C-WSL: Count-guided Weakly Supervised Localization

Mingfei Gao[1], Ang Li[2⋆], Ruichi Yu[1], Vlad I. Morariu[3⋆], and Larry S. Davis[1]

[1]University of Maryland, College Park [2]DeepMind [3]Adobe Research
{mgao,richyu,lsd}@umiacs.umd.edu anglili@google.com morariu@adobe.com

Abstract. We introduce count-guided weakly supervised localization (C-WSL), an approach that uses per-class object count as a new form of supervision to improve weakly supervised localization (WSL). C-WSL uses a simple count-based region selection algorithm to select high-quality regions, each of which covers a single object instance during training, and improves existing WSL methods by training with the selected regions. To demonstrate the effectiveness of C-WSL, we integrate it into two WSL architectures and conduct extensive experiments on VOC2007 and VOC2012. Experimental results show that C-WSL leads to large improvements in WSL and that the proposed approach significantly outperforms the state-of-the-art methods. The results of annotation experiments on VOC2007 suggest that a modest extra time is needed to obtain per-class object counts compared to labeling only object categories in an image. Furthermore, we reduce the annotation time by more than 2× and 38× compared to center-click and bounding-box annotations.

Keywords: Weakly supervised localization · Count supervision.

1 Introduction

Convolutional neural networks (CNN) have achieved state-of-the-art performance on the object detection task [29, 23, 27, 28, 32, 21, 12, 20, 37, 33, 38, 39]. However, these detectors are trained in a strongly supervised setting, requiring a large number of bounding box annotations and huge amounts of human labor.

To ease the burden of human annotation, weakly supervised localization (WSL) methods train a detector using weak supervision, *e.g.*, image-level supervision, instead of tight object bounding boxes. The presence of an object category in an image can be obtained on the Internet nearly for free, so most existing WSL architectures require only object categories as supervision.

Existing methods [1, 3, 5, 15, 24, 36, 35, 19, 14, 40, 16, 30, 34] have proposed different architectures to address the WSL problem. However, there is still a large performance gap between weakly and strongly supervised detectors [29, 28, 23] on standard object detection benchmarks [9, 10, 22]. Often, this is due to the limited information provided by object-category supervision. One major unsolved

⋆ The work was done while the author was at the University of Maryland

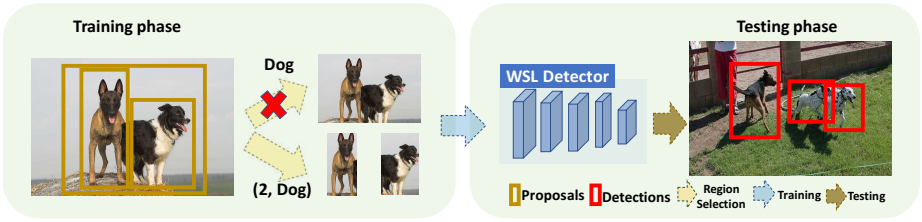


Fig. 1. Given a set of object proposals and the per-class object count label, we select high-quality positive regions (that tightly cover a single object) to train a WSL detector. Count information significantly reduces detected bounding boxes that are loose and contain two or more object instances, one of the most common errors produced by weakly supervised detectors

problem of WSL is that high confidence detections tend to include multiple objects instead of one. As shown in Fig. 1 (red cross branch), since training images containing multiple dogs are labeled just as "Dog", detectors tend to learn the composite appearance of multiple dogs as if they were one dog and group multiple dogs as a single instance at test time. To resolve this ambiguity, we use per-class object count information to supervise detector training.

Object count is a type of image-level supervision which is much weaker and cheaper than instance-level supervisions, such as center clicks [26] and bounding boxes. Unlike center click and bounding box annotations, which require several well-trained annotators to specify the center and tight box of each object, object count contains no location information and can be obtained without actually clicking on an object. Moreover, a widely studied phenomenon in psychology, called subitizing [4] suggests that humans are able to determine the number of objects without pointing to or fixating on each object sequentially if the total number of objects in the image is small (typically 1-4) [2]. Thus, people may be able to specify the object count with just a glance. To demonstrate the inexpensiveness of count annotation, we conduct annotation experiments on Pascal VOC2007. Experimental results show that only a small amount of extra time is needed to obtain per-class object counts compared to labeling just object categories in an image and the response time of the count annotation is much less than that of object center and bounding box.

Our proposed method, Count-guided WSL (C-WSL), is illustrated in Fig. 1. During the training process, C-WSL makes use of per-class object count supervision to identify the correct high-scoring object bounding boxes from a set of object proposals. Then, a weakly supervised detector is refined with these high-quality regions as pseudo ground-truth (GT) bounding boxes. This strategy is similar to existing WSL methods that refine detectors using automatically identified bounding boxes [19, 14, 35]. However, since these methods do not make use of object count supervision, they treat only the top-scoring region as the pseudo GT box, regardless of the number of object instances present in the image. This sometimes leads to multiple object instances being grouped into a single pseudo

GT box, which hurts the detector's ability to localize individual objects. With the guidance of the object count label, C-WSL selects tight box regions that cover individual objects as shown in Fig. 1 (the "(2, Dog)" branch).

The main contribution of C-WSL is that it uses per-class object count, a cheap and effective form of image-level supervision, to address a common failure case in WSL where one detected bounding box contains multiple object instances. To implement C-WSL, we develop a simple Count-based Region Selection (CRS) algorithm and integrate it into two existing architectures—alternating detector refinement (ADR) and online detector refinement (ODR)—to significantly improve WSL. Experimental results on Pascal VOC2007 [9] and VOC2012 [10] show that C-WSL significantly improves WSL detection and outperforms state-of-the-art methods.

2 Related Works

MIL-based CNN Methods. Most existing WSL methods [1, 3, 5, 15, 24, 36, 35, 19, 14] are based on multiple instance learning (MIL) [6]. In the MIL setting, a bag is defined as a collection of regions within an image. A bag is labeled as positive if at least one instance in the bag is positive and labeled as negative if all of its samples are negative. Bilen *et al.* [1] proposed a two-stream CNN architecture to classify and localize simultaneously and train the network in an end-to-end manner. Following [1], Kantorov *et al.* [15] added *additive* and *contrastive* models to improve localization on object boundaries instead of local parts. Singh *et al.* [34] proposed the 'Hide-and-Seek' framework which hides informative patches to encourage WSL to detect complete object instances. In [19], Li *et al.* conducted progressive domain adaption and significantly improved the localization ability of the baseline detector. Diba *et al.* [5] performed WSL in two/three cascaded stages to find the best candidate location based on a generated class activation map. Jie *et al.* proposed a self-taught learning approach in [14] which alternates between classifier training and online supportive sample harvesting. Similarly, in [35], Tang *et al.* designed an online classifier refinement pipeline to progressively locate the most discriminative region of an image. [14] and [35] are most related to our approach since we also conduct alternating and online detector refinement. However, instead of using the top-scoring detection as the positive label [35] or mining confident regions by solving a complex dense subgraph discovery problem [14], we use per-class object count, a cheap form of supervision, to guide region selection and progressively obtain better positive training regions.

