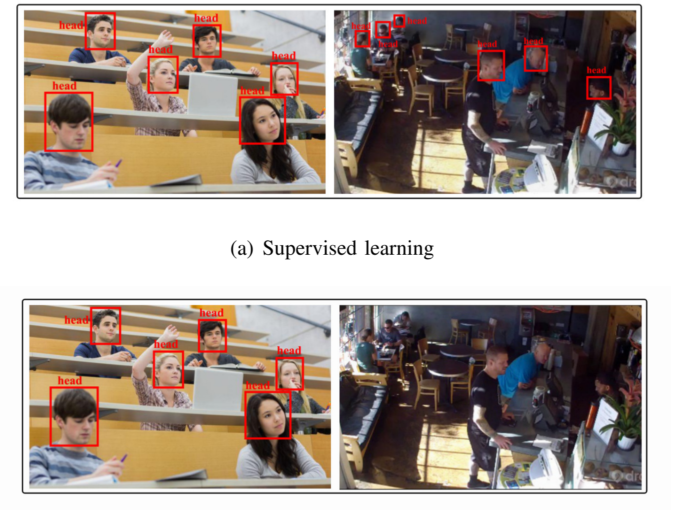
Towards Precise End-to-end Semi-Supervised Human Head Detection Network

Rongchun Li*, Junjie Zhang*, Yuntao Liu†, Yong Dou‡
National Laboratory for Parallel and Distributed Processing
National University of Defense Technology
Changsha Hunan, 410073, China
{*rongchunli, ‡yongdou}@nudt.edu.cn, {*junjie.zhang.cs, †liuyuntao.me}@gmail.com

***Abstract*—Head detection, as a fundamental task in practice for many head-related problems, requires an enormous number of annotated boxes to maintain the performance. To alleviate the time and cost of labeling each image in the dataset, we propose an end-to-end semi-supervised head detection framework, which shows competitive results with only a small set of data. Specifically, under the setting of semi-supervised, we introduce a weak boxes generate branch and a weak boxes refine branch to produce pseudo ground truth label for unlabeled images with the guidance of annotated images. The weak boxes generate branch is embedded in the detection framework taking the proposals as input and outputting the initial weak boxes that coarsely locate the place of the head. Then, the weak boxes refine branch adjusts the weak boxes more accurate gradually by training a transferred sub-network with the established relation between proposals, weak boxes and labeled boxes. In the training process, we jointly train the two branches in an end-to-end manner, which can generate better pseudo bounding boxes with a small dataset online to avoid over-fitting and obtain a more precise head detector. The results on the public head detection benchmark Brainwash and SCUT-HEAD show the effectiveness of our method.**

***Index Terms*—semi-supervise, head detection**



(a) Supervised learning

(b) Semi-supervised learning

Fig. 1. Different types of head detection settings

I. Introduction

Head detection, which is a sub-task of object detection, is fundamental in many head-related problems such as visual tracking [20], action recognition [3], and crowd understanding [23], aiming to localize spatial extents of all heads. The state-of-the-art head detection methods typically trained deep Convolutional Neural Networks(CNNs) from large scale datasets with bounding boxes labels [8] [2] [7]. As shown in Fig. 1(a), the annotations in the level of the bounding box are essential to the effectiveness of these methods. However, It is known that it takes about 10 seconds to label an object [15]. As a result, labeling for detection tasks with bounding box locations requires enormous cost and time.

To reduce the cost of such labeling, we mainly study how to learn a precise head detector using unlabeled data in combination with the box-level labeled data, as shown in Fig. 1(b). This problem can be thought of as a subtask of semi-supervised object detection. Compared to object detection, head detection only contains one category. But there still some similarities between the two tasks. The authors in [11] try to solve the incomplete boxes problems in weakly supervised detection by learning a box correction network from a small portion of box annotations. They have improved the performance of semi-supervised object detection. However, in head detection, suitable additional image-level labels in their works are not available. Besides, Wang [19] utilize high-confident samples with pseudo-labels in training to provide more data. The disadvantage of the method is large time consumption because of their iterative training process. Recently, Jeong [6] propose a novel consistency based semi-supervised learning algorithm, which creatively uses horizontally flipped images to avoid the inappropriate place when applying consistency regularization to object detection. But it still has a limitation that the distribution of unlabeled data should be similar to that of the labeled data.

Follow semi-supervising setting in [19], we define the semi-supervised head detection task is: given a large dataset without any labels, and only a small subset of it has bounding box annotation of the head, testing the performance by labeled images.

In this paper, we propose an end-to-end semi-supervised

This research is founded by the National Key R&D Program of China [grant numbers 2018YFB1003400] and the National Natural Science Foundation of China [grant numbers 61802419].

978-1-7281-6926-2/20/$31.00 ©2020 IEEE

head detection framework called Weak Boxes Generate and Refine network(WBGR), which can generate pseudo ground truth for unlabeled images and train the detector online. Here, the weak boxes are those boxes predict by our proposed embedded network. Furthermore, we utilize the partially annotated bounding boxes as strong boxes to guide the refined process of the weak boxes. As shown in Fig. 2, our method consists of three components, a detection branch as the base, a weak boxes generate(WBG) branch to initialize the weak boxes, and a weak boxes refine(WBR) branch to adjust the boxes. Specifically, we used a typical two-stage detector Faster R-CNN to generate proposals and trained the detector by the annotated and pseudo labels. The WBG branch, which takes the proposals without post-processing as input and outputs weak boxes, is a simple classification and bounding-box regression branch embedded in the based detection backbone. To further enhance the quality of weak boxes, the WBR branch utilizes annotated bounding boxes to transfer the weak boxes more accurate by a greedy method and a regression net. During the training process, we make the image with annotation and unlabeled as a pair. The generated pseudo ground truth from our branch is online training together with annotated bounding boxes to alleviate the detector over-fitting into the small subset and maintain the generalization of the model.

