Sparse Generation: Making Pseudo Labels Sparse for weakly supervision with points

Tian Ma [*1] **Chuyang Shang** [*1 2] **Wanzhu Ren** [3] **Yuancheng Li** [3] **Jiayi Yang** [3] **Jiali Qian** [3] [3]

Abstract

In recent years, research on point weakly supervised object detection (PWSOD) methods in the field of computer vision has attracted people's attention. However, existing pseudo labels generation methods perform poorly in a small amount of supervised annotation data and dense object detection tasks. We consider the generation of weakly supervised pseudo labels as the result of model's sparse output, and propose a method called Sparse Generation to make pseudo labels sparse. It constructs dense tensors through the relationship between data and detector model, optimizes three of its parameters, and obtains a sparse tensor via coordinated calculation, thereby indirectly obtaining higher quality pseudo labels, and solving the model's density problem in the situation of only a small amount of supervised annotation data can be used. On two broadly used open-source datasets (RSOD, SIMD) and a self-built dataset (BulletHole), the experimental results showed that the proposed method has a significant advantage in terms of overall performance metrics, comparing to that state-of-the-art method.

1. Introduction

In recent years, methods based on PWSOD (Point Weakly Supervised Object Detection) (Chen et al., 2021; 2022) have aroused research interests in academia. Due to the fact that it only needs to annotate a very small amount of supervised annotation data, other data can use the low-cost weakly supervised annotation format, which can greatly reduce the workload (Fu et al., 2023; Ren et al., 2020) of model training. Compared with semi-supervised object detection(SSOD) (Sohn et al., 2020b;a; Tarvainen & Valpola, 2017; Zhou et al., 2022) methods, its advantage lies in utilizing these weakly supervised annotations information could better guide the model training. While the PWSOD methods using the pseudo labels (Lee et al., 2013) face the same problem as the SSOD method: **how to generate high-quality pseudo labels as the supervision for model training?**

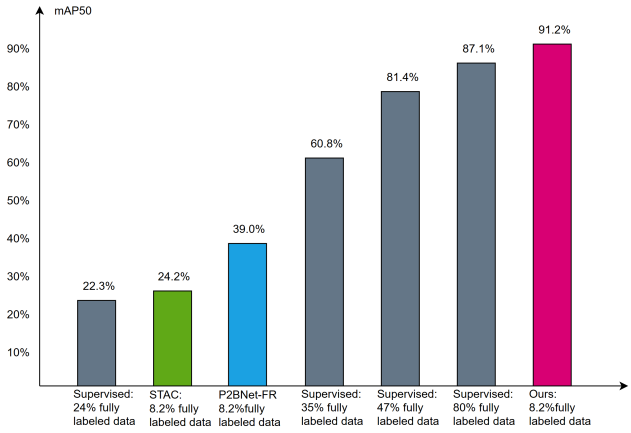

Figure 1. The experimental results with STAC (Sohn et al., 2020b), P2BNet (Chen et al., 2022) and supervised training.

Let's take a different approach to this issue. The output process of the object detector currently using CNN (Long et al., 2015) as the backbone, **can be viewed as a region selection process from dense to sparse**, similar to the concept (Sun et al., 2021) first proposed in 2021. The model's output result of the image after feature extraction through the backbone network, usually contains thousands or even hundreds of thousands of region proposals, which will be filtered out under the selection of the detection head.

However, in weakly supervised methods, due to the inability to directly obtain the Bounding Box (bbox) from weakly supervised annotation data, it is often necessary to rely on additional networks or detectors themselves to assist in generating labels. At this step, for the limited use of supervised annotation data, the accuracy from trained network obtained is very low, and the detection head, to some extent degrades its functionality. In addition to the inability to obtain accurate instance areas, the number of pseudo labels generated greatly exceeds the number of instances in the detected image, which is particularly prominent in dense instances detection tasks. **At this point, the generation of pseudo labels seems to have become a process from dense to sparse again!**

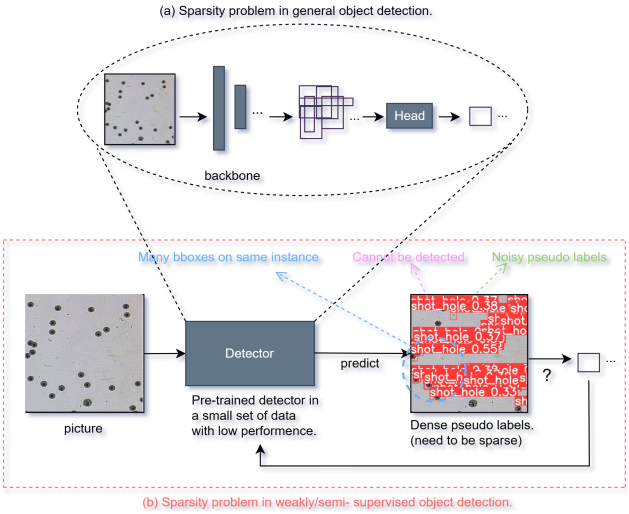

Figure 2. The sparsity problem between general object detection and weakly/semi- supervised object detection.