WSL with Different Supervisions. [25] proposed a novel framework where an annotator verifies predicted results instead of manually drawing boxes. Kolesnikov *et al.* [17] assigned object or distractor labels to co-occuring objects in images to improve WSL. Papadopoulos *et al.* [26] proposed click supervision and integrated it into existing MIL-based methods to improve localization performance. However, these methods either highly depend on the produced results and require frequent interactions with annotators or require annotators to search for

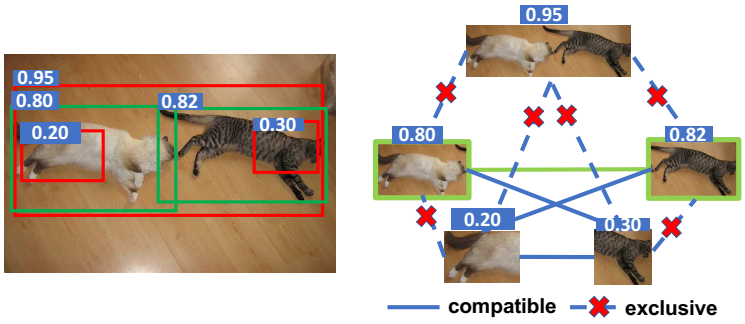


Fig. 2. A common failure case of WSL methods (left) and graph representation of our region selection formulation (right). Our goal is to select the two green boxes, each of which tightly covers one object, as the positive training samples for WSL detectors. We achieve this by analyzing the confidence scores and spatial constraints among regions

and click on each instance in an image. In contrast, object count is an image-level annotation which contains no location information at all. It can be obtained with no clicks and few interactions, thus requires much less annotation time.

3 Proposed Approach

C-WSL selects regions covering a single object with the help of per-class object count supervision and then refines the WSL detector using these regions as the pesudo GT bounding boxes. We first introduce a simple Count-based Region Selection (CRS) algorithm that C-WSL relies on to select high-quality regions from object proposals on training images. Then, we integrate CRS into two detector refinement structures to improve weakly supervised detectors.

3.1 Count-based Region Selection (CRS)

As shown in Fig. 2 (left), without object count information, previous methods often select the top-scoring box in training images as the positive training sample to refine the WSL detector [35, 19, 14]. Their detection performance is degraded because in many cases the top-scoring box contains multiple objects from the same category, *e.g.*, two cats. Our goal is to select distinct regions, each covering a single object as positive training samples with the help of object count constraints so that the detector will learn the appearance of a single cat.

We formulate the problem as a region selection problem. Given a set of boxes $\mathbf{B} = \{b_1, ..., b_N\}$ and the corresponding confidence scores $\mathbf{P} = \{p_1, ..., p_N\}$ (*e.g.*, the detection score of a region in each detector refinement iteration), a subset $\mathbf{G}$ is selected as the set of positive training regions where $|\mathbf{G}| = C$ and C indicates the per-class object count. We identify a good subset $\mathbf{G}$ using a greedy algorithm applied to a graphical representation of the set of boxes. Each box is represented

as a node in the graph, and two nodes are connected if the spatial overlap of their corresponding boxes is below a threshold (See solid line in Fig. 2). The greedy algorithm provides an approximation to the following optimization problem:

$$\begin{aligned} &\mathbf{G}^* = \arg\max_{\mathbf{G}} \sum_{b_k \in \mathbf{G}} p_k, \\ &s.t.\ |\mathbf{G}| = C,\ a_o(b_i, b_j) < T\ \forall b_i, b_j \in \mathbf{G}, i \neq j. \end{aligned} \tag{1}$$

To encourage selecting regions containing just one object, we use the asymmetric area of overlap, i.e, $a_o(b_i, b_j) = \frac{area(b_i \cap b_j)}{area(b_j)}$, which has been proposed in [7, 8] to model spatial overlap between two boxes, where b_i is a box previously selected by the greedy algorithm and b_j indicates a box considered for selection. T is the overlap threshold. If the algorithm has previously added a large box to the solution, thresholding on a_o will discourage the selection of its subregions, regardless of their sizes.[1] So, to deliver a high total score, the algorithm prefers C small high-scoring boxes to one large box, even though the large box may have the highest score.

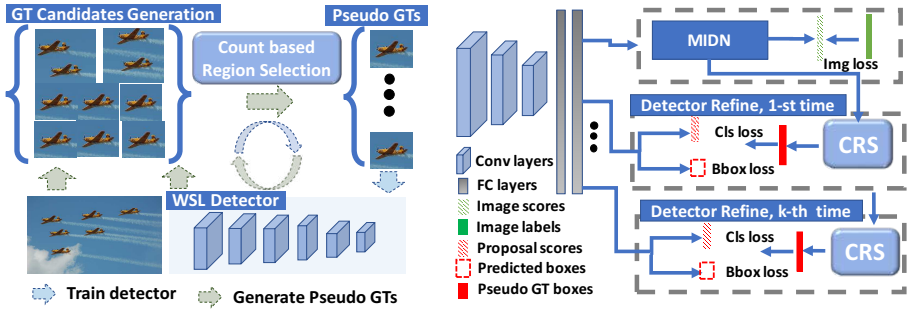
We conduct region selection after applying non-maximum suppression on a complete set of the detection boxes, so the number of nodes is limited to a reasonable number, and the computation cost is low in practice. The algorithm is summarized in Alg. 1.

Algorithm 1: Count-based Region Selection (CRS)

Input : $\mathbf{B} = \{b_1, ..., b_N\}$, $\mathbf{P} = \{p_1, ..., p_N\}$, T, C;
$\mathbf{B}$ is a list of candidate boxes;
$\mathbf{P}$ is the corresponding scores;
T is the overlap threshold;
C indicates the object count;

Initialization: Sort (descend) $\mathbf{B}$ based on $\mathbf{P}$;
$\mathbf{G}^* \leftarrow \emptyset$; $s_{max} \leftarrow 0$;
Output: $\mathbf{G}^*$
for $i \in \{1, ..., N\}$ **do**
  $\mathbf{G} \leftarrow b_i$; $s \leftarrow p_i$;
  for $j \in \{i+1, ..., N\}$ **do**
    if $a_o(b_k, b_j) < T(\forall b_k \in \mathbf{G})$ **then**
      $\mathbf{G} \leftarrow \mathbf{G} \cup \{b_j\}$; $s \leftarrow s + p_j$
      if $|\mathbf{G}| == C$ *or* $j == N$ **then**
        if $s > s_{max}$ **then**
          $s_{max} \leftarrow s$; $\mathbf{G}^* \leftarrow \mathbf{G}$
        break;

[1] The commonly used symmetric intersection-over-union measure would select sufficiently small regions even if they were fully overlapped by an existing large box.



(a) Alternating detector refinement (b) Online detector refinement w/ CRS

Fig. 3. (a): Count-based Region Selection (*CRS*) is applied to select high-quality positive training regions from the ground-truth (GT) candidate boxes generated by a WSL detector. The WSL detector is then refined using these regions. (b): The Multiple Instance Detection Network(*MIDN*) [1, 35] and multiple detector networks share the same feature representation to refine the detector at all stages together. *Cls loss* indicates the classification loss and *Bbox loss* indicates bounding box regression loss

3.2 Detector Refinement Structures with CRS

Alternating Detector Refinement (ADR). We first integrate CRS into an alternating WSL refinement architecture, where a poor weakly supervised detector can be refined iteratively. The architecture is shown in Fig. 3, where a WSL detector alternates between generating high-quality regions as pseudo ground-truth (GT) boxes and refining itself using these GT boxes. Some WSL methods are based on a strategy like this [3, 14]. The major difference is that we use CRS to select multiple high-quality regions as the GT boxes.

