P2P-Loc: Point to Point Tiny Person Localization
Xuehui Yu, Di Wu, Qixiang Ye, Jianbin Jiao and Zhenjun Han

Abstract—Bounding-box annotation form has been the most frequently used method for visual object localization tasks. However, bounding-box annotation relies on a large amount of precisely annotating bounding boxes, and it is expensive and laborious. It is impossible to be employed in practical scenarios and even redundant for some applications (such as tiny person localization) that the size would not matter. Therefore, we propose a novel point-based framework for the person localization task by annotating each person as a coarse point (CoarsePoint) instead of an accurate bounding box that can be any point within the object extent. Then, the network predicts the person’s location as a 2D coordinate in the image. Although this greatly simplifies the data annotation pipeline, the CoarsePoint annotation inevitably decreases label reliability (label uncertainty) and causes network confusion during training. As a result, we propose a point self-refinement approach that iteratively updates point annotations in a self-paced way. The proposed refinement system alleviates the label uncertainty and progressively improves localization performance. Experimental results show that our approach has achieved comparable object localization performance while saving up to 80% of annotation cost.

Index Terms—Tiny person localization, pedestrian localization, detection, tiny person models, point to point.

I. INTRODUCTION

Object localization is essential in the computer vision community with various systems, including visual surveillance, driving assistance, and mobile robotics [1–10]. Object localization has achieved unprecedented progress with the rise of deep learning and large-scale bounding-box or precision point annotations.

Precise object location usually requires precise annotation, as shown in Fig. 1. To pursue high-precision positioning, a general object detection presents each target as a tight bounding rectangle. For example, the pose estimation models the human body as seventeen key points in fixed positions. In scenarios where targets are far away or of low resolution [11–13], it is time-consuming and even infeasible to implement precise annotation due to the small object size and low signal-to-noise ratio.

Therefore, we creatively propose a new framework that can localize objects with easily acquired supervision signals. In the low-resolution scenarios, the framework is intended to locate the object’s position with needless bounding boxes.

In this paper, we propose a novel computer vision task to achieve object localization with coarsely annotated point supervisions, referred to as CoarsePoint. Because any point in the object area can serve as a CoarsePoint, the required human effort is greatly reduced for data annotation. However, training with inaccurate supervision information will deteriorate the model due to the label uncertainty, such as the significant variance of annotated instance appearance and features uncertainties. For example, different parts of a human body (e.g., head, torso, or foot) could be labeled as positive instances in the training set, and other parts will be set as negative examples, shown in Fig. 2. The heads of different instances may be labeled as positive or negative samples. The label uncertainty brings two negative impacts: On the one hand, the same parts in different instances could be annotated as either positives or negatives, which misleads the localization model toward undesirable tendencies; On the other hand, different object parts may be labeled as positives for objects of the same kind, which will propel the network to predict different parts of the same object, aggregating the risk of false positives.

To tackle the negative impact of CoarsePoint, we propose a simple-yet-effective strategy point self-refinement, aiming at narrowing the performance gap between the coarse point supervision and the precision point supervision.

The idea behind self-refinement is that when coupled with localization model learning (e.g., CNN training), the coarsely annotated points can be iteratively promoted in a self-paced fashion. Self-refinement investigates the statistical characteristics of annotations and semantic features, pursuing a gradual decrease of label uncertainty and better convergence of model training.
The contributions of this study include that:

- We propose a CoarsePoint vision task, and under the relaxed point supervision, we set the first solid baseline for object localization;
- We propose the self-refinement approach to promote coarse points, implement the statistical stability of supervision signals, and improve the localization model in a self-paced fashion;
- CoarsePoint achieves competitive experimental results with precise bounding-box annotation-based methods while saving up to 80% of annotation cost.

II. RELATED WORK

Objects can be modeled in different fashions based on semantic and geometric characteristics, e.g., bounding-box, point, and so on.

To better explain our proposed task, we review related works from different supervision and evaluation fashions, respectively.

A. Object as Bounding-box

Different from the CoarsePoint, the bounding-box-based modeling represents the position and scale information. It can be categorized into fully supervised and weakly supervised methods based on the different supervisions.