Fig. 2 shows the sparsity problem within the detector in general object detection scenarios and weakly/semi- supervised object detection scenarios. **This is like a chain loop**, in this case, if an additional network (Chen et al., 2022; Zeng et al., 2019; Tang et al., 2017; Cheng et al., 2020) is used specifically for pseudo labels generation, it can be foreseen that its network output will still be a dense set, and prone to localized focusing problem (Zeng et al., 2019; Tang et al., 2017; Cheng et al., 2020). **Repeated optimization in these imprecise subsets** may not be able to tap into more potential of the algorithm.

In WSOD (weakly supervised object detection) dense pseudo label (DPL) generation, there are four significant issues that cannot be ignored: ① Multiple overlapping bboxes are often generated on the same detection object. ② Due to the low accuracy of the pre-trained network, some instances are not detected. ③ There are cases where noisy output is used as a pseudo label. ④ Pseudo boxes with higher confidence sometimes choose more incorrect regions of instances, as this also mentioned in two works (Xu et al., 2021; Liu et al., 2023).

In summary, we have explored a non-networked approach to avoid these issues, and simplify the process of training for weakly supervised object detection. Fig. 1 shows the comparison of various methods on the Bullet-Hole dataset, and our method has a leading mAP50 metric of over 120% compared to P2BNet-FR (Chen et al., 2022).

2. Related Work

Existing works have proposed some solutions to the problem of pseudo labels generation, **but they are not based on the idea of transitioning from dense to sparse**, and they also perform poorly in dense detection tasks. Weakly supervised methods (Zhang et al., 2021) such as WSOD2 (Zeng et al., 2019) and OICR (Tang et al., 2017) designed complex cascaded optimization networks to generate pseudo labels. A previous work (Cheng et al., 2020) improved the quality of pseudo labels by learning positive and negative samples separately during weakly supervised processes. Soft Teacher (Xu et al., 2021) refined pseudo labels generation by generating jitter around the pseudo boxes. Dense Teacher (Zhou et al., 2022) intentionally made pseudo labels dense to obtain pseudo labels. Point DETR (Chen et al., 2021) modified the DETR (Carion et al., 2020) network as a pseudo label generation method for point weakly supervised detection (PWSOD), however the transformer (Vaswani et al., 2017) network itself performs not very well for high-density detection tasks. P2BNet (Chen et al., 2022), as an existing SOTA method of generating pseudo labels for PWSOD, generates multiple sets of recommended pseudo labels by designing an additional network. In weakly supervised object detection tasks with a small amount of supervised annotation data and dense instance, the network itself could output dense results, and the pseudo labels recommended by the network are only subsets from many dense labels. Liu (Liu et al., 2022) used the positional relationship between instances as a reference for the size of pseudo boxes, and used a detection confidence map in the detection process to fuse information with the regression network. Two previous works (Sohn et al., 2020a;b) also used the confidence as a guide for selecting pseudo labels, a previous work (Liu et al., 2023) pointed out that pseudo labels with high confidence cannot accurately reflect the true position of the instance. Our proposed method focuses on the sparsity of pseudo labels, combining weakly supervised annotations with pseudo label generation, to compose the dense pseudo label (DPL) generated on instance into a sparse pseudo label (SPL). It does not use additional networks to avoid the risk of density, the entire calculation process only requires three parameters, and our method can be directly applied to any CNN architecture detector to achieve PWSOD training.

3. Sparse Generation

Firstly, as shown in Fig. 3, only a small amount of supervised annotation data is needed to train an initial model for CNN based detectors to predict pseudo labels, provided that these small amount of supervised annotation data follow an overall independent and identically distributed distribution. Afterwards, these pseudo labels were mapped into tensors through a staircase function, Fig. 4 and Fig. 5 show the stair-

