FI-WSOD: Foreground Information Guided Weakly Supervised Object Detection

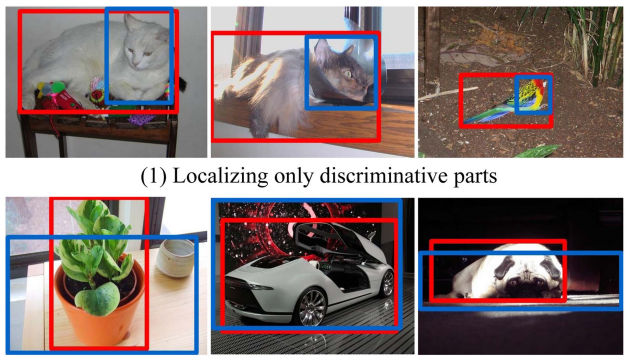
Yufei Yin, Jiajun Deng, *Graduate Student Member, IEEE*, Wengang Zhou, *Senior Member, IEEE*, Li Li, *Member, IEEE*, and Houqiang Li, *Fellow, IEEE*

***Abstract*—Existing solutions for weakly supervised object detection (WSOD) generally follow the multiple instance learning (MIL) paradigm to formulate WSOD as a multi-class classification problem over a set of region proposals. However, without the supervision signal of ground-truth boxes, the training objective of multi-class classification makes the detectors devote main efforts to finding the most common pattern of each class, as the common pattern is always the most discriminative evidence for classification. In addition, although learning from distinguishing multiple foreground classes, the detectors can still ignore to differentiate foreground regions from the background ones, which causes false alarm in prediction. These two points account for the limited localization capability of MIL-based WSOD methods. To this end, we propose foreground information guided WSOD (FI-WSOD), a novel framework that introduces an extra foreground-background binary classification (F-BBC) sub-task to the original MIL-based WSOD paradigm. At the training stage, the involvement of F-BBC task not only improves the feature representation of the network, but also provides extra information from the foreground-background perspective. By leveraging the learnt foreground information, a Foreground Guided Self-Training (FGST) module is further proposed to filter out noisy samples, and to mine representative seeds from the remaining proposals. Moreover, a Multi-Seed Training strategy is performed to reduce the impact of noisy labels when training the self-training networks in FGST. We have conducted extensive experiments on the prevalent Pascal VOC 2007, Pascal VOC 2012 and MSCOCO datasets, and report a series of state-of-the-art records achieved by our proposed framework.**

***Index Terms*—Object detection, weakly supervised learning.**

I. Introduction

OBJECT detection aims to spatially localize objects and identify their categories. The recent success of object detection [1], [2], [3], [4], [5], [6], [7], [8], [9] heavily relies on the box-level annotations with a fully supervised setting. However, it is laborious to collect such well-annotated data, which is not desired in real applications. Such facts motivate the exploration of weakly supervised object detection (WSOD), which aims to train an object detector with image-level labels [10], [11], [12], [13], [14], [15], [16], [17].



(1) Localizing only discriminative parts

(2) Containing background

Fig. 1. Two typical issues of to-date weakly supervised object detectors: (1) locating at the most discriminative part of an object, instead of the whole instance; (2) containing background noises in the predicted bounding boxes. The blue boxes represent the detection results, and the red boxes denote the ground-truth boxes.

The existing approaches in the literature generally formulate WSOD as a Multiple Instance Learning (MIL) problem. The early work WSDDN [12] combines MIL with a CNN model following a two-stream structure. OICR [10] extends WSDDN with cascaded online self-training module to facilitate pseudo label generation. To further boost the performance of weakly supervised object detectors, many recent works have been proposed to improve the self-training procedure [11], [17], [18], [19], the MIL strategy [14], [20], and feature representation learning [16], [17]. However, despite the non-trivial improvement, there are typically two issues repeatedly shown with different kinds of approaches shown in Fig. 1: (1) the predicted bounding box tends to locate at the most discriminative part of an object, instead of the whole instance; (2) some parts of the predicted bounding box are filled with background noises.

We conduct an in-depth analysis to discover the causes behind the above phenomenon. Particularly, we find that the methods in the literature are similar in that they are barely developed with multi-class classification (M-CC), which exactly accounts for these two typical issues. Specifically, on the one hand, M-CC inclines to find the similarities within each category, which tends to mislead the detector to localize the most discriminative part of an object, instead of the whole entity

Manuscript received 15 August 2021; revised 15 June 2022; accepted 25 July 2022. Date of publication 10 August 2022; date of current version 7 June 2023. The guest editor coordinating the review of this manuscript and approving it for publication was Mr. Dingwen Zhang. *(Corresponding authors: Wengang Zhou; Houqiang Li.)*

The authors are with the CAS Key Laboratory of Technology in Geo-Spatial Information Processing and Application System, Department of Electronic Engineering and Information Science, University of Science and Technology of China, Hefei 230026, China (e-mail: yinyufei@mail.ustc.edu.cn; dengjj@mail.ustc.edu.cn; zhwg@ustc.edu.cn; lil1@ustc.edu.cn; lihq@ustc.edu.cn).

Digital Object Identifier 10.1109/TMM.2022.3198018

(shown in the top row of Fig. 1). On the other hand, M-CC mainly focuses on mining the inter-class diversity among foreground categories, but not excels at distinguishing differences between the foreground and background, making the detectors failed to filter out the boxes with plenty of background components (shown in the bottom row of Fig. 1). These two factors together affect the localization capability of the previous methods.

In this work, to alleviate these issues, we propose a novel Foreground Information Guided Weakly Supervised Object Detection (FI-WSOD) framework. FI-WSOD introduces an online Foreground-Background Binary Classification (F-BBC) task to WSOD. F-BBC has played a significant role in two-stage fully supervised object detectors [2], [3] to filter out region proposals without foreground objects, while it is seldom explored in WSOD. However, we find that F-BBC has benefits in WSOD as well. Concretely, it can effectively mitigate these typical issues in two aspects: (1) Compared to M-CC, F-BBC inclines to find the similarities within foreground categories rather than within each foreground category, and thus will facilitate the detection of more complete objects. (2) In contrast to M-CC, F-BBC guides the network to discriminate foreground features from the background ones, thus helping the detector to distinguish the incorrect detections with background parts.

By consolidating the idea of leveraging foreground information from F-BBC to improve the detection capability, we design a Foreground Guided Self-Training (FGST) module to refine the detection results in an online fashion. In this module, we first obtain the foreground scores from the F-BBC task, and filter out negative instances with scores below a soft threshold. Next, with the remaining instances, the Foreground Guided Seeds Mining (FGSM) algorithm takes the information from both M-CC and F-BBC as input, and applies an iterative strategy to mine multiple positive seeds. Then, pseudo labels, *i.e.*, pseudo ground-truth boxes, are generated according to these credible seeds. Finally, when updating the FGST module, our FI-WSOD employs a Multi-Seed Training (MST) strategy to largely reduces the effect of noisy pseudo labels. During inference, scores from the F-BBC network and the FGST module are combined together according to a carefully-designed algorithm to produce the final results.

Some recent works [13], [15], [21] also exploit foreground score to facilitate object detection. Compared to these algorithms, we jointly optimize F-BBC task and WSOD task in an end-to-end manner. Thus, we can obtain foreground scores without offline hand-craft algorithms [15], [21] or models optimized on other datasets [13], [21]. Moreover, the capability of our model to extract feature representation can also be improved by jointly optimizing multiple tasks.

In summary, we make three-fold contributions:

- We propose a unified framework FI-WSOD, in which we introduce the F-BBC task to the WSOD and design a simple yet effective F-BBC network to achieve the task.
- Utilizing the F-BBC information, we apply a Foreground Guided Self-Training (FGST) module to mine more accurate instances online and improve the detection capability of the trained CNN classifier to a great extent.
- We conduct extensive experiments on the benchmark datasets, *i.e.* Pascal VOC 2007 & 2012 and MSCOCO, to validate the merits of our method. Experimental results demonstrate the effectiveness of our proposed method.

II. Related Work

In this section, we briefly review the related methods including fully supervised object detection, weakly supervised object detection and foreground-background classification.

A. Fully Supervised Object Detection

Object Detection is a fundamental computer vision task, whose purpose is to locate and classify desired objects from images. With the advent of deep learning, many high-quality CNN-based object detectors have been proposed for fully supervised object detection. In general, these detectors can be divided into two categories. One is two-stage detector, such as Fast R-CNN [1], Faster R-CNN [2] and FPN [3]. They first generate a number of high-quality foreground proposals, and then predict their categories and refine their locations. The other is one-stage detector, such as YOLO [6], SSD [5] and RetinaNet [7]. They directly predict the categories and locations of anchor boxes. Besides, anchor-free detectors [8], [9], [22] have been proposed in recent years to avoid the artificial design of the anchor boxes. Furthermore, DETR [23] utilizes transformers and discards some hand-crafted post-processing procedures (*e.g.*, NMS) to make the detector end-to-end. However, all these methods rely on fine-grained box-level annotations, which usually cost a lot of labor.

B. Weakly Supervised Object Detection

To alleviate the cost to collect well-annotated data, how to use less data to achieve the original task [10], [12], [24], [25], [26], [27] have received great attention.

In the object detection area, weakly supervised object detection (WSOD) is of great interest in recent years [10], [11], [12], [13], [14], [15], [16], [17], [18], [19], [20], [21], [28], [29], [30], [31], [32]. Compared with the fully supervised one, it only needs image-level annotations, which can be obtained easily. Many recent methods formulate WSOD as a Multiple Instance Learning (MIL) problem [10], [11], [16]. These methods regard the set of object proposals in each image as a bag, and train the weakly supervised detector under MIL constraints to approach the object detection task. As one of the most popular basic structure for weakly supervised object detection, WSDDN [12] first applies MIL with a CNN model to accomplish the WSOD task. It first utilizes Selective Search [33] to generate thousands of proposals and uses RoI pooling to obtain their feature vectors. Then, WSDDN [12] feeds these vectors into two sub-branches and combines their outputs to generate the proposal-level classification scores. Finally, WSDDN sums the scores for all proposals of each class to obtain the image-level scores, and utilizes cross entropy loss to train the whole model.

Based on WSDDN [12], OICR [10] adds several classification branches to refine the scores online. For each refinement branch,

OICR uses the top-scoring proposal as the pseudo ground-truth box to train the next branch. To find more credible boxes for online refinement, PCL [11] constructs a spatial graph between the top-scoring proposals, WSOD^2 [15] combines both bottom-up and top-down features, and OIM [18] introduces an extra appearance graph. To refine the candidate boxes from Selective Search [15], some works [15], [16], [28] add a regression branch to regress the initial boxes. To detect more complete objects, C-MIDN [14] and P-MIDN [20] add extra Multiple Instance Detection Network branch(es) for supplement and MIST [17] applies DropBlock to drop the most discriminative parts. Otherwise, OCRepr [21] and PG-PS [30] utilize weakly supervised semantic segmentation methods for assistance to obtain reliable proposal scores [21] or generate high-quality proposals [30]. PSLR [29] uses graph convolutional networks to identify the object location and semantic existence.

