Object Localization under Single Coarse Point Supervision

Xuehui Yu1*, Pengfei Chen1*, Di Wu1, Najmul Hassan2, Guorong Li1, Junchi Yan3, Humphrey Shi2,4, Qixiang Ye1, Zhenjun Han1†

1University of Chinese Academy of Sciences, 2U of Oregon, 3Shanghai Jiao Tong University, 4Picsart AI Research (PAIR)

Abstract

Point-based object localization (POL), which pursues high-performance object sensing under low-cost data annotation, has attracted increased attention. However, the point annotation mode inevitably introduces semantic variance for the inconsistency of annotated points. Existing POL methods heavily rely on accurate key-point annotations which are difficult to define. In this study, we propose a POL method using coarse point annotations, relaxing the supervision signals from accurate key points to freely spotted points. To this end, we propose a coarse point refinement (CPR) approach, which to our best knowledge is the first attempt to alleviate semantic variance from the perspective of algorithm. CPR constructs point bags, selects semantic-correlated points, and produces semantic center points through multiple instance learning (MIL). In this way, CPR defines a weakly supervised evolution procedure, which ensures training high-performance object localizer under coarse point supervision. Experimental results on COCO, DOTA and our proposed SeaPerson dataset validate the effectiveness of the CPR approach. The dataset and code will be available at https://github.com/ucas-vg/PointTinyBenchmark/.

1. Introduction

Humans can recognize and easily achieve a sense of the objects present in their eye-sight. In computer-vision, this is usually framed as drawing bounding boxes around objects [21,25,40,41] or dense annotations of the entire scene [13, 16]. However, one inevitable circumstance for training such models is that they require high-quality densely annotated data which is expensive and difficult to obtain. In some applications [27], just the object’s location is necessary while costly annotation (e.g. bounding box) is redundant or even undesirable (e.g. a robotic arm aims at a single point to pick up an object [27]).

Hence point-based object localization (POL) is studied. Due to the simple and time-efficient annotation, point-based object localization has attracted increasing attention in recent years [26,29]. POL based methods require point-level object annotations and can predict the object’s location as a 2D coordinate in the original image.

However, while annotating an object as a point, there can be multiple candidate points. One problem that arises with optional candidate points is that multiple regions of varying semantic information are labeled positive for the same class. Conversely, identical regions with similar semantic information are labeled differently. Take bird category as an example, during annotation, we label the bird’s different body parts (e.g. neck and tail etc.) as positive based on the visible regions in the image. Based on the annotation, for different images in the dataset, we have labeled same body part (e.g. neck) of the bird as both positive and negative (see Fig. 1). Therefore, during training, the model has to consider the neck region as positive for one image and negative in another (the image where tail is annotated). This phenomenon introduces ambiguity and confuses the model which results in poor performance.

Previous works [29, 37] addressed this issue by setting strict annotation rules by annotating only the pre-defined key-point areas of the object. As a result, they suffer from the following challenges: i) the key points are not easy to define, especially for some broadly defined categories where they do not have a specific shape (Fig. 2 (a)); ii) the key point may not exist in the image due to the different poses of objects and different camera views (Fig. 2 (b)); iii) when objects have large scale variance, it is difficult to decide the appropriate granularity of the key points (Fig. 2 (c)). For a person, if the head is a key point [29] (coarse-grained) then there remains a large semantic variance for the
In this paper, we formulate the coarse point-based localization (CPL) paradigm for training a localizer of a general POL, as shown in Fig. 3. We firstly adopt a coarse point annotation strategy, which allows to annotate any point on an object. Then the coarse point refinement (CPR) algorithm is proposed to refine the initialized annotated coarse point to the semantic center in the training set. Finally, the refined points instead of the annotated points are used as supervision to train a localizer. The proposed CPR is the first attempt to alleviate semantic variance from the perspective of algorithm rather than annotation. Specifically, CPR finds the semantic points around the annotated point through multiple instance learning (MIL) [9], then weighted averages the semantic points to obtain the semantic center, which has a smaller semantic variance and a higher tolerance for prediction errors. The contributions are:

1) We dive into point-based object localization (POL) task, and formulate the coarse point based localization (CPL) paradigm for general object localization, extending the previous works to a multi-class/multi-scale POL task;
2) The coarse point refinement (CPR) algorithm is proposed to alleviate the semantic variance from the perspective of algorithm rather than rigid annotation rules;
3) The experimental results show the CPR is effective for CPL, which obtains a comparable performance with the center point (approximate key point) based object localization, and improves the performance over 10 points compared with the baseline;
4) A new dataset with more than 600,000 annotations, named SeaPerson, is introduced in this paper. This dataset can be used for tiny person detection and localization.

2. Related Work

In this section, we review the relevant point-based vision tasks and the vision tasks with multiple instance learning.

2.1. Vision Tasks under Point Supervision

**Pose Estimation.** Human or animal pose estimation aims to locate the position of joint points of persons or animals accurately [31,48,50]. There are several benchmarks built for the task, e.g. COCO [22] and the Human3.6M [17] datasets are the most well-known ones for 2D and 3D pose estimation and AP-10k [45] for animal pose estimation. In these datasets, annotations are a set of accurate key points, and the predicted results are human or animal poses rather than the location of person or animal instances.

**Crowd Counting.** In this task, accurate head annotation is utilized as point supervision [29,37,52]. The crowd density map [15,18,19], generated by head annotation, is chosen as the optimization objective of the network. Furthermore, crowd counting focuses on the number of people rather than each person’s position. It depends on precise key points such as the human head, while the coarse point object localization task only requires the coarse position annotation on the human body.

**Object Localization.** Unlike object detection [12,21,25], especially for rotation detection [42,43], requiring the exact bounding box information, object localization applications [26] are often agnostic to object’s scale. The works [26,29] train a localizer with points instead of bounding boxes. These tasks are summarized as POL in our paper. However, they heavily rely on key-point annotations to reduce the semantic variance.

Different from the above mentioned tasks, our CPL relies upon a coarse point instead of keypoints and deals the semantic variance problem with a novel approach.

2.2. Vision Tasks with Multiple Instance Learning

The paradigm of MIL [9] is that a bag is positively labeled if it contains at least one positive instance; otherwise, it is labeled as a negative bag. Inspired by weakly supervised object detection task, the proposed CPR method follows the MIL paradigm. With the object category and the coarse point annotation, we consider sampled points around each annotated point as a bag and utilize MIL for training.