Our contributions can be summarized as follows.

- We design an end-to-end semi-supervised head detection network than can optimize the head detector with unlabelled images by WBG and WBR branch generating high-quality pseudo ground truth.
- Our proposed network significantly outperforms the original model on head detection benchmarks Brainwash and SCUT-HEAD.

II. Related Work

A. Head Detection

Object detection, including head detection algorithms, can be divided into two categories, depending on whether the detector does classification and regression twice. Single-stage detectors usually process the proposals to output boxes directly with the consideration of balance training strategies, such as SSD [10] and Retinanet [9]. Two-stage detectors are region proposed network(RPN) based algorithms, which generate a set of proposed regions and modifies them further, represented by Faster RCNN [14]. Due to the better performance of two-stage detectors, we utilize them as the baseline.

Furthermore, as a subtask of object detection, head detection has inherited some methods from generic object detection. Chen [2] present an approach that learns a semantic connection between the head and body parts. Additional annotation makes this method applicable to a few datasets. After that, to alleviate the problem of false alarms, Li [7] add a saliency attention network on the two-stage detector with a feature fusion strategy. Besides, Li [8] propose an adaptive relational framework with the local relation and global priors. To verify the effectiveness of our method under a semi-supervised setting, we use Faster R-CNN with adjusted hyper-parameters for head detection as our detection branch.

Moreover, based on the two-stage detector, multi-stage works represented by Cascade R-CNN [1] and Cascade RPN [24] have appeared recently. They perform a cascade process on the first and second stage of the two-stage detector, respectively. Cascade R-CNN is trained with increasing IOU thresholds to avoid the problems of overfitting and mismatch. Different from Cascade R-CNN, Cascade RPN optimize the anchor generation strategy and utilize generated proposals to guide the feature learning process. Motivated by these two work, our WBG branch generated weak boxes with an uncomplicated cascading network.

B. Semi supervised Detection

Since we have not found work for semi-supervised head detection, in this section, we mainly introduce semi-supervised learning methods for object detection. Most of the works in this area are divided into two kinds. One focuses on combining image-level labels and instance-level labels in object detection, such as [17], [21], [18], [11]. Another is based on the self-training scheme with completed unlabeled images, such as [19], [6], which we follow due to the useless of image-level labels in the head detection task. Wang [19] stitch high confidence patches from unlabeled data to labeled data. Jeong [6] utilize consistency regularization to fine-tune the location of the predicted box. Our method also focuses on how to use the unlabeled images, but with end-to-end online training strategy.

III. Method

In this section, we firstly present an overview of our network architecture, then elaborate weak boxes generation(WBG) branch and weak boxes refine(WBR) branch in details, finally give the training details .

A. Overall Architecture

The overall architecture of proposed semi-supervised object detection network is shown in Fig. 2, which consists of three major components: a two-stage detection backbone, WBG branch to generate weak boxes and WBR branch to refine the coarse weak boxes more accurate.

We split all images $\mathcal{I}$ into two parts, $\mathcal{I} = \mathcal{A} \cup \mathcal{B}$ and $\mathcal{A} \cap \mathcal{B} = \emptyset$, where images in small part A(i.e. 10%) has box annotations while images in B don't contain any annotations. For images in $\mathcal{A}$, their manually labeled bounding boxes are denoted as strong boxes, b_s. The pseudo bounding boxes generated by WBG branch are denoted as weak boxes, b_w.

In the training process, our method starts with preparing paired training images which contains an image form A and an image from B. Then, we extract the features of the images by a deep full conventional network(FCN). The region proposal network(RPN) is built on the top of the FCN, which produce transferred anchors and postprocessing proposals. Continuously, in order to generate original weak boxes, we utilize a simple WBG branch which is shown in the yellow