Initialization phase. We first generate a set of box candidates from the training data using a pre-trained WSL detector. This set of box candidates is treated as the initialized pseudo GTs and will be refined iteratively afterwards.

Alternating training phase. We use Fast R-CNN [13] as our WSL network. Starting from the initialized pseudo GT boxes, Fast R-CNN alternates between improving itself via retraining with the pseudo GT boxes generated by CRS and generating a refined set of GT candidate boxes on the training images.

Online Detector Refinement (ODR). As argued in [35], the alternating strategy has two potential limitations: 1) it is time consuming to alternate between training on the fixed labels and generating labels by the trained model; 2) separating refinements into different iterations might harm performance since it hinders the procedure from sharing image representations across iterations.

Based on [35], we propose an online detector refinement framework integrated with CRS. An illustration of the proposed method is shown in Fig. 3. A Multiple Instance Detection Network (MIDN) and several detector refinement stages share the same feature representation extracted from a backbone structure. The

MIDN utilizes an object-category label to supervise its training as in [35, 1]. Each detector refinement network outputs the classification score and predicted bounding box for each region proposal. The predicted boxes with scores at each stage will be used to select pseudo GTs for the next stage refinement. Compared to [35], we have two major differences: 1) we use CRS to generate high-quality regions as pseudo GTs rather than just choosing the top-scoring region; 2) we use both classification loss and bounding box regression loss for detector refinement, just as RCNNs do. Note that the inputs to CRS produced by MIDN are the proposals with scores before the summation over proposals.

4 Experiments

We compare with the existing WSL methods which are trained by object class labels to show the advantage of per-class count supervision. It may seem an 'unfair' comparison, since the per-class count provides more information compared to object class. However, we demonstrate via our annotation experiment that the cost of the additional information is very low, which makes it reasonable to determine how much improvement can be gained by adding this information.

4.1 Experimental Setup

Datasets and Evaluate Metrics. Comparisons with state-of-the-art methods are conducted on VOC2007 [9] and VOC2012 [10] which contain 20 object categories. For VOC2007, all the models are trained on the *trainval* set which contains 5,011 images and evaluated on *test* set which includes 4,952 images. For VOC2012, models are trained on 5,717 images of the *train* set and evaluated on 5,823 images in the *val* set. We use two widely used metrics for localization evaluation: Correct localization (CorLoc) [24] and Average Precision (AP) [11]. CorLoc evaluates localization accuracy by measuring if the maximum response point of a detection is inside the ground truth bounding box. AP evaluates models by comparing IoU between output and ground truth bounding boxes.

Implementation Details. We fix $T = 0.1$ for all models at all the iterations on both datasets. Note that our experiments show that the method is robust to T, *e.g.,* varying T from 0.1 to 1 with step 0.1, we achieved (Mean, Std) = (47.2%, 0.42%) mAP. Following [14, 35], we set the total iteration number to 3 and use *VGG16* [31] as the backbone structure for both ADR and ODR. For fair comparison, the existing works also use *VGG16* except for [3] which utilizes *AlexNet*. In ADR, we strictly follow the steps of training Fast-RCNN at each iteration and use all the released default training parameters except that we use the generated pseudo GT boxes instead of the bounding box labels. In ODR, we follow the basic MIDN structure and training process from [35], and use the parameters released by the author. Note that we use the same classification and bounding box regression loss in ODR as in [13].

Variants of Our Approach. C-WSL:WSLPDA/OICR+ADR indicates ADR initialized with a pre-trained WSLPDA [19] (or OICR [35]) model where CRS is

used to select confident GT boxes in each iteration. Then, a Fast-RCNN is alternatively refined as we mentioned in Sec. 3.2. *C-WSL:ODR* indicates the structure shown in Fig. 3(b). *C-WSL:ODR+FRCNN* denotes a Fast RCNN trained with the top-scoring region generated by *C-WSL:ODR* to improve results (inspired by [19, 35]). *C-WSL** indicates models trained by our annotated counts.

4.2 Annotation Time vs. Detection Accuracy

Object counting is very straightforward. The user interface includes an image and 15 buttons indicating the count numbers. We cap object count with 15 since it is very rare to have a count of the same class bigger than 15. Similar to the click experiments [21], an annotator was given a category and was asked to click the count corresponding to that category. Following [26], given an object category, we measure the response time of counting the object instances from the moment the image appears until the count is determined.

Annotation evaluations are conducted on the full *trainval* set with 20 categories of VOC2007 [9]. The average response time of counting a single object per class per image is $0.90s$. Average response time per image of annotating a single image class is from $1.5s$ to $1.9s$ [18] and that of annotating count given object class is $1.48s$, so obtaining per-class object count from an image only needs $1.48/1.9 = 78\%$ to $1.48/1.5 = 99\%$ more time compared to annotating just the object class.

Annotation time of object counts per image increases as the number of objects increases. However, it might not always be helpful to count all the objects, especially for images with many objects, since these images are more likely to depict complex scenes, *e.g.*, significant occlusions and small object instances, and for such images the generated GT candidates might not include all the objects in the first place. Thus, we evaluate the detection accuracy of our model using at most K per-class objects annotation, where K is the upper bound of per-class object instances that are counted for each image. Obviously, K has positive correlation with annotation time, since annotators may not be able to subitize for high values of K and will need to spend an amount of time proportional to K in order to produce an accurate count. Analysis of mAP and average *CorLoc* vs. K is shown in Fig. 4. The results suggest that the detection accuracy reaches the highest point when at most 3 per-class objects are counted per image. Average annotation time per image for images with at most 3 per-class objects is $1.20s$ which is $63\% \sim 80\%$ overhead compared to object category annotations.

We compare our models trained by our annotated counts and those obtained

Table 1. Accuracy vs. cost among bounding box, clicks and count supervisions on VOC2007. We use [29] as a reference of fully supervised detector

Method	Faster-RCNN [29]	Two-clicks [26]	One-click [26]	C-WSL*:ODR+FRCNN
mAP(%)	69.9	49.1(AlexNet)/57.5(VGG16)	45.9(AlexNet)	48.2(VGG16)
Annotation cost	34.5s/img+anno. train +re-draw rejected boxes	3.74s/img+anno. train +re-click rejected clicks	1.87s/img+anno. train +re-click rejected clicks	0.90s/img

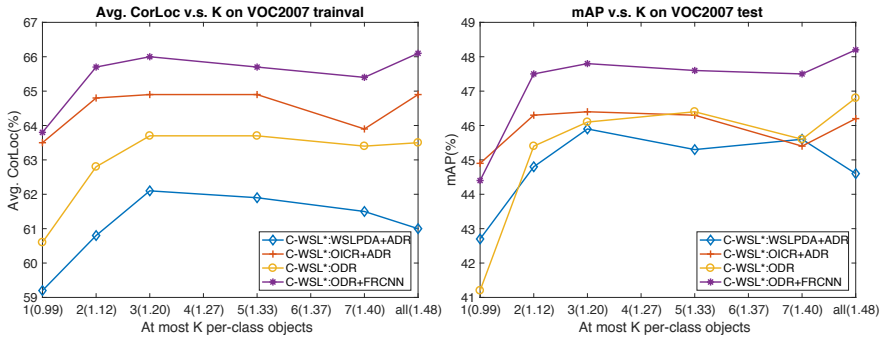


Fig. 4. Detection accuracy analysis when at most K per-class objects are counted in an image. Average annotation time (in seconds) per image under each K is shown in the parentheses. Detection accuracy becomes stable when K=3

from the VOC2007 annotations in Tab. 2 and 3. The results demonstrate that models trained by the two sets of annotations have comparable performance, which suggests that our annotation is as useful as the VOC2007 annotations. Thus, in the following analysis, we just use (*C-WSL*) VOC2007 annotations.