**Image-level Tasks.** An image is divided into patches, where patches are seemed as instances and the entire im-
age as a bag. Content-based image retrieval [47, 49] is a
conventional MIL task, which just classifies images by their
content. If the image contains at least one object of a class,
the whole bag can be seen as a positive sample for that class.
Otherwise, the bag will be regarded as a negative sample.

**Video-level Tasks.** Firstly, the video is divided into
segments, which will be classified separately and then the
whole video is seemed as a bag. Following the above
pre-processing, MIL is used to identify specific events in
videos [10, 24, 30]. Additionally, some researchers have
applied MIL to video object tracking [1, 2, 2] also achieves
a robust tracker by constructing instance-level bag from box,
but different from our work, it does not handle the negative
samples to suppress background for MIL.

**Object-level Tasks.** MIL is widely used in weakly
supervised object localization and detection (WSOL [8, 38]
and WSOD [4, 7, 32, 34–36]), where only the image-level
annotation is utilized. Firstly, Select Search [33] or Edge
Box [53] methods are used to produce proposal boxes,
which are then used as a bag and each of them as an
instance. Finally, they classified positive and negative samples
by judging whether the image contains at least one object
of a specific class. WSOL/WSOD, only with image-level
annotation, focus on local regions and can not distin-
guish instances due to the lack of object-level annotation.

Annotation of CPL is a coarse point position and the cat-
egory of each object. CPR views sampled points around
the annotated point as a bag and trains object-level MIL to find
a better and stable semantic center.

### 3. Coarse Point Refinement

As shown in Fig. 3, CPR can be regarded as a pre-
processing that transforms the annotations on the training
set to a more conductive form for the subsequent tasks. The
main purpose of CPR is to find a semantic point, which has
a smaller semantic variance and a higher tolerance for pre-
diction errors, to replace the initial annotated point.

In Fig 4 and Algorithm 1, there are three key steps in
CPR: i) Point Sampling: points in the neighborhood of the
annotated point are sampled; ii) CPRNet Training: based on
the sampled points, a network is trained to classify whether
the points are in the same category with the annotated point
or not; iii) Point Refinement: based on the scores obtained
by CPRNet and the constraints (details in Sec. 3.3), the

### Algorithm 1 Coarse Point Refinement

**Input:** Training set $D_{\text{train}}$, CPRNet $E$.

**Output:** Refined points $\hat{A}_{\text{train}}$.

**Note:** $A$ and $C$ are 2D coordinates and category label of annotated points in image $I$ respectively.

1. $L_{\text{CPR}}^{\text{train}} \leftarrow \{\}$;
2. for $(I, A, C) \in D_{\text{train}}$ do
3. Extract feature map $F$ of $I$ with $E$;
4. // Step1: point sampling
5. $B_j \leftarrow \text{bag sampling}(a_j)$ for each $a_j \in A$, Eq. 1;
6. $\text{Neg}_k \leftarrow \text{neg sampling}(k)$ for each category $k \in \{1, 2, ..., K\}$, Eq. 2;
7. // Step2: CPRNet training
8. Calculate $L_{\text{MIL}}$ with $B_j$ and $F$, Eq. 7;
9. Calculate $L_{\text{ann}}$ with $A$ and $F$, Eq. 8;
10. Calculate $L_{\text{neg}}$ with $\text{Neg}_k$ and $F$, Eq. 10;
11. Sum $L_{\text{MIL}}, L_{\text{ann}}$ and $L_{\text{neg}}$ to obtain the CPR loss $L_{\text{CPR}}$,
12. $L_{\text{CPR}}^{\text{train}} \leftarrow L_{\text{CPR}}^{\text{train}} + L_{\text{CPR}}$;
13. end for
14. Train $E$ by minimizing $L_{\text{CPR}}^{\text{train}}$ to obtain $\hat{E}$.
15. // Step3: point refinement
16. $A_{\text{train}} \leftarrow \{\}$;
17. for $(I, A, C) \in D_{\text{train}}$ do
18. $A \leftarrow \text{Point Refinement}(\hat{E}, I, A, C)$, Algorithm 2;
19. $A_{\text{train}} \leftarrow A_{\text{train}} \cup \{A\}$;
20. end for

points, having similar semantic information with the an-
notated point, are chosen as the semantic points and then
weighted with their scores to obtain the semantic center as
the final refined point.

### 3.1. Point Sampling

In this paper, $K$ denotes the number of categories, $a_j \in R^2$ and $c_j \in \{0, 1\}^K$ denote the annotated point’s 2D co-
dinate and the category label of $j$-th instance. $p = (p_x, p_y)$
denotes a point on a feature map.

**Point Bag Construction.** In Fig. 4, to sample points
uniformly in the neighborhood of $a_j$, we define $R$ circles
with $a_j$ as the center, where the radius of the $r$-th ($1 \leq r \leq R, r \in N^+$) circle is set as $r$. Then we sample $r \cdot u_0 \cdot 8$ (by default) points with equal intervals around
the circumference of the $r$-th circle, and obtain $\text{Circle}(a_j, r)$.
All sampled points of the $R$ circles are defined as points’
bag of $a_j$, denoted as $B_j$ in Eq. 1. The points outside the
feature map are excluded.

$$\text{Circle}(p, r) = \left\{ \left( p_x + r \cdot \cos \left( 2\pi \cdot \frac{i}{u_0 \cdot r} \right), p_y + r \cdot \sin \left( 2\pi \cdot \frac{i}{u_0 \cdot r} \right) \right) \mid 0 \leq i < r \cdot u_0, i \in N^+ \right\}; \quad (1)$$

$B_j = \cup_{1 \leq r \leq R} \text{Circle}(a_j, r)$.

$B_j$ is used for calculating the MIL loss for CPRNet train-
and obtaining the semantic points for point refinement.

Negative Point Sampling. All integer points on the feature map, outside the circles with radius $R$ of all annotated points of a given category, will be selected as negative samples. The negative samples of category $k$ can be defined as:

$$Neg_k = \{(p_x, p_y)|p_x \leq w, p_y \leq h, p_x \in N^+, p_y \in N^+\} \quad \forall (a_x, a_y),$$

where $||p-a_j||$ is the Euclidean distance between $p$ and $a_j$, $w$ and $h$ are the width and height of a given feature map.