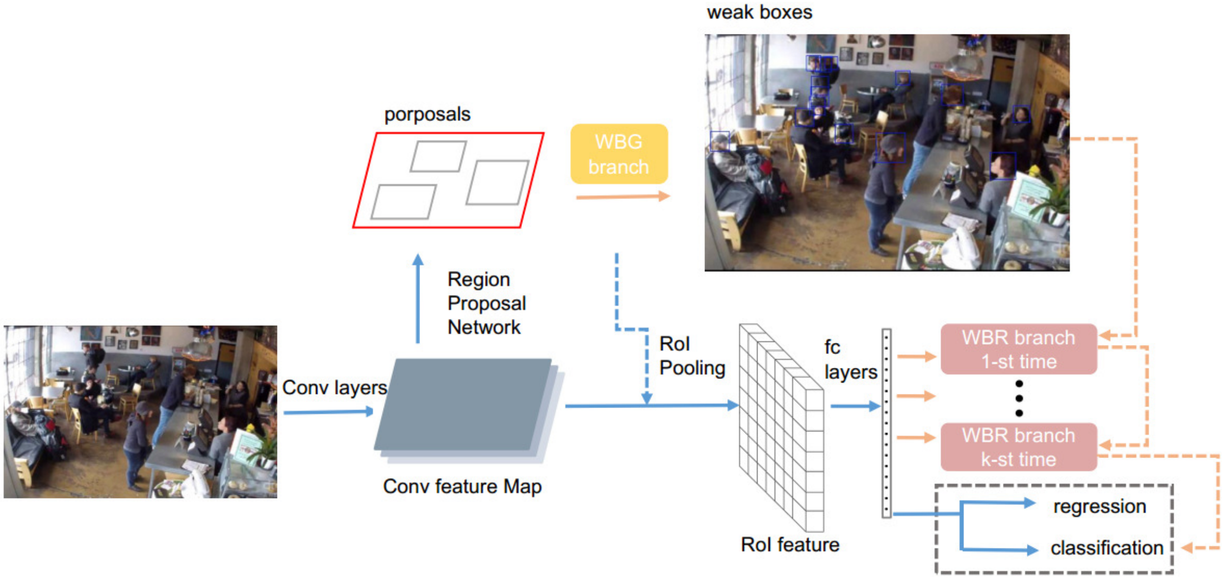


Fig. 2. An overview of our WBGR architecture. (1)Generate proposals and RoI features from conv feature map by RPN and RoI pooling. (2)WBG branch: Feed the proposals not yet post-processed into a simple network for weak boxes annotation initialization. (3) WBR branch: Feed the RoI features and generated weak boxes to the branch for weak boxes refine. (4) Perform proposals classification and regression with the supervision of weak and strong boxes.

part in Fig. 2. The WBG branch is trained by existed strong boxes and takes the transferred proposals without non-maxima suppression or eliminate and incoming feature maps for RPN as input.

After generating the weak boxes, we utilize the region of interest(ROI) pooling to extract the feature of proposals from RPN. Our weak boxes refine branch will match the proposals with the weak boxes and existing strong boxes, and gradually optimize the weak ones. The weak boxes refine branch is shown in the pink part in Fig. 2, which get roughly weak boxes and output more accurate weak boxes. Finally, adjusted weak boxes will be employed as fully supervised labels for classifier and regressor in detection backbone. The weak boxes can be seen as extra samples to make the network training more adequate and robust. More details will be introduced respectively in remain sections.

B. WBG Branch

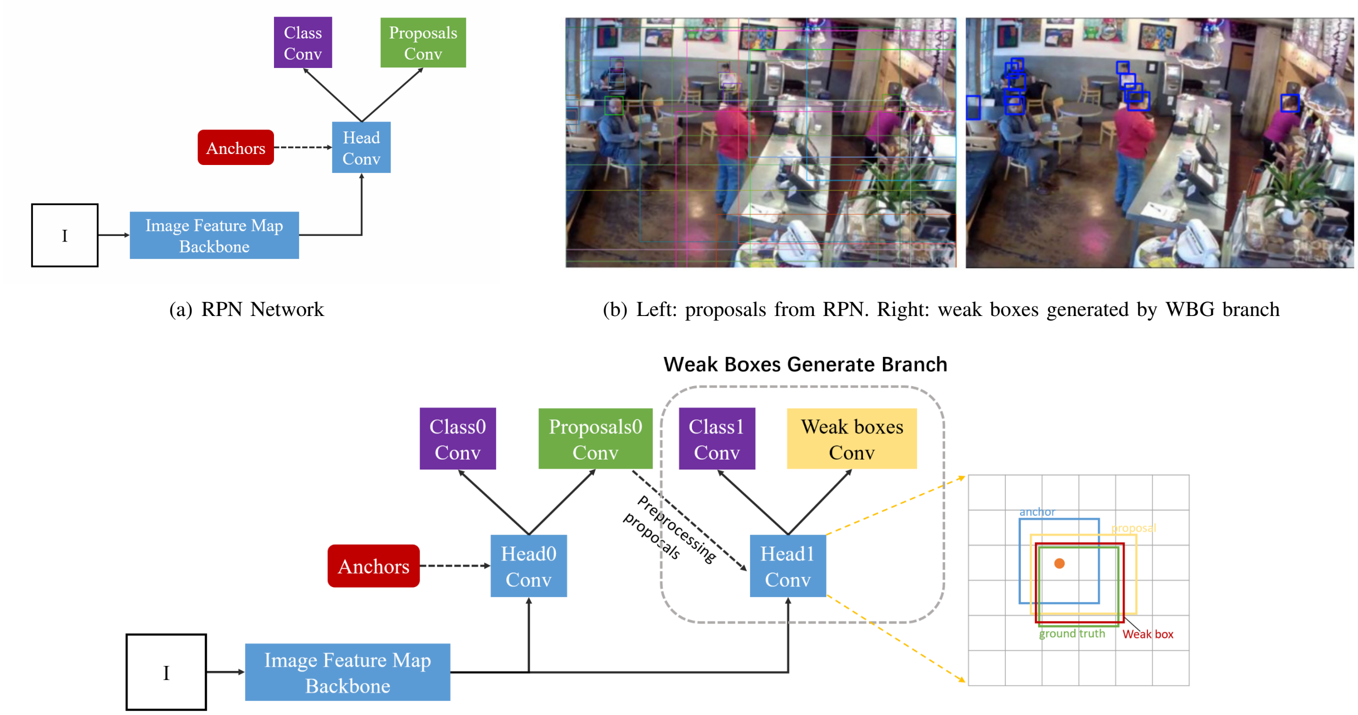
In semi-supervised object detection setting, we only have a part of images with box-level labels. To train a standard object detection with regression, previous works introduce a MIL learning method to initialize the pseudo GT annotations. In [11], the authors utilize a pretrained weakly supervised object detection model and use it to generate weak boxes for all images. However, head detection is different from generic object detection. There are only one category in head detection, making it challenging to guide the score of the proposals by the distribution of category scores. As a result, we think of using the existing partial annotation to generate roughly weak boxes for the remaining unlabeled pictures.