Accuracy and cost comparisons among box, clicks and count supervisions are shown in Tab. 1. Although the accuracy of our approach does not outperform supervised and two-click methods, we have achieved a significant reduction in annotation cost. We are 38× and 4× faster regarding to response time for labeling a single image. In addition, box and clicks annotations require additional repeated annotator training to accurately locate objects and lengthy quality control processes. Our annotation does not require knowing the location of an object so it avoids the sensitivity to location noise. Consequently, we do not need annotator training and quality control in our experiments.

4.3 Comparison with State-of-the-art (SOTA) Approaches

Comparison in terms of *mAP* on the VOC2007 *test* set and *CorLoc* on the VOC2007 *trainval* set are shown in Tab. 2 and 3, respectively. Overall, the proposed *C-WSL:ODR+FRCNN* outperforms all the existing SOTA methods using both *CorLoc* and *mAP* measurements.

Tab. 4 and 5 compare our variants with the two baseline detectors, *i.e.*, WSLPDA [19] and OICR [35]. The results suggest that even the simple ADR strategy can significantly improve the results. Moreover, if we use object count information, we can largely improve WSLPDA by 6.2% *mAP* (9.5% average *CorLoc*) and OICR by 5.2% *mAP* (4.0% average *CorLoc*). C-WSL improves the results of *WSLPDA+ADR* on 17 (15) out of 20 categories and the results of *OICR+ADR* on 10 (10) out of 20 categories in terms of *mAP* on the VOC2007 *test* set (in terms of *CorLoc* on the VOC2007 *trainval* set).

[2] The numbers are reproduced by using the code released by the author.

Table 2. Comparison with the state-of-the-art in terms of mAP on the VOC2007 *test* set. Our number is marked in red if it is the best in the column

Methods	are	bik	brd	boa	btl	bus	car	cat	cha	cow	tbl	dog	hrs	mbk	prs	plt	shp	sfa	trn	tv	mAP
Cinbis *et al.* [3]	39.3	43.0	28.8	20.4	8.0	45.5	47.9	22.1	8.4	33.5	23.6	29.2	38.5	47.9	20.3	20.0	35.8	30.8	41.0	20.1	30.2
Wang *et al.* [36]	48.8	41.0	23.6	12.1	11.1	42.7	40.9	35.5	11.1	36.6	18.4	35.3	34.8	51.3	17.2	17.4	26.8	32.8	35.1	45.6	30.9
Jie *et al.* [14]	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
WSDDN [1]	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
WSDDN+Context [15]	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
WSDDN-Ens. [1]	46.4	58.3	35.5	25.9	14.0	66.7	53.0	39.2	8.9	41.8	26.6	38.6	44.7	59.0	10.8	17.3	40.7	49.6	56.9	50.8	39.3
WCCN-3stage [5]	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
WSLPDA [19]	54.5	47.4	41.3	20.8	17.7	51.9	63.5	46.1	21.8	57.1	22.1	34.4	50.5	61.8	16.2	29.9	40.7	15.9	55.3	40.2	39.5
OICR [35]	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
OICR-Ens.+FRCNN^2 [35]	64.5	64.4	44.1	25.9	16.9	67.8	68.4	33.2	9.0	57.5	46.4	21.7	57.8	64.3	10.0	23.7	50.6	60.9	64.7	58.0	45.5
C-WSL:ODR	**62.7**	**63.7**	**40.0**	**25.5**	**17.7**	**70.1**	**68.3**	**38.9**	**25.4**	**54.5**	**41.6**	**29.9**	**37.9**	**64.2**	**11.3**	**27.4**	**49.3**	**54.7**	**61.4**	**67.4**	**45.6**
C-WSL*:ODR	**62.9**	**64.8**	**39.8**	**28.1**	**16.4**	**69.5**	**68.2**	**47.0**	**27.9**	**55.8**	**43.7**	**31.2**	**43.8**	**65.0**	**10.9**	**26.1**	**52.7**	**55.3**	**60.2**	**66.6**	**46.8**
C-WSL:ODR+FRCNN	**61.9**	**61.9**	**48.6**	**28.7**	**23.3**	**71.1**	**71.3**	**38.7**	**28.5**	**60.6**	**45.4**	**26.3**	**49.7**	**65.5**	**7.2**	**27.3**	**54.7**	**61.6**	**63.2**	**59.5**	**47.8**
C-WSL*:ODR+FRCNN	**62.9**	**68.3**	**52.9**	**25.8**	**16.5**	**71.1**	**69.5**	**48.2**	**26.0**	**58.6**	**44.5**	**28.2**	**49.6**	**66.4**	**10.2**	**26.4**	**55.3**	**59.9**	**61.6**	**62.2**	**48.2**

Table 3. Comparison with the state-of-the-art in terms of CorLoc (%) on the VOC2007 *trainval* set. Our number is marked in red if it is the best in the column

Methods	are	bik	brd	boa	btl	bus	car	cat	cha	cow	tbl	dog	hrs	mbk	prs	plt	shp	sfa	trn	tv	Avg.
Cinbis *et al.* [3]	65.3	55.0	52.4	48.3	18.2	66.4	77.8	35.6	26.5	67.0	46.9	48.4	70.5	69.1	35.2	35.2	69.6	43.4	64.6	43.7	52.0
Wang *et al.* [36]	80.1	63.9	51.5	14.9	21.0	55.7	74.2	43.5	26.2	53.4	16.3	56.7	58.3	69.5	14.1	38.3	58.8	47.2	49.1	60.9	48.5
Jie *et al.* [14]	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
WSDDN [1]	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
WSDDN+Context [15]	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
WSDDN-Ens. [1]	68.9	68.7	65.2	42.5	40.6	72.6	75.2	53.7	29.7	68.1	33.5	45.6	65.9	86.1	27.5	44.9	76.0	62.4	66.3	66.8	58.0
WCCN-3stage [5]	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
SP-VGGNet [40]	85.3	64.2	67.0	42.0	16.4	71.0	64.7	88.7	20.7	63.8	58.0	84.1	84.7	80.0	60.0	29.4	56.3	68.1	77.4	30.5	60.6
WSLPDA [19]	78.2	67.1	61.8	38.1	36.1	61.8	78.8	55.2	28.5	68.8	18.5	49.2	64.1	73.5	21.4	47.4	64.6	22.3	60.9	52.3	52.4
OICR [35]	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
OICR-Ens.+FRCNN^2 [35]	88.3	78.8	62.8	48.9	38.9	83.2	85.4	50.0	21.9	77.4	45.6	41.9	79.3	91.6	12.6	60.8	86.6	70.2	80.2	79.9	64.2
C-WSL:ODR	**86.3**	**80.4**	**58.3**	**50.0**	**36.6**	**85.8**	**86.2**	**47.1**	**42.7**	**81.5**	**42.2**	**42.6**	**50.7**	**90.0**	**14.3**	**61.9**	**85.6**	**64.2**	**77.2**	**82.4**	**63.3**
C-WSL*:ODR	**85.8**	**81.2**	**64.9**	**50.5**	**32.1**	**84.3**	**85.9**	**54.7**	**43.4**	**80.1**	**42.2**	**42.6**	**60.5**	**90.4**	**13.7**	**57.5**	**82.5**	**61.8**	**74.1**	**82.4**	**63.5**
C-WSL:ODR+FRCNN	**85.8**	**78.0**	**61.6**	**52.1**	**44.7**	**81.7**	**88.4**	**49.1**	**50.0**	**82.9**	**44.1**	**44.4**	**63.9**	**92.4**	**14.3**	**60.4**	**86.6**	**68.3**	**80.6**	**82.8**	**65.6**
C-WSL*:ODR+FRCNN	**87.5**	**81.6**	**65.5**	**52.1**	**37.4**	**83.8**	**87.9**	**57.6**	**50.3**	**80.8**	**44.9**	**44.4**	**65.6**	**92.8**	**14.9**	**61.2**	**83.5**	**68.5**	**77.6**	**83.5**	**66.1**