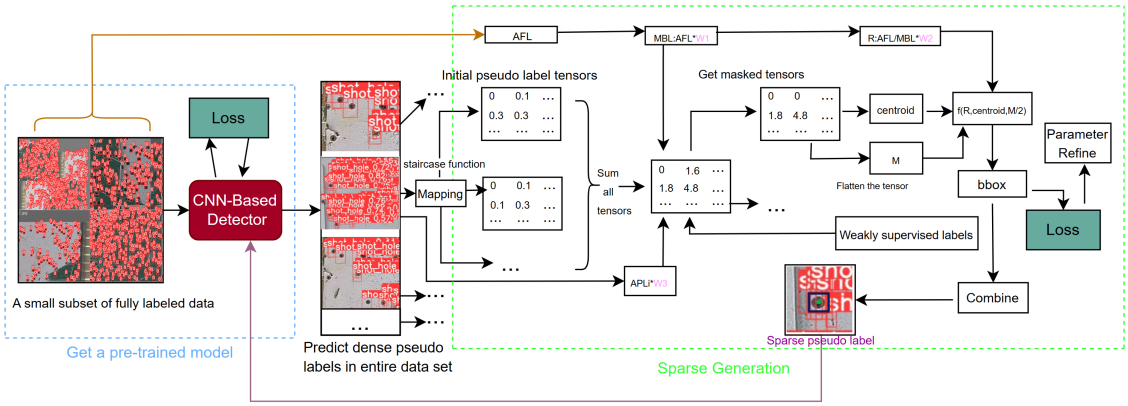

Figure 3. **The pipeline of Sparse Generation**, where AFL represents the average bounding box (bbox) length in supervised annotation data, MBL represents the constraint length used to generate mask tensors (MT), R represents the constraint factor used to generate pseudo boxes in function f, APL represents the average bbox (bounding box) length of dense pseudo labels (DPL) in each image for generating pseudo labels, and centroid (MacDougal, 2012) represents the centroid of the MT, M represents the consequence of flattening the MT into 0-dimension, f is the bbox position mapping function, which will be explained later. W1, W2, and W3 are the three parameters of this architecture. The reason for using the average bbox length here is based on the assumption of homogenization trend of adjacent labels of the same class, which was proposed in a previous work (Ouali et al., 2020), and our experimental results also confirmed this.

case function and quadrant definitions used in the mapping rules during this process.

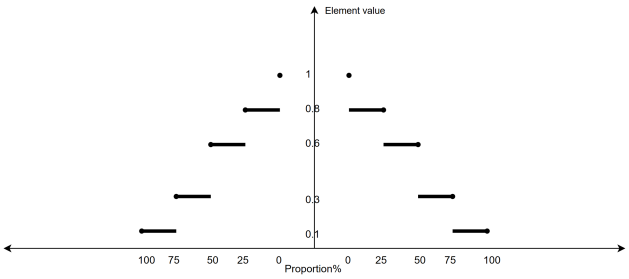

Figure 4. Staircase mapping function.

A two-dimensional tensor is defined, with its central value set to 1. The tensor's size, in terms of rows and columns, are determined by the bounding box from DPL, specifically the length (h_i) and width (w_i), where two pixels in the bounding box of scale of entire image correspond to one tensor element. The tensor's center serves as the origin of Cartesian coordinate, dividing the tensor into eight quadrants. The value of staircase function within the tensor varies according to the Euclidean distance of each element from the center, and its relative position along the semi-axis of the respective quadrant, with a range from 0 to 1. From this, the set of initial tensors (IT) is derived.

Step 2, is to pad (Girshick et al., 2014) each mapped IT on the scale of the entire image, summing all padded tensors to obtain a tensor ST that can reflect the thermal distribution

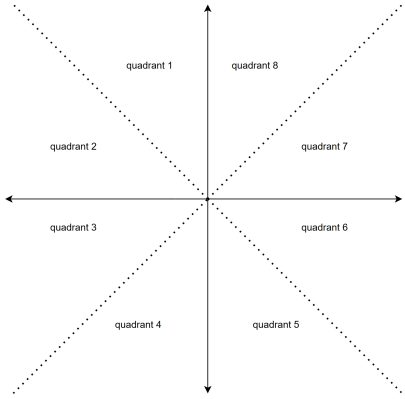

Figure 5. Two-dimensional quadrant definition specification.

(Zhou et al., 2019) on the scale of the entire image. This process is given by:

$$ST = \sum_{0}^{n} \mathrm{Pad}(IT_i(h_i/2, w_i/2), h_i/4, w_i/4), \quad (1)$$

where n represents the number of elements in the set of dense pseudo labels (DPL) from a picture, h_i represents the height of each DPL, w_i represents the width of each DPL.

Step 3, covering the tensor obtained in step two with a mask tensor (MT). The size of the mask tensor is determined by the MBL, which need to ensure that it is bigger than the length and width of the instances in processed images at this

time.

The generation of mask tensor: The mask tensor (MT) is obtained by using the same length which is two pixels for the corresponding element of a tensor in the first initialization step, with the number of rows and columns of this tensor equal to MBL. Afterwards, the obtained MT will be padded to the size corresponding to the entire image scale using the coordinates (x_i, y_i) annotated with point supervision. The padded tensors will be Hadamard product with each summed tensor (ST) to obtain the set of tensors after mask coverage (AMT). This process is given by:

$$AMT = \sum_{0}^{n} \mathrm{Pad}(MT(l,l), x_i - l/2, y_i + l/2) \odot ST, \quad (2)$$

where l represents the constraint lenth of MBL used to generate mask tensors (MT). Pad represents the padding operation.

Step 4, flattening the tensor covered by AMT onto two one-dimensional tensors, M_x, M_y, respectively. Than flatting the tensor into a 0-dimensional tensor ***M***. Using the tensor ***M*** and M_x, M_y to obtain *x* and *y* coordinates of centroid. The process is given by:

$$M\_x = \mathrm{Flatten}(AMT_i, 0), \quad (3)$$

$$M\_y = \mathrm{Flatten}(AMT_i, 1), \quad (4)$$

$$M = \mathrm{Flatten}(AMT_i, 0, 1), \quad (5)$$

$$x_i = \frac{\sum_0^n M\_x(j)}{M/2}, \quad (6)$$

$$y_i = \frac{\sum_0^n M\_y(j)}{M/2}, \quad (7)$$

where Flatten represents the flattening operation. By inputting the centroid coordinates into mapping function f, the width and height of the pseudo boxes are obtained. Then, the information of point annotation is integrated with the width and height to obtain the pseudo label for a single instance. The specification of function f is provided by the pseudo codes in Algorithm 1.

