.5	60.6	38.6	29.2	16.2	70.8	56.9	42.5	10.9	44.1	
SGWSOD-VGG16 [21]	48.4	61.5	33.3	30.0	15.3	72.4	62.4	59.1	10.9	42.3	
OICR-Ens [39]	58.5	63.0	35.1	16.9	17.4	63.2	60.8	34.4	8.2	49.7	
OICR-Ens+FRCNN [39]	65.5	67.2	47.2	21.6	22.1	68.0	68.5	35.9	5.7	63.1	
GAL300-VGG16+SSD* [33]	52.0	60.5	44.6	26.1	20.6	63.1	66.2	65.3	15.0	50.1	
ZLDN-VGG16+FRCNN* [44]	55.4	68.5	50.1	16.8	20.8	62.7	66.8	56.5	2.1	57.8	
OICR-VGG16+FRCNN [39]	60.9	62.9	50.5	28.9	17.1	70.3	68.1	27.0	25.7	58.8	
Ours-3RL-VGG16	62.1	55.7	42.0	31.1	17.2	67.6	65.2	50.8	20.4	51.5	
Ours-3RL-VGG16+FRCNN	59.8	62.8	45.6	33.2	21.8	70.2	68.6	56.6	22.8	55.9	
Method	table	dog	horse	mbike	person	plant	sheep	sofa	train	tv	Avg.
WSDDN-VGG16 [2]	24.9	38.2	34.4	55.6	9.4	14.7	30.2	40.7	54.7	46.9	34.8
WSDDN+context [18]	49.2	42.0	47.3	56.6	15.3	12.8	24.8	48.9	44.4	47.8	36.3
OICR-VGG16 [39]	30.6	25.3	37.8	65.5	15.7	24.1	41.7	46.9	64.3	62.6	41.2
SelfTaught-VGG16 [17]	24.7	43.6	48.4	65.8	6.6	18.8	51.9	43.6	53.6	62.4	41.7
WCCN-VGG16 [7]	29.9	42.2	47.9	64.1	13.8	23.5	45.9	54.1	60.8	54.5	42.8
SGWSOD-VGG16 [21]	34.3	53.1	48.4	65.0	20.5	16.6	40.6	46.5	54.6	55.1	43.5
OICR-Ens [39]	41.0	31.3	51.9	64.8	13.6	23.1	41.6	48.4	58.9	58.7	42.0
OICR-Ens+FRCNN [39]	49.5	30.3	64.7	66.1	13.0	25.6	50.0	57.1	60.2	59.0	47.0
GAL300-VGG16+SSD* [33]	52.8	56.7	21.3	63.4	36.8	22.7	47.9	51.7	68.9	54.1	47.0
ZLDN-VGG16+FRCNN* [44]	47.5	40.1	69.7	68.2	21.6	27.2	53.4	56.1	52.5	58.2	<u>47.6</u>
OICR-VGG16+FRCNN [39]	41.9	20.7	42.4	65.5	7.1	24.6	51.5	61.9	62.7	56.5	45.3
Ours-3RL-VGG16	36.3	34.1	46.2	65.8	12.3	21.9	48.8	55.4	60.2	65.7	45.4
Ours-3RL-VGG16+FRCNN	47.5	40.8	59.0	65.0	9.1	22.4	49.5	64.6	57.8	57.3	**48.8**

5.4 Results on PASCAL VOC Datasets

We compare the proposed algorithm with existing state-of-the art methods for weakly supervised object detection including WSDDN [2], WSDDN+context [18], OICR [39], SelfTaught [17], WCCN [7], SGWSOD [21], ZLDN [44], GAL300 [33]. Tables 4 and 5 present performance of all compared algorithms on PASCAL VOC 2007 dataset in terms of mean of APs and CorLoc, respectively. We also present the performances on PASCAL VOC 2012 dataset in Table 6. Best performance of each measure is marked with bold and second best is marked with underline.

To obtain the final results, we use the models trained for four rounds with refinement layer reinitialization. Our model with 3 refinement layers based on VGG16, which is denoted by Ours-3RL-VGG16 in the table, achieves significantly improved accuracy compared to OICR-VGG16 [39]. This result suggests that our training method is very effective because the two models have the exactly same network architecture. We also train a Fast-RCNN [12] (FRCNN) detector based on the labels of the proposals with the highest scores given by

Table 5. CorLoc (%) of all compared algorithms on VOC 2007 trainval set. Asterisk (*) denotes the method that uses an external detector such as Fast-RCNN or SSD within its framework.