Table 4. Comparison with baselines in terms of mAP on the VOC2007 *test* set. The table contains two comparison groups separated by double solid lines. Each group shows how much ADR and C-WSL improve each baseline. Underline is used if the C-WSL variant outperforms its baselines

Methods	are	bik	brd	boa	btl	bus	car	cat	cha	cow	tbl	dog	hrs	mbk	prs	plt	shp	sfa	trn	tv	mAP
WSLPDA [19]	54.5	47.4	41.3	20.8	17.7	51.9	63.5	46.1	21.8	57.1	22.1	34.4	50.5	61.8	16.2	29.9	40.7	15.9	55.3	40.2	39.5
WSLPDA+ADR	57.9	68.3	47.8	20.3	12.2	52.9	67.6	68.8	24.6	50.0	24.9	49.8	54.8	63.5	14.1	27.4	41.2	19.5	57.1	30.7	42.7
C-WSL:WSLPDA+ADR	**60.5**	**70.1**	**52.5**	**24.7**	**24.4**	**63.6**	**71.8**	**58.1**	**26.0**	**66.4**	**26.5**	**34.7**	**55.0**	**65.8**	**8.8**	**31.9**	**51.6**	**20.4**	**60.0**	**41.8**	**45.7**
OICR [35]	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
OICR+ADR	58.1	61.2	43.3	24.4	19.4	65.5	67.1	34.3	3.6	56.5	45.5	26.4	61.9	60.7	10.4	23.6	49.2	62.1	61.4	64.2	44.9
C-WSL:OICR+ADR	**61.7**	**66.8**	**45.6**	**21.1**	**23.5**	**67.2**	**73.8**	**32.5**	**10.6**	**54.6**	**42.9**	**16.6**	**59.2**	**63.3**	**11.0**	**25.4**	**55.3**	**61.3**	**67.4**	**67.8**	**46.4**

Table 5. Comparison with the baseline detectors in terms of CorLoc (%) on the VOC2007 *trainval* set. The table contains two comparison groups separated by double solid lines. Each group shows how much ADR and C-WSL improve each baseline. Underline is used if the C-WSL variant outperforms its baselines

Methods	are	bik	brd	boa	btl	bus	car	cat	cha	cow	tbl	dog	hrs	mbk	prs	plt	shp	sfa	trn	tv	Avg.
WSLPDA [19]	78.2	67.1	61.8	38.1	36.1	61.8	78.8	55.2	28.5	68.8	18.5	49.2	64.1	73.5	21.4	47.4	64.6	22.3	60.9	52.3	52.4
WSLPDA+ADR	84.6	76.9	69.7	41.0	21.8	68.5	83.2	77.6	34.4	76.7	19.8	73.7	75.2	84.7	26.3	53.8	70.1	22.3	73.8	50.9	59.2
C-WSL:WSLPDA+ADR	**83.3**	**80.0**	**70.9**	**51.6**	**41.2**	**73.6**	**85.3**	**67.7**	**40.7**	**79.5**	**20.9**	**54.7**	**79.6**	**87.1**	**24.5**	**56.8**	**83.5**	**20.7**	**76.0**	**60.2**	**61.9**
OICR [35]	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
OICR+ADR	85.8	76.9	65.8	49.5	38.5	83.2	84.8	49.7	14.0	79.5	46.8	41.2	80.3	89.2	15.0	60.1	84.5	66.4	78.3	80.6	63.5
C-WSL:OICR+ADR	**85.4**	**78.0**	**65.5**	**49.5**	**43.5**	**84.3**	**87.5**	**48.0**	**23.6**	**80.8**	**43.3**	**38.8**	**79.9**	**92.8**	**15.8**	**60.1**	**87.6**	**66.4**	**81.0**	**80.3**	**64.6**

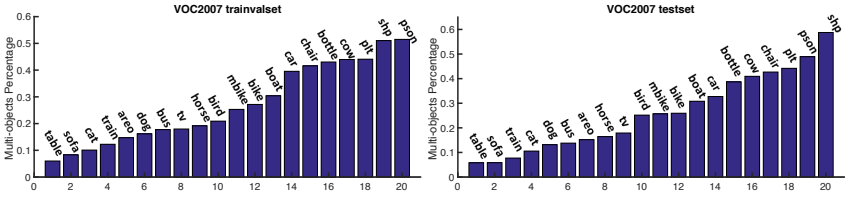


Fig. 5. Image number of multiple-objects over image number of non-zero objects. Note that "pson" means "person", "plt" means "plant" and "shp" denotes "sheep". C-WSL works better on most classes with high multiple-objects percentage. See Sec. 4.3

As stated in Sec. 1, the object count information is helpful to avoid a detector localizing on multiple objects. To demonstrate this point, we first calculate the percentage of images that have more than one per-class object (multi-objects percentage) in VOC2007. As shown in Fig. 5, "bottle", "car", "chair", "cow", "person", "plant" and "sheep" have a high percentage of images which include more than one object in the corresponding category. As shown in Tab. 2 and 3, *C-WSL:ODR+FRCNN* outperforms SOTA methods for 5 out of these 7 categories. When looking into the effect of object count supervision on WSLPDA and OICR, we see significant improvement on these categories as shown in Tab. 4 and 5. Consider the "sheep" category for example. *C-WSL:WSLPDA+ADR* improves *WSLPDA+ADR* by 13.4% *CorLoc* and 10.4% *AP*. *C-WSL:OICR+ADR* improves *OICR+ADR* by 3.1% *CorLoc* and 6.1% *AP*. Fig. 6 shows some examples of training regions selected by *OICR+CRS* and *OICR*. *OICR* tends to select regions containing multiple instances, while object count helps to obtain regions including a single instance. Qualitative comparison between our *C-WSL:ODR+FRCNN* and *OICR-Ens.+FRCNN* on the VOC2007 *test* set is shown in Fig. 8, demonstrating that our approach achieves more precise localization when multiple per-class objects appear in an image. We will further analyze our approach on images with different numbers of objects in Sec. 4.4.

Tab. 6 and 7 show the comparison of C-WSL with the SOTA on VOC2012. Note that results of WSLPDA and OICR models are reproduced by running the pretrained model and the code released by the authors. The results suggest that our method outperforms the SOTA method (*OICR-Ens.+FRCNN*) by 2.6% in *mAP* on the VOC2012 *val* set and by 2.8% in *CorLoc* on the VOC2012 *train* set. C-WSL improves the results of *WSLPDA+ADR* on 12 (10) out of 20 categories and the results of *OICR+ADR* on 10 (12) out of 20 categories in terms of *mAP* on the VOC2012 *val* set (in terms of *CorLoc* on the VOC2012 *train* set).