**Fully Supervised Methods.** General object detection models object as a bounding box. As instance-level annotation, a bounding box is created to render the center and scale of the object. Under this setting, more and more fully supervised detectors\[14–22\] are expected to represent the object accurately. However, such supervision form would consume huge workforce physical resources, and it takes much time and heavy workload to annotate the object at the instance level.

**Weakly Supervised Methods.** To lower the annotation cost and make full use of massive network data, the weakly supervised object detection (WSOD) \[23–27\] trains a detector when only image-level labels are available, which will cause WSOD to focus only on local areas and lack the ability to distinguish instances due to the lack of instance-level constraints. Different from WSOD, weakly supervised object localization utilizes the activation map from the last convolution layer to generate semantic-aware localization maps for object bounding-box estimation \[28\].

The evaluation of the above tasks is all based on the bounding box. Average precision (AP) and the Correct Localization (CorLoc) \[29\] are used to evaluate detector performance. Unlike the methods mentioned above, CoarsePoint has no downstream task and does not focus on the object geometry. We only focus on the location points and evaluate the performance according to the position of the predicted point, avoiding bounding box annotation.

B. Object as Point

For some vision tasks, it is not necessary to model an object in the form of a bounding box. Researchers began to pay attention to the annotation form of points that serve to represent part of or the entire object.

**Human Keypoint Detection.** Human keypoint detection, also known as human posture estimation, aims to locate the position of joint human points accurately. The COCO dataset \[30\] contains over 200,000 images and 250,000 person instances labeled with 17 key points. During inference, the pose detector predicts the position of every visible keypoint. Similar to COCO, the Human3.6M \[31\] is one of the most extensive 3D human pose benchmarks. With this benchmark, the detector needs to predict the position of each keypoint in the 3D coordinate system. Three-point annotation form datasets used in \[32\] are intended to locate the object accurately without considering other redundant information or reduce annotation burden by using the form of the center point \[33\].

**Object Localization.** A new task \[32\] has been proposed to estimate object locations without annotated bounding boxes. In object localization, a true positive is counted when the distance between the estimated location and a ground truth point is less...
than d, and vice versa. Then the evaluation results of precision and recall can be calculated. In addition, several evaluation metrics are adopted to indicate if the number of location points is incorrect, such as Mean Absolute Error (MAE), Root Mean Squared Error (RMSE), and Mean Absolute Percent Error (MAPE) [34].

Different from the above tasks, CoarsePoint does not require precise point annotations, which significantly reduces the annotation time. Meanwhile, the expectation of Coarse only aims at locating objects without extensive human effort, which does not require an extremely accurate localization effect like the above tasks. Therefore, the object is considered to be successfully located if the point falls into the corresponding bounding box.

### C. Object as Counting

Crowd counting focuses on the number of people in the current scene rather than the position of human targets. In crowd counting, an accurate head annotation is utilized as point supervision. Generated by head annotation, a crowd density map is chosen as the optimization objective of the network. More importantly, On the contrary, CoarsePoint only focuses on the coarse position of the human body. The precision requirement of the annotation is relatively low.

### III. METHODOLOGY

Considering the statistical stability of the semantic center, we use the semantic center of the object to replace the coarse point annotation during the self-refinement process. First, on the training set, an estimator is trained under the coarse point supervision. Then, the estimator is used to predict the location of semantic statistic points (SSPs) on each image of the training set. Finally, to reduce the uncertainty of coarse points, the center of the SSPs’ location of each object is chosen as the refined point annotation. To simplify the description, the following formulation in this paper is only for localizing one category object. In the case of multiple categories, object localization is treated as multiple tasks of one-category object localization.