C. Foreground-Background Binary Classification

Foreground-background binary classification (F-BBC) is widely used in various tasks. At the pixel level, the task of salient object detection [34], [35] is to generate a saliency map for each image, where distinguishing the salient pixels can be seen as an F-BBC for each pixel. At the box level, F-BBC plays an important role in many two-stage fully-supervised object detection approaches [2], [3]. For example, in Faster R-CNN [2], the Region Proposal Network acts as an F-BBC network, which is utilized to select foreground proposals and filter unnecessary background proposals.

In the WSOD task, recent works [13], [15], [21] introduce some similar concepts (*e.g.*, objectness score), but these scores are obtained either from the hand-craft features [15] or from the results of other offline pretrained task [13], [21]. In this work, we instead exhibit how to generate foreground score online without additional information or offline processes, and we are the first to illustrate the importance of the F-BBC task for the WSOD task and to jointly optimize them in an end-to-end manner.

III. Proposed Method

In this section, we introduce our FI-WSOD framework for the WSOD task, which consists of four major components: a Multiple Instance Learning (MIL) branch (Section III-B), a Foreground-Background Binary Classification (F-BBC) network (Section III-C), Foreground Guided Self-Training (FGST) modules (Section III-D), and the detection branch (Section III-E).

A. Framework Overview

The overall framework is illustrated in Fig. 2. For each input image I, we first utilize Selective Search [33] to generate a set of proposals. Features of all proposals are then extracted through a ConvNet pretrained on ImageNet [36] and RoI Pooling, followed by two fully connected layers. During the training phase, these features are first fed into the MIL branch to obtain the initial classification scores for each proposal. Next, we utilize these scores to train the F-BBC network online. Then, in each Foreground Guided Self-Training (FGST) module, the classification scores and the F-BBC results are integrated to mine a set of accurate instances. These instances are then used to train a multi-class classifier in this module. Finally, the results from the last FGST module are fed into the detection branch for further refinement.

B. Multiple Instance Learning Branch

In the WSOD task, it is hard to distinguish the positive samples from the negative ones with no box-level annotations are available. To tackle this problem, many existing works [12], [37], [38] apply Multiple Instance Learning (MIL) with a CNN model to accomplish the detection task. In this paper, we choose WSDDN [12] as our base detector to generate the initial detection results.

Specifically, given an image I, we denote its image-level label as $Y = [y_1, y_2, \ldots, y_C] \in \mathbb{R}^{C\times 1}$, where $y_c = 1$ or 0 indicates the presence or absence of the class c.

A set of proposals $R = \{R_1, R_2, \ldots, R_N\}$ for image I is first generated by Selective Search [33]. Features of these proposals are then extracted through a CNN backbone and an RoI Pooling layer, followed by two fully connected layers. Then, the proposal features are fed into two sub-branches, *i.e.*, classification branch and detection branch. In the classification branch, the score matrix $x^{\text{cls}} \in \mathbb{R}^{C\times|R|}$ is first obtained through a fully connected layer, where C denotes the number of object categories and $|R|$ denotes the number of proposals. Then, a softmax function is applied on x^{cls} along the categories to produce $\sigma_{\text{cls}}(x^{\text{cls}})$. Similarly, in the detection branch, the score matrix $x^{\text{det}} \in \mathbb{R}^{C\times|R|}$ is obtained through another fully connected layer. A softmax function is then performed on x^{det} along the proposals to produce $\sigma_{\text{det}}(x^{\text{det}})$. The two softmax operations are defined as follows:

$$\begin{cases} \left[\sigma_{\text{cls}}\left(x^{\text{cls}}\right)\right]_{ij} = \frac{e^{x^{\text{cls}}_{ij}}}{\sum_{k=1}^{C} e^{x^{\text{cls}}_{kj}}}, \\ \left[\sigma_{\text{det}}\left(x^{\text{det}}\right)\right]_{ij} = \frac{e^{x^{\text{det}}_{ij}}}{\sum_{k=1}^{|R|} e^{x^{\text{det}}_{ik}}}. \end{cases} \tag{1}$$

After that, the score of each proposal is formed as the element-wise product of the results from these two branches: $x^{\text{box}} = \sigma_{\text{cls}}(x^{\text{cls}}) \odot \sigma_{\text{det}}(x^{\text{det}})$. Finally, considering only image-level labels are available, x^{box} is aggregated over the proposal dimensions to obtain the image-level scores: $x_c^{\text{img}} = \sum_{i=1}^{|R|} x_{c,i}^{\text{box}}$. In this way, we are able to use the image-level label Y as supervision to optimize the MIL branch using binary cross-entropy loss, which is shown in (2).

$$\mathcal{L}_{mil} = -\sum_{c=1}^{C} \left[y_c log x_c^{\text{img}} + (1 - y_c)\, log\left(1 - x_c^{\text{img}}\right)\right]. \tag{2}$$

C. Foreground-Background Binary Classification Network

The MIL-based detector applies multi-class classification on proposals, focusing mainly on the differences among the foreground categories. However, as illustrated in Section I, foreground-background binary classification (F-BBC)

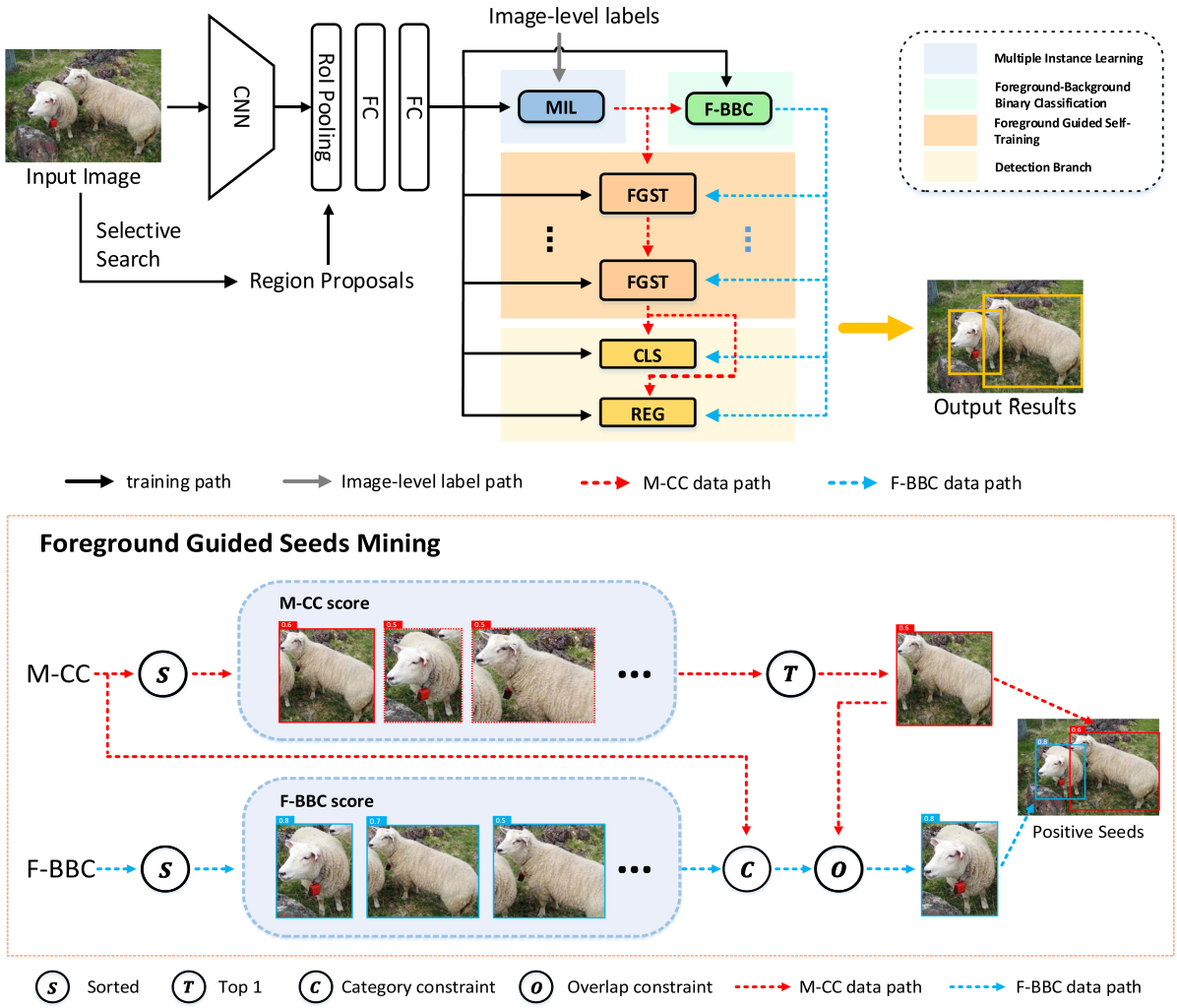


Fig. 2. Overall framework of our proposed method. Proposal features obtained from FC layers are first fed into the MIL branch to obtain the initial classification scores for each proposal. Next, these scores are utilized to train the F-BBC network online. Then, in each FGST module, Foreground Guided Seeds Mining (FGSM) algorithm integrates the F-BBC scores and the M-CC scores from the last branch to mine a set of positive seeds. These positive seeds are used to train a multi-class classifier in FGST. Finally, the results from the last FGST module are fed into the detection branch for further refinement.

also shows its benefits for the WSOD task. To this end, we integrate the F-BBC task into the WSOD framework and design a simple yet effective network to achieve it.

Given the feature $f_i \in \mathbb{R}^D$ of a proposal, its foreground score can be predicted through a binary classifier to apply foreground-background binary classification and a sigmoid function to limit the range of its value:

$$x_i^{\text{fg}} = \sigma(BC(f)), \quad x_i^{\text{fg}} \in [0, 1], \tag{3}$$

where BC represents the binary classifier and σ represents the sigmoid function.