We also evaluated our methods and baselines (pre-trained on the VOC2007 trainval set) on the common 20 classes in MS COCO [22] 35k-val2014 set using COCO mAP@0.5 metric. Although not fine-tuned on COCO, our approaches still outperform the baseline methods. The results are that C-WSL:WSLPDA improves WSLPDA [19] from 17.9% to 19.6%. C-WSL:OICR+ADR improves OICR [35] from 18.7% to 20.1% and C-WSL:ODR+FRCNN improves OICR-Ens.+FRCNN [35] from 19.0% to 20.0%.

Table 6. Comparison with the state-of-the-art in terms of mAP on the VOC2012 *val* set. Our number is marked in red if it is the best in the column. Underline is used if the C-WSL variant outperforms its baselines

Methods	are	bik	brd	boa	btl	bus	car	cat	cha	cow	tbl	dog	hrs	mbk	prs	plt	shp	sfa	trn	tv	mAP
Jie *et al.* [14]	60.9	53.3	31.0	16.4	18.2	58.2	50.5	55.6	9.1	42.1	12.1	43.4	45.3	64.6	7.4	19.3	44.8	39.3	51.4	57.2	39.0
OICR-Ens.+FRCNN [35]	71.0	68.2	52.7	20.1	27.2	57.3	57.1	19.0	8.0	50.6	30.2	34.5	63.3	69.5	1.2	20.5	48.5	55.2	41.1	60.4	42.8
WSLPDA [19]	42.2	27.8	32.7	4.2	13.7	52.1	35.8	48.3	11.8	31.7	4.9	30.4	45.3	51.8	11.5	13.4	33.5	7.2	45.6	38.4	29.1
WSLPDA+ADR	70.0	65.6	46.3	14.4	22.8	57.5	54.2	67.5	16.1	45.0	4.4	40.0	51.7	71.8	5.8	27.7	38.3	11.7	55.2	34.1	40.0
C-WSL:WSLPDA+ADR	**69.8**	**62.8**	**52.7**	**16.7**	**28.3**	**61.1**	**56.6**	**58.0**	**18.5**	**47.8**	**5.1**	**36.3**	**53.3**	**66.8**	**6.8**	**24.2**	**47.1**	**11.0**	**60.1**	**43.4**	**41.3**
OICR [35]	71.0	59.1	42.3	27.4	20.2	58.7	46.4	18.6	18.1	45.7	21.7	20.5	53.1	68.5	1.8	15.7	42.7	40.0	41.0	61.5	38.7
OICR+ADR	67.0	63.1	50.8	12.8	23.8	55.3	55.1	16.1	5.2	47.2	23.4	28.2	55.9	69.2	1.9	21.5	46.5	49.9	35.9	63.8	39.6
C-WSL:OICR+ADR	**71.3**	**68.3**	**50.9**	**17.1**	**24.8**	**60.9**	**56.4**	**13.9**	**14.5**	**54.6**	**22.2**	**25.7**	**57.7**	**70.4**	**1.6**	**20.0**	**55.8**	**46.0**	**35.7**	**62.9**	**41.5**
C-WSL:ODR	**74.0**	**67.3**	**45.6**	**29.2**	**26.8**	**62.5**	**54.8**	**21.5**	**22.6**	**50.6**	**24.7**	**25.6**	**57.4**	**71.0**	**2.4**	**22.8**	**44.5**	**44.2**	**45.2**	**66.9**	**43.0**
C-WSL:ODR+FRCNN	**75.3**	**71.6**	**52.6**	**32.5**	**29.9**	**62.9**	**56.9**	**16.9**	**24.5**	**59.0**	**28.9**	**27.6**	**65.4**	**72.6**	**1.4**	**23.0**	**49.4**	**52.3**	**42.4**	**62.2**	**45.4**

Table 7. Comparison with the state-of-the-art in terms of *CorLoc* on the VOC2012 *train* set. Our number is marked in red if it is the best in the column. Underline is used if the C-WSL variant outperforms its baselines

Methods	are	bik	brd	boa	btl	bus	car	cat	cha	cow	tbl	dog	hrs	mbk	prs	plt	shp	sfa	trn	tv	Avg.
OICR-Ens.+FRCNN [35]	85.4	81.5	70.4	44.7	46.6	83.6	78.4	33.9	29.3	83.2	51.6	50.5	86.1	88.0	11.0	56.7	82.5	69.1	65.1	83.6	64.1
WSLPDA [19]	80.5	63.7	64.4	34.1	29.3	76.7	71.5	62.8	30.3	76.1	23.0	55.3	75.2	77.7	18.7	56.4	66.7	25.1	66.5	54.8	55.4
WSLPDA+ADR	87.2	79.7	72.4	38.6	40.9	82.6	75.2	79.8	35.1	81.3	18.9	62.1	82.4	83.9	21.6	60.9	75.4	29.5	74.5	55.5	61.9
C-WSL:WSLPDA+ADR	**85.7**	**77.2**	**73.4**	**38.6**	**46.4**	**84.9**	**75.8**	**69.1**	**43.0**	**76.8**	**20.1**	**58.6**	**79.8**	**79.6**	**20.3**	**57.8**	**79.5**	**35.4**	**76.4**	**61.9**	**62.0**
OICR [35]	86.6	80.4	65.2	57.6	42.1	85.4	72.5	28.0	45.7	79.4	46.2	34.0	78.2	87.2	7.5	55.0	83.6	58.5	62.2	84.3	62.0
OICR+ADR	84.5	79.0	72.4	39.0	47.1	83.6	79.9	31.9	25.0	84.5	48.7	48.3	87.8	88.7	13.3	55.0	82.5	67.4	65.1	83.9	63.4
C-WSL:OICR+ADR	**86.6**	**80.8**	**73.9**	**43.2**	**44.4**	**87.7**	**76.2**	**32.2**	**34.0**	**87.1**	**49.1**	**46.2**	**88.2**	**91.2**	**12.1**	**57.1**	**78.4**	**65.5**	**65.1**	**85.3**	**64.2**
C-WSL:ODR	**90.9**	**81.1**	**64.9**	**57.6**	**50.6**	**84.9**	**78.1**	**29.8**	**49.7**	**83.9**	**50.9**	**42.6**	**78.6**	**87.6**	**10.4**	**58.1**	**85.4**	**61.0**	**64.7**	**86.6**	**64.9**
C-WSL:ODR+FRCNN	**92.1**	**84.3**	**69.9**	**58.3**	**53.9**	**86.8**	**80.4**	**30.6**	**52.6**	**83.9**	**54.7**	**45.8**	**83.2**	**90.1**	**12.7**	**56.4**	**86.0**	**64.9**	**66.5**	**84.3**	**66.9**

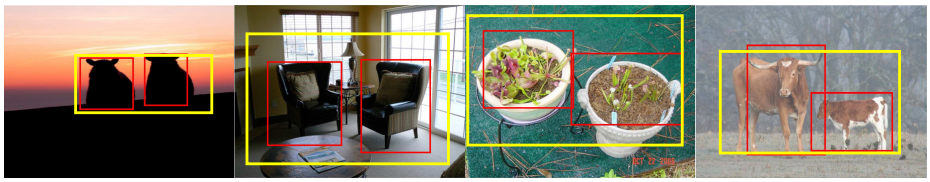


Fig. 6. Examples of the training regions selected by *OICR+CRS* (red) and *OICR* (yellow). The regions selected by *OICR* contain multiple object instances. Object count information helps to select regions, each covering a single instance