Method	aero	bike	bird	boat	bottle	bus	car	cat	chair	cow	
WSDDN-VGG16 [2]	65.1	58.8	58.5	33.1	39.8	68.3	60.2	59.6	34.8	64.5	
WSDDN+context [18]	83.3	68.6	54.7	23.4	18.3	73.6	74.1	54.1	8.6	65.1	
OICR-VGG16 [39]	81.7	80.4	48.7	49.5	32.8	81.7	85.4	40.1	40.6	79.5	
SelfTaught-VGG16 [17]	72.7	55.3	53.0	27.8	35.2	68.6	81.9	60.7	11.6	71.6	
WCCN-VGG16 [7]	83.9	72.8	64.5	44.1	40.1	65.7	82.5	58.9	33.7	72.5	
SGWSOD-VGG16 [21]	71.0	76.5	54.9	49.7	54.1	78.0	87.4	68.8	32.4	75.2	
OICR-Ens [39]	85.4	78.0	61.6	40.4	38.2	82.2	84.2	46.5	15.2	80.1	
OICR-Ens+FRCNN [39]	85.8	82.7	62.8	45.2	43.5	84.8	87.0	46.8	15.7	82.2	
GAL300-VGG16+SSD* [33]	76.5	76.1	64.2	48.1	52.5	80.7	86.1	73.9	30.8	78.7	
ZLDN-VGG16+FRCNN* [44]	74.0	77.8	65.2	37.0	46.7	75.8	83.7	58.8	17.5	73.1	
OICR-VGG16+FRCNN [39]	86.7	81.2	64.0	50.5	30.9	83.2	85.3	38.7	45.1	80.1	
Ours-3RL-VGG16	85.4	71.4	61.6	55.9	37.0	83.2	84.2	61.3	29.7	77.4	
Ours-3RL-VGG16+FRCNN	86.3	77.6	65.5	55.9	41.6	82.7	86.7	61.6	39.7	80.8	
Method	table	dog	horse	mbike	person	plant	sheep	sofa	train	tv	Avg.
WSDDN-VGG16 [2]	30.5	43.0	56.8	82.4	25.5	41.6	61.5	55.9	65.9	63.7	53.5
WSDDN+context [18]	47.1	59.5	67.0	83.5	35.3	39.9	67.0	49.7	63.5	65.2	55.1
OICR-VGG16 [39]	35.7	33.7	60.5	88.8	21.8	57.9	76.3	59.9	75.3	81.4	60.6
SelfTaught-VGG16 [17]	29.7	54.3	64.3	88.2	22.2	53.7	72.2	52.67	68.9	75.5	56.1
WCCN-VGG16 [7]	25.6	53.7	67.4	77.4	26.8	49.1	68.1	27.9	64.5	55.7	56.7
SGWSOD-VGG16 [21]	29.5	58.0	67.3	84.5	41.5	49.0	78.1	60.3	62.8	78.9	62.9
OICR-Ens [39]	45.2	41.9	73.8	89.6	18.9	56.0	74.2	62.1	73.0	77.4	61.2
OICR-Ens+FRCNN [39]	51.0	45.6	83.7	91.2	22.2	59.7	75.3	65.1	76.8	78.1	64.3
GAL300-VGG16+SSD* [33]	62.0	71.5	46.7	86.1	60.7	47.8	82.3	74.7	83.1	79.3	**68.1**
ZLDN-VGG16+FRCNN* [44]	49.0	51.3	76.7	87.4	30.6	47.8	75.0	62.5	64.8	68.8	61.2
OICR-VGG16+FRCNN [39]	41.4	32.3	67.0	91.2	12.7	60.4	76.3	66.4	80.2	78.9	62.6
Ours-3RL-VGG16	28.1	46.3	66.0	88.0	16.6	51.3	70.1	59.7	73.8	79.2	61.3
Ours-3RL-VGG16+FRCNN	47.5	57.4	82.3	90.8	20.3	55.7	77.3	69.6	74.9	79.2	66.7

Table 6. Comparison between the proposed algorithm and the existing ones on PASCAL VOC 2012 dataset in terms of mAP (%) and CorLoc (%).