3.2. CPRNet Training

This section gives the details of the objective function of training CPRNet based on the sampled points bag $B_j$ ($j \in \{1, 2, \ldots, M\}$) and the negative points $Neg_k$ ($k \in \{1, 2, \ldots, K\}$), where $M$ and $K$ are the amount of instances and categories. $U$ is defined as the amount of points in $B_j$.

CPRNet. CPRNet adopts FPN [20] with ResNet [14] as the backbone. Only P2 or P3 is used due to the lack of scale information in point annotation. After four $3 \times 3$ conv layers followed by the ReLU [11] activation, the final feature map $F \in \mathbb{R}^{h \times w \times d}$ is obtained, where $h \times w$ is the corresponding spatial size and $d$ is the dimension of channel. For a given point $p = (p_x, p_y)$, $F_p \in \mathbb{R}^d$ denotes the feature vector of $p$ on $F$. If $p$ is not an integer point, the bilinear interpolation is used to obtain $F_p$.

CPR Loss. Object-level MIL loss is introduced to endrow CPRNet the ability of finding semantic points around each annotated point. Then to overcome the overfitting problem of MIL when the data is insufficient, we further introduce the instance-level prior as supervision by designing annotation and negative loss. The objective function of CPRNet is a weighted summation of the three losses:

$$L_{CPR} = L_{MIL} + \alpha_{ann}L_{ann} + \alpha_{neg}L_{neg},$$

where $\alpha_{ann} = 0.5$ and $\alpha_{neg} = 3$ (by default in this paper). And $L_{MIL}$, $L_{ann}$ and $L_{neg}$ are based on the focal loss [21]:

$$FL(S_p, c_j) = \sum_{k=1}^{K} c_{j,k}(1 - S_{p,k})^\gamma \log(S_{p,k}) + \alpha(1 - c_{j,k})S_{p,k}^\gamma \log(1 - S_{p,k}),$$

where $\gamma$ is set as 2 in Eq. 4 following the standard focal loss, and $S_{p,k} \in \mathbb{R}^K$ and $c_{j,k} \in \{0, 1\}^K$ are the predicted scores on all categories and the category label, respectively.

Object-level MIL Loss. To find the semantic points during refinement, we refer to WSOD [4] and design a MIL loss to enable the CPRNet justify whether the points in $B_j$ are in the same category with $a_j$. Based on $B_j$, the feature vectors $\{F_p|p \in B_j\}$ are extracted. As Eq. 5 shows, for each $p \in B_j$, a classification branch $f_{cls}$ is applied to obtain the logits $[O^{cls}_{B_j}]_p$, which is then utilized as an input of an activation function $\sigma_1$ to obtain $[S^{cls}_{B_j}]_p$. Besides, an instance selection branch $f_{ins}$ is applied to $F_p$ to obtain $[O^{ins}_{B_j}]_p$, which is then utilized as an input of an activation function $\sigma_2$ to obtain the selection score $[S^{ins}_{B_j}]_p$. The score $[S^{over}_{B_j}]_p$ is obtained by taking the element-wise product of $[S^{ins}_{B_j}]_p$ and $[S^{cls}_{B_j}]_p$.

$$[O^{cls}_{B_j}]_p = f_{cls}(F_p) \in \mathbb{R}^K, \quad [O^{ins}_{B_j}]_p = f_{ins}(F_p) \in \mathbb{R}^K;$$

$$[S^{cls}_{B_j}]_p = \sigma_1([O^{cls}_{B_j}]_p) = 1/(1 + e^{-[O^{cls}_{B_j}]_p});$$

$$[S^{ins}_{B_j}]_p = \sigma_2([O^{ins}_{B_j}]_p) = e^{[O^{ins}_{B_j}]_p} \sum_{p' \in B_j} e^{-[O^{ins}_{B_j}]_{p'}} \in \mathbb{R}^K;$$

$$[S^{over}_{B_j}]_p = [S^{ins}_{B_j}]_p \cdot [S^{cls}_{B_j}]_p \in \mathbb{R}^K,$n

where $\sigma_2$ is a softmax function. Different from MIL in WSOD, the sigmoid activation function is applied for $\sigma_1$ due to its suitability for binary task compared with the softmax function. Furthermore, the sigmoid activation function allows to perform multi-label classification (for the overlapping area of multiply objects’ neighborhood) for points and is more compatible with focal loss.
The bag-level score $S_{B_j}$ is obtained by the summation of all points’ scores in $B_j$ by Eq. 6. $S_{B_j}$ can be seen as the weighted summation of the classification score $[S^{cls}_{B_j}]_p$ of $p$ in $B_j$ by the corresponding selection score $[S^{inc}_{B_j}]_p$.

$$S_{B_j} = \sum_{p \in B_j} [S^{inc}_{B_j}]_p p \in \mathbb{R}^K.$$  
(6)

The MIL is finally given by the focal loss on the predicted bag-level scores $S_{B_j}$ and the category label $c_j$ of $a_j$:

$$L_{MIL} = \frac{1}{M} \sum_{j=1}^M FL(S_{B_j}, c_j).$$  
(7)

**Annotation Loss.** Due to the lack of explicit positive samples for supervision in MIL, the network sometimes focuses on the points outside the instance region and mistakenly regards them as the foreground. Therefore, we introduce the annotation loss $L_{ann}$, that gives the network accurate positive samples for supervision via annotated points, to guide MIL training. $L_{ann}$ can guarantee a high score of the annotated point and mitigate mis-classification to some extent. Firstly, the classification score of $S_{a_j}(j \in 1, 2, ... M)$ of $a_j$ is calculated as:

$$S_{a_j} = \sigma_1(f_{cls}(F_{a_j})) \in \mathbb{R}^K.$$  
(8)

$L_{ann}$ is calculated with focal loss as:

$$L_{ann} = \frac{1}{M} \sum_{j=1}^M FL(S_{a_j}, c_j).$$  
(9)

**Negative Loss.** The conventional MIL adopts binary log loss, and it views the proposals belonging to other categories as negative samples. For lacking of explicit supervision from samples in background, the negative samples are not well suppressed during MIL training. Therefore, based on $N_{neg}$, the negative loss $L_{neg}$, the negative part of focal loss, is calculated as follows, where we set $\gamma = 2$.

$$S_p = \sigma_1(f_{cls}(F_p)) \in \mathbb{R}^K;$$  
$$L_{neg} = \frac{1}{M} \sum_{k=1}^K \sum_{p \in N_{neg}} c_j k S_{p,k} \log(1 - S_{p,k}).$$  
(10)

### 3.3 Point Refinement

As described in Algorithm 2, the trained CPRNet $\hat{E}$ is used to refine the annotated point. Based on $B_j$, $[S^{cls}_{B_j}]_p$ predicted by $\hat{E}$ and the constrains (details given in following), points with the same category (similar semantic) with the annotated point are selected, denoted as $B_j^+$. Then, the semantic center (final refined point), used to replace the annotated point, is set as the weighted mean of points in $B_j^+$.