#### A. Point Estimation

**Formulation of Semantic Parts.** The image set for training is \( I = \{I_1, I_2, ..., I_N\} \), coarse points sets are defined as \( A = \{A_1, A_2, ..., A_N\} \), in which \( N \) is the number of images. \( A_i = \{a_{i1}, a_{i2}, ..., a_{im_i}\} \). \( a_{ij} = (x, y) \) is the 2d coordinate of the annotated coarse point of the \( j \)-th object, and \( M_i \) is the number of objects in the image \( I_i \). The set of all points in \( I \) is \( \omega \) that is defined as \( \Omega = \bigcup_{i=1}^{N} \{a|a is the point in I_i\} = \{\omega_1, \omega_2, ..., \omega_M\} \). Whether \( \omega_h \) is annotated or not is denoted by \( l_h; l_h = 1 \) means \( \omega_h \) is annotated, whereas \( l_h = 0 \) means it is not. Accordingly, the annotation set can be denoted as \( L = \{l_1, l_2, ..., l_{M}\} \).

From a semantic perspective, objects can be divided into several semantic parts (e.g., a human object can be divided into hand, leg, etc.). \( T \) denotes the number of divided semantic parts, and \( S_t \) \((t = 1, 2, ..., T)\) represents the set of points within the \( t \)-th part of all these category objects in \( I \). \( S_0 \) is the set of points that are not in any part of the category object in \( I \). All these point sets constitute \( S = \{S_0, S_1, S_2, ..., S_T\} \), and \( \sum_{0 \leq t \leq T} S_t = \Omega \) is the set of all points in \( I \).

**Objective Function.** The set of annotated points in \( S_t \) is represented as \( S_t^+ \), and the annotated frequency of the semantic part \( S_t \) is defined as \( Q(S_t) \).

\[
S_t^+ = \{\omega_h|l_h = 1, \omega_h \in S_t\},
\]

\[
Q(S_t) = P\{l_h = 1|\omega_h \in S_t\} = \frac{|S_t^+|}{|S_t|},
\]

where \(|S_t^+|, |S_t|\) indicates the number of elements in \( S_t^+, |S_t| \).

The objective function for training the estimator \( E(\omega; \theta) \) under the supervision of \( L \) is defined with Eq (2):

\[
\text{loss}(\theta; \Omega, L) = \frac{1}{|\Omega|} \sum_{1 \leq h \leq |\Omega|} FL(E(\omega_h; \theta), l_h)
\]

\[
= \sum_{S_t \in S} \frac{|S_t|}{|\Omega|} FL(E(\omega_h; \theta), Q(S_t)),
\]

where \( FL \) is the focal loss, and the derivation process details of Eq (2) can be reached as follows with Eq (3) (4) (5):

\[
SL_t^+ = \{(f_h, s_h, l_h)|l_h = 0, (f_h, s_h, l_h) \in S_t\},
\]
Fig. 3: The pipeline of the self-refinement algorithm. An estimator is trained under the coarse points supervision. The estimator is then used to find all SSPs in each image of the training set. Finally, the center of the SSPs of each object is chosen as the refined point annotation. After the first iteration, the refined annotation is the input annotation of the next iteration. A point is initially modeled as uniform distribution. With the iteration of the estimator, the point distribution gradually converges to the center of SSPs.

\[
\text{loss}(\theta; SL) = \frac{1}{|\Omega|} \sum_{(f_h,s_h,l_h) \in SL} CE(E(f_h; \theta), l_h)
\]
\[
= - \frac{1}{|\Omega|} \sum_{(f_h,s_h,l_h) \in SL} \left[ l_h \cdot \log(E(f_h; \theta)) + (1 - l_h) \cdot \log(1 - E(f_h; \theta)) \right]
\]
\[
= - \sum_{S_t \in S} \left( \sum_{(f_h,s_h,l_h) \in SL^+} \log(E(f_h; \theta)) + \sum_{(f_h,s_h,l_h) \in SL^-} \log(1 - E^k(f_h; \theta)) \right)
\]
\[
= - \sum_{S_t \in S} \left[ Q(S_t) \log(E(f_h; \theta)) + (1 - Q(S_t)) \log(1 - E^k(f_h; \theta)) \right]
\]
\[
= \sum_{S_t \in S} \left[ \frac{|SL_t|}{|\Omega|} \right] CE(E(f_h; \theta), Q(S_t)),
\]
\[
\theta^* = \arg\min_{\theta} \text{loss}(\theta; \Omega, L).
\]

**Statistic Semantic Points.** For a point \( \omega_h \in S_t \), the learning objective is to minimize the distance between the estimator \( E(\omega_h; \theta) \) and the probability \( Q(S_t) \) (instead of hard supervision, 1 or 0). To achieve a high score predicted by \( E(\omega; \theta^*) \), there are two constraints: (1) High \( Q(S_t) \); (2) \( S_t \) is semantically distinguishable from points of other categories and background \( (S_0) \). Such as SSPs.