Inspired by OICR [10], we utilize the MIL results to generate the labels for training the F-BBC network. Specifically, suppose the label of proposal j is $y_j^{\text{fg}} \in \mathbb{R}$, for each existing class c (*i.e.*, $y_c = 1$), we first select the top-scoring proposal j_{max}^c, and label it as foreground: $y_{j_{max}^c}^{\text{fg}} = 1$. Then we label other proposals according to their overlaps with these top-scoring proposals j_{max}. For each proposal j, we calculate its maximum overlaps I_j with j_{max}. If $I_j \geq 0.5$, we label it as foreground ($y_j^{\text{fg}} = 1$), and if $0.1 \leq I_j < 0.5$, we label the proposal as background ($y_j^{\text{fg}} = 0$). The remaining proposals are ignored during the training phase. Finally we optimize the F-BBC network utilizing a weighted cross-entropy loss,

$$\mathcal{L}_{F-BBC} = -\frac{1}{|R|}\sum_{i=1}^{|R|} w_i y_i^{\text{fg}} log x_i^{\text{fg}}, \tag{4}$$

where $|R|$ represents the number of proposals, and w_i is the loss weight of the proposal i. Following OICR [10], we apply the classification score of its corresponding top-scoring proposal as w_i.

In this paper, we simply construct the binary classifier with a fully connected layer to maintain structural similarity with subsequent networks. A more complex network could be designed to better implement the F-BBC task, but its not the main focus of this paper. Despite its simplicity, we find that the proposed

F-BBC network is effective enough in providing useful foreground information for the subsequent parts of our framework.

D. Foreground Guided Self-Training

Since the detection results of MIL branch tend to surround the most discriminative parts rather than covering entire instances, we apply a self-training strategy [10], [11], [17] to improve the detection performance. Self-training will benefit the original weakly-supervised detector since the process can be regarded as a teacher-student distillation, which is beneficial for improving the representation of the student model.

To succeed in self-training, two key issues needs to be properly addressed: *(1) How to generate high-quality pseudo labels for self-training? (2) How to effectively train the self-training network?* To this end, we propose a Foreground Guided Self-Training (FGST) module to accomplish the self-training task. In FGST, a Foreground Guided Seeds Mining (FGSM) algorithm and a Multi-Seed Training (MST) strategy are introduced to solve these two key issues. We will explain them in the following subsections.

1) Foreground Guided Seeds Mining: The regular process of pseudo labels generation can be divided into two parts, *i.e.*, positive seeds mining and proposals labeling, where the former is the most critical due to the absence of box-level annotations. Many previous works [10], [11], [16] directly select the high-scoring proposals of the last branch as positive seeds. However, this selection criteria utilizes only information from a single perspective (*i.e.*, M-CC), which may lead to many inaccurate seeds being selected. In the contrast, our proposed F-BBC network provides information from a different perspective, which focuses on distinguishing the foreground proposals from the negative ones. To this end, we purpose to combine information from F-BBC and M-CC to select more diverse yet representative seeds.

Specifically, given an image I and a set of region proposals $R = \{R_1, R_2, \ldots, R_N\}$, their M-CC scores $x^{\text{mc}} \in \mathbb{R}^{C\times|R|}$ and F-BBC scores $x^{\text{fg}} \in \mathbb{R}^{1\times|R|}$ can be obtained from the MIL branch and the F-BBC network, respectively.

First, we aim to narrow the search range of the positive seeds. It's a natural way to set a threshold on the F-BBC score to achieve it, since the F-BBC score represents how likely a proposal belongs to an object. Considering that the distributions of scores will change among different iterations or images, we choose to set a soft threshold according to the number of proposals. Specifically, we rank all the object proposals in descending order by the F-BBC score and select the top $p\%$ ones R^t as candidate positive proposals. In this way, large numbers of negative proposals are filtered out.

Then, we apply a Foreground Guided Seeds Mining (FGSM) algorithm to mine credible seeds from these remained top-ranking proposals R^t. In the FGSM algorithm, we take information both from M-CC and F-BBC into consideration and apply an iterative strategy to mine credible seeds. Concretely, for each existing class c, we first select the proposal $R_c^{\text{mc}} \in R^t$ which has the highest M-CC score and regard it as an initial seed. As mentioned before, F-BBC score reflects the possibility of a proposal being an object, hence we utilize it as the main standard for selecting positive seeds. We sort the remaining proposals in descending order by their F-BBC scores and pick them in turn.

Algorithm 1: Foreground Guided Seeds Mining.

Input: Filtered proposals $R^t = \{R_1^t, R_2^t, \ldots, R_{N'}^t\}$; image labels $Y = [y_1, y_2, \ldots, y_C] \in \mathbb{R}^{C\times 1}$; M-CC scores $x^{\text{mc}} \in \mathbb{R}^{C\times|R^t|}$; F-BBC scores $x^{\text{fg}} \in \mathbb{R}^{1\times|R^t|}$

Output: Positive seed set R_{seed} and their labels Y_{seed}

/*Select initial seed from M-CC results*/

for $cin[1, 2, \ldots, C]$ **do**
  if $y_c == 1$ **then**
    $R_c^{\text{mc}} = \arg\max_j x_{c,j}^{\text{mc}}$;
    $R_{\text{seed}} \leftarrow R_c^{\text{mc}},\ Y_{\text{seed}} \leftarrow c$;

/*Select more credible seeds combining F-BBC and M-CC results*/

$R_s^t \leftarrow SORT(x^{\text{fg}})$ //sort by F-BBC score
for i *in* $[1, 2, \ldots, |R^t|]$ **do**
  $r \leftarrow R_s^t(i)$;
  $C_r = \arg\max_c x_{c,r}^{\text{mc}}$;
  if $y_{C_r} == 1$ **then** //category constraint
    $IoU_r \leftarrow \max(IoU(r, R_{\text{seed}}))$;
    if $IoU_r < 0.5$ **then** //overlap constraint
      $R_{\text{seed}} \leftarrow r,\ Y_{\text{seed}} \leftarrow C_r$;
    else
      break;

We select a proposal j as a positive seed following two rules: **(1) It should belong to an existing class of this image.** Since the F-BBC mainly focuses on distinguishing the foreground and background instances, sometimes it may pick proposals containing objects that do not belong to any existing class. In contrast, M-CC can tell the probability of each proposal belonging to a certain class, hence we apply M-CC information to constrain the selection. **(2) It does not have high overlaps with existing selected seeds.** It's widely recognized that the proposals surrounding a high-scoring proposal are likely to have high scores as well. Therefore, we add this constraint to avoid choosing spatially-similar proposals, which deviates from our original intention to select representative seeds. Finally, a set of credible positive seeds can be collected through the process above. As shown in Fig. 3, seeds selected from F-BBC scores act as a refinement or supplement for the original ones obtained by M-CC scores. The detailed algorithm is described in Algorithm 1.

After obtaining positive seeds using the FGSM algorithm, we continue to generate pseudo labels for all the proposals. Specifically, for each proposal $R_i \in R$, we calculate its IoU with all the selected seeds and obtain its overlap IoU_i with the closest seed. Then, we denote the positive proposals as $R_{\text{pos}} = \{R_i | IoU_i \geq 0.5\}$ and the negative proposals as $R_{\text{neg}} = \{R_i | 0.1 \leq IoU_i < 0.5\}$. Suppose the pseudo label of proposal i is denoted as $Y_i = [y_{1,i}, y_{2,i}, \ldots, y_{C+1,i}]$. For the positive proposals, we label them as the same class as their closest seeds. For the negative proposals, we label them as class $C + 1$, which represents the background category. The remaining proposals are ignored during the training process.

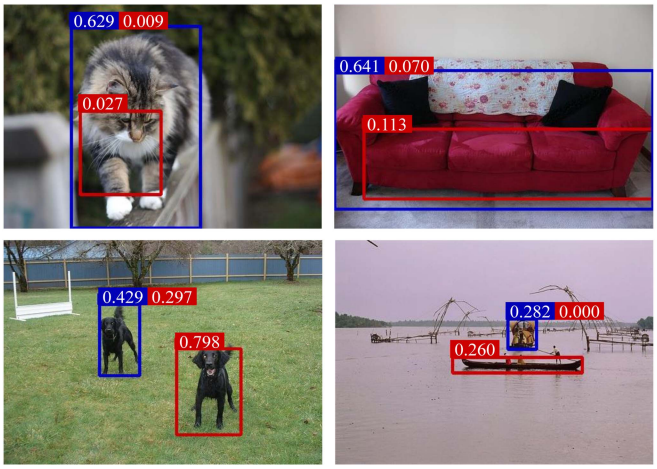


Fig. 3. Seeds selected from FGSM at 25 k iteration. The red boxes are selected from M-CC results while the blue boxes are selected from F-BBC scores. Their M-CC scores and F-BBC scores are marked with red and blue, respectively.

2) Multi-Seed Training: The generated seeds can be broadly regarded as an estimate of the underlying box-level ground truth. Though our proposed FGSM algorithm is able to mine more accurate seeds and leads to significant performance improvements (shown in Section IV), noisy samples are still inevitable. To alleviate this problem, inspired by [39], we apply a Multi-Seed Training (MST) strategy to train the self-training network, which employs different seeds simultaneously during the training phase. MST is well capable of reducing the influence of noisy labels. Suppose two kinds of seeds are given, we can obtain two sets of proposal labels, denoted by Y_1, Y_2, according to them. If a proposal is assigned to the same label in both Y_1 and Y_2, the label is more likely to be correct, since the probability of generating a correct pseudo-label is higher than that of generating a wrong one. In contrast, if the proposal labels are different in the two sets, the two labels will provide gradients of different directions, thus weakening the influence of each other when updating the network.

Here, besides the seeds from the proposed FGSM algorithm, we employ the top-scoring strategy [10], [16] to generate another kind of seeds. Concretely, for each existing class c, we simply choose the proposal with the highest M-CC score as the positive seed. Similarly, we obtain another set of pseudo labels using these seeds.

The self-training network consists of a fully connected layer and a softmax operation along the categories. Given the proposal features, the network outputs the classification score of each proposal $x^{\text{st}} \in \mathbb{R}^{(C+1)\times|R|}$. For each label set, we utilize the weighted cross-entropy loss, which is similar with $\mathcal{L}_{BBC}$:

$$\mathcal{L}_{st} = -\frac{1}{|R|}\sum_{i=1}^{|R|}\sum_{c=1}^{C+1} w_i y_{c,i} log x^{\text{st}}_{c,i}. \tag{5}$$

Finally, we combine the losses calculated by the two kinds of label sets:

$$\mathcal{L}_{mst} = \mathcal{L}_{st}^{fgsm} + \mathcal{L}_{st}^{top}, \tag{6}$$

where $\mathcal{L}_{st}^{fgsm}$ represents the L_{st} calculated using the first pseudo label set, which is generated based on the seeds from the FGSM algorithm. Similarly, L_{st}^{top} represents the L_{st} calculated using the second pseudo label set, which is generated based on the seeds using the top-scoring strategy.

It is worth mentioning that, we apply only $\mathcal{L}_{st}^{top}$ to train the self-training network at the first $k \times max\_iter$ iterations, since the F-BBC results are not accurate enough at the beginning of the training stage. In the following iterations, we combine the two losses to update the network as in (6).