In the following, we introduce two methods to generate weak boxes.

1) Generating weak boxes using a decouple two-stage detector: A solution for generating weak boxes is to pretrain a two-stage head detector like Faster RCNN with the existing partial label in Image set $\mathcal{A}$. After the training process of network completed, we select the images without bounding box labels as input and utilize a score threshold and top K number of boxes to filter the network output as the generated weak boxes.

2) Generating weak boxes using a simple embedded network: Although we can generate initial weak boxes using the decouple head detector, this method requires alternating optimization, and it is difficult to directly train a convergent network when there are fewer samples. Therefore, we propose a new method to generate weak boxes utilising an embedded network which can be trained end-to-end.

Our method is motivated by the work [24] and [1]. They propose a cascaded bounding box regression framework to get high-quality results and a cascade architecture to refine score and location, respectively. Different from their work, we incorporate a cascade-style network to generate weak boxes with the extracted base feature map and proposals from RPN. As shown in Fig. 3(c), the left part is a standard RPN, which utilizes a sliding network to predict multiple region proposal from anchors. Based on RPN, we add our lightweight architecture, WBG branch, which is in the middle part of Fig. 3(c) framed by a dotted line. After post-processing such as sorting and non-maximum suppression and truncating in number, the output of our WBG branch will be considered as weak boxes in our setting.



(a) RPN Network

(b) Left: proposals from RPN. Right: weak boxes generated by WBG branch

(c) WBG Branch

Fig. 3. Components and examples in WBG Branch. (a) The original RPN network. (b)The weak boxes are less and more accurate than the proposals. (c)Our lightweight WBG Branch's architecture.

The architecture of our WBG branch is a fully convolutional network. The WBG branch takes as input an $n \times n$ spatial window of the input feature map. Each window is mapped to a lower-dimensional feature(1024-d to 512-d for ResNet). Then, we feed the feature into two 1×1 convolution layers: a box classification layer and a box regression layer. The scores and deltas are corresponding to the proposals form RPN. After box transferred, we utilize a more strict filter with higher IOU and score threshold, and a smaller top K truncating number. This lightweight network is illustrated in Fig. 3(c). Our network structure is similar to RPN shown in Fig. 3(a), but with entirely different purposes (generating weak boxes) and stricter restrictions to filter the resulting boxes. Proposals from RPN and initial weak boxes are compared in the Fig. 3(b).

C. WBR Branch

After the weak boxes are generated, we design a weak box refine branch to make the initial weak boxes more accurate and can be used as training examples of detection head.

Here we will expound how to refine the weak boxes. A natural way to improve weak boxes quality is an alternative strategy, that is, fixing the weak boxes and training a regressor for transffering weak boxes to strong boxes, fixing the regressor and generating refined boxes. But is have a limitation: the number of weak boxes will generally be few, which is challenging to train the network till convergence. Hence we integrate the proposals, weak boxes and strong boxes and group them in one-to-one correspondence as samples for our branch. After we obtained the trained regressor, we convert those weak boxes without corresponding strong boxes into more accurate ones.

To make the weak boxes refine branch more clear, we summarize the process to obtain more accurate weak boxes in Algorithm 1. In training iteration, each proposal is assigned a corresponding weak box which owns the largest overlap. If the largest overlap is smaller than a threshold, this pair will be removed. Furthermore, the weak box will be assigned to a strong box in the same way. Then we can map a proposal to a strong box and calculate the offset which is used to train the weak-to-strong regressor indirectly. The offset targets are similar to the regression in Faster RCNN while we employ it to transfer boxes. Besides, the architecture of our WBR branch can be implemented as a fully connected neural network. As the box-annotated set $\mathcal{A}$ is a small part of $\mathcal{I}$, we also add regressor together with the classification of proposals under the supervised of weak boxes. After obtaining supervision and the architecture, we can get the loss of transfer function to learn the WBR branch.

$$\begin{aligned} L_{WBR}\left(p_i, p_w, t_i, t_{si}\right) = & \frac{1}{N_{cls}} \sum_i L_{cls}\left(p_i, p_{wi}\right) \\ & + \lambda \frac{1}{N_{reg}} \sum_i p_{wi} L_{reg}\left(t_i, t_{si}\right) \end{aligned} . \quad (1)$$

Algorithm 1 Weak boxes refine branch process

```
Input:
    The conv image feature map $X$ and its proposals;
    The roughly weak boxes $b_w$ and strong boxes $b_s$;
Output:
    More accurate weak boxes $b_w^{'}$;
Feed $X$ and its proposals into the network with weak boxes $b_w$ and labeled strong boxes $b_s$ to produce refined weak boxes $b_w^{'}$;
for $i = 0 \to K - 1$ do
    Get the weak boxes $b_w$ from WBG branch or last WBR instance
    for $r = 0 \to |R| - 1$ do
        Compute IoU vectors $\mathbf{I} = \left[I_1, \ldots, I_{|N|}\right]$ between proposal $j_r$ and weak boxes $b_w$.
        Choose the top-scoring IOU $I_{rn}$.
        Compute IoU vectors $\mathbf{S} = \left[S_1, \ldots, S_{|M|}\right]$ between weak box $b_w n$ and strong boxes $b_s$.
        Choose the top-scoring IOU $S_{nm}$.
        Compute the deltas $d_r$ between $j_r$ and strong box $b_{sm}$
    end for
    Train the weak boxes refine network by deltas vector $\mathbf{D} = \left[D_1, \ldots, D_{|R|}\right]$
    for $n = 0 \to N - 1$ do
        Transfer weak boxes $b_{wn}$ to $b_{wn}^{'}$ by trained network
    end for
end for
return $b_w^{'}$;
```