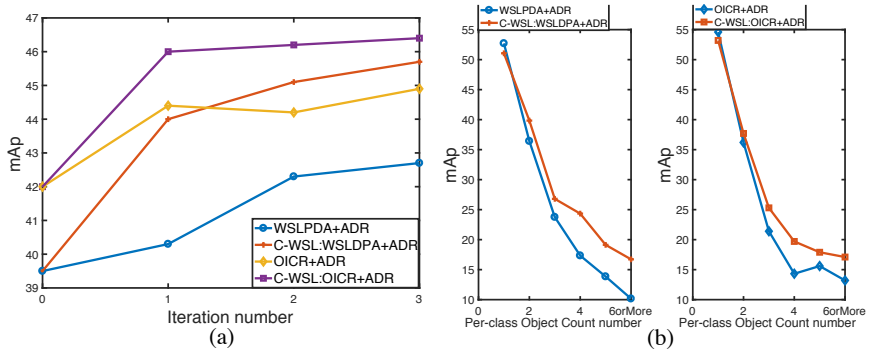


Fig. 7. (a): model improvement as the number of *ADR* iterations increases on the VOC2007 *test* set. *C-WSL* approaches improve faster than others. (b): Evaluation on images with different per-class object counts on VOC2007. Our approach outperforms the WSL detectors in the presence of multiple instances in a test image

4.4 Ablation Analysis

Two major components contribute to the success of our approach. One is the iterative training process (alternating/online) and the other one is the per-class object count supervision. In Tab. 4 and 5, we can see the improvement by adding ADR and object count into the system. For WSLPDA [19], iterative training (ADR) improves *mAP* by 3.2% and the count information (CRS) increases it by 3%. For OICR [35], ADR helps by increasing 3.7% *mAP* and CRS contributes 1.5%. In the following, we analyze each component in detail.

Number of iterations. ADR performances as a function of the number of iterations using the WSLDPA and OICR models is shown in Fig. 7(a). Generally, models improve as the number of iterations increases. When adding object count supervision into the framework, the results of both WSLDPA and OICR models improve faster, which demonstrates the advantage of count information in WSL.

Number of object instances per image. Adding the object count constraint helps a detector focus on a single object rather than multiple objects. To demonstrate this, we partition images in the VOC2007 *test* set based on their per-class object count and re-evaluate our approaches on each subset.

The results are shown in Fig. 7(b). For both WSLPDA and OICR, the performance is much better under C-WSL. Generally, the gaps between curves of with and without C-WSL are bigger as the object count number increases.

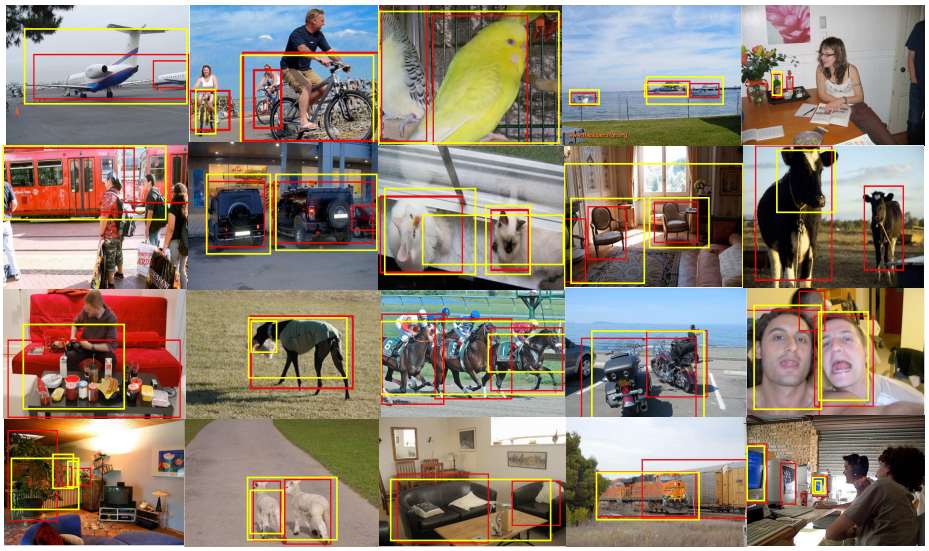


Fig. 8. Qualitative comparison between our *CWSL:ODR+FRCNN* (red boxes) and *OICR+FRCNN* (yellow boxes) on the VOC2007 *test* set over the 20 classes. Our detector detects much tighter bounding boxes, yields much fewer boxes with multiple objects in them, and finds instances more accurately

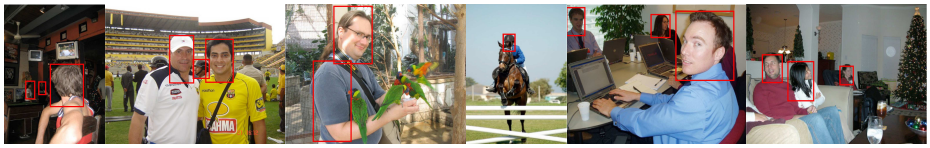


Fig. 9. Some examples of the common failure cases of our approach (*C-WSL:ODR+FRCNN*) on the "person" category of the VOC2007 *test* set

4.5 Error Analysis

The results shown in Tab. 2, 3, 6 and 7 suggest that most existing WSL detectors perform poorly on the "person" category: strongly supervised detectors achieve more than 76% AP on the VOC2007 *test* set (*e.g.*, 76.6% [23] and 76.3% [29]), while the best WSL detection result on "person" is 20.3% (see Tab. 2). This result is likely due to the large appearance variations of persons in the dataset. Without constraints provided by tight bounding boxes, rigid parts are easier to learn and mostly sufficient to differentiate the object from others. So, WSL detectors focus on local parts instead of the whole object as shown in Fig. 9.

Intuitively, this can be overcome if we can roughly estimate the size of object instances. We conducted a preliminary experiment as follows. Suppose that we know the size of the smallest instance of an object category in an image and assume all the object parts are smaller than the smallest object. This assumption is not generally true and we use it just as a proof-of-concept. We preprocess the region candidates by removing all boxes whose size is smaller than the smallest object and then conduct *C-WSL:WSLPDA+ADR* on VOC2007. The *AP* on "person" improves to 40.0% and the *mAP* over all the classes improves to 52.7%.

5 Conclusions

We proposed a Count-guided Weakly Supervised Localization (C-WSL) framework where a cheap and effective form of image-level supervision, *i.e.*, per-class object count, is used to select training regions each of which tightly covers a single object instance for detector refinement. As a part of C-WSL, we proposed a Count-based Region Selection (CRS) algorithm to perform high-quality region selection. We integrated CRS into two detector refinement architectures to improve WSL detectors. Experimental results demonstrate the effectiveness of C-WSL. To prove the inexpensiveness of the per-class object count annotation, we conduct annotation experiments on VOC2007. The results show that only a small amount of time is needed to obtain the count information in an image and that we reduce the annotation time of center click and bounding box by more than 2× and 38× respectively.

Acknowledgement. The research was supported by the Office of Naval Research under Grant N000141612713: Visual Common Sense Reasoning for Multi-agent Activity Prediction and Recognition. The authors would like to thank Eddie Kessler for proofreading the manuscript.