To obtain $B_j^+$, three constraints (line in blue in Algorithm 2) are introduced. Constraint I is to delete points with small classification scores. We filter out the point $p \in B_j$ whose $S_{p,k_j}$ is smaller than the threshold $\delta_1$ (set as 0.1 by default) or $\delta_2 * S_{a_j,k_j}$, where $\delta_2$ is set as 0.5 by default and $k_j$ is category label (not one-hot format) of $j$-th object.

**Algorithm 2 Point Refinement**

**Input:** Trained CPRNet $\hat{E}$, input image $I$, annotated points $A$, category label of annotated points $C$.

**Output:** Refined points $\hat{A}$.

**Note:** $\delta_1$, $\delta_2$ are thresholds, $K$ is the number of categories, $S_{p,k}$ is the predicted score on $k$-th category of point $p$, $k_j \in \{1, 2, ... K\}$ is the category label (not one-hot format) of the $j$-th object.

1: $\hat{A} \leftarrow \emptyset$;
2: $F \leftarrow extract\_feature(I; \hat{E})$ according to Sec. 3.2;
3: for $a_j \in A$, $c_j \in C$ do
4:     find $k_j \in \{1, 2, ... K\}$ s.t. $c_{j,k_j} = 1$;
5:     $B_j^+ \leftarrow \{a_j\}$;
6:     $S_{a_j} \leftarrow \sigma_1(f_{cls}(F_{a_j}; \hat{E})) \in \mathbb{R}^K$;
7:     $B_j \leftarrow bag\_sampling(a_j)$ according to Eq. 1;
8:     for $p \in B_j$ do
9:         $S_p \leftarrow \sigma_1(f_{cls}(F_p; \hat{E})) \in \mathbb{R}^K$;
10:        $s_p \leftarrow S_{p,k_j}$;
11:       if $s_p > \delta_1$ and $s_p > \delta_2 * S_{a_j,k_j}$ and $k_j = \text{argmax}_{1 \leq k \leq K} S_{p,k}$ and $a_j = \text{argmin}_{c \in C} ||p - a||$ then
12:           $B_j^+ \leftarrow B_j^+ \cup \{p\}$;
13:       end if
14:     end for
15:     $\hat{a}_j \leftarrow (\sum_{p \in B_j^+} S_p * p) / (\sum_{p \in B_j^+} S_p)$;
16:     $\hat{A} \leftarrow \hat{A} \cup \{\hat{a}_j\}$;
17: end for

With the point sampling, CPRNet training and Point refinement mentioned above, the CPR can effectively mitigate semantic variance, as shown in Fig. 5.

### 4. Experiment

**4.1. Experimental Settings**

**Datasets.** For experimental comparisons, three public available datasets are used for point supervised localization task: COCO [22], DOTA-v1.0 [39] and SeaPerson. COCO is MSCOCO 2017, and it has 118k training and 5k validation images with 80 common categories. Since the ground-truth on the test set are not released, we train our model on the training set and evaluate it on the validation set.
DOTA (v1.0) provides 2,806 images with 15 object categories. We utilize training set for the training and validation set for evaluation. SeaPerson\textsuperscript{1} is a dataset for tiny person detection collected through a UAV camera at the seaside. The dataset contains 12,032 images and 619,627 annotated persons with low resolution. The images in the SeaPerson are randomly selected as training, validation, and test sets with the proportion of 10:1:10. The details are given in the supplemental materials.

Coarse Point Annotation In practical scenarios, the coarse point can be obtained through annotating any single point on an object. However, since datasets in the experiment are already annotated with masks or bounding boxes, according to the law of large numbers, it is reasonable that the manually annotated points follow Gaussian distribution. Furthermore, since the annotated points must be inside the bounding box or mask of the object, then an improved Gaussian distribution, named as Rectified Gaussian (RG) Distribution, is utilized for annotation. $RG(p; \mu, \sigma)$ is chosen to generate the point annotations for the experiments.

$$\phi(p; \mu, \sigma) = Gauss(p; \mu, \sigma) \cdot Mask(p);$$

$$RG(p; \mu, \sigma) = \int_{p} \phi(p; \mu, \sigma).$$

where $\mu$ and $\sigma$ are the mean and standard deviation of Gaussian distribution, respectively. $Mask(p) \in \{0, 1\}$ denotes whether point $p$ falls inside the mask of an object. If it is generated from the bounding box annotation, then the box is treated as a mask.

Evaluation. Similar to WSOD, a point-box distance, calculated between the point and box, is used for evaluation. Specifically, the distance $d$ between point $a = (x, y)$ and bounding box $b = (x^c, y^c, w, h)$ is defined as:

$$d(a, b) = \sqrt{(\frac{x - x^c}{w})^2 + (\frac{y - y^c}{h})^2}.$$  

where $(x^c, y^c), w, h$ are the center point, width, and height of the bounding box, respectively. The distance $d$ is used as the matching criterion for POL performance. A point and the object’s bounding box are matched if the distance $d$ is smaller than a predefined threshold $\tau$. (e.g. $\tau = 1.0$ means that as long as the point falls within an matched ground-truth box, the point successfully matches the ground-truth box.) If a bounding box has multiple matched points, the point with the highest score is chosen. While a point has multiple matched objects, the object with the smallest point-box distance is selected. A true positive (TP) is counted if a point matches an object. Otherwise, a false positive (FP) is counted. Neither TP nor FP will be counted if a point matches an object that is annotated as ignore, which follows the evaluation criteria of pedestrian detection [51] and TinyPerson benchmark [46]. We adopt mean Average Precision with $\tau = 1.0$ ($mAP_{1.0}$) as the main metric for experimental comparisons. We do not consider a small $\tau$ because it makes the task more like center localization instead of object localization. Here we also report the results of $\tau = 0.5$ and $\tau = 2.0$ in Table 3, which can be informative.