Method	mAP (%)	CorLoc (%)
WSDDN+context [18]	34.9	56.1
OICR-VGG16 [39]	37.9	62.1
SelfTaught-VGG16 [17]	38.3	58.8
WCCN-VGG16 [7]	37.9	-
SGWSOD-VGG16 [21]	39.6	62.9
SGWSOD-Ens [21]	40.6	64.2
OICR-Ens [39]	38.2	63.5
OICR-Ens+FRCNN [39]	42.5	65.6
ZLDN-VGG16+FRCNN* [44]	42.9	61.5
GAL300-VGG16+SSD* [33]	43.1	67.2
Ours-3RL-VGG16	41.2	64.1
Ours-3RL-VGG16+FRCNN	**44.1**	**68.5**

our method in individual images. Our final model (Ours-3RL-VGG16+FRCNN) shows higher mAP score than the state-of-the-art methods in both datasets. It is also noticeable that even our method without using FRCNN (Our-3RL-VGG16) outperforms even the ensemble OICR model (OICR-Ens) and the OICR-VGG16-FRCNN method. In terms of CorLoc, we achieve the second best score among the comparison methods on PASCAL VOC 2007 dataset and the top score on 2012 dataset.

Figures 2 and 3 illustrate qualitative examples and failure cases, respectively. Our method is effective in finding more accurate bounding boxes of the objects compared to OICR, but still confused with the objects that have similar appearance and background. Also, detecting highly non-rigid objects (*e.g.* `person`) is still challenging and limited to finding discriminative parts such as human faces.

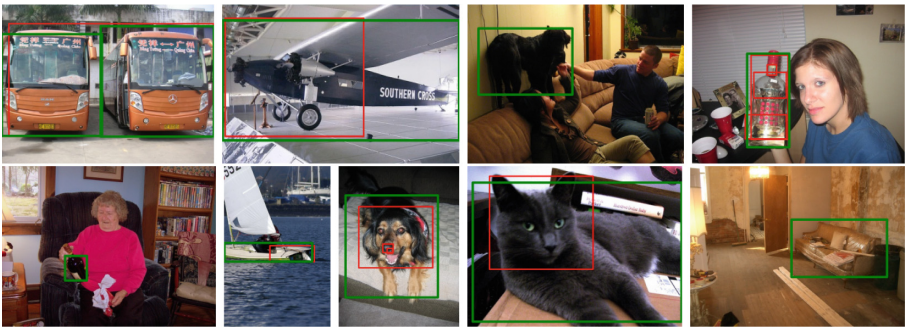

Fig. 2. Qualitative examples on PASCAL VOC 2007 test set. Red boxes indicate detection results from OICR [39] and green ones present our results. (Color figure online)

Fig. 3. Examples of failure cases. Our method is often confused with the objects with similar appearances.

6 Conclusion

We presented simple but effective regularization techniques with a graph-based labeling method for weakly supervised object detection. The proposed regularization algorithms—refinement layer reinitialization and labeling perturbation during iterative training procedure—are helpful to avoid overfitting to local optima

by forgetting biased weights and diversifying pseudo-labels. A mode-seeking algorithm on a graph of object proposals contributes to identifying multiple target instances and improving detection accuracy. Our method illustrates outstanding performances on PASCAL VOC 2007 and 2012 datasets compared to existing state-of-the-art weakly supervised object detection techniques.

Acknowledgements. This research was supported in part by Naver Labs., the Institute for Information & Communications Technology Promotion (IITP) grant [2014-0-00059, 2017-0-01778] and the National Research Foundation of Korea (NRF) grant [NRF-2017R1E1A1A01077999, NRF-2018R1A5A1060031, NRF-2018R1C1B6001223] funded by the Korea government (MSIT).