Step 5, the algorithm will optimize parameters based on a small amount of supervised annotation data. Its loss function is defined by:

Algorithm 1 Function f

Input: R, x_i, y_i, $M/2$, APL_i, M_x, M_y
Initialize $width = 0, height = 0, sum = 0$.
if $M\_x == 0$ **then**
  $width = APL_i * W3$
end if
for $j = 0$ **to** $len(M\_x)$ **do**
  if $sum >= (M/2) * R$ **then**
    break
  end if
  $sum+ = M\_x[j]$
  $j+ = 1$
end for
...
#similar process get k
...
$width = j - k$
...
#get $height$
...
return $width, height$

$$Loss = \tanh(\sum_{0}^{n} (SPL_i\_bbox - GT_j\_bbox)/n), \quad (8)$$

where *n* is the number of supervised annotation data, SPL_i is the i-th sparse label, and GT_j is the corresponding j-th supervised annotation data.

The above steps will complete the sparsity of pseudo labels for most instances, so that each individual instance corresponds to only one pseudo label, and solve the overlap problem caused by dense pseudo labels. For the other two problems in PWSOD that we proposed, we also provided effective solutions: ① For instances where DPL was not detected, make the row and column numbers of the initialization tensor corresponding to this instance equal to $APL*W3$, so as to more reasonably reflect the bbox length distribution of instances in single image. ② For the case of identifying noise as a positive example, due to the previous mask coverage, this problem was directly avoided.

The bbox position information obtained by this way reduces the degree of linear or nonlinear positional changes for the size of boxes due to the coverage of many DPL in the same instance. Greatly reducing the occurrence which mentioned in work (Liu et al., 2023) of misleading SPL results due to high confidence DPL. Effectively avoiding localized focusing problem (Zeng et al., 2019; Tang et al., 2017; Cheng et al., 2020). Each instance has only one pseudo label generated, which is also the key to complete the dense to sparse process.

4. Experiments

4.1. Dataset

Bullet-Hole dataset: Due to the need to obtain instances as dense as possible, we conducted Bullet-Hole data set collection in a real shooting range. Selecting shooting targets with bullets which were very dense on them, removed the shooting targets from equipment, and used cameras to collect data at different distances in different lighting environments. The model of the camera was Hikvision DS-2CD3356FWDA3-IS, with a resolution of 5 million pixels. Each image obtained was segmented to 3×3, and after selection, 85 photos with a resolution of 320×320 were obtained. In these photos with the highest number of bullet holes contain over 502 bullet holes.

In addition, **two different widely used open-source datasets were selected**: RSOD (Li et al., 2020) and SIMD (Haroon et al., 2020) remote sensing datasets, where the SIMD dataset contains 15 categories, to test the algorithm's ability to detect multiple categories and instances.

In the data annotation stage, using Labelimg software for annotating supervised data on the self-built dataset, as they need to be used for comparison with supervised training methods. Selecting random points around the center point within a range of 20% of the average bbox length from supervised annotation as the point annotation data. For the partitioning of the dataset, the comparative experiments to be explained later used the same partitioning ratio, and the data in the validation set have been randomly sampled.

4.2. Model Training and Contrasts Setting

In this section, the results of different methods tested on different datasets are presented. All epochs during experimental training were obtained without increasing the mAP metric.

Sparse Generation: Selecting the classical YOLOV5s (Bochkovskiy et al., 2020) and Faster RCNN to represent detectors based on CNN architecture networks. On three datasets (Bullet-Hole, RSOD (Li et al., 2020), SIMD (Haroon et al., 2020)), a pre-trained model was obtained by training randomly selected images with a single card RTX4070 GPU, which accounted for approximately 8.2%, 5.5%, 4.5%, and 3.4% of the total data, respectively. This model was used to predict the images from the entire dataset and obtained dense pseudo labels (DPL). These DPL were fed into the Sparse Generation method to obtain Sparse Pseudo labels (SPL), and then the detector was trained only once using these SPL to get the final training result.

STAC (Sohn et al., 2020b): Selecting STAC as the representative of semi-supervised methods. Weakly enhancing 100% of the images in Bullet-Hole dataset, including flipping 50% horizontally and 50% vertically; Strong enhancement was applied to 50% of the randomly selected photos, including color block replacement and hard cropping two colors at random positions around the instances. 35% and 8% of supervised annotation data were selected for pre-training of its Teacher Model. Up to 3 rounds of Student Model and Teacher Model updating were conducted on a single card RTX4070 GPU under 35% supervised annotation data until the validation accuracy of the model no longer improved.