Implementation Details for CPRNet. Our codes are based on MMDetection [6]. Same to the default setting of the object detection on COCO, the stochastic gradient descent (SGD [5]) algorithm is used to optimize in 1x training schedule. The learning rate is set to 0.0025 and decays by 0.1 at the 8-th and 11-th epoch, respectively.

4.2. Experimental Comparisons

The POL task with coarse point annotation is divided into two key parts: refining the coarse point annotation and training the point localizer with refined points.

Detector with Pseudo Box. For training a point localizer, an intuitive idea is to convert the point-to-point (POL) to a box-to-box (object detection) problem. Firstly, a fixed-size pseudo-box is generated with each annotated point as the center. Next, the pseudo-box is used to train a detector. Finally, during inference, the center points of the boxes predicted by the trained detector are used as the final output. Following [26], we conduct the pseudo box for localization and give the performance in row 1 of Table 1. The difference to [26] is that the RepPoint [44] is utilized rather than Faster RCNN [25], due to its efficiency.

Multi-Class P2PNet. We adopt P2PNet\textsuperscript{2}, to train with point annotation and predict point for each object during inference, as a stronger baseline for POL task. Improvement

\textsuperscript{1}SeaPerson is a low-resolution tiny person dataset and disclose little personal privacy from the appearance.

\textsuperscript{2}In our experiments, we re-implement P2PNet and further endow it the new ability of handling multi-class prediction, which to our best efforts, aligns the results with those reported in the raw paper [29].
Figure 6. The top-left and top-right figures are the relative position distribution (Eq. 14) heatmap of annotated points and final refined points in bounding boxes. The top-middle figure is RSV curve of refined points (Eq. 13)) in each epoch during CPRNet training. The bottom four figures give the annotated points (green) and refined points (yellow) in four epochs, showing the refined points gradually converge to the semantic center points.

Table 1. The experimental comparisons (mAP\text{all}) of localizers in three datasets: COCO, DOTA and SeaPerson. RepPoint" means RepPoint with pseudo box (details in Sec. 4.2).

| refinement                  | COCO       | DOTA       | SeaPerson  |
|-----------------------------|------------|------------|------------|
| -                           | RepPoint   | 57.42      | 47.22      | 47.72      |
|                             | P2PNet     | 38.48      | 48.34      | 76.52      |
| Self-Refinement (ours)      | P2PNet     | 50.36      | 60.39      | 84.96      |
| CPR (ours)                  | P2PNet     | 55.46      | 63.81      | 85.86      |

Table 2. The effect of training loss in CPRNet: MIL loss, annotation loss, negative loss. The pos loss is for comparison.

| pos | MIL | ann  | neg | mAP\text{1/1} | mAP\text{1/2} | mAP\text{2/2} |
|-----|-----|------|-----|---------------|---------------|---------------|
| ✓   | ✓   | ✓    | ✓   | 39.07         | 28.37         | 46.90         |
| ✓   | ✓   | ✓    | ✓   | 39.45         | 28.22         | 47.42         |
| ✓   | ✓   | ✓    | ✓   | 54.24         | 46.56         | 58.94         |
| ✓   | ✓   | ✓    | ✓   | 51.82         | 46.24         | 55.67         |
| ✓   | ✓   | ✓    | ✓   | 42.72         | 33.07         | 48.81         |
| ✓   | ✓   | ✓    | ✓   | 55.46         | 50.23         | 59.49         |

Table 3. mAP\text{all} of more architectures on the three datasets.

| detector                      | COCO       | DOTA       | SeaPerson  |
|-------------------------------|------------|------------|------------|
| RetinaNet                     | 52.61      | 51.53      | 48.50      |
| FasterRCNN                    | 35.29      | 51.15      | 47.93      |
| RepPoint                      | 37.42      | 47.26      | 47.72      |
| RetinaNet w. CPR              | 51.35      | 63.69      | 77.90      |
| FasterRCNN w. CPR             | 53.21      | 63.00      | 77.80      |
| RepPoint w. CPR               | 53.97      | 60.37      | 78.94      |

With these refined points as supervision, the performance of P2PNet as point localizer is given in row 3 in Table 1, where it alleviates the semantic variance problem. **CPR.** Compared with self-refinement, CPR (shown in row 5 of Table 1) obtains more performance gain, indicating it is more efficient for dealing with the semantic variance.

To quantify the semantic variance of point annotation, the relative semantic variance (RSV), which is calculated based on the relative distance of the point to the center point:

\[
R_{SV} = |\text{Var}(x') + \text{Var}(y')| \frac{1}{w} \frac{1}{h}
\]

\[(13)\]

where \((x', y')\) is an annotated point or refined point and \((x^c, y^c)\) is the corresponding center point in the bounding box of an object. \(\text{Var}(x')\) and \(\text{Var}(y')\) are the variance of \(x'\) and \(y'\) of all objects in the dataset, respectively. Statistically, the smaller RSV means the \((x, y)\) holds a more stable relative position to its corresponding \((x^c, y^c)\), as shown in Eq. 13. Considering the \((x^c, y^c)\) as a strict key point, the intuition behind the RSV is that a small RSV for a category is equivalent to a strict annotation, which can effectively reduce the semantic variance of annotations. As shown in Fig. 6, the annotated coarse point holds a larger RSV, while the refined point via CPR obtains a smaller RSV.

To show the relative position distribution of annotated points in the bounding boxes, we calculate \(\text{Prob}(x', y')\) as:

\[
\text{Prob}(x', y') = \frac{\sum_{1 \leq j \leq M} \mathbb{1}\{x'_{j} = p_{x} \text{ and } y'_{j} = p_{y}\}}{M}
\]

\[(14)\]

where \((x'_{j}, y'_{j})\) is relative position of annotated point or refined point for object \(j\) in dataset, \(\mathbb{1}\{\text{something}\}\) is 1 if something is true, otherwise 0. \(\text{Prob}(x', y')\) is shown as the heatmap in Fig. 6.

**Performance Analysis.** Pseudo box based localizer is almost equivalent to train a detector that treats the points in the neighborhood of the annotated points as positive samples and others as negative samples. The general detectors perform label assignment with IoU, which depends heavily on the scale information of the given bounding box. However, precise bounding box can not be obtained from point annotation of POL, leading to poor performance of Pseudo box based localizer. P2PNet adopts hungarian algorithm to achieve a purely point-to-point assignment, obtaining better performance than pseudo box based localizer. However, P2PNet is much sensitive to the accuracy of annotated
4.3. Ablation Studies

To further analyze CPR’s effectiveness and robustness, we conduct more experiments on COCO.