In our framework, we construct several FGST modules sequentially after the MIL branch. For the first FGST module, the M-CC scores are obtained from the outputs of the MIL branch, and for the remaining ones, the M-CC scores are obtained from their respective previous modules.

E. Detection Branch

Additionally, we add an extra detection branch to further refine the detection results following [16], [40]. The detection branch has two sibling branches, *i.e.*, a classification branch and a regression branch. The classification branch predicts the categories of all the proposals, and we denote its output as $x^{\text{dcls}} \in \mathbb{R}^{(C+1)\times|R|}$. For each object class c, the second sibling branch predicts the offsets of the positions and shapes for proposals $t^c = (t_x^c, t_y^c, t_w^c, t_h^c)$.

Then, we apply the FGSM algorithm to generate the positive seeds utilizing the outputs of the last FGST module, and obtain the pseudo labels for each proposal i, *i.e.*, classification label $u_i = [u_{1,i}, u_{2,i}, \ldots, u_{C+1,i}]$ and regression label $v_i = (v_x, v_y, v_w, v_h)$. Finally, we utilize the weighted cross-entropy loss to train the classification branch (*i.e.*, $\mathcal{L}_{dcls}$), which has a similar formulation with $\mathcal{L}_{st}$. The regression branch is optimized by a weighted smooth-L1 loss, which can be formulated as follows:

$$\mathcal{L}_{reg} = -\frac{1}{|R|}\sum_{i=1}^{|R|}\sum_{c=1}^{C}\mathbb{I}(u_{c,i}=1) w_i \cdot \text{smooth}_{\text{L1}}(t_i^c, v_i). \tag{7}$$

The loss for the detection branch $\mathcal{L}_{det}$ can be obtained by combining $\mathcal{L}_{dcls}$ and $\mathcal{L}_{reg}$.

In summary, the overall network is trained end-to-end by combining all the losses mentioned above:

$$\mathcal{L}_{total} = \mathcal{L}_{mil} + \lambda\mathcal{L}_{F-BBC} + \sum_{t=1}^{T}\mathcal{L}_{mst}^{t} + \mathcal{L}_{det}, \tag{8}$$

where $\mathcal{L}_{mil}$ is the multi-class classification loss for the MIL branch, $\mathcal{L}_{F-BBC}$ is the F-BBC loss, $\mathcal{L}_{mst}^{t}$ represents the multi-seed training loss for the t-th FGST module, and $\mathcal{L}_{det}$ is the loss for the detection branch. λ is set to 3 for all experiments.

F. Inference With F-BBC Results

In contrast to the previous works [10], [11], [14] that average the outputs of all the self-training networks to generate the classification results at the inference stage, we involve the F-BBC prediction in our inference procedure to ameliorate detection. However, F-BBC score only indicates whether the proposals

belong to foreground or background, thus we cannot directly combine it with those multi-class classification scores. To address this problem, we apply the results from the MIL branch for supplement. The MIL branch is usually not considered during inference, since it tends to give higher scores to the proposals only containing the discriminative parts. Nevertheless, though poor in scoring proposals, it can provide the classification information accurately, *i.e.*, which category a proposal belongs to. Therefore, we choose to apply the F-BBC results to update the MIL scores. Specifically, for each proposal r, we first classify it by utilizing the MIL scores through an $argmax$ operation over classes. Then we replace its MIL score for this class with its F-BBC score, and its other MIL scores remain unchanged:

$$x^{\text{mil}}_{C_r,r} = x^{\text{fg}}_{r}, \quad where\ C_r = \arg\max_{c} x^{\text{mil}}_{c,r}. \tag{9}$$

Finally, we combine the updated MIL scores with the outputs from all the T FGST modules and the classification scores in the detection branch to generate the final scores for all proposals, and adjust their positions and sizes according to the regression outputs.

IV. Experiments and Analysis

In this section, we first introduce the datasets and implementation details of our FI-WSOD. Then, we compare our approach with several state-of-the-art methods. Finally, we conduct ablative experiments to demonstrate the effectiveness of each component in our work.

A. Datasets

We evaluate our proposed method on the popular PASCAL VOC 2007, PASCAL VOC 2012 and MSCOCO datasets [41], [42], which contain 20, 20 and 80 object categories, respectively.

For VOC 2007 and 2012, following the common practice, we train on the *trainval* split (5,011 images for VOC 2007 and 11,540 images for VOC 2012), and two kinds of metrics are applied to evaluate the performance: (1) Average Precision (AP) and the mean of AP (mAP) on the *test* split; (2) Correct localization (CorLoc) on the *trainval* split. Both metrics are evaluated under the condition of IoU ≥ 0.5 following the PASCAL criteria.

For MSCOCO, we train on the *train* split (about 80 K images) and use the *val* split (about 40 K images) for testing. For evaluation, we apply two metrics mAP@0.5 and mAP@[.5,.95] following the standard PASCAL criteria and the standard MSCOCO criteria, respectively.

B. Implementation Details

Following a widely-used setting, we adopt VGG16 [43] pretrained on ImageNet [36] as the backbone and utilize Selective Search [33] and MCG [44] to generate proposals for Pascal VOC benchmarks and MSCOCO, respectively. The whole framework is end-to-end optimized using stochastic gradient descent (SGD), with the momentum and weight decay set as 0.9 and 5×10^{-4}, respectively. We set the batch size to 4. The initial learning rate is set as 1×10^{-3} for the first 60 K, 150 K, 240 K iterations, and it is dropped by a factor of 10 for the following 20 K, 40 K, 80 K iterations for VOC 2007, VOC 2012 and MSCOCO, respectively. We set p to 14 and k to 0.3 in the FGST module. T is set to 3 following the previous work. For data augmentation, we random re-scale the short side of each image to one of these five scales, *i.e.*, $\{480, 576, 688, 864, 1200\}$ and apply random horizontal flipping during training. During inference, all these five scales and horizontal flip are applied to each image, and the average score of these 10 augmented images is utilized as the final score. Our experiments are implemented on the deep learning framework of PyTorch [45] and we run all the experiments on an NVIDIA GTX 1080Ti GPU.

TABLE I
Ablation Study of Different Components of Our Method on VOC 2007 Dataset

OICR_{reg}	F-BBC	FGSM	MST	Inf.	mAP (%)
✓					53.0
✓	✓				53.7
✓	✓			✓	54.1
✓	✓	✓			55.3
✓	✓	✓	✓		56.1
✓	✓	✓	✓	✓	**56.4**

C. Comparison With State-of-The-Art Methods

The performances of different models on the VOC 2007 dataset for weakly supervised object detection task are summarized in Tables II and III, where 'FRCNN' denotes re-training a Fast-RCNN detector and 'Ens.' represents model ensemble. In general, our method exhibits better performances than other single models on both evaluation metrics. Our method achieves an mAP of 56.4%, which is to-date the best performance and outperforms the other single model methods by at least 1.5%. Moreover, our single model even surpasses all the previous methods with post-process (*i.e.*, 'FRCNN' and 'Ens.') by at least 0.6%. Besides, in terms of CorLoc, our method obtains 72.4% with a single model, which is also state-of-the-art. We also evaluate our work on the VOC 2012 dataset. Tables IV and V show the mAP and CorLoc results with a single model, which demonstrate the effectiveness of our work as well.

On MSCOCO dataset, as shown in Table VI, our method also achieves the state-of-art performances on both two metrics, and outperforms the best competitor MIST [17] by 1.4% on mAP@0.5.

Table VII shows the comparison of different methods on computational complexities and the time costs on one NVIDIA GTX 1080Ti GPU, where "OIM*" represents the OIM model reproduced by us. Train Time and Test Time represent the time cost per image during training and inference, respectively. Considering that the multi-scaling strategy is widely used in the WSOD methods, we also adopt it when calculating the Test Time. Compared with previous works [11], [18], [40] and our baseline OICR_{reg}, our method achieves better performance with negligible additional computational complexities and comparable time cost on training and inference. We attribute this to the proposed simple