P2BNet (Chen et al., 2022): P2BNet was chosen as the main method for comparison, which is currently the representative of the most advanced PWSOD pseudo labels recommendation method. On a dual card RTXA5000 GPU system, making the proportions of supervised annotations selected in three different datasets same with other methods, with all other point annotation data for the first round of P2BNet training. The pseudo labels recommended by the first round of P2BNet were used as supervision to train Faster RCNN (Ren et al., 2015) or YoloV5s. After the second round of P2BNet training was completed, Faster RCNN was trained again using the pseudo labels recommended by them. Using random points coordinates within a Gaussian distribution range of 0.2 around the center point from the boxes of supervised annotation as the point annotation data, to match the previous settings.

Supervised Learning: As a comparison with WSOD and PWSOD training methods, single card training on RTX4070 GPU was conducted using approximately 5.9%, 24%, 35%, 47%, and 80% of the Bullet-Hole dataset, respectively.

4.3. Experiment Results on Bullet-Hole Dataset.

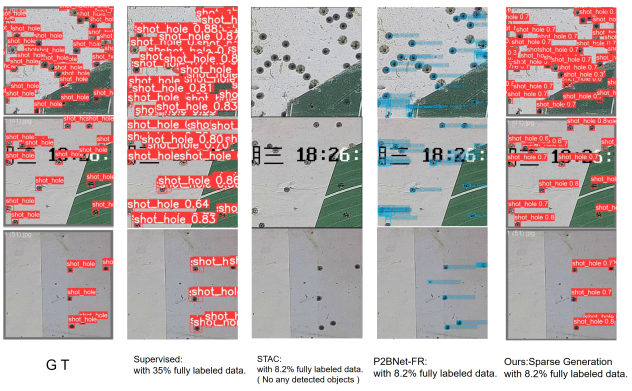

Figure 6. The performance comparing on Bullet-Hole dataset.

Table 1 shows the comparison of various methods in the Bullet-Hole dataset. In supervised training, 24% of the training data did not make it exceed the performance metrics of semi-supervised (STAC (Sohn et al., 2020b)) and two weakly supervised methods. Using the STAC method to

Table 1. Experimental results on Bullet-Hole dataset.

METHOD	DETECTOR	FEATURE	EPOCHS	MAP50	MAP50-95
SUPERVISED	YOLOV5S	5.9% FULLY LABELED DATA	80	4.67	0.74
SUPERVISED	YOLOV5S	24.0% FULLY LABELED DATA	80	22.30	5.40
SUPERVISED	YOLOV5S	35.0% FULLY LABELED DATA	80	60.91	24.08
SUPERVISED	YOLOV5S	80.0% FULLY LABELED DATA	80	87.68	42.25
STAC (SOHN ET AL., 2020B)	YOLOV5S	8.2% FULLY LABELED DATA	80	24.16	10.72
STAC	YOLOV5S	35.0% FULLY LABELED DATA	3×80	79.14	**58.73**
P2BNET (CHEN ET AL., 2022)	FASTER RCNN	8.2% FULLY LABELED DATA	12+2×12	38.99	16.43
P2BNET	YOLOV5S	8.2% FULLY LABELED DATA	12+80	52.66	18.07
OURS: SPARSE GENERATION	FASTER RCNN	8.2% FULLY LABELED DATA	36	55.06	19.64
OURS: SPARSE GENERATION	YOLOV5S	8.2% FULLY LABELED DATA	80	**91.20**	42.10

Table 2. Experimental results on RSOD OIL TANK (Li et al., 2020) dataset, * means P2BNet use the mask filter.

METHOD	DETECTOR	FEATURE	EPOCHS	MAP50	MAP50-95
P2BNET*	YOLOV5S	5.0% FULLY LABELED DATA	80	33.08	13.73
P2BNET (CHEN ET AL., 2022)	YOLOV5S	5.0% FULLY LABELED DATA	80	61.82	23.07
OURS:SPARSE GENERATION	YOLOV5S	5.0% FULLY LABELED DATA	80	**89.72**	**29.46**

train 35% of supervised annotated data, the mAP50 metric still did not exceed 80. P2BNet was trained using the same 8.2% supervised labeled data as our Sparse Generation method. When Faster RCNN (Ren et al., 2015) was used as the detector, our method outperformed the mAP50 metric by **16.07**; While using YoloV5s (Redmon & Farhadi, 2018; Bochkovskiy et al., 2020) as a detector, our method leaded the mAP50 metric of **38.54**.

As shown in Fig. 6, under supervised training, 35% of the supervised annotation data clearly did not make the model training sparser. In its detection results, although the model had undergone NMS, there were a large number of duplicate detection boxes on a single instance. P2BNet, due to its recommendation network sampling the area around several instances, **made the whole process denser through the network**, and considered some noise as positive examples for classification results. Due to the high density of pseudo labels, the STAC semi-supervised method did not have the ability to detect any objects on the three images shown in Fig. 6. The Sparse Generation method significantly improved the recall and precision of point weakly supervised algorithms in such applications.