**Training Loss in CPRNet.** Ablation study of the training loss is given in Table 2. The CPR loss given in row 6 in Table 2 obtains 55.46 mAP. i) **MIL loss.** If the MIL loss is removed (row 4), the CPRNet training relies on the annotation loss and the negative loss, the performance drops 3.64 points (51.82 vs 55.46). When we replace the MIL loss with the pos loss, which treats all the sampled points in the MIL bag as positive samples (line 5), the performance sharply declines by 12.74 points (42.72 vs 55.46), showing that MIL can autonomously discern points belonging to the object. ii) **Annotation loss.** Lacking of the annotation loss (row 3), the performance of localization decreases 1.22 points (54.24 vs 55.46). The annotation loss guides the training through an given accurate positive supervision. iii) **Negative loss.** With the negative loss (row 2), the performance improves by 16.01 points (55.46 vs 39.45), indicating that only MIL loss is not enough to suppress the background, and the negative loss is inevitable.

**Feature Map Level.** The CPRNet is established based on single level feature map of FPN. Table 4a and 4b show the performance with different feature map levels. Since the performance on P3 is similar to that of P2, P3 is chosen for our experiments in COCO if not otherwise specified.

**Sampling Scope.** Table 4a and 4b show the performance of different radius R, where R is a sensitive hyperparameter in CPRNet. On P3, the best performance 55.46 is obtained when R is set as 8. If the sampling scope reduces, such as R = 5, the performance significantly declines to 53.32, since the sampling scope is limited to a small local region, leading to a worse refinement. While the scope getting larger, the performance becomes steady but drops slowly until R is over 25 (53.85), since the bag B̂ j for MIL introduces more noise, which degrades the performance.

**Point Refinement Policy.** For point refinement, there are three constraints (described in Sec. 3.3). δ1 and δ2 are threshold of constraint I. In Table 4c, it shows that the three constraints together obtain performance gain.

**Upper Bound Analysis.** To further validate the CPR, a comparison between CPR and a strict annotation based localizer, which can be seen as the upper bound for CPR, is conducted on COCO. Since it is hard to annotate the objects in general dataset (e.g. COCO) with key points. Therefore, we approximately use the center point of each objects’ bounding box as a kind of strict point annotation. The experiment results in Table 4d show that CPR can achieve a comparable performance to center point annotation based localizer (55.46 vs 57.47).

**Localizer Architecture.** Table 3 shows that CPR can further improve the performance of different localizers, such as Faster-RCNN, RetinaNet and RepPoint.

**Backbone.** As shown in Table 4e, due to the stronger backbone Resnet-101 for CPRNet and P2PNet, it obtains a better performance 56.43.

5. Conclusion and Outlook

In this paper, we rethink the semantic variance problem in point-based annotation caused by the non-uniqueness of optional annotated points. The proposed CPR samples points in neighbourhood, finds the semantic points on the object by introducing MIL, and then weighted averages these semantic points to obtain the semantic center of the object as the supervision for the localizer. CPR alleviates semantic variance and facilitates the extension of POL task to multi-class and multi-scale. Comprehensive ablations on multiple datasets further verify the effectiveness of our model. In future, we will study on an adaptive R and explore the possibilities of extending CPR to other tasks.

**Limitation.** The performance is sensitive to R, which is not an adaptive value in this paper and may limit CPR to better deal with the multi-scale of objects to some extent.

**Broader Impact.** Similar to most of object detection and localization task, the bias of dataset from the intrinsic artifacts are not considered.

6. Acknowledgements

This work was supported in part by the Youth Innovation Promotion Association CAS, the National Natural Science Foundation of China (NSFC) under Grant No. 61836012 and 61774417, and the Strategic Priority Research Program of the Chinese Academy of Sciences under Grant No.XDA27000000. The work was partially done during Xuehui’s internship at JD Explore Academy, China. We would like to thank Jing Zhang for helpful discussion and suggestions.
References

[1] Boris Babenko, Ming-Hsuan Yang, and Serge J. Belongie. Visual tracking with online multiple instance learning. In *CVPR*, 2009. 3

[2] Boris Babenko, Ming-Hsuan Yang, and Serge J. Belongie. Robust object tracking with online multiple instance learning. *IEEE TPAMI*, 2011. 3

[3] Hessam Bagherinezhad, Maxwell Horton, and Mohammad Rastegari et al. Label refinery: Improving imagenet classification through label progression. *CoRR*, 2018. 7

[4] Hakan Bilen and Andrea Vedaldi. Weakly supervised deep detection networks. In *CVPR*, 2016. 3, 4

[5] Léon Bottou. Stochastic gradient descent tricks. In *Neural Networks: Tricks of the Trade - Second Edition*. Springer, 2012. 6

[6] Kai Chen, Jiaqi Wang, Jiangmiao Pang, Yuhang Cao, Yu Xiong, Xiaoxiao Li, Shuyang Sun, Wansen Feng, Ziwei Liu, Jiaru Xu, Zheng Zhang, Dazhi Cheng, Chenchen Zhu, Tianheng Cheng, Qijie Zhao, Buyu Li, Cheng Xiu, Rui Zhu, Yue Wu, Jifeng Dai, Jingdong Wang, Jianping Shi, Wanli Ouyang, Chen Change Loy, and Dahua Lin. MMDetection: Open mmlab detection toolbox and benchmark. *arXiv preprint arXiv:1906.07155*, 2019. 6

[7] Ze Chen, Zhihang Fu, and Rongxin Jiang. SLV: spatial likelihood voting for weakly supervised object detection. In *CVPR*, 2020. 3

[8] Ramazan Gokberk Cinbis, Jakob J. Verbeek, and Cordelia Schmid. Weakly supervised object localization with multi-fold multiple instance learning. *IEEE TPAMI*, 2017. 3

[9] Thomas G. Dietterich, Richard H. Lathrop, and Tomás Lozano-Pérez. Solving the multiple instance problem with axis-parallel rectangles. *Artificial Intelligence*, 1997. 2

[10] Jia-Chang Feng, Fa-Ting Hong, and Wei-Shi Zheng. MIST: multiple instance self-training framework for video anomaly detection. In *CVPR*, 2021. 3