TABLE II
COMPARISON WITH THE STATE-OF-THE-ART METHODS IN TERMS OF MAP (%) ON THE VOC 2007 *TEST* SET

Methods	aero	bike	bird	boat	bottle	bus	car	cat	chair	cow	table	dog	horse	mbike	person	plant	sheep	sofa	train	tv	mAP
WSDDN [12]	39.4	50.1	31.5	16.3	12.6	64.5	42.8	42.6	10.1	35.7	24.9	38.2	34.4	55.6	9.4	14.7	30.2	40.7	54.7	46.9	34.8
ContextLocNet [43]	57.1	52.0	31.5	7.6	11.5	55.0	53.1	34.1	1.7	33.1	49.2	42.0	47.3	56.6	15.3	12.8	24.8	48.9	44.4	47.8	36.3
OICR [10]	58.0	62.4	31.1	19.4	13.0	65.1	62.2	28.4	24.8	44.7	30.6	25.3	37.8	65.5	15.7	24.1	41.7	46.9	64.3	62.6	41.2
Self-taught [49]	52.2	47.1	35.0	26.7	15.4	61.3	66.0	54.3	3.0	53.6	24.7	43.6	48.4	65.8	6.6	18.8	51.9	43.6	53.6	62.4	41.7
WCCN [42]	49.5	60.6	38.6	29.2	16.2	70.8	56.9	42.5	10.9	44.1	29.9	42.2	47.9	64.1	13.8	23.5	45.9	54.1	60.8	54.5	42.8
PCL [11]	54.4	69.0	39.3	19.2	15.7	62.9	64.4	30.0	25.1	52.5	44.4	19.6	39.3	67.7	17.8	22.9	46.6	57.5	58.6	63.0	43.5
TS^{2}C [13]	59.3	57.5	43.7	27.3	13.5	63.9	61.7	59.9	24.1	46.9	36.7	45.6	39.9	62.6	10.3	23.6	41.7	52.4	58.7	56.6	44.3
WSRPN [50]	57.9	70.5	37.8	5.7	21.0	66.1	69.2	59.4	3.4	57.1	57.3	35.2	64.2	68.6	32.8	28.6	50.8	49.5	41.1	30.0	45.3
WS-JDS [51]	52.0	64.5	45.5	26.7	27.9	60.5	47.8	59.7	13.0	50.4	46.4	56.3	49.6	60.7	25.4	28.2	50.0	51.4	66.5	29.7	45.6
MELM [52]	55.6	66.9	34.2	29.1	16.4	68.8	68.1	43.0	25.0	65.6	45.3	53.2	49.6	68.6	2.0	25.4	52.5	56.8	62.1	57.1	47.3
SDCN [53]	59.8	67.1	32.0	34.7	22.8	67.1	63.8	67.9	22.5	48.9	47.8	60.5	51.7	65.2	11.8	20.6	42.1	54.7	60.8	64.3	48.3
OIM [18]	55.6	67.0	45.8	27.9	21.1	69.0	68.3	70.5	21.3	60.2	40.3	54.5	56.5	70.1	12.5	25.0	52.9	55.2	65.0	63.7	50.1
C-MIL [54]	62.5	58.4	49.5	32.1	19.8	70.5	66.1	63.4	20.0	60.5	52.9	53.5	57.4	68.9	8.4	24.6	51.8	58.7	66.7	63.5	50.5
OCRepr [21]	59.4	66.4	45.8	21.5	22.1	70.1	67.3	66.1	24.2	58.8	48.5	60.5	62.4	66.7	17.9	26.0	47.5	57.5	60.5	63.5	50.6
PG-PS [35]	63.0	64.4	50.1	27.5	17.1	70.6	66.0	71.1	25.8	55.9	43.2	62.7	65.9	64.1	10.2	22.5	48.1	53.8	72.2	67.4	51.1
PSLR [36]	62.2	61.1	51.1	33.8	18.0	66.7	66.5	65.0	18.5	59.4	44.8	60.9	65.6	66.9	24.7	26.0	51.0	53.2	66.0	62.2	51.2
Yang *et al.* [16]	57.6	70.8	50.7	28.3	27.2	72.5	69.1	65.0	26.9	64.5	47.4	47.7	53.5	66.9	13.7	29.3	56.0	54.9	63.4	65.2	51.5
C-MIDN [14]	53.3	71.5	49.8	26.1	20.3	70.3	69.9	68.3	28.7	65.3	45.1	64.6	58.0	71.2	20.0	27.5	54.9	54.9	69.4	63.5	52.6
Pred Net [55]	66.7	69.5	52.8	31.4	24.7	74.5	74.1	67.3	14.6	53.0	46.1	52.9	69.9	70.8	18.5	28.4	54.6	60.7	67.1	60.4	52.9
WSOD^2 [15]	65.1	64.8	57.2	39.2	24.3	69.8	66.2	61.0	29.8	64.6	42.5	60.1	71.2	70.7	21.9	28.1	58.6	59.7	52.2	64.8	53.6
SLV [19]	65.6	71.4	49.0	37.1	24.6	69.6	70.3	70.6	30.8	63.1	36.0	61.4	65.3	68.4	12.4	29.9	52.4	60.0	67.6	64.5	53.5
P-MIDN [20]	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	53.9
IM-CFB [45]	64.1	74.6	44.7	29.4	26.9	73.3	72.0	71.2	28.1	66.7	48.1	63.8	55.5	68.3	17.8	27.7	54.4	62.7	70.5	66.6	54.3
MIST [17]	68.8	77.7	57.0	27.7	28.9	69.1	74.5	67.0	32.1	73.2	48.1	45.2	54.4	73.7	35.0	29.3	64.1	53.8	65.3	65.2	**54.9**
OIM [18]+FRCNN	53.4	72.0	51.4	26.0	27.7	69.8	69.7	74.8	21.4	67.1	45.7	63.7	63.7	67.4	10.9	25.3	53.5	60.4	70.8	58.1	52.6
C-MIDN [14] +FRCNN	54.1	74.5	56.9	26.4	22.2	68.7	68.9	74.8	25.2	64.8	46.4	70.3	66.3	67.5	21.6	24.4	53.0	59.7	68.7	58.9	53.6
Pred Net [55]+Ens.	67.7	70.4	52.9	31.3	26.1	75.5	73.7	68.6	14.9	54.0	47.3	53.7	70.8	70.2	19.7	29.2	54.9	61.3	67.6	61.2	53.6
PSLR [36]+FRCNN	62.3	63.1	53.5	42.1	19.0	64.8	68.2	71.0	17.2	64.3	56.0	72.6	67.9	64.1	20.8	23.0	50.3	69.5	65.8	59.5	53.8
SLV [19]+FRCNN	62.1	72.1	54.1	34.5	25.6	66.7	67.4	77.2	24.2	61.6	47.5	71.6	72.0	67.2	12.1	24.6	51.7	61.1	65.3	60.1	53.9
Yang *et al.* [16]+Ens.	59.8	72.8	54.4	35.6	30.2	74.4	70.6	74.5	27.7	68.0	51.7	46.3	63.7	68.6	14.8	27.8	54.9	60.9	65.1	67.4	54.5
OCRepr [21]+Ens.	64.7	71.6	53.7	30.2	27.9	71.3	68.2	73.7	27.6	62.4	50.7	63.9	69.3	67.5	15.5	29.4	53.3	61.4	63.4	66.1	54.6
PG-PS [35]+FRCNN	59.3	66.2	55.4	35.2	22.3	69.7	70.2	73.8	29.4	63.6	47.9	78.1	67.9	68.2	12.2	24.9	43.2	63.7	73.2	66.8	54.6
P-MIDN [20]+FRCNN	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	55.0
IM-CFB [45]+FRCNN	63.3	77.5	48.3	36.0	32.6	70.8	71.9	73.1	29.1	68.7	47.1	69.4	56.6	70.9	22.8	24.8	56.0	59.8	73.2	64.6	55.8
Ours (single)	66.1	75.5	57.1	29.8	29.9	72.0	73.8	79.5	26.9	66.7	49.7	71.6	60.4	68.5	16.0	27.0	59.5	58.3	74.0	66.4	**56.4**

TABLE III
COMPARISON WITH THE STATE-OF-THE-ART METHODS IN TERMS OF CORLOC (%) ON THE VOC 2007 *TRAINVAL* SET

Methods	aero	bike	bird	boat	bottle	bus	car	cat	chair	cow	table	dog	horse	mbike	person	plant	sheep	sofa	train	tv	CorLoc
WSDDN [12]	65.1	58.8	58.5	33.1	39.8	68.3	60.2	59.6	34.8	64.5	30.5	43.0	56.8	82.4	25.5	41.6	61.5	55.9	65.9	63.7	53.5
ContextLocNet [43]	83.3	68.6	54.7	23.4	18.3	73.6	74.1	54.1	8.6	65.1	47.1	59.5	67.0	83.5	35.3	39.9	67.0	49.7	63.5	65.2	55.1
Self-taught [49]	72.7	55.3	53.0	27.8	35.2	68.6	81.9	60.7	11.6	71.6	29.7	54.3	64.3	88.2	22.2	53.7	72.2	52.6	68.9	75.5	56.1
WCCN [42]	83.9	72.8	64.5	44.1	40.1	65.7	82.5	58.9	33.7	72.5	25.6	53.7	67.4	77.4	26.8	49.1	68.1	27.9	64.5	55.7	56.7
OICR [10]	81.7	80.4	48.7	49.5	32.8	81.7	85.4	40.1	40.6	79.5	35.7	33.7	60.5	88.8	21.8	57.9	76.3	59.9	75.3	81.4	60.6
TS^{2}C [13]	84.2	74.1	61.3	52.1	32.1	76.7	82.9	66.6	42.3	70.6	39.5	57.0	61.2	88.4	9.3	54.6	72.2	60.0	65.0	70.3	61.0
MELM [52]	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	61.4
PCL [11]	79.6	85.5	62.2	47.9	37.0	83.8	83.4	43.0	38.3	80.1	50.6	30.9	57.8	90.8	27.0	58.2	75.3	68.5	75.7	78.9	62.7
WSRPN [50]	77.5	81.2	55.3	19.7	44.3	80.2	86.6	69.5	10.1	87.7	68.4	52.1	84.4	91.6	57.4	63.4	77.3	58.1	57.0	53.8	63.8
WS-JDS [51]	82.9	74.0	73.4	47.1	60.9	80.4	77.5	78.8	18.6	70.0	56.7	67.0	64.5	84.0	47.0	50.1	71.9	57.6	83.3	43.5	64.5
C-MIL [54]	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	65.0
OIM [18]	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	67.2
OCRepr [21]	85.4	79.2	65.2	47.9	42.4	84.3	83.3	76.2	37.8	79.5	47.9	71.4	83.7	90.8	25.8	57.9	71.1	64.5	75.3	80.6	67.5
Yang *et al.* [16]	80.0	83.9	74.2	53.2	48.5	82.7	86.2	69.5	39.3	82.9	53.6	61.4	72.4	91.2	22.4	57.5	83.5	64.8	75.7	77.1	68.0
PSLR [36]	86.3	72.9	71.2	59.0	36.3	80.2	84.4	75.6	30.8	83.6	53.2	75.1	82.7	87.1	37.7	54.6	74.2	59.1	79.8	78.9	68.1
SDCN [53]	85.0	83.9	58.9	59.6	43.1	79.7	85.2	77.9	31.3	78.1	50.6	75.6	76.2	88.4	49.7	56.4	73.2	62.6	77.2	79.9	68.6
C-MIDN [14]	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	68.7
MIST [17]	87.5	82.4	76.0	58.0	44.7	82.2	87.5	71.2	49.1	81.5	51.7	53.3	71.4	92.8	38.2	52.8	79.4	61.0	78.3	76.0	68.8
PG-PS [35]	85.4	80.4	69.1	58.0	35.9	82.7	86.7	82.6	45.5	84.9	44.1	80.2	84.0	89.2	12.3	55.7	79.4	63.4	82.1	82.1	69.2
WSOD^2 [15]	87.1	80.0	74.8	60.1	36.6	79.2	83.8	70.6	43.5	88.4	46.0	74.7	87.4	90.8	44.2	52.4	81.4	61.8	67.7	79.9	69.5
P-MIDN [20]	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	69.8
IM-CFB [45]	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	70.7
Pred Net [55]	88.6	86.3	71.8	53.4	51.2	87.6	89.0	65.3	33.2	86.6	58.8	65.9	87.7	93.3	30.9	58.9	83.4	67.8	78.7	80.2	70.9
SLV [19]	84.6	84.3	73.3	58.5	49.2	80.2	87.0	79.4	46.8	83.6	41.8	79.3	88.8	90.4	19.5	59.7	79.4	67.7	82.9	83.2	71.0
Ours	84.6	87.1	69.7	55.3	50.4	86.8	88.3	84.0	47.7	84.2	51.0	84.9	79.9	94.0	22.5	64.5	82.5	60.5	87.5	83.2	**72.4**