In the \( k \)-th iteration, with images \( I_i \) and the estimator \( E^k(\omega; \theta) \) trained in \( I \) under the supervision of \( A^k \), the estimated SSPs can be obtained as Eq (6):

\[
\hat{A}^k_i = \{ \omega_h | E^k(\omega_h; \theta^*) > \delta, \omega_h \in \Omega \},
\]

in which \( \delta \) is a threshold value ranging from 0 to 1. In the experiment, \( \delta \) is set to 0.2.

**B. Point Refinement**

The estimated semantic points \( \hat{A}^k_i \) may come from different objects in the image \( I_i \). They are assigned to the \( j \)-th object in \( I_i \), noted as \( G_{ij} \) in Eq (7):

\[
G_{ij} = \{ a | a \in \hat{A}^k_i, j = \arg\min_{1 \leq j' \leq M_i} \text{dis}(a, A_{ij'}) \},
\]

in which \( \text{dis} \) is a distance function to measure the Euclidean distance between two points.

As Fig. 4 shows, the top \( K \) points in \( G_{ij} \) ranked by the predicted score during the merging process will be kept. To obtain the final refined point annotations of the \( j \)-th object in
In order to avoid incomplete objects caused by cutting, there is a constant overlap between adjacent sub-images. Then, the NMS strategy is used to fuse the same results of an image. **Visdrone-Person.** As a large-scale dataset captured by UAV, Visdrone contains four tracks: (1) Image object detection, (2) Video object detection, (3) Single object tracking, and (4) Multi-object tracking. The experiments in this paper are conducted on the image object detection track of Visdrone. Considering the application scenarios of localization, we mainly focus on the object with a relatively small size. Therefore, the images containing human beings are used to construct a new human detection dataset named Visdrone-Person, and only the labels of pedestrians and persons are applied in this dataset. Visdrone-Person contains 10,209 images, with 6,471 images used for training, 548 for validation, and 3,190 for testing. The same cutting strategy used by TinyPerson is employed to get the sub-images with the appropriate size.

### B. Point Annotation Initialization

We obtained a point \((x, y)\) by Uniform Distribution or with the Rectified Gaussian Distribution within a bounding box \((x_c, y_c, w, h)\) to generate point annotations.

**Uniform Distribution.** Supposing manual annotating workers do not have any preference when labeling objects, the position of an annotated point follows the uniform distribution. According to the large number theorem, it is reasonable to assume that the annotation labeled by a man is like a Gaussian distribution. However, Gaussian distribution has two problems when modelling the position distribution of labeled points. One is how to determine the variance of a Gaussian distribution. The other is that the range of the labeled points is bounded ((−0.5, 0.5)), while the Gaussian is unbounded. Therefore, the RG distribution \((\text{RG}(x', y'; \mu, \sigma))\) is adopted to handle these problems.

In this paper, the Uniform and RG \((\text{RG}(x', y'; 0, 1/4))\) distributions are chosen to generate point annotations for most experiments. \(\text{RG}(x', y'; -0.26, 1/8), \text{RG}(x', y'; -0.28, 1/6), \text{RG}(x', y'; -0.31, 1/5),\) and \(\text{RG}(x', y'; -0.38, 1/4),\) whose mean are −0.25, are used for ablation study.