[11] Xavier Glorot, Antoine Bordes, and Yoshua Bengio. Deep sparse rectifier neural networks. In *Proceedings of the Fourteenth International Conference on Artificial Intelligence and Statistics, AISTATS 2011*, 2011. 4

[12] Yuqi Gong, Xuehui Yu, Yao Ding, Xiaoke Peng, Jian Zhao, and Zhenjun Han. Effective fusion factor in FPN for tiny object detection. In *IEEE WACV*, 2021. 2

[13] Kaiming He, Georgia Gkioxari, Piotr Dollár, and Ross Girshick. Mask r-cnn. In *ICCV*, 2017. 1

[14] Kaiming He, Xiangyu Zhang, and Shaoqing Ren et al. Deep residual learning for large-scale image recognition. In *CVPR*, 2016. 4

[15] Yutao Hu, Xiaolong Jiang, and Xuhui Liu et al. Nasc-count: Counting-by-density with neural architecture search. In *ICCV*, 2020. 2

[16] Zilong Huang, Xinggang Wang, Lichao Huang, Chang Huang, Yunchao Wei, and Wenyu Liu. Cenet: Criss-cross attention for semantic segmentation. In *ICCV*, 2019. 1

[17] Catalin Ionescu, Dragos Papava, Vlad Olaru, and Cristian Sminchisescu. Human3.6m: Large scale datasets and predictive methods for 3d human sensing in natural environments. *IEEE TPAMI*, 2014. 2

[18] Xiaoheng Jiang, Li Zhang, and Mingliang Xu et al. Attention scaling for crowd counting. In *CVPR*, 2020. 2

[19] Victor S. Lempitsky and Andrew Zisserman. Learning to count objects in images. In *NeurIPS*, 2010. 2

[20] Tsung-Yi Lin, Piotr Dollár, and Ross B. Girshick et al. Feature pyramid networks for object detection. In *CVPR*, 2017. 4

[21] Tsung-Yi Lin, Priya Goyal, Ross B. Girshick, Kaiming He, and Piotr Dollár. Focal loss for dense object detection. In *ICCV*, 2017. 1, 2, 4

[22] Tsung-Yi Lin, Michael Maire, and Serge et al. Belongie. Microsoft coco: Common objects in context. In *ECCV*, 2014. 2, 5

[23] Alexander Neubeck and Luc Van Gool. Efficient non-maximum suppression. In *ICPR*, 2006. 7

[24] Phuc Nguyen, Ting Liu, and Gautam Prasad et al. Weakly supervised action localization by sparse temporal pooling network. In *CVPR*, 2018. 3

[25] Shaoqing Ren, Kaiming He, and Ross B. Girshick et al. Faster R-CNN: towards real-time object detection with region proposal networks. In *NeurIPS*, 2015. 1, 2, 6

[26] Javier Ribera, David Guera, Yuhaao Chen, and Edward J. Delp. Locating objects without bounding boxes. In *CVPR*, 2019. 1, 2, 6

[27] Björn Runge. Deep learning for point detection in images, 2020. 1

[28] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. In *ICLR*, 2015. 7

[29] Qingyu Song, Changan Wang, Zhengkai Jiang, Yabiao Wang, Ying Tai, Chengjie Wang, Jilin Li, Feiyue Huang, and Yang Wu. Rethinking counting and localization in crowds: A purely point-based framework. In *ICCV*, 2021. 1, 2, 5, 6

[30] Waqas Sultani, Chen Chen, and Mubarak Shah. Real-world anomaly detection in surveillance videos. In *CVPR*, 2018. 3

[31] Ke Sun, Bin Xiao, Dong Liu, and Jingdong Wang. Deep high-resolution representation learning for human pose estimation. In *CVPR*, pages 5693–5703, 2019. 2

[32] Peng Tang, Xinggang Wang, and Song Bai et al. PCL: proposal cluster learning for weakly supervised object detection. *IEEE TPAMI*, 2020. 3

[33] Koen E. A. van de Sande, Jasper R. R. Uijlings, and Theo Gevers et al. Segmentation as selective search for object recognition. In *ICCV*, 2011. 2

[34] Fang Wang, Pengxiu Wei, and Wei Ke et al. C-MIL: continuation multiple instance learning for weakly supervised object detection. In *CVPR*, 2019. 3

[35] Fang Wang, Pengxiu Wei, Zhenjun Han, Jianbin Jiao, and Qixiang Ye. Min-entropy latent model for weakly supervised object detection. *IEEE PAMI*, 2019. 3

[36] Jiajie Wang, Jiangchao Yao, and Ya Zhang et al. Collaborative learning for weakly supervised object detection. In *IJCAI*, 2018. 3

[37] Qi Wang, Junyu Gao, and Wei Lin et al. NWPU-crowd: A large-scale benchmark for crowd counting and localization. *IEEE TPAMI*, 2021. 1, 2
[38] Pingyu Wu, Wei Zhai, and Yang Cao. Background activation suppression for weakly supervised object localization. *arXiv preprint arXiv:2112.00580*, 2021.

[39] Gui-Song Xia, Xiang Bai, and Jian Ding et al. DOTA: A large-scale dataset for object detection in aerial images. In *CVPR*, 2018.

[40] Xue Yang, Liping Hou, Yue Zhou, Wentao Wang, and Junchi Yan. Dense label encoding for boundary discontinuity free rotation detection. In *CVPR*, 2019.

[41] Xue Yang and Junchi Yan. Arbitrary-oriented object detection with circular smooth label. In *ECCV*, 2019.

[42] Xue Yang, Junchi Yan, Qi Ming, Wentao Wang, Xiaopeng Zhang, and Qi Tian. Rethinking rotated object detection with gaussian wasserstein distance loss. In *ICML*, 2021.

[43] Xue Yang, Xiaojiang Yang, Jirui Yang, Qi Ming, Wentao Wang Qi Tian, and Junchi Yan. Learning high-precision bounding box for rotated object detection via kullback-leibler divergence. In *NeurIPS*, 2021.

[44] Ze Yang, Shaohui Liu, and Han Hu et al. Reppoints: Point set representation for object detection. In *ICCV*, 2019.

[45] Hang Yu, Yufei Xu, Jing Zhang, Wei Zhao, Ziyu Guan, and Dacheng Tao. Ap-10k: A benchmark for animal pose estimation in the wild. In *Thirty-fifth Conference on Neural Information Processing Systems Datasets and Benchmarks Track (Round 2)*, 2021.