TABLE IV
COMPARISON WITH THE STATE-OF-THE-ART METHODS IN TERMS OF MAP (%) ON THE VOC 2012 *TEST* SET

Methods	aero	bike	bird	boat	bottle	bus	car	cat	chair	cow	table	dog	horse	mbike	person	plant	sheep	sofa	train	tv	mAP
ContextLocNet [43]	64.0	54.9	36.4	8.1	12.6	53.1	40.5	28.4	6.6	35.3	34.4	49.1	42.6	62.4	19.8	15.2	27.0	33.1	33.0	50.0	35.3
OICR [10]	67.7	61.2	41.5	25.6	22.2	54.6	49.7	25.4	19.9	47.0	18.1	26.0	38.9	67.7	2.0	22.6	41.1	34.3	37.9	55.3	37.9
Self-taught [49]	60.8	54.2	34.1	14.9	13.1	54.3	53.4	58.6	3.7	53.1	8.3	43.4	49.8	69.2	4.1	17.5	43.8	25.6	55.0	50.1	38.3
WCCN [42]	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	37.9
WS-JDS [51]	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	39.1
PCL [11]	58.2	66.0	41.8	24.8	27.2	55.7	55.2	28.5	16.6	51.0	17.5	28.6	49.7	70.5	7.1	25.7	47.5	36.6	44.1	59.2	40.6
TS^{2}C [13]	67.4	57.0	37.7	23.7	15.2	56.9	49.1	64.8	15.1	39.4	19.3	48.4	44.5	67.2	2.1	23.3	35.1	40.2	46.6	45.8	40.0
WSRPN [50]	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	40.8
MELM [52]	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	42.4
SDCN [53]	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	43.5
PSLR [36]	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	46.3
Yang *et al.* [16]	64.7	66.3	46.8	28.5	28.4	59.8	58.6	70.9	13.8	55.0	15.7	60.5	63.9	69.2	8.7	23.8	44.7	52.7	41.5	62.6	46.8
C-MIL [54]	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	46.7
PG-PS [35]	68.3	60.0	47.4	26.4	20.6	61.5	59.9	82.1	23.7	50.4	20.1	78.8	52.7	67.7	2.6	21.5	43.8	50.1	67.2	60.5	48.3
Pred Net [55]	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	48.4
SLV [19]	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	49.2
IM-CFB [45]	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	49.4
C-MIDN [14]	72.9	68.9	53.9	25.3	29.7	60.9	56.0	78.3	23.0	57.8	25.7	73.0	63.5	73.7	13.1	28.7	51.5	35.0	56.1	57.5	50.2
Ours	73.9	74.5	57.4	32.2	35.1	60.1	55.2	66.1	23.7	61.8	26.2	61.1	62.6	73.3	9.6	25.9	53.0	40.7	66.8	62.1	**51.1**

TABLE V
COMPARISON WITH THE STATE-OF-THE-ART METHODS IN TERMS OF CORLOC (%) ON THE VOC 2012 *TRAINVAL* SET

Methods	aero	bike	bird	boat	bottle	bus	car	cat	chair	cow	table	dog	horse	mbike	person	plant	sheep	sofa	train	tv	CorLoc
ContextLocNet [43]	78.3	70.8	52.5	34.7	36.6	80.0	58.7	38.6	27.7	71.2	32.3	48.7	76.2	77.4	16.0	48.4	69.9	47.5	66.9	62.9	54.8
OICR [10]	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	62.1
Self-taught [49]	82.4	68.1	54.5	38.9	35.9	84.7	73.1	64.8	17.1	78.3	22.5	57.0	70.8	86.6	18.7	49.7	80.7	45.3	70.1	77.3	58.8
PCL [11]	77.2	83.0	62.1	55.0	49.3	83.0	75.8	37.7	43.2	81.6	46.8	42.9	73.3	90.3	21.4	56.7	84.4	55.0	62.9	82.5	63.2
TS^2C [13]	79.1	83.9	64.6	50.6	37.8	87.4	74.0	74.1	40.4	80.6	42.6	53.6	66.5	88.8	18.8	54.9	80.4	60.4	70.7	79.3	64.4
WS-JDS [51]	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	63.5
WSRPN [50]	85.5	60.8	62.5	36.6	53.8	82.1	80.1	48.2	14.9	87.7	68.5	60.7	85.7	89.2	62.9	62.1	87.1	54.0	45.1	70.6	64.9
SDCN [53]	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	67.9
PSLR [36]	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	68.7
Yang *et al.* [16]	82.4	83.7	72.4	57.9	52.9	86.5	78.2	78.6	40.1	86.4	37.9	67.9	87.6	90.5	25.6	53.9	85.0	71.9	66.2	84.7	69.5
C-MIL [54]	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	67.4
PG-PS [35]	85.5	81.1	69.2	54.3	37.6	86.7	81.7	84.0	44.6	83.3	45.8	80.2	84.2	87.2	11.5	52.1	78.9	63.9	81.0	80.9	68.7
Pred Net [55]	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	69.5
SLV [19]	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	69.2
IM-CFB [45]	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	69.6
C-MIDN [14]	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	71.2
Ours	90.1	86.7	74.6	58.7	57.3	90.0	74.2	68.2	54.9	85.4	54.6	67.5	80.5	93.5	26.1	60.6	88.3	54.2	81.3	84.5	**71.6**

TABLE VI
COMPARISON WITH THE STATE-OF-THE-ART METHODS ON MSCOCO DATASET

Methods	mAP@0.5	mAP@[.5, .95]
PCL [11]	19.4	8.5
PG-PS [35]	20.7	-
C-MIDN [14]	21.4	9.6
$WSOD^2$ [15]	22.7	10.8
PSLR [36]	23.6	11.1
MIST [17]	24.3	11.4
Ours	**25.7**	**12.0**

TABLE VII
COMPARISON OF DIFFERENT METHODS IN TERMS OF COMPUTATIONAL COMPLEXITIES AND THE TIME COSTS ON THE VOC 2007 DATASET. THE MULTI-SCALING STRATEGY IS ADOPTED DURING INFERENCE

Methods	Para. (M)	Train Time (s)	Test Time (s)	mAP
PCL [11]	134.68	2.48	2.25	48.0
OIM* [18]	134.68	5.40	2.17	50.1
IM-CFB [45]	135.11	2.64	2.20	54.3
$OICR_{reg}$	135.11	1.97	2.11	53.0
Ours	135.12	2.17	2.24	**56.4**

yet effective F-BBC network to provide useful foreground information and the efficient FGSM algorithm to generate positive seeds.

Fig. 4 shows the detection results on VOC 2007 *test* set. The first three rows indicate that our method can correctly detect diverse objects, *e.g.*, "cat", "dog", "car," even if they are in some complex scenes. Some failure cases are shown in the last row, containing localizing the most discriminative parts and grouping multiple objects. These cases are particularly reflected in the "person" and "bottle" class, since they always exist in relatively complex scenes with various objects or appear in groups bringing unavoidable occlusion problems. Without box-level annotations, our method is struggling to detect individual objects accurately in these conditions.

D. Ablation Study

We conduct ablation experiments on PASCAL VOC 2007 to analyze how each component of our proposed method influences the detection performance.

Baseline: We construct our baseline model following the settings of OICR [10], which contains the MIL branch and several self-training branches. In each self-training branch, the top-scoring proposals are chosen as positive seeds directly. Additionally, we add an extra regression branch after the last self-training branch, as described in Section III-E. We denote the baseline model as $OICR_{reg}$.

Effect of Each Component: To fully examine the impact of each component of our method, we conduct different ablation experiments, which are shown in Table I. 'Inf.' means utilizing F-BBC results during inference (see Section III-F). We start from a base model $OICR_{reg}$. Next, we extend the base model by adding an F-BBC network, bringing a 0.7% mAP gain. It is worth mentioning that, the information from the F-BBC network has not been used in this setting, but the performance is obviously improved by the extra F-BBC task. Hence, the improvement indicates that the F-BBC task acts as a useful complement for the original M-CC task and is beneficial for the network representation.

Then, we replace the original approach (*i.e.*, OICR [10]) with our proposed FGSM algorithm (including filtering negative proposals) to mine accurate seeds for self-training networks. Utilizing FGSM improves the mAP from 53.7% to 55.3 %. The result indicates that combining the M-CC and F-BBC information helps localize more reliable seeds. Additionally, we apply the multi-seed training (MST) strategy to train the self-training networks, bringing a 0.8% improvement. Overall, the whole FGST module (containing FGSM and MST) leads to a boost of 2.4% mAP relatively. We attribute it to that our FGST utilizes the extra F-BBC results and takes full advantage of these two kinds of information.

Finally, we bring in the F-BBC results during the inference. It brings a 0.4% mAP improvement on a weak baseline where the FGST is not applied. When based on a more complete model (*i.e.*, containing all the components before), it improves the mAP from 56.1% to 56.4%. These results further indicate the effectiveness of the F-BBC information.

Fig. 5 shows the detection results of our method and the baseline method $OICR_{reg}$ on VOC 2007 *test* set. Compared with

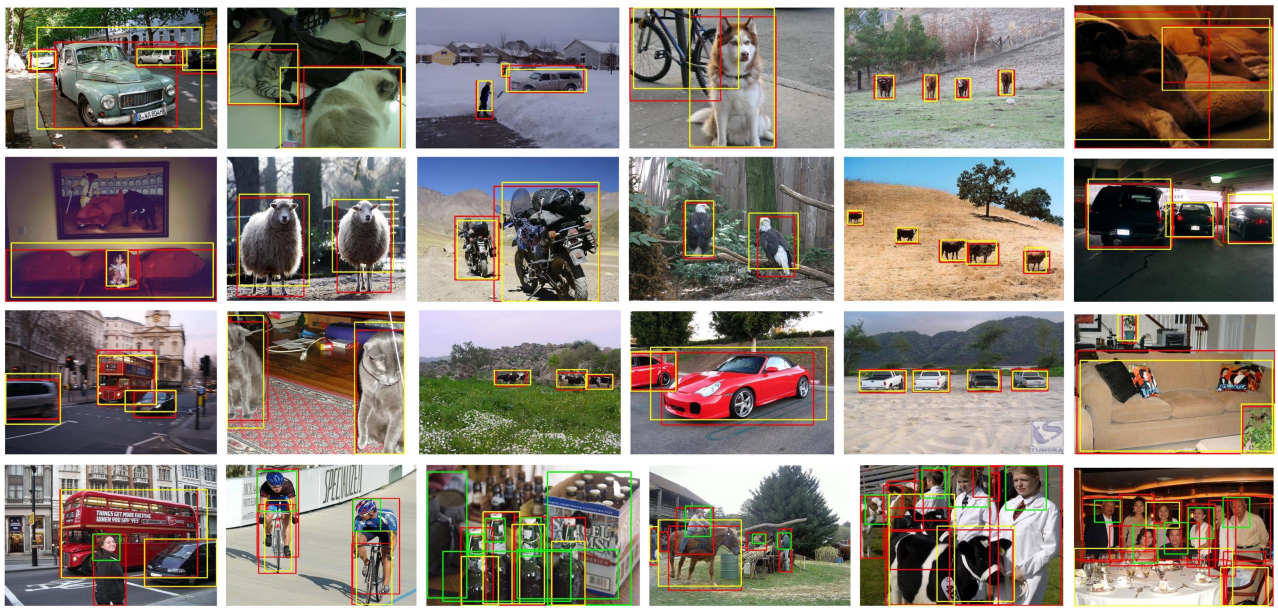


Fig. 4. Detection results on VOC 2007 *test* set. Boxes in red, yellow, and green represent ground-truth boxes, successful predictions, and failure cases, respectively.