References

1. Bilen, H., Pedersoli, M., Tuytelaars, T.: Weakly supervised object detection with posterior regularization. In: BMVC (2014)
2. Bilen, H., Vedaldi, A.: Weakly supervised deep detection networks. In: CVPR (2016)
3. Cho, M., Lee, K.M.: Mode-seeking on graphs via random walks. In: CVPR (2012)
4. Cinbis, R.G., Verbeek, J., Schmid, C.: Weakly supervised object localization with multi-fold multiple instance learning. TPAMI **39**, 189–203 (2017)
5. Dai, J., Li, Y., He, K., Sun, J.: R-FCN: object detection via region-based fully convolutional networks. In: NIPS, pp. 379–387 (2016)
6. Deng, J., Dong, W., Socher, R., Li, L.J., Li, K., Fei-Fei, L.: ImageNet: a large-scale hierarchical image database. In: CVPR (2009)
7. Diba, A., Sharma, V., Pazandeh, A., Pirsiavash, H., Van Gool, L.: Weakly supervised cascaded convolutional networks. In: CVPR (2017)
8. Dietterich, T.G., Lathrop, R.H., Lozano-Pérez, T.: Solving the multiple instance problem with axis-parallel rectangles. Artif. Intell. **89**(1), 31–71 (1997)
9. Edelsbrunner, H., Letscher, D., Zomorodian, A.: Topological persistence and simplification. Discrete Comput. Geom. **28**, 511–533 (2002)
10. Everingham, M., Eslami, S.A., Van Gool, L., Williams, C.K., Winn, J., Zisserman, A.: The Pascal visual object classes challenge: a retrospective. IJCV **111**(1), 98–136 (2015)
11. Everingham, M., Van Gool, L., Williams, C.K., Winn, J., Zisserman, A.: The Pascal visual object classes (VOC) challenge. IJCV **88**(2), 303–338 (2010)
12. Girshick, R.: Fast R-CNN. In: ICCV, pp. 1440–1448 (2015)
13. Girshick, R., Donahue, J., Darrell, T., Malik, J.: Region-based convolutional networks for accurate object detection and segmentation. TPAMI **38**(1), 142–158 (2016)
14. Han, B., Sim, J., Adam, H.: Branchout: regularization for online ensemble tracking with convolutional neural networks. In: CVPR (2017)
15. Huang, G., Sun, Y., Liu, Z., Sedra, D., Weinberger, K.Q.: Deep networks with stochastic depth. In: Leibe, B., Matas, J., Sebe, N., Welling, M. (eds.) ECCV 2016. LNCS, vol. 9908, pp. 646–661. Springer, Cham (2016). https://doi.org/10.1007/978-3-319-46493-0_39
16. Jia, Y., et al.: Caffe: Convolutional architecture for fast feature embedding. arXiv preprint arXiv:1408.5093 (2014)