### C. RG Distribution

First, as for the probability density function \(G(x, y; \mu, \sigma)\) of Gaussian distribution, \(\mu = 0\). Following \(G(x, y; \mu, \sigma)\) while sampling, the physical meaning of the sample falling outside of the range \([-0.5, 0.5]\) indicates that there is a certain probability that the position annotated by a man will be labeled outside the bounding box of objects that is often very small. According to the two sigma-rule and the three-sigma rule, it is generally believed that the sampling from Gaussian distribution falls on \([\mu - 2\sigma, \mu + 2\sigma]\) or \([\mu - 3\sigma, \mu + 3\sigma]\) is a low probability event, the probabilities of which are 4.56% and 0.26% respectively. Consequently, we assume that the probability \(P(|x| > 0.5) = P(|x| > \sigma \ast r),\) in which \(r = 2\) or 3 depending on the labeling accuracy of the worker men. The variance has been determined as \(\sigma = \frac{1}{\sqrt{r}} = 1/4, 1/6.\)
Second, think back to the process of human labeling. When humans label a point that falls outside an object, it is easy to find out. In this case, the man who labels the wrong annotated point should modify and relabel it until the point falls on the object. To lessen the problem, we assume that the distribution of the labeled point’s position still conforms to the same Gaussian distribution. The probability of the annotated point fallings outside the object (bounding box is adopted to approximate object) of the first labeling $t$ can be obtained as $t = 1 - \int_{-0.5}^{0.5} \int_{-0.5}^{0.5} G(x, y; \mu, \sigma) dxdy$. We name the probability density function of the final annotated points’ distribution as RG distribution ($RG(x, y; \mu, \sigma)$). For $|x| \leq 0.5$ and $|y| \leq 0.5$, the RG distribution can be deduced as follow:

$$RG(x, y; \mu, \sigma) = G(x, y; \mu, \sigma) + t * G(x, y; \mu, \sigma) + \ldots + t^n * G(x, y; \mu, \sigma) + \ldots$$

$$= G(x, y; \mu, \sigma) * \lim_{n \to \infty} \frac{(1 - t^n)}{1 - t}$$

$$= G(x, y; \mu, \sigma) / (1 - t)$$

$$= \int_{-0.5}^{0.5} \int_{-0.5}^{0.5} G(x, y; \mu, \sigma) dxdy.$$

Otherwise, for $|x| > 0.5$ or $|y| > 0.5$, $RG(x, y; \mu, \sigma) = 0$. Therefore, we adopt the RG distribution bounded on $[-0.5, 0.5]$ when generating point annotation from each object. (For $RG(x, y; \mu, \sigma)$, $\mu$ and $\sigma^2$ are not means and variance anymore.)

### D. Experimental Details

**Implementation.** As a baseline model, RepPoints [36] is suitable for coarse point supervised object localization. This comprehensive framework consists of one estimator and one locator. To ensure that the proportions of positive and negative samples in training are appropriate, we regard some points around the labeled points as positive samples during label assigning in training. The network outputs the classification results and the position offset of each point. In practice, simply treat the input point as a pseudo box with a fixed length and width centered on the input point. Then, for self-refinement, the pseudo box is regarded as the supervision to train an estimator, after which the refined pseudo bounding box is fed to the locator. Finally, predicted bounding boxes produced by the locator are converted to the localization points in the evaluation.

**Setting.** ResNet-50 is used as a backbone network. The training epoch is set to 12. The initial learning rate is 0.01, and it decreases with the factor 0.1 in the 8th and 11th epoch. We choose Faster RCNN with FPN and Sparse RCNN[37] for comparison.

**Metric.** We adopt the Average Precision (AP) as the metric. According to the bounding-box size, the scale interval is divided into tiny[2, 20], small[20, 32], normal[32, +∞], and all[2, +∞]. In point evaluation, the point-box distance, instead of IOU, is used as the matching criterion. Specifically, the threshold of point-box distance $\tau$ is set as 1. It means that as long as the predicted point falls within an unmatched ground truth, the match between the predicted point and the ground truth is successful.

### E. Experimental Analysis

#### Decline of Model Confusion.
As mentioned in Sec. III, the uncertainty of annotation semantics limits the localization performance during the self-refinement. To quantitatively analyze this uncertainty, we collect the refined points in each iteration and calculate their relative positions $(x', y')$ within the object’s bounding box. The distribution is defined as Eq (10):

$$P(x', y'; A^k) = \frac{\text{number of instances annotated on } (x', y')}{\text{number of instances}}$$

in which $|x'| \leq 0.5$ and $|y'| \leq 0.5$.