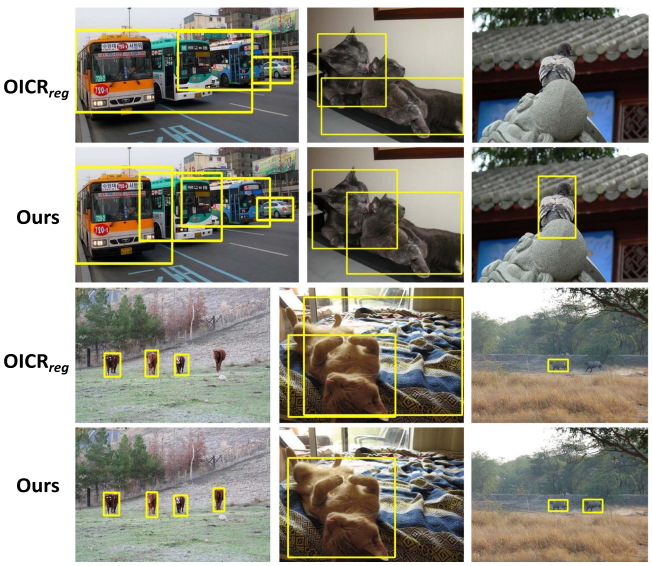


Fig. 5. Detection results of the baseline model OICR$_{reg}$ (the first and third row) and our framework (the second and fourth row).

OICR$_{reg}$, our proposed FI-WSOD can detect more existing objects (3rd, 4th, 6th images), more complete objects (5th image), and filter out proposals containing background (2nd, 5nd images). Moreover, FI-WSOD can localize objects more accurately even if they are in some complex scenes (1st image). These results demonstrate the effectiveness of the F-BBC network and the proposed FI-WSOD model.

Influence of p in FGST: We conduct experiments to analyze the influence of the selection rate p, which determines the number of negative proposals filtered out in the FGST module, on the detection performance. The results are shown in Table VIII.

TABLE VIII
PERFORMANCE COMPARISONS BY USING DIFFERENT SELECTION RATE p IN THE FGST MODULE. THE BASELINE MODEL IS DENOTED AS '-'. THE FIRST ROW REPRESENTS THE DIFFERENT SETTINGS OF p AND THE SECOND ROW REPRESENTS THE CORRESPONDING MAP RESULTS

p (%)	-	2	5	10	12	14	16	18	20	30	40	50
mAP	54.1	55.5	55.7	56.1	56.0	**56.4**	55.8	56.0	55.7	55.9	55.9	55.2

TABLE IX
PERFORMANCE COMPARISONS BY USING DIFFERENT k IN THE FGST MODULE

k	0.0	0.1	0.2	0.3	0.4	0.5	0.7	1.0
mAP (%)	53.2	54.6	56.0	**56.4**	56.2	55.9	55.2	54.1

Directly choosing the top-scoring proposal without filtering (*i.e.*, the 3 rd row in Table I) results in an mAP of 54.1% (denoted as '-' in Table VIII), which can be seen as the baseline approximately. When the p is too small, the FGST module filters out the majority of proposals, resulting in less positive seeds being mined. When the p is too large, too many proposals remain, which will cause some background samples to be selected incorrectly, hence exerting a negative influence on the quality of the seeds. In general, the FGST is insensitive to p in a wide range, and it brings at least 1.6% mAP improvements in most cases. Among different settings, $p = 14$ performs best and leads to a boost of 2.3% mAP, hence we choose it in our experiments.

Influence of k in FGST: We conduct experiments to analyze the influence of k, which controls the usage time of $\mathcal{L}_{st}^{fgsm}$ (*i.e.*, FGSM), on the detection performance. The results are shown in Table IX. At the beginning of the training stage, the F-BBC results are not accurate enough, hence applying FGSM will not bring many improvements if the k is too small. If the k is too

TABLE X
IMPACT OF DIFFERENT SOURCES OF FOREGROUND SCORES ON VOC 2007 DATASET. WSOD^2* REPRESENTS THE SINGLE-SCALE TESTING RESULTS OF WSOD^2[15], AS IT CONDUCTS ABLATION STUDIES FOLLOWING THIS SETTING

Methods	Extra info.	Source	mAP	mAP gain
TS^{2}C [13]	SOD	Seg. branch	44.3	+3.0
WSOD^2* [15]	-	Offline	50.3	**+4.4**
OCRepr [21]	WSSS	Offline	50.6	+3.3
WSOD^2 [15]	-	Offline	53.6	-
Ours-0	-	M-CC	55.0	+2.0
Ours	-	F-BBC	**56.4**	**+3.4**

large, the effect of FGSM will be greatly reduced, and the performance gained from it will be limited. In general, compared with the model trained without the FGSM algorithm (*i.e.*, $k = 1.0$), applying FGSM improves mAP by at least 1.3% in most cases, and $k = 0.3$ performs best among different settings.

FI-WSOD v.s. Previous Methods Using Foreground Scores: We compare our work with previous methods [13], [15], [21] that using foreground scores as well. The results are shown in the 1st-4th and 6 rd rows in Table X. Considering that different works apply different baselines or differ in replicating the baseline module, we further report the mAP gain for a fair comparison. TS ^{2} C [13] uses extra information from Saliency Object Detection (SOD) method which is fully supervised trained with binary mask. However, we only use image-level labels and do not need additional a priori knowledge. OCRepr [21] applies information from Weakly Supervised Semantic Segmentation (WSSS) task and generates foreground scores offline. WSOD^2[15] applies four extra hand-crafted algorithms on low-level features to obtain bottom-up objectness scores during training. However, these four algorithms need to be applied on all regressed proposals at each iteration, which will cost much time. In contrast, we can directly obtain foreground (objectness) scores from the F-BBC branch without any extra post-processes or comprehensive algorithms. Furthermore, despite only using these scores to select confident positive proposals as [15], [21], performing the F-BBC task online will guide the network to better discriminate foreground features from the background ones, hence further improving the network representation. As shown in the 2nd row in Table I, the F-BBC task will benefit the M-CC task even if the foreground scores are not used. Overall, compared with these previous works, our method obtains better performance and brings more mAP gain in most cases even from a stronger baseline, which demonstrates the effectiveness of F-BBC.

F-BBC Scores v.s. Foreground Scores from M-CC: In addition to the F-BBC, scores of "background" category in M-CC in self-training branches can also be used as foreground guidance. In the 5th row of the Table X, we show the performance when the "background" scores (or summation of all foreground category scores) in M-CC are used directly for foreground guidance. Compared to the F-BBC (6th row), the use of M-CC will result in a performance degradation of about 1.4% for mAP. We attribute it to that although M-CC considers "background" category, it still focuses primarily on mining the inter-class diversity among foreground categories, and thus it does not accurately reflect the information of F-BBC.

V. CONCLUSION

In this paper, we present a novel Foreground Information Guided Weakly Supervised Object Detection (FI-WSOD) framework for weakly supervised object detection (WSOD), which integrates the foreground-background binary classification task into WSOD to localize more accurate instances from images. We first design a simple yet effective Foreground-Background Binary Classification (F-BBC) network to provide the foreground information for each proposal. Taking advantage of the extra information, the proposed Foreground Guided Self-Training (FGST) module greatly improves the detection capability of the trained CNN classifier. In the FGST module, the Foreground Guided Seeds Mining (FGSM) algorithm leverages the foreground information to mine accurate and representative seeds. When training the self-training networks with these seeds, a Multi-Seed Training (MST) strategy is applied to reduce the impact of noisy labels. Additionally, the foreground information from F-BBC is also utilized during the inference stage. Extensive experiments conducted on two benchmark datasets, *i.e.*, Pascal VOC 2007, Pascal VOC 2012 and MSCOCO, demonstrate the effectiveness of our method.

REFERENCES

[1] R. Girshick, "Fast R-CNN," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2015, pp. 1440–1448.
[2] S. Ren, K. He, R. Girshick, and J. Sun, "Faster R-CNN: Towards real-time object detection with region proposal networks," in *Proc. Adv. Neural Inf. Process. Syst.*, 2015, pp. 91–99.
[3] T.-Y. Lin et al., "Feature pyramid networks for object detection," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2017, pp. 2117–2125.
[4] J. Yu, Y. Jiang, Z. Wang, Z. Cao, and T. Huang, "UnitBox: An advanced object detection network," in *Proc. 24th ACM Int. Conf. Multimedia*, 2016, pp. 516–520.
[5] W. Liu et al., "SSD: Single shot multibox detector," in *Proc. Eur. Conf. Comput. Vis.*, 2016, pp. 21–37.
[6] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, "You only look once: Unified, real-time object detection," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2016, pp. 779–788.
[7] T.-Y. Lin, P. Goyal, R. Girshick, K. He, and P. Dollár, "Focal loss for dense object detection," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2017, pp. 2980–2988.
[8] H. Law and J. Deng, "CornerNet: Detecting objects as paired keypoints," in *Proc. Eur. Conf. Comput. Vis.*, 2018, pp. 734–750.
[9] K. Duan et al., "CenterNet: Keypoint triplets for object detection," in *Proc. IEEE/CVF Int. Conf. Comput. Vis.*, 2019, pp. 6569–6578.
[10] P. Tang, X. Wang, X. Bai, and W. Liu, "Multiple instance detection network with online instance classifier refinement," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2017, pp. 2843–2851.
[11] P. Tang et al., "PCL: Proposal cluster learning for weakly supervised object detection," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 42, no. 1, pp. 176–191, Jan. 2020.
[12] H. Bilen and A. Vedaldi, "Weakly supervised deep detection networks," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2016, pp. 2846–2854.
[13] Y. Wei et al., "TS2C: Tight box mining with surrounding segmentation context for weakly supervised object detection," in *Proc. Eur. Conf. Comput. Vis.*, 2018, pp. 434–450.
[14] G. Yan et al., "C-MIDN: Coupled multiple instance detection network with segmentation guidance for weakly supervised object detection," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2019, pp. 9834–9843.