References

1. Bilen, H., Vedaldi, A.: Weakly supervised deep detection networks. In: The IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (June 2016)
2. Chattopadhyay, P., Vedantam, R., RS, R., Batra, D., Parikh, D.: Counting everyday objects in everyday scenes. CVPR (2017)
3. Cinbis, R.G., Verbeek, J., Schmid, C.: Weakly supervised object localization with multi-fold multiple instance learning. IEEE transactions on pattern analysis and machine intelligence **39**(1), 189–203 (2017)
4. Clements, D.H.: Subitizing: What is it? why teach it? Teaching children mathematics **5**(7), 400 (1999)
5. Diba, A., Sharma, V., Pazandeh, A., Pirsiavash, H., Van Gool, L.: Weakly supervised cascaded convolutional networks. In: Proceedings of the 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR). pp. 5131–5139 (2017)
6. Dietterich, T.G., Lathrop, R.H., Lozano-Pérez, T.: Solving the multiple instance problem with axis-parallel rectangles. Artificial intelligence **89**(1), 31–71 (1997)
7. Dollár, P., Tu, Z., Perona, P., Belongie, S.: Integral channel features. In: BMVC (2009)
8. Dollar, P., Wojek, C., Schiele, B., Perona, P.: Pedestrian detection: An evaluation of the state of the art. IEEE transactions on pattern analysis and machine intelligence **34**(4), 743–761 (2012)
9. Everingham, M., Van Gool, L., Williams, C.K.I., Winn, J., Zisserman, A.: The PASCAL Visual Object Classes Challenge 2007 (VOC2007) Results. http://www.pascal-network.org/challenges/VOC/voc2007/workshop/index.html
10. Everingham, M., Van Gool, L., Williams, C.K.I., Winn, J., Zisserman, A.: The PASCAL Visual Object Classes Challenge 2012 (VOC2012) Results. http://www.pascal-network.org/challenges/VOC/voc2012/workshop/index.html
11. Everingham, M., Van Gool, L., Williams, C.K., Winn, J., Zisserman, A.: The pascal visual object classes (voc) challenge. International journal of computer vision **88**(2), 303–338 (2010)
12. Gao, M., Yu, R., Li, A., Morariu, V.I., Davis, L.S.: Dynamic zoom-in network for fast object detection in large images. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (2018)
13. Girshick, R.: Fast r-cnn. In: Proceedings of the IEEE International Conference on Computer Vision. pp. 1440–1448 (2015)
14. Jie, Z., Wei, Y., Jin, X., Feng, J., Liu, W.: Deep self-taught learning for weakly supervised object localization. IEEE CVPR (2017)
15. Kantorov, V., Oquab, M., Cho, M., Laptev, I.: Contextlocnet: Context-aware deep network models for weakly supervised localization. In: European Conference on Computer Vision. pp. 350–365. Springer (2016)
16. Kim, D., Yoo, D., Kweon, I.S., et al.: Two-phase learning for weakly supervised object localization. Proc. IEEE Int. Conf. Comput. Vis.(ICCV) (2017)
17. Kolesnikov, A., Lampert, C.H.: Improving weakly-supervised object localization by micro-annotation. BMVC (2016)
18. Krishna, R.A., Hata, K., Chen, S., Kravitz, J., Shamma, D.A., Fei-Fei, L., Bernstein, M.S.: Embracing error to enable rapid crowdsourcing. In: Proceedings of the 2016 CHI conference on human factors in computing systems. pp. 3167–3179. ACM (2016)
19. Li, D., Huang, J.B., Li, Y., Wang, S., Yang, M.H.: Weakly supervised object localization with progressive domain adaptation. In: The IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (June 2016)

20. Lin, T.Y., Dollár, P., Girshick, R., He, K., Hariharan, B., Belongie, S.: Feature pyramid networks for object detection. In: CVPR (2017)
21. Lin, T.Y., Goyal, P., Girshick, R., He, K., Dollár, P.: Focal loss for dense object detection. In: ICCV (2017)
22. Lin, T.Y., Maire, M., Belongie, S., Hays, J., Perona, P., Ramanan, D., Dollár, P., Zitnick, C.L.: Microsoft coco: Common objects in context. In: European conference on computer vision. pp. 740–755. Springer (2014)
23. Liu, W., Anguelov, D., Erhan, D., Szegedy, C., Reed, S., Fu, C.Y., Berg, A.C.: SSD: Single shot multibox detector. In: ECCV (2016)
24. Oquab, M., Bottou, L., Laptev, I., Sivic, J.: Is object localization for free?-weakly-supervised learning with convolutional neural networks. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 685–694 (2015)
25. Papadopoulos, D.P., Uijlings, J.R., Keller, F., Ferrari, V.: We don't need no bounding-boxes: Training object class detectors using only human verification. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 854–863 (2016)
26. Papadopoulos, D.P., Uijlings, J.R., Keller, F., Ferrari, V.: Training object class detectors with click supervision. CVPR (2017)
27. Redmon, J., Divvala, S., Girshick, R., Farhadi, A.: You only look once: Unified, real-time object detection. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 779–788 (2016)
28. Redmon, J., Farhadi, A.: YOLO9000: better, faster, stronger. In: 2017 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2017, Honolulu, HI, USA, July 21-26, 2017. pp. 6517–6525 (2017)
29. Ren, S., He, K., Girshick, R., Sun, J.: Faster r-cnn: Towards real-time object detection with region proposal networks. In: Advances in neural information processing systems. pp. 91–99 (2015)
30. Shi, M., Caesar, H., Ferrari, V.: Weakly supervised object localization using things and stuff transfer. Proc. IEEE Int. Conf. Comput. Vis.(ICCV) (2017)
31. Simonyan, K., Zisserman, A.: Very deep convolutional networks for large-scale image recognition. CoRR **abs/1409.1556** (2014)
32. Singh, B., Davis, L.S.: An analysis of scale invariance in object detection–snip. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 3578–3587 (2018)
33. Singh, B., Li, H., Sharma, A., Davis, L.S.: R-fcn-3000 at 30fps: Decoupling detection and classification. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 1081–1090 (2018)
34. Singh, K.K., Lee, Y.J.: Hide-and-seek: Forcing a network to be meticulous for weakly-supervised object and action localization. The IEEE International Conference on Computer Vision (ICCV) (2017)
35. Tang, P., Wang, X., Bai, X., Liu, W.: Multiple instance detection network with online instance classifier refinement. CVPR (2017)
36. Wang, C., Ren, W., Huang, K., Tan, T.: Weakly supervised object localization with latent category learning. In: European Conference on Computer Vision. pp. 431–445. Springer (2014)
37. Yang, F., Choi, W., Lin, Y.: Exploit all the layers: Fast and accurate cnn object detector with scale dependent pooling and cascaded rejection classifiers. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 2129–2137 (2016)
38. Yu, R., Chen, X., Morariu, V.I., Davis, L.S.: The role of context selection in object detection. In: British Machine Vision Conference (BMVC) (2016)

39. Yu, R., Li, A., Morariu, V.I., Davis, L.S.: Visual relationship detection with internal and external linguistic knowledge distillation. IEEE International Conference on Computer Vision (ICCV) (2017)
40. Zhu, Y., Zhou, Y., Ye, Q., Qiu, Q., Jiao, J.: Soft proposal networks for weakly supervised object localization. In: Proc. IEEE Int. Conf. Comput. Vis.(ICCV). pp. 1841–1850 (2017)