17. Jie, Z., Wei, Y., Jin, X., Feng, J., Liu, W.: Deep self-taught learning for weakly supervised object localization. In: CVPR (2017)
18. Kantorov, V., Oquab, M., Cho, M., Laptev, I.: ContextLocNet: context-aware deep network models for weakly supervised localization. In: Leibe, B., Matas, J., Sebe, N., Welling, M. (eds.) ECCV 2016. LNCS, vol. 9909, pp. 350–365. Springer, Cham (2016). https://doi.org/10.1007/978-3-319-46454-1_22
19. Kingma, D.P., Salimans, T., Welling, M.: Variational dropout and the local reparameterization trick. In: NIPS (2015)
20. Krasin, I., et al.: Openimages: A public dataset for large-scale multi-label and multi-class image classification (2017). Dataset available from https://storage.googleapis.com/openimages/web/index.html
21. Lai, B., Gong, X.: Saliency guided end-to-end learning for weakly supervised object detection. In: IJCAI (2017)
22. Li, D., Huang, J., Li, Y., Wang, S., Yang, M.H.: Weakly supervised object localization with progressive domain adaptation. In: CVPR (2016)
23. Lin, T.-Y., et al.: Microsoft COCO: common objects in context. In: Fleet, D., Pajdla, T., Schiele, B., Tuytelaars, T. (eds.) ECCV 2014. LNCS, vol. 8693, pp. 740–755. Springer, Cham (2014). https://doi.org/10.1007/978-3-319-10602-1_48
24. Liu, W., et al.: SSD: single shot multibox detector. In: Leibe, B., Matas, J., Sebe, N., Welling, M. (eds.) ECCV 2016. LNCS, vol. 9905, pp. 21–37. Springer, Cham (2016). https://doi.org/10.1007/978-3-319-46448-0_2
25. Noh, H., You, T., Mun, J., Han, B.: Regularizing deep neural networks by noise: its interpretation and optimization. In: NIPS (2017)
26. Oquab, M., Bottou, L., Laptev, I., Sivic, J.: Is object localization for free? - weakly-supervised learning with convolutional neural networks. In: CVPR (2015)
27. Redmon, J., Farhadi, A.: Yolo9000: better, faster, stronger. In: CVPR, pp. 6517–6525 (2017)
28. Redmon, J., Divvala, S., Girshick, R., Farhadi, A.: You only look once: unified, real-time object detection. In: CVPR, June 2016
29. Reed, R., Oh, S., Marks, R.: Regularization using jittered training data. In: IJCNN (1992)
30. Ren, S., He, K., Girshick, R., Sun, J.: Faster R-CNN: towards real-time object detection with region proposal networks. In: NIPS (2015)
31. Seo, S., Seo, P.H., Han, B.: Confidence calibration in deep neural networks through stochastic inferences. In: arXiv preprint arXiv:1809.10877 (2018)
32. Sheikh, Y.A., Khan, E.A., Kanade, T.: Mode-seeking by medoidshifts. In: ICCV (2007)
33. Shen, Y., Ji, R., Zhang, S., Zuo, W., Wang, Y.: Generative adversarial learning towards fast weakly supervised detection. In: CVPR (2018)
34. Simonyan, K., Zisserman, A.: Very deep convolutional networks for large-scale image recognition. CoRR abs/1409.1556 (2014)
35. Singh, S., Gupta, A., Efros, A.A.: Unsupervised discovery of mid-level discriminative patches. In: Fitzgibbon, A., Lazebnik, S., Perona, P., Sato, Y., Schmid, C. (eds.) ECCV 2012. LNCS, pp. 73–86. Springer, Heidelberg (2012). https://doi.org/10.1007/978-3-642-33709-3_6
36. Siva, P., Russell, C., Xiang, T.: In defence of negative mining for annotating weakly labelled data. In: Fitzgibbon, A., Lazebnik, S., Perona, P., Sato, Y., Schmid, C. (eds.) ECCV 2012. LNCS, vol. 7574, pp. 594–608. Springer, Heidelberg (2012). https://doi.org/10.1007/978-3-642-33712-3_43
37. Song, H.O., Lee, Y.J., Jegelka, S., Darrell, T.: Weakly-supervised discovery of visual pattern configurations. In: NIPS (2014)

38. Srivastava, N., Hinton, G., Krizhevsky, A., Sutskever, I., Salakhutdinov, R.: Dropout: a simple way to prevent neural networks from overfitting. J. Mach. Learn. Res. **15**(1), 1929–1958 (2014)
39. Tang, P., Wang, X., Bai, X., Liu, W.: Multiple instance detection network with online instance classifier refinement. In: CVPR (2017)
40. Uijlings, J.R.R., van de Sande, K.E.A., Gevers, T., Smeulders, A.W.M.: Selective search for object recognition. IJCV **104**, 154–171 (2013)
41. Wan, F., Wei, P., Jiao, J., Han, Z., Ye, Q.: Min-entropy latent model for weakly supervised object detection. In: CVPR (2018)
42. Wan, L., Zeiler, M., Zhang, S., Le Cun, Y., Fergus, R.: Regularization of neural networks using dropconnect. In: ICML (2013)
43. Xie, L., Wang, J., Wei, Z., Wang, M., Tian, Q.: Disturblabel: regularizing CNN on the loss layer. In: CVPR (2016)
44. Zhang, X., Feng, J., Xiong, H., Tian, Q.: Zigzag learning for weakly supervised object detection. In: CVPR (2018)
45. Zhang, Y., Bai, Y., Ding, M., Li, Y., Ghanem, B.: W2f: a weakly-supervised to fully-supervised framework for object detection. In: CVPR (2018)
46. Zhou, B., Khosla, A., Lapedriza, À., Oliva, A., Torralba, A.: Learning deep features for discriminative localization. In: CVPR (2016)