4.4. Experiment Results on RSOD Dataset.

Table 2 shows the comparison results on the RSOD OIL TANK (Li et al., 2020) dataset. Compared to P2BNet using mask filtration, the mAP50 metric of our method was higher than P2BNet by **56.64**; When the detector directly used the predicted results of P2BNet as pseudo labels for training, the mAP50 metric of our method leaded **27.9**. Table 3 shows the comparison results on the RSOD AIRCRAFT data set. Our method outperformed P2BNet in both mAP50 and mAP50-95 metrics.

Figure 7. The performance comparing on RSOD AIRCRAFT (Li et al., 2020) dataset. P2BNet (Chen et al., 2022) and Sparse Generation use the same detector.

As shown in Fig. 7, the detector trained by our method predicted bounding boxes on instances, were closer to the GT (reflecting the true position of the instance), while the prediction of detector trained by P2BNet generated multiple boxes on the same instance.

Table 3. Experimental results on RSOD AIRCRAFT (Li et al., 2020) dataset, * means P2BNet use the mask filter.

METHOD	DETECTOR	FEATURE	EPOCHS	MAP50	MAP50-95
P2BNET*	YOLOV5S	4.5% FULLY LABELED DATA	12+30	32.02	8.65
P2BNET (CHEN ET AL., 2022)	YOLOV5S	4.5% FULLY LABELED DATA	12+30	55.99	18.93
OURS:SPARSE GENERATION	YOLOV5S	4.5% FULLY LABELED DATA	30	**65.70**	**26.47**

Table 4. Experimental results on SIMD (Haroon et al., 2020) dataset, * means P2BNet use the mask filter.

METHOD	DETECTOR	FEATURE	EPOCHS	MAP50	MAP50-95
P2BNET*	YOLOV5S	3.4% FULLY LABELED DATA	12+30	6.72	2.16
P2BNET (CHEN ET AL., 2022)	YOLOV5S	3.4% FULLY LABELED DATA	12+30	6.08	1.95
OURS:SPARSE GENERATION	YOLOV5S	3.4% FULLY LABELED DATA	30	**29.11**	**8.45**

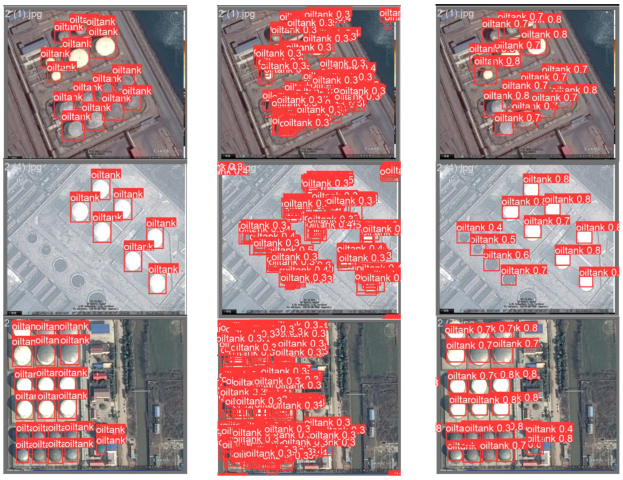

Figure 8. The performance comparing on RSOD OIL TANK (Li et al., 2020) dataset using 5.0% fully labeled data for training. P2BNet (Chen et al., 2022) and Sparse Generation use the same detector. Where GT represents the true boxes of instances.

As shown in Fig. 8, during oil tanks object detection, after training the detector with pseudo labels generated by P2BNet, **a large number of duplicate boxes appeared** in the predicted results, due to its problem of density for using additional network.

4.5. Experiment Results on SIMD Dataset.

Table 4 shows the comparison results on the SIMD (Haroon et al., 2020) dataset. Our method achieved a mAP50 metric which was **4.78 times** performance of P2BNet without using mask filtering; While using mask filtering in P2BNet, our method's mAP50 metric reached **4.33 times** performance of it.

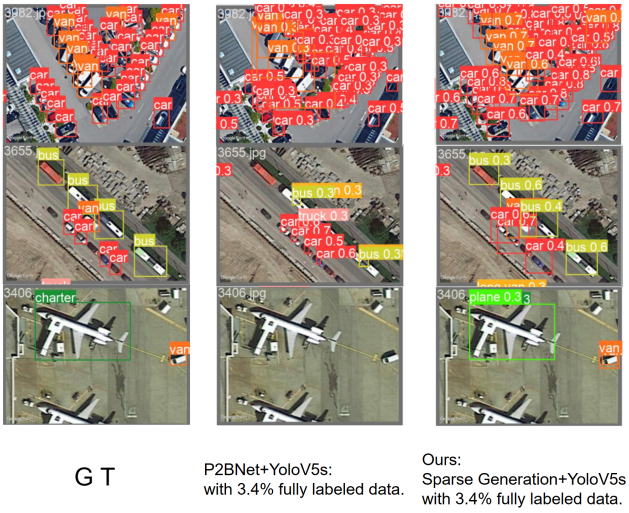

Figure 9. With 3.4% fully labeled data for training, the performance comparing on SIMD (Haroon et al., 2020) dataset, Where GT represents the true boxes of instances.