As shown in Fig. 5, after the self-refinement, the variance of the distribution decreases, which results in less annotation uncertainty and alleviates the model confusion. In the bottom line of Fig. 5, it is noticeable that with a small distribution variance and the initial annotation concentrated on the weaker semantic region, such as the top-left parts of persons, the proposed approach can refine the annotation to the neighboring region with strong semantic information, such as the head of a person.

#### Reduction of False Positives.
Fig. 7 shows that the self-refinement significantly reduces the number of false positives, leading to the improvement of the network’s discriminative ability.

**Analysis of Upper Bound.** As shown in Tab. VII, we perform experiments based on VisDrone-Person with RepPoints as both the estimator and the locator. The real box means that precision bounding box annotations are used as the supervision.
Fig. 5: Heatmaps of the point distribution in an object’s bounding box. These three rows of heat maps indicate that the initial coarse point annotation obeys Uniform, RG(0, 1/4), and RG(-0.28, 1/6) distribution. The relevant performance data of these distributions that the initial coarse points obey can be seen in Tab. 5.

| iteration | $AP_{\text{all}}$ | $AP_{\text{tiny}}$ | $AP_{\text{small}}$ | $AP_{normal}$ |
|-----------|-----------------|-----------------|-----------------|-------------|
| 0         | 60.2            | 61.9            | 46.9            | 19.1        |
| 1         | 68.2            | 61.3            | 72.3            | 55.2        |
| 2         | 68.9            | 60.7            | 71.9            | 62.9        |

TABLE III: The performance in VisDrone-Person with coarse point annotation following Uniform distribution, and RepPoints as the estimator and the locator.

| Est  | Loc  | $AP_{\text{all}}$ | $AP_{\text{tiny}}$ | $AP_{\text{small}}$ | $AP_{normal}$ |
|------|------|-----------------|-----------------|-----------------|-------------|
|      | RP   | 59.7            | 62.5            | 44.5            | 14.0        |
| RP   | RP   | 68.5            | 60.6            | 72.4            | 56.9        |
|      | SR   | 54.3            | 48.2            | 33.8            | 19.4        |
| RP   | SR   | 67.5            | 57.1            | 59.7            | 43.6        |

TABLE IV: Comparisons of APs with different locators on VisDrone-Person w self-refinement vs. w/o self-refinement. The estimator is simplified as Est, the locator is simplified as Loc, RP represents RepPoints, SR represents Sparse RCNN, and - represents no self-refinement policy.

| Est  | Loc  | $AP_{\text{all}}$ | $AP_{\text{tiny}}$ | $AP_{\text{small}}$ | $AP_{normal}$ |
|------|------|-----------------|-----------------|-----------------|-------------|
| FR   |      | 45.8            | 49.3            | 15.3            | 10.2        |
| RP   | FR   | 63.4            | 63.2            | 41.5            | 19.9        |
| FR   | FR   | 66.1            | 66.3            | 47.6            | 18.9        |

TABLE V: Comparisons of APs with different estimators on TinyPerson. FR represents Faster RCNN.
F. Ablation Study

Locators. In Tab. IV, self-refinement improves $AP_{1,0}^{all}$ and $AP_{1,0}^{normal}$ by 8.82 and 42.9 points respectively under RP+RP conditions. Meanwhile, $AP_{1,0}^{all}$ and $AP_{1,0}^{normal}$ are increased by 13.2 and 24.2 points respectively over RP+SR.

Estimators. It is shown in Tab. V that even with different estimators, self-refinement improves $AP_{1,0}^{all}$ by 20.27 points and $AP_{1,0}^{small}$ by 32.28 points with FR+FR. Our self-refinement algorithm is compatible with different frameworks. When different detectors are used as estimators or locators, the performance under various scales has almost been improved.

Dataset. As shown in Tab. IV and Tab. V, the performance of the self-refinement showed 8.2 and 20.3 points increase on VisDrone-Person and TinyPerson, respectively.

Number of Iteration. The results in Tab. VI reveal that the performance significantly improves as the number of iterations increases. Then, $AP_{1,0}^{all}$ gains 30.6, 16.87 and 7.33 points in the initial pseudo box size setting with 8*8, 16*16 and 32*32 pixels, respectively. Alternatively stated, the greater the number of iterations is, the slower the performance grows.