$$L_{reg}(t, t_s) = \sum_{j \in \{x,y,w,h\}} \operatorname{smooth}_{L1}\left(t^j - t^j\right), \tag{2}$$

in which i is the index of a proposal, p_i is the class prediction of proposal i and p_{wi} is the class label from weak boxes, t_i is the predicted offset and t_{si} is the offset targets calculated from strong boxes. The regression loss for our refine branch is the smooth L1 loss, which is not sensitive to outliers with smaller gradient changes.

D. Multi-task training

Our method can be trained in an end-to-end manner using multi-task loss which contains following composite loss function from the four components with stochastic gradient descent:

$$\mathcal{L} = \mathcal{L}_{WBG} + \mathcal{L}_{WBR} + \mathcal{L}_{rpn} + \mathcal{L}_{det}. \tag{3}$$

Here, $\mathcal{L}_{rpn}$ and $\mathcal{L}_{det}$ are multi-task loss of detection subnetwork. Since we get the annotations from strong boxes in A and refined weak boxes in B, each proposal now has a ground truth bounding-box regression target and classification target. For localization, smooth L1 loss is uesd. For classification, a binary cross entropy loss is used. $\mathcal{L}_{WBR}$ is the weak boxes refinement loss of WBR branch aforementioned.

For $\mathcal{L}_{WBG}$, as mentioned in Section. III-B, the weak boxes generation branch has two sibling branches. The first branch predicts a discrete probability distribution, $p \in \mathbb{R}^{(1+1)\times 1}$, over 1+1 categories, where one denotes head, plus one for the background. The second sibling branch computes the offset for transferring input to weak boxes. The loss function of WBG branch is:

$$\mathcal{L}_{WBG} = \mathcal{L}_{cls} + \lambda \mathcal{L}_{loc}, \tag{4}$$

in which λ is the loss weight balance parameters, $\mathcal{L}_{cls}$ is classification loss, and $\mathcal{L}_{loc}$ is regression loss. Since we only utilize detection branch without the weak boxes generate branch and weak boxes refine branch on test images, our methods brings little increase in computation during the inference time.

IV. EXPERIMENT

In this section, we first introduce our experiment details, such as datasets, evaluation matrics, and hyperparameter settings. Then we conduct ablation experiments to explore the contributions of each proposed module. After that, we give some qualitative results for further analyses. Finally, we compare the performance of our method with some leading results in head detection.

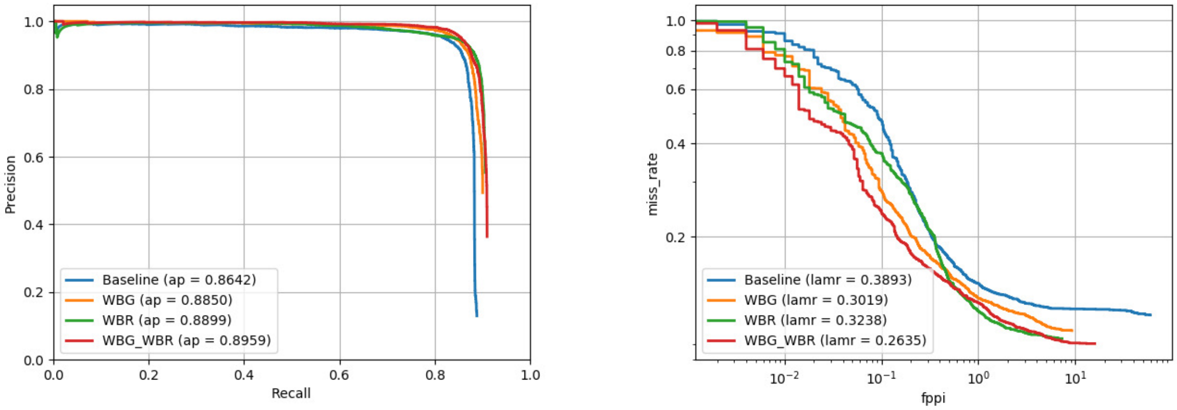
A. Experiment settings

1) Datasets and Evaluation Metrics: We evaluate our method on the two primary head detection datasets: Brainwash [16], SCUT-HEAD [12]. The Brainwash dataset is a frequently-used head detection benchmark, which contains 91146 labeled people in 11917 images. Our ablation studies are performed on the test set of Brainwash, of which 1000 images are used for evaluation and testing, remaining for training. Another dataset is SCUT-HEAD dataset aims to reflect the robustness of head detector under difficult scenarios. It contains 4405 images with 111251 annotated head collected from video and Internet.

For testing, there are two metrics for evaluation: AP and MR^{-2}. Following the standard PASCAL VOC protocal, Average Precision(AP) computes the average precision value for recall value over 0 to 1, which is commonly used in object detection. MR^{-2} calculates the log-average miss rate over 9 points ranging from 10^{-2} to 10^{0} FPPI. Both metrics are based on the PASCAL criterion, ie., IOU>0.5 between ground truth boxes and predicted boxes.