Fig. 9 shows the partial images prediction results after training using two methods on the same detector. The SIMD dataset has 15 different instance categories, with many instances from different categories appearing in the same image, which requires a high overall ability for weakly supervised method. On the predicted image with instances of buses at the middle position, the detector trained by P2BNet **encountered localized focusing problem** (Tang et al., 2018), which previous works (Zeng et al., 2019; Tang et al., 2018; Cheng et al., 2020) diligently tried to avoid.

However, it inevitably has the tendency to this for using neural networks in this case with a lack of training data. In the bottom image, P2BNet did not detect the instances of chart and van. Our method has a higher ability to detect dense instances of different categories in different scenarios.

Table 5. On the RSOD OIL TANK (Li et al., 2020) dataset, remove the function f and parameter optimization parts from the proposed method and conduct experiments on their functionality.

FUNCTION F	PARAMETER REFINE	FEATURE	MAP50	MAP50-95
-	-	4.5% FULLY LABELED DATA	23.33	6.59
✓	-	4.5% FULLY LABELED DATA	83.68	24.97
✓	✓	4.5% FULLY LABELED DATA	89.72	29.46

Table 6. On the Bullet-Hole dataset, comparison using point annotations to rank the confidence of the dense pseudo labels (DPL) with the maximum confidence filter in the mask, and select the pseudo labels with the highest confidence for the supervised training.

METHOD	DETECTOR	FEATURE	MAP50	MAP50-95
SUPERVISED	YOLOV5S	8.2% FULLY LABELED DATA	8.13	1.43
PSEUDO LABELS WITHOUT HANDLING	YOLOV5S	8.2% FULLY LABELED DATA	25.94	7.74
PWSOD+MAXIMUM CONFIDENCE FILTER	YOLOV5S	8.2% FULLY LABELED DATA	32.26	10.56
SPARSE GENERATION	YOLOV5S	8.2% FULLY LABELED DATA	91.20	42.10

4.6. Ablation Study

As shown in Table 5, on the RSOD OIL TANK (Li et al., 2020) dataset, when the algorithm did not use tensor flattening and function f calculation, that was, the biggest bounding box in the horizontal and vertical directions after mask filtering was combined into one pseudo box. Compared with the results trained using the complete method, both mAP50 and mAP50-95 metrics showed a significant decline.

The main role of function f is to eliminate the impact of some pseudo labels being too large or too small compared to the size of the instance itself when there are lots of dense pseudo labels on the instance.

After experimental verification, the parameters conform to the characteristics of regression optimization, using the initial parameter values we have summarized, we only need to optimize the parameter R for different datasets to achieve decent pseudo labels generation quality. The algorithm will automatically implement optimization iterations to further improve performance.

As shown in Table 6, selecting the pseudo labels with the highest confidence in the mask corresponding to each instance as supervision for detector training, could not obtain good performance metric.

5. Conclusion

In this paper, we analyzed the shortcomings of previous methods in a small amount of supervised annotation data and weakly supervised object detection for dense instances tasks. Using CNN-based architecture network for pseudo labels generation, the output results will still be a relatively dense set, and prone to localized focusing problem. It will reduce the performance of the model for using these pseudo labels subsets as the supervision for model training, and sparsity of pseudo labels is the key to solving these problems. In addition, four problems in such detection tasks were pointed out, and a non-networked pseudo label sparsity method: Sparse Generation was proposed. This method only has three parameters and could achieve good mAP performance metric for training only once when we have the pre-trained model.

6. Impact Statements

For this work for research, there are none potential societal consequences which we feel must be specifically highlighted.

References

Bochkovskiy, A., Wang, C.-Y., and Liao, H.-Y. M. Yolov4: Optimal speed and accuracy of object detection. *arXiv preprint arXiv:2004.10934*, 2020.

Carion, N., Massa, F., Synnaeve, G., Usunier, N., Kirillov, A., and Zagoruyko, S. End-to-end object detection with transformers. In *European conference on computer vision*, pp. 213–229. Springer, 2020.

Chen, L., Yang, T., Zhang, X., Zhang, W., and Sun, J. Points as queries: Weakly semi-supervised object detection by points. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 8823–8832, 2021.

Chen, P., Yu, X., Han, X., Hassan, N., Wang, K., Li, J., Zhao, J., Shi, H., Han, Z., and Ye, Q. Point-to-box network for accurate object detection via single point supervision. In *European Conference on Computer Vision*, pp. 51–67. Springer, 2022.

Cheng, G., Yang, J., Gao, D., Guo, L., and Han, J. High-quality proposals for weakly supervised object detection. *IEEE Transactions on Image Processing*, 29:5794–5804, 2020.