Pseudo Box Size. Tab. VI also depicts the effect of different initial pseudo box sizes. Increasing the pseudo box size from 8*8 pixels to 32*32 pixels shows that a larger pseudo box size achieves a better performance. However, self-refinement with more iterations drives a performance boost.

Annotation Initialization. In Tab. I, the results show that the initial distribution of generated points affects performance. Manual annotation is more consistent with the RG distribution. The improvement brought by self-refinement is greater when encountering coarser annotation. In addition, the performance becomes relatively stable and insensitive to the initial annotation distribution through self-refinement.

G. Annotation Efficiency

To quantitatively compare annotation efficiencies mentioned above, we randomly select some images in different scenarios from TinyPerson and VisDrone-Person with 1021 objects, and nine persons are chosen for conducting a manual annotation.
Fig. 7: The visualization of the estimator $E$’s results during different iterations on TinyPerson when the Recall was set as 0.5. We obtain the first column results by training $E$ without the self-refinement strategy. The results of the second column are obtained by $E$ after one iteration. Also, the results of the last column are obtained by $E$ after two iterations. Red points are the detected results, and blue points are the center of the ground-truth bounding boxes.

| Iteration | $A^{{\text{all}}}_{1.0}$ | $A^{{\text{tiny}}}_{1.0}$ | $A^{{\text{small}}}_{1.0}$ | $A^{{\text{normal}}}_{1.0}$ |
|-----------|----------------|----------------|----------------|----------------|
| 0         | 21.4           | 16.1           | 58.92          |                 |
| 1         | 52.0           | 62.69         | 66.25          |                 |
| 2         | 60.0           | 66.09         | 66.08          |                 |
| 3         | 62.8           | -             | -              | -              |

TABLE VI: Results on TinyPerson after different iterations and in the different initial pseudo box sizes with Faster RCNN as the estimator and Faster RCNN as the locator.

| Annotation Type | Average Time | $A^{{\text{all}}}_{1.0}$ |
|-----------------|--------------|-------------------------|
| bounding box    | 9.4s         | 76.2                    |
| precision point | 5.3s         | 71.2                    |
| coarse point    | 1.8s         | 70.8                    |

TABLE VII: Upper bound analysis. This experiment shows the performance of different kinds of semantic points generated from the bounding box on VisDrone-Person.

| Initial Point   | $A^{{\text{all}}}_{1.0}$ | $A^{{\text{tiny}}}_{1.0}$ | $A^{{\text{small}}}_{1.0}$ | $A^{{\text{normal}}}_{1.0}$ |
|-----------------|----------------|----------------|----------------|----------------|
| real box        | 76.2           | 65.0           | 80.7            | 82.6            |
| box center      | 74.1           | 63.5           | 79.1            | 76.8            |
| box head        | 71.2           | 60.2           | 75.4            | 72.5            |
| box foot        | 72.7           | 61.0           | 77.7            | 74.0            |
| box corner      | 68.6           | 63.6           | 68.2            | 38.2            |

TABLE VIII: Annotation efficiency. We counted the average time and evaluated the final performance of the same group of people using different annotation types to label images from VisDrone-Person. The performance of coarse points is obtained with reference to the RG(0,1/4) distribution. It only lost 5, 0.4 points of $A^{{\text{all}}}_{1.0}$ while saving the annotation cost up to 80% and 66% in comparison with bounding-box and precision point correspondingly.

In order to avoid the influence of the annotation order, the testers are randomly divided into three groups. Any group is selected to annotate these images following the order of coarse point, precision point, bounding box. Another group follows the order of precision point, coarse point, bounding box. The third group follows the order of bounding box, precision point, coarse point. Finally, we calculate the single object’s average annotation time of different annotations, shown in Tab. VIII.