2) Implementation Details: Our network is base on Faster R-CNN, of which the implementation is the same as [22]. We maintain most of the hyper-parameter in Faster R-CNN such as RPN batch size, pooling model to keep the fairness of experimental results when compared with the baseline.

During training, we adopt ResNet-50 pre-trained on ImageNet [4] as the backbone of our proposed network. For the newly added layers, the parameters are randomly initialized with a Gaussian distribution $\mathcal{N}(\mu, \delta)(\mu = 0, \delta = 0.01)$. Furthermore, for the hyper-parameter in Section. III-B, the top K truncating number is set to 30. Besides, for the parameters in Section. III-C, if not specified, our weak boxes refine branch will transfer the initial weak boxes twice time; the proposals nums is set to 2000 per-image; the IOU threshold to make the pair between weak and strong boxes is 0.2. In finetuning,



(a) The PR curves on Brainwash test set

(b) The MR-FPPI curves on Brainwash test set

Fig. 4. Miss rates versus false positive per-image(MR-FPPI) curves and Precision-recall(PR) curves on Brainwash test set

TABLE I
THE ABLATION STUDIES UNDER BRAINWASH DATASET

Method	Methodology		AP(Brainwash)
	WBG branch	WBR branch	
Faster R-CNN (10%)			0.8642
Ours(10%)	✓		0.8850±0.0022
Ours(10%)		✓	0.8899±0.0019
Ours(10%)	✓	✓	0.8959±0.0039
Faster R-CNN (5%)			0.8392
Ours(5%)	✓	✓	0.8647±0.0033
Faster R-CNN(100%)			0.9196

TABLE II
THE ABLATION STUDIES UNDER SCUT-HEAD DATASET

Method	Methodology		AP
	WBG branch	WBR branch	(SCUT-HEAD)
Faster R-CNN (20%)			0.8017
Ours(20%)	✓	✓	0.8128±0.0021
Faster R-CNN(100%)			0.8566

the mini-batch size for SGD is set to be 1. The learning rate is set to 0.0005 for the first 15 epochs and divided by 10 in the following 5 epochs. The trade-off parameters between loss are set to 1 all. And we set the momentum and weight decay to 0.9 and 0.0001, respectively. For semi-supervised settings, we randomly select a limited number of images with boxes annotation, remaining images without any labels.

B. Ablation studies

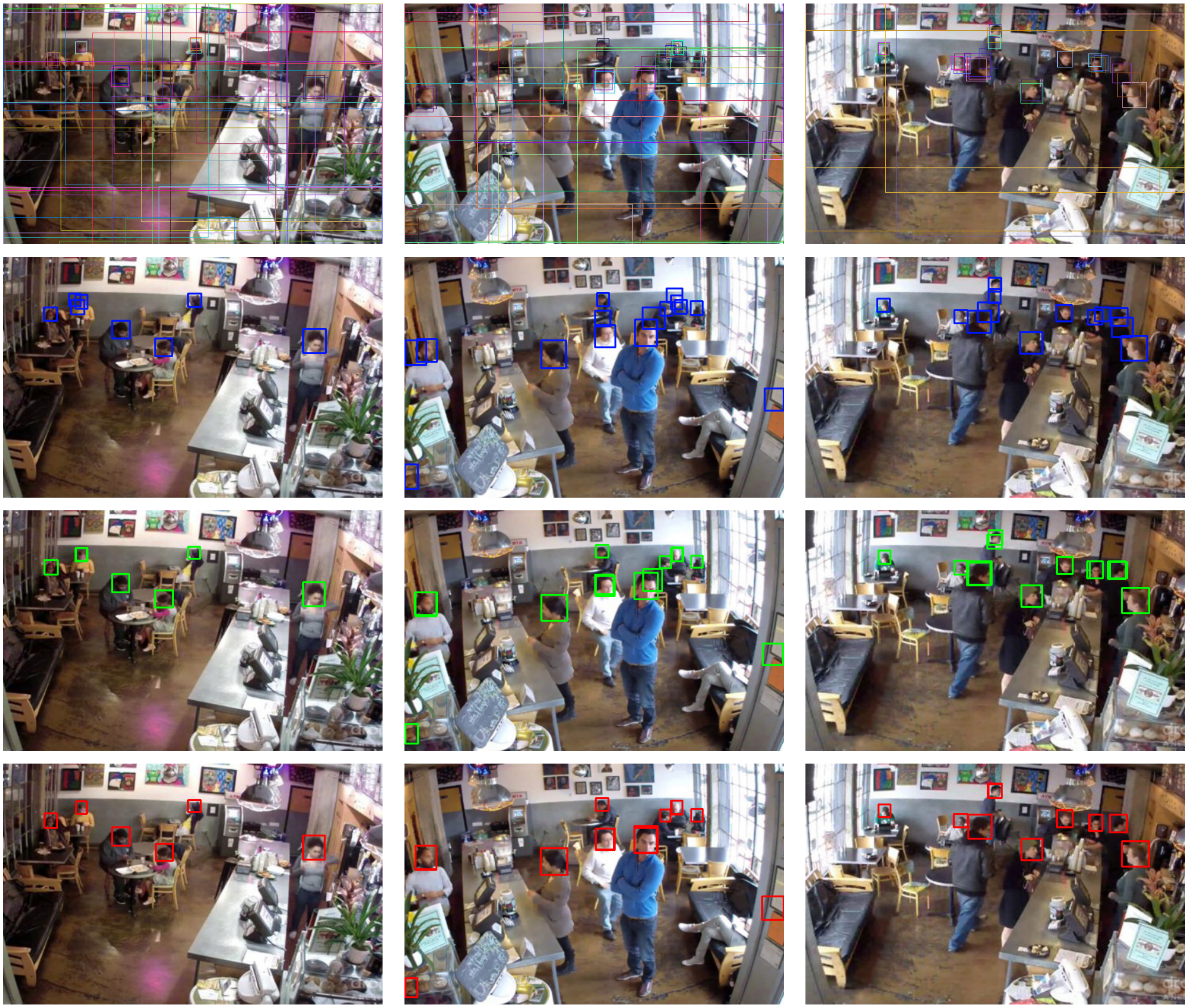
We conduct ablation studies experiment on Brainwash to illustrate the effectiveness of our proposed network. We validate the contribution of each component including WBG and WBR branch. To verify the validity of our results under semi-supervised settings reasonably, each experiment in ablation studies selects the same 10% of all box annotations.