Fu, L., Li, S., Li, Q., Deng, L., Li, F., Fan, L., Chen, M., and He, X. Ufo2: A unified pre-training framework for online and offline speech recognition. In *ICASSP 2023-2023 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 1–5. IEEE, 2023.

Girshick, R., Donahue, J., Darrell, T., and Malik, J. Rich feature hierarchies for accurate object detection and semantic segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 580–587, 2014.

Haroon, M., Shahzad, M., and Fraz, M. M. Multisized object detection using spaceborne optical imagery. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, 13:3032–3046, 2020.

Lee, D.-H. et al. Pseudo-label: The simple and efficient semi-supervised learning method for deep neural networks. In *Workshop on challenges in representation learning, ICML*, volume 3, pp. 896. Atlanta, 2013.

Li, K., Wan, G., Cheng, G., Meng, L., and Han, J. Object detection in optical remote sensing images: A survey and a new benchmark. *ISPRS journal of photogrammetry and remote sensing*, 159:296–307, 2020.

Liu, C., Zhang, W., Lin, X., Zhang, W., Tan, X., Han, J., Li, X., Ding, E., and Wang, J. Ambiguity-resistant semi-supervised learning for dense object detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 15579–15588, 2023.

Liu, Y., Wang, Z., Shi, M., Satoh, S., Zhao, Q., and Yang, H. Discovering regression-detection bi-knowledge transfer for unsupervised cross-domain crowd counting. *Neurocomputing*, 494:418–431, 2022.

Long, J., Shelhamer, E., and Darrell, T. Fully convolutional networks for semantic segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 3431–3440, 2015.

MacDougal, D. W. *Newton's gravity: an introductory guide to the mechanics of the universe*. Springer Science & Business Media, 2012.

Ouali, Y., Hudelot, C., and Tami, M. An overview of deep semi-supervised learning. *arXiv preprint arXiv:2006.05278*, 2020.

Redmon, J. and Farhadi, A. Yolov3: An incremental improvement. *arXiv preprint arXiv:1804.02767*, 2018.

Ren, S., He, K., Girshick, R., and Sun, J. Faster r-cnn: Towards real-time object detection with region proposal networks. *Advances in neural information processing systems*, 28, 2015.

Ren, Z., Yu, Z., Yang, X., Liu, M.-Y., Schwing, A. G., and Kautz, J. Ufo 2: A unified framework towards omni-supervised object detection. In *European conference on computer vision*, pp. 288–313. Springer, 2020.

Sohn, K., Berthelot, D., Carlini, N., Zhang, Z., Zhang, H., Raffel, C. A., Cubuk, E. D., Kurakin, A., and Li, C.-L. Fixmatch: Simplifying semi-supervised learning with consistency and confidence. *Advances in neural information processing systems*, 33:596–608, 2020a.

Sohn, K., Zhang, Z., Li, C.-L., Zhang, H., Lee, C.-Y., and Pfister, T. A simple semi-supervised learning framework for object detection. *arXiv preprint arXiv:2005.04757*, 2020b.

Sun, P., Zhang, R., Jiang, Y., Kong, T., Xu, C., Zhan, W., Tomizuka, M., Li, L., Yuan, Z., Wang, C., et al. Sparse r-cnn: End-to-end object detection with learnable proposals. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 14454–14463, 2021.

Tang, P., Wang, X., Bai, X., and Liu, W. Multiple instance detection network with online instance classifier refinement. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 2843–2851, 2017.

Tang, P., Wang, X., Bai, S., Shen, W., Bai, X., Liu, W., and Yuille, A. Pcl: Proposal cluster learning for weakly supervised object detection. *IEEE transactions on pattern analysis and machine intelligence*, 42(1):176–191, 2018.

Tarvainen, A. and Valpola, H. Mean teachers are better role models: Weight-averaged consistency targets improve semi-supervised deep learning results. *Advances in neural information processing systems*, 30, 2017.

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, Ł., and Polosukhin, I. Attention is all you need. *Advances in neural information processing systems*, 30, 2017.

Xu, M., Zhang, Z., Hu, H., Wang, J., Wang, L., Wei, F., Bai, X., and Liu, Z. End-to-end semi-supervised object detection with soft teacher. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 3060–3069, 2021.

Zeng, Z., Liu, B., Fu, J., Chao, H., and Zhang, L. Wsod2: Learning bottom-up and top-down objectness distillation for weakly-supervised object detection. In *Proceedings of the IEEE/CVF international conference on computer vision*, pp. 8292–8300, 2019.

Zhang, D., Han, J., Cheng, G., and Yang, M.-H. Weakly supervised object localization and detection: A survey. *IEEE transactions on pattern analysis and machine intelligence*, 44(9):5866–5885, 2021.

Zhou, H., Ge, Z., Liu, S., Mao, W., Li, Z., Yu, H., and Sun, J. Dense teacher: Dense pseudo-labels for semi-supervised object detection. In *European Conference on Computer Vision*, pp. 35–50. Springer, 2022.

Zhou, X., Wang, D., and Krähenbühl, P. Objects as points. *arXiv preprint arXiv:1904.07850*, 2019.