[46] Xuehui Yu, Yuqi Gong, and Nan Jiang et al. Scale match for tiny person detection. In *IEEE WACV*, 2020.

[47] Dan Zhang, Fei Wang, and Zhenwei Shi et al. Interactive localized content based image retrieval with multiple-instance active learning. *Pattern Recognition*, 2010.

[48] Jing Zhang, Zhe Chen, and Dacheng Tao. Towards high performance human keypoint detection. *IJCV*, 129(9):2639–2662, 2021.

[49] Qi Zhang, Sally A. Goldman, and Wei Yu et al. Content-based image retrieval using multiple-instance learning. In *ICML*, 2002.

[50] Qiming Zhang, Yufei Xu, Jing Zhang, and Dacheng Tao. Vitaev2: Vision transformer advanced by exploring inductive bias for image recognition and beyond. *arXiv preprint arXiv:2202.10108*, 2022.

[51] Shanshan Zhang, Rodrigo Benenson, and Bernt Schiele. Citypersons: A diverse dataset for pedestrian detection. In *CVPR*, 2017.

[52] Yingying Zhang, Desen Zhou, and Sigin Chen et al. Single-image crowd counting via multi-column convolutional neural network. In *CVPR*, 2016.

[53] C. Lawrence Zitnick and Piotr Dollár. Edge boxes: Locating object proposals from edges. In *ECCV*, 2014.
Appendix

A. More experiment.

Sample shape and sample density. In Table 5b, we ablate on sample density \(u_0\) and validate that \(u_0\) has little impact. Theoretically, circles are orientation equivalent, which makes it suitable to model multi-view object. Experimentally, the sampling results of circle and rectangle are very close due to the discrete sampling, leading to comparable performance (Table 5a, where different aspect ratio rect. \(a\) \& \(b\) are studied).

Threshold of refinement. \(\delta_1\) and \(\delta_2\) are thresholds, well defined in detection and localization tasks. \(\delta_1, \delta_2, u_0\) have little impact to performance (Table 5).

B. SeaPerson

SeaPerson is building for tiny person localization, which can help maritime quick rescue, beach safety inspection and so on. The resolution of images are mainly 1920 × 1080 and the person size is extremely low (about 22.6 pixels). Therefore, there is no privacy sensitive information.

Dataset Collection. SeaPerson is collected as: i) Videos are recorded in various seaside scenes by a RGB camera on a Unmanned Aerial Vehicle. ii) We sample an image of every 50 frames from video and remove images with high homogeneity. iii) We annotated all persons in all sampled images with bounding boxes. iv) Following the rules of coarse point annotation in Sec 4.1, coarse point annotation is obtained on SeaPerson for POL task.

Dataset Splitting. We randomly split dataset into three subsets (training set, valid set and test set), while images from the same video sequence cannot be separated into different subsets. As shown in Table 6, the ratio of images’ number in training set, valid set and test set is about 10:1:10.

Dataset Properties. Our proposed SeaPerson is similar with TinyPerson while the volume of SeaPerson is about 7 times that of TinyPerson. The absolute size and relative size of objects are very small as shown in Table 7 and Fig. 7. In such scenario, we only care about the position of the object rather than the size of the object, which makes it very suitable for POL task. In addition, SeaPerson can also be used as a dataset for tiny object detection due to the bounding box annotation.

C. Implementation Details for CPRNet

ResNet-50 is used as the backbone network unless otherwise specified and FPN is adopted for feature fusion. P2 (stride is 4) is used for SeaPerson and P3 (stride is 8) is used for COCO and DOTA. The mini-batch is 64/4/4 images and all models are trained with 8/2/4 GPUs for COCO/DOTA/SeaPerson. The training epoch numbers are set as 12/12/6, the learning rate are set as 0.0025, 0.00025, 0.0125 and decays by 0.1 at the 8-th/8-th/4-th and 11-th/11-th/5-th epoch for COCO/DOTA/SeaPerson, respectively. In default settings, the backbone is initialized with the pre-trained weights on ImageNet and other newly added layers are initialized with Xavier. The sampling radius \(R\) is set as 8/7/5 for COCO/DOTA/SeaPerson by default.

In dataset pre-processing, for COCO dataset, the short side of the images is resized to 400, and the ratio of width and height is kept. In dota dataset, images are split into sub-images (1024 × 1024 pixels) with overlap (200 × 200 pixels). And in SeaPerson, images are split into sub-images (640 × 640 pixels) with overlap (100 × 100 pixels). For data augmentation, only random horizontal is utilized in our CPRNet training.
D. Details of Semantic Variance

The $\text{Var}(x')$ and $\text{Var}(y')$ in Eq. 13 are calculate as:

$$\text{Mean}(x') = \frac{1}{M} \sum_{1 \leq j \leq M} x'_j;$$
$$\text{Mean}(y') = \frac{1}{M} \sum_{1 \leq j \leq M} y'_j;$$
$$\text{Var}(x') = \frac{1}{M} (x'_j - \text{Mean}(x'))^2;$$
$$\text{Var}(y') = \frac{1}{M} (y'_j - \text{Mean}(y'))^2;$$

where $(x'_j, y'_j)$ is relative position of $j$-th object, $M$ is the number of objects in dataset. For the objects whose annotated points or refined points are out of the bounding box, they often are regarded as wild points during the learning procedure. The wild points account for a small proportion and will not be learned by the network. However, their $RSV$ will be very large since the $RSV$ is relative to the height and width. Therefore, to better reflect the semantic variance of the points that really affect network learning, only the object whose annotated point or refined point is inside bounding box is used for calculating $RSV$.

E. Visualization of CPR

The visualization of CPR on COCO, DOTA and SeaPerson are shown as Fig. 9, Fig. 10 and Fig. 8, respectively.

Figure 8. Visualization of CPR on SeaPerson. The images are cut from original images for better visualization. Semantic points (red) around the annotated point (green) are weighted averaged to obtain the semantic center (yellow) as final refined point (see Sec. 3.3).
Figure 9. Visualization of CPR on COCO. The images are cut from original images for better visualization. Semantic points (red) around the annotated point (green) are weighted averaged to obtain the semantic center (yellow) as final refined point (see Sec. 3.3).
Figure 10. Visualization of CPR on DOTA. The images are cut from original images for better visualization. Semantic points (red) around the annotated point (green) are weighted averaged to obtain the semantic center (yellow) as final refined point (details in Sec. 3.3).