1) Baseline: The baseline is the detection backbone without WBG and WBR branch, which is same as Faster R-CNN. We re-run the experiment and get a higher result of 0.9196 AP with 100% box annotations while 0.8642 AP with 10% box annotations.

2) WBG Branch : To verify the effect of WBG, we conduct experiment with and w/o WBG. The boxes annotation generated by our WBG branch will be used to train the detector. From Table. I and Fig. 4, we can conclude that WBG branch does generate rough but valid box annotation for images in $\mathcal{B}$ and improve the AP of baseline by 2.4%.

3) WBR Branch: When we measure the effectiveness of our WBR branch, we find that if there are no existed weak boxes, it is unable to refine them. For separately verify the result of WBR, we utilize the boxes generation method mentioned in Section. III-B, which generated pseudo ground truth as weak boxes for unlabelled images with a decouple trained detector. From Table. I and the green curve in Fig. 4, we can see that this branch is essential for our method.

4) Joint Optimization: To further explain the effectiveness of our method, we optimize the proposed WBG and WBR branches jointly. The experiments are summarized in Table I and Fig. 4. From the results, we can find that comparing to train a decouple two-stage detector to generate boxes separately, our proposed simple embedded network performs better with efficient training strategy. Another conclusion is that the accuracy of joint optimization with the refining stage is higher than the result only with the WBG branch. We attribute the improvement to more accurate weak boxes adjusted by the WBR branch. In general, what our method does is effectively generate higher quality training samples for a better detector. We also carry the exploration studies on fewer boxes annotation, SCUT-Head data, and the MR metric, as reported in Table I, Table II, and Fig . 4, respectively. The



(a) Scene at night

(b) Scene with strong light in day

(c) Another scene with strong light in day

Fig. 5. Examples of head detection results compared between Different scenes. First Row: Proposals from RPN. Second Row: Initial weak boxes. Third Row: Weak boxes refine for the first time. Fourth Row: Pseudo ground truth from weak boxes. See details in Section. IV-C.

percentage of boxes annotation in SCUT-HEAD is 20% result from the size of data set much smaller than Brainwash. From these results, we can draw the same conclusion.

C. Qualitative results

We show some qualitative comparison among the proposal from RPN, weak boxes generated by WBG, weak boxes refined by WBR for the first time, and weak boxes refined by WBR for the second time, both of which use the same basic network. As shown in Fig. 5, each column represents a scene. The first row contains the top 30 proposals filtered by RPN. The blue boxes in the second row indicate initial weak boxes created by the WBG branch, While the green boxes and red boxes in the last two rows denote refined weak boxes from the WBR branch. From these examples, we can observe that our proposed WBG can generate coarse but practical week boxes, and our WBR branch does adjust the weak boxes better gradually. Although the boxes still have some problems, we believe our method can further improve by incorporating segmentation context in WSOD.

D. Comparison with other methods

In the field of semi-supervision of head detection, we have not found any existing work. And there are few articles with public experiment results on the SCUT-HEAD dataset. So in this section, we evaluate our model on Brainwash test benchmark and compare it with some existed leading methods such as SSD [10], E2PD [16], HeadNet [8], etc. The results are summarized in Table. III. From the table, we can see that our method achieves a competitive result, which outperforms some full-supervised methods even with only 10% of the box annotations. This result demonstrates the superiority of our proposed method. Trough our methods, we can make better use of the unlabeled images to help the detector being more robust and effective. Furthermore, after our extended experiment, we find that our method performs not better enough in fewer samples like only 2% of the box annotations. The main reason is that the samples are too few to train a better-generalized detector under the detection framework in this paper. An ideal solution is yet wanted because there is still room for improvement.

TABLE III
THE RESULT OF OUR METHOD COMPARED WITH OTHERS ON THE BRAINWASH DATASET

	Method	mAP
10%	Faster R-CNN [14]	0.864
	Ours	0.896
100 %	YOLO9000 [13]	0.625
	SSD [10]	0.741
	TINY [5]	0.893
	E2PD [16]	0.821
	Faster R-CNN [14]	0.919
	HeadNet [8]	0.913

V. Conclusion

In this paper, we proposed an end-to-end semi-supervised head detection framework that can obtain competitive performance with a small subset. Under the semi-supervised setting, our method focuses on how to generate quality pseudo ground truth for the unlabeled image with the help of annotated boxes to obtain more precise detector. We firstly proposed a WBG branch to produce weak boxes that coarsely located the head. Then, we introduce a WBR branch aiming to transfer weak boxes more accurate with the guidance of strong boxes. The pseudo labels are generated online and trained with labeled images, which avoid head detector over-fitting to the small set. Experiments show the effectiveness of our method.