The annotation of tight bounding boxes is time-consuming, especially for finding small objects in large-scale images. The annotation of the coarse point is quite efficient, and it just needs to click on the object. Under such annotation, our proposed self-refinement algorithm can still achieve excellent localization performance while saving much annotation time up to 80%.
V. CONCLUSION

Object localization is important in the computer vision community. In this paper, we propose CoarsePoint, a new vision task for object localization with coarse point supervision, setting the first solid baseline for the community. Furthermore, we propose the self-refinement approach to promote coarse points iteratively, implementing the statistical stability of supervision signals and improving the localization model in a self-paced fashion. CoarsePoint achieves comparable results with precise bounding-box annotation-based methods and saves up to 80% annotation cost.

VI. ACKNOWLEDGMENT

This work was supported by the National Natural Science Foundation of China (NSFC) under Grant No. 61836012 and 61771447, and the Strategic Priority Research Program of the Chinese Academy of Sciences under Grant No. XDA27000000.

REFERENCES

[1] M. Enzweiler and D. M. Gavrila, “Monocular pedestrian detection: Survey and experiments,” TPAMI, 2008.
[2] P. Dollar, C. Wojek, B. Schiele, and P. Perona, “Pedestrian detection: An evaluation of the state of the art,” TPAMI, 2011.
[3] A. Geiger, P. Lenz, and R. Urtasun, “Are we ready for autonomous driving? the kitti vision benchmark suite,” in CVPR, 2012.
[4] S. Zhang, R. Benenson, and B. Schiele, “Citypersons: A diverse dataset for pedestrian detection,” in CVPR, 2017.
[5] J. Mao, T. Xiao, Y. Jiang, and Z. Cao, “What can help pedestrian detection?” in CVPR, 2017.
[6] V. Havaryarima, Z. Xiao, A. Sibomana, D. Wu, and J. Bai, “A fusion framework based on sparse gaussian-wigner prediction for vehicle localization using GDOP of GPS satellites,” IEEE Trans. Intell. Transp. Syst., vol. 21, no. 2, pp. 680–689, 2020.
[7] H. Yin, Y. Wang, X. Ding, L. Tang, S. Huang, and R. Xiong, “3d lidar-based global localization using siamese neural network,” IEEE Trans. Intell. Transp. Syst., vol. 21, no. 4, pp. 1380–1392, 2020.
[8] S. Choi and J. Kim, “Leveraging localization accuracy with off-centered GPS,” IEEE Trans. Intell. Transp. Syst., vol. 21, no. 6, pp. 2277–2286, 2020.
[9] A. Thangarajah and Q. M. J. Wu, “sendec: An improved image to image CNN for foreground localization,” IEEE Trans. Intell. Transp. Syst., vol. 21, no. 10, pp. 4435–4443, 2020.
[10] S. Zhang, R. Benenson, M. Omran, J. Hosang, and B. Schiele, “Towards reaching human performance in pedestrian detection,” TPAMI, 2017.
[11] X. Yu, Y. Gong, N. Jiang, Q. Ye, and Z. Han, “Scale match for tiny person detection,” in WACV, 2020.
[12] G. Xia, X. Bai, J. Ding, Z. Zhu, S. J. Belongie, J. Luo, M. Datcu, M. Pelillo, and L. Zhang, “DOTA: A large-scale dataset for object detection in aerial images,” in CVPR, 2018.
[13] B. Han, Y. Wang, Z. Yang, and X. Gao, “Small-scale pedestrian detection based on deep neural network,” IEEE Trans. Intell. Transp. Syst., vol. 21, no. 7, pp. 3046–3055, 2020.
[14] S. Ren, K. He, R. B. Girshick, and J. Sun, “Faster R-CNN: towards real-time object detection with region proposal networks,” in NeuralIPS, 2015, pp. 91–99.
[15] T. Lin, P. Dollar, R. B. Girshick, K. He, B. Hariharan, and S. J. Belongie, “Feature pyramid networks for object detection,” in IEEE CVPR, 2017, pp. 936–944.
[16] X. Zhou, D. Wang, and P. Krähenbühl, “Objects as points,” arXiv preprint arXiv:1904.07850, 2019.
[17] T. Ye, X. Zhang, Y. Zhang, and J. Liu, “Railway traffic object detection using differential feature fusion convolution neural network,” IEEE Trans. Intell. Transp. Syst., vol. 