[15] Z. Zeng, B. Liu, J. Fu, H. Chao, and L. Zhang, "WSOD2: Learning bottom-up and top-down objectness distillation for weakly-supervised object detection," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2019, pp. 8292–8300.
[16] K. Yang, D. Li, and Y. Dou, "Towards precise end-to-end weakly supervised object detection network," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2019, pp. 8372–8381.
[17] Z. Ren et al., "Instance-aware, context-focused, and memory-efficient weakly supervised object detection," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2020, pp. 10598–10607.
[18] C. Lin, S. Wang, D. Xu, Y. Lu, and W. Zhang, "Object instance mining for weakly supervised object detection," in *Proc. AAAI Conf. Artif. Intell.*, 2020, pp. 11482–11489.
[19] Z. Chen, Z. Fu, R. Jiang, Y. Chen, and X.-S. Hua, "SLV: Spatial likelihood voting for weakly supervised object detection," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2020, pp. 12995–13004.
[20] Y. Xu, C. Zhou, X. Yu, B. Xiao, and Y. Yang, "Pyramidal multiple instance detection network with mask guided self-correction for weakly supervised object detection," *IEEE Trans. Image Process.*, vol. 30, pp. 3029–3040, 2021.
[21] K. Yang et al., "Objectness consistent representation for weakly supervised object detection," in *Proc. 28th ACM Int. Conf. Multimedia*, 2020, pp. 1688–1696.
[22] Z. Tian, C. Shen, H. Chen, and T. He, "FCOS: Fully convolutional one-stage object detection," in *Proc. IEEE/CVF Int. Conf. Comput. Vis.*, 2019, pp. 9627–9636.
[23] N. Carion et al., "End-to-end object detection with transformers," in *Proc. Eur. Conf. Comput. Vis.*, 2020, pp. 213–229.
[24] J. Ahn and S. Kwak, "Learning pixel-level semantic affinity with image-level supervision for weakly supervised semantic segmentation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2018, pp. 4981–4990.
[25] J. Ahn, S. Cho, and S. Kwak, "Weakly supervised learning of instance segmentation with inter-pixel relations," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2019, pp. 2209–2218.
[26] E. Xie et al., "DetCo: Unsupervised contrastive learning for object detection," in *Proc. IEEE/CVF Int. Conf. Comput. Vis.*, 2021, pp. 8392–8401.
[27] D. Cheng, J. Zhou, N. Wang, and X. Gao, "Hybrid dynamic contrast and probability distillation for unsupervised person Re-Id," *IEEE Trans. Image Process.*, vol. 31, pp. 3334–3346, Apr. 2022.
[28] M. Gao, A. Li, R. Yu, V. I. Morariu, and L. S. Davis, "C-WSL: Count-guided weakly supervised localization," in *Proc. Eur. Conf. Comput. Vis.*, 2018, pp. 152–168.
[29] D. Zhang, J. Han, G. Cheng, and M.-H. Yang, "Weakly supervised object localization and detection: A survey," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 44, no. 9, pp. 5866–5885, Sep. 2022.
[30] G. Cheng, J. Yang, D. Gao, L. Guo, and J. Han, "High-quality proposals for weakly supervised object detection," *IEEE Trans. Image Process.*, vol. 29, pp. 5794–5804, Apr. 2020.
[31] D. Zhang, W. Zeng, J. Yao, and J. Han, "Weakly supervised object detection using proposal-and semantic-level relationships," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 44, no. 6, pp. 3349–3363, Jun. 2022.
[32] X. Feng, J. Han, X. Yao, and G. Cheng, "TCANet: Triple context-aware network for weakly supervised object detection in remote sensing images," *IEEE Trans. Geosci. Remote Sens.*, vol. 59, no. 8, pp. 6946–6955, Aug. 2021.
[33] J. R. Uijlings, K. EA Van De Sande, T. Gevers, and A. W. Smeulders, "Selective search for object recognition," *Int. J. Comput. Vis.*, vol. 104, no. 2, pp. 154–171, 2013.
[34] Q. Ren, S. Lu, J. Zhang, and R. Hu, "Salient object detection by fusing local and global contexts," *IEEE Trans. Multimedia*, vol. 23, pp. 1442–1453, 2021.
[35] X. Lin, Z.-J. Wang, L. Ma, and X. Wu, "Saliency detection via multi-scale global cues," *IEEE Trans. Multimedia*, vol. 21, no. 7, pp. 1646–1659, Jul. 2019.
[36] J. Deng et al., "ImageNet: A large-scale hierarchical image database," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2009, pp. 248–255.
[37] A. Diba, V. Sharma, A. Pazandeh, H. Pirsiavash, and L. Van Gool, "Weakly supervised cascaded convolutional networks," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2017, pp. 914–922.
[38] V. Kantorov, M. Oquab, M. Cho, and I. Laptev, "ContextLocNet: Context-aware deep network models for weakly supervised localization," in *Proc. Eur. Conf. Comput. Vis.*, 2016, pp. 350–365.
[39] J. Fan, Z. Zhang, and T. Tan, "Employing multi-estimations for weakly-supervised semantic segmentation," in *Proc. Eur. Conf. Comput. Vis.*, 2020, pp. 332–348.
[40] Y. Yin, J. Deng, W. Zhou, and H. Li, "Instance mining with class feature banks for weakly supervised object detection," in *Proc. AAAI Conf. Artif. Intell.*, 2021, pp. 3190–3198.
[41] M. Everingham, L. Van Gool, C. K. Williams, J. Winn, and A. Zisserman, "The pascal visual object classes (VOC) challenge," *Int. J. Comput. Vis.*, vol. 88, no. 2, pp. 303–338, 2010.
[42] T.-Y. Lin et al., "Microsoft COCO: Common objects in context," in *Proc. Eur. Conf. Comput. Vis.*, 2014, pp. 740–755.
[43] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," 2014, *arXiv:1409.1556*.
[44] J. Pont-Tuset, P. Arbelaez, J. T. Barron, F. Marques, and J. Malik, "Multiscale combinatorial grouping for image segmentation and object proposal generation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 39, no. 1, pp. 128–140, Jan. 2017.
[45] Y. Yin, J. Deng, W. Zhou, and H. Li, "Instance mining with class feature banks for weakly supervised object detection," in *Proc. Conf. Artif. Intell.*, 2021, vol. 35, no. 4, pp. 3190–3198.
[46] Z. Jie, Y. Wei, X. Jin, J. Feng, and W. Liu, "Deep self-taught learning for weakly supervised object localization," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2017, pp. 1377–1385.
[47] P. Tang et al., "Weakly supervised region proposal network and object detection," in *Proc. Eur. Conf. Comput. Vis.*, 2018, pp. 352–368.
[48] Y. Shen, R. Ji, Y. Wang, Y. Wu, and L. Cao, "Cyclic guidance for weakly supervised joint detection and segmentation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2019, pp. 697–707.
[49] F. Wan, P. Wei, J. Jiao, Z. Han, and Q. Ye, "Min-entropy latent model for weakly supervised object detection," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2018, pp. 1297–1306.
[50] X. Li, M. Kan, S. Shan, and X. Chen, "Weakly supervised object detection with segmentation collaboration," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2019, pp. 9735–9744.
[51] F. Wan et al., "C-MIL: Continuation multiple instance learning for weakly supervised object detection," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2019, pp. 2199–2208.
[52] A. Arun, C. V. Jawahar, and M. P. Kumar, "Dissimilarity coefficient based weakly supervised object detection," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2019, pp. 9432–9441.

Yufei Yin received the B.E. degree in electronic information engineering from the University of Science and Technology of China (USTC), Hefei, China, in 2019, where he is currently working toward the Ph.D. degree in information and communication engineering with the Department of Data Science. His research interests include computer vision, weakly supervised object detection, and open-set panoptic segmentation.

Jiajun Deng (Graduate Student Member, IEEE) received the B.E. degree in electrical engineering and information science from the The University of Science and Technology of China, Hefei, China, in 2016, and the Ph.D. degree in information and communication engineering from The University of Science and Technology of China in 2021. His research interests include object detection, 3D scene perception, image rectification, and vision-language understanding.

Wengang Zhou (Senior Member, IEEE) received the B.E. degree in electronic information engineering from Wuhan University, Wuhan, China, in 2006, and the Ph.D. degree in electronic engineering and information science from the University of Science and Technology of China (USTC), Hefei, China, in 2011. From September 2011 to September 2013, he was a Postdoctoral Researcher in Computer Science Department with the University of Texas at San Antonio, San Antonio, TX, USA. He is currently a Professor with the EEIS Department, USTC. He has authored or coauthored more than 100 papers in IEEE/ACM TRANSACTIONS and CCF Tier-A International Conferences. His research interests include multimedia information retrieval, computer vision, and computer game. He was the recipient of the Best Paper Award for ICIMCS 2012. He received the award for the Excellent Ph.D Supervisor of Chinese Society of Image and Graphics in 2021, and the award for the Excellent Ph.D Supervisor of Chinese Academy of Sciences in 2022. He won the First Class Wu-Wenjun Award for Progress in Artificial Intelligence Technology in 2021. He was the Publication Chair of the IEEE ICME 2021 and won 2021 ICME Outstanding Service Award. He is currently an Associate Editor for the IEEE TRANSACTIONS ON MULTIMEDIA.

Li Li (Member, IEEE) received the B.S. and Ph.D. degrees in electronic engineering from the University of Science and Technology of China (USTC), Hefei, Anhui, China, in 2011 and 2016, respectively. He was a Visiting Assistant Professor with the University of Missouri-Kansas City, Kansas City, MO, USA, from 2016 to 2020. He joined the Department of Electronic Engineering and Information Science of USTC as a Research Fellow in 2020 and became a Professor in 2022. His research interests include image/video/point cloud coding, and processing. He was the recipient of the Best 10% Paper Award at the 2016 IEEE Visual Communications and Image Processing (VCIP) and the 2019 IEEE International Conference on Image Processing.

Houqiang Li (Fellow, IEEE) received the B.S., M.Eng., and Ph.D. degrees in electronic engineering from the University of Science and Technology of China, Hefei, China, in 1992, 1997, and 2000, respectively, where he is currently a Professor with the Department of Electronic Engineering and Information Science. He has authored or coauthored more than 200 papers in journals and conferences. His research interests include image/video coding, image/video analysis, computer vision, reinforcement learning. He is the winner of National Science Funds for Distinguished Young Scientists, the Distinguished Professor of Changjiang Scholars Program of China, and the Leading Scientist of Ten Thousand Talent Program of China. He is an Associate Editor for the IEEE TRANSACTIONS ON MULTIMEDIA, and was the AE of IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS FOR VIDEO TECHNOLOGY from 2010 to 2013. He was the General Co-Chair of ICME 2021 and the TPC Co-Chair of VCIP 2010. He was the recipient of the Second Class Award of China National Award for Technological Invention in 2019, Second Class Award of China National Award for Natural Sciences in 2015, and the First Class Prize of Science and Technology Award of Anhui Province in 2012, Award for the Excellent Ph.D Supervisor of Chinese Academy of Sciences for four times from 2013 to 2016, Best Paper Award for VCIP 2012, Best Paper Award for ICIMCS 2012, and the Best Paper Award for ACM MUM in 2011.