References

[1] Zhaowei Cai and Nuno Vasconcelos. Cascade r-cnn: Delving into high quality object detection. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 6154–6162, 2018.

[2] Gang Chen, Xufen Cai, Hu Han, Shiguang Shan, and Xilin Chen. Headnet: pedestrian head detection utilizing body in context. In *2018 13th IEEE International Conference on Automatic Face & Gesture Recognition (FG 2018)*, pages 556–563. IEEE, 2018.

[3] Shuanglu Dai and Hong Man. Mixture statistic metric learning for robust human action and expression recognition. *IEEE Transactions on Circuits and Systems for Video Technology*, 28(10):2484–2499, 2017.

[4] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *2009 IEEE conference on computer vision and pattern recognition*, pages 248–255. Ieee, 2009.

[5] Peiyun Hu and Deva Ramanan. Finding tiny faces. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 951–959, 2017.

[6] Jisoo Jeong, Seungeui Lee, Jeesoo Kim, and Nojun Kwak. Consistency-based semi-supervised learning for object detection. In *Advances in Neural Information Processing Systems*, pages 10758–10767, 2019.

[7] Rongchun Li, Biao Zhang, Zhen Huang, Xiang Zhao, Peng Qiao, and Yong Dou. *Spatial Attention Network for Head Detection: 19th Pacific-Rim Conference on Multimedia, Hefei, China, September 21-22, 2018, Proceedings, Part II*, pages 547–557. 09 2018.

[8] Wei Li, Hongliang Li, Qingbo Wu, Fanman Meng, Linfeng Xu, and King Ngi Ngan. Headnet: An end-to-end adaptive relational network for head detection. *IEEE Transactions on Circuits and Systems for Video Technology*, 2019.

[9] Tsung-Yi Lin, Priya Goyal, Ross Girshick, Kaiming He, and Piotr Dollár. Focal loss for dense object detection. In *Proceedings of the IEEE international conference on computer vision*, pages 2980–2988, 2017.

[10] Wei Liu, Dragomir Anguelov, Dumitru Erhan, Christian Szegedy, Scott Reed, Cheng-Yang Fu, and Alexander C Berg. Ssd: Single shot multibox detector. In *European conference on computer vision*, pages 21–37. Springer, 2016.

[11] Tianxiang Pan, Bin Wang, Guiguang Ding, Jungong Han, and Junhai Yong. Low shot box correction for weakly supervised object detection. In *Proceedings of the 28th International Joint Conference on Artificial Intelligence*, pages 890–896. AAAI Press, 2019.

[12] Dezhi Peng, Zikai Sun, Zirong Chen, Zirui Cai, Lele Xie, and Lianwen Jin. Detecting heads using feature refine net and cascaded multi-scale architecture. In *2018 24th International Conference on Pattern Recognition (ICPR)*, pages 2528–2533. IEEE, 2018.

[13] Joseph Redmon and Ali Farhadi. Yolo9000: better, faster, stronger. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 7263–7271, 2017.

[14] Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. Faster R-CNN: Towards real-time object detection with region proposal networks. In *Advances in Neural Information Processing Systems (NIPS)*, 2015.

[15] Olga Russakovsky, Li-Jia Li, and Li Fei-Fei. Best of both worlds: human-machine collaboration for object annotation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2121–2131, 2015.

[16] Russell Stewart, Mykhaylo Andriluka, and Andrew Y Ng. End-to-end people detection in crowded scenes. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2325–2333, 2016.

[17] Yuxing Tang, Josiah Wang, Boyang Gao, Emmanuel Dellandréa, Robert Gaizauskas, and Liming Chen. Large scale semi-supervised object detection using visual and semantic knowledge transfer. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2119–2128, 2016.

[18] Jiasi Wang, Xinggang Wang, and Wenyu Liu. Weakly-and semi-supervised faster r-cnn with curriculum learning. In *2018 24th International Conference on Pattern Recognition (ICPR)*, pages 2416–2421. IEEE, 2018.

[19] Keze Wang, Xiaopeng Yan, Dongyu Zhang, Lei Zhang, and Liang Lin. Towards human-machine cooperation: Self-supervised sample mining for object detection. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1605–1613, 2018.

[20] Jian Xiong, Xianzhong Long, Ran Shi, Miaohui Wang, Jie Yang, and Guan Gui. Background error propagation model based rdo in hevc for surveillance and conference video coding. *IEEE Access*, 6:67206–67216, 2018.

[21] Ziang Yan, Jian Liang, Weishen Pan, Jin Li, and Changshui Zhang. Weakly-and semi-supervised object detection with expectation-maximization algorithm. *arXiv preprint arXiv:1702.08740*, 2017.

[22] Jianwei Yang, Jiasen Lu, Dhruv Batra, and Devi Parikh. A faster pytorch implementation of faster r-cnn. *https://github.com/jwyang/faster-rcnn.pytorch*, 2017.

[23] Shuai Yi, Hongsheng Li, and Xiaogang Wang. Understanding pedestrian behaviors from stationary crowd groups. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3488–3496, 2015.

[24] Qiaoyong Zhong, Chao Li, Yingying Zhang, Di Xie, Shicai Yang, and Shiliang Pu. Cascade region proposal and global context for deep object detection. *Neurocomputing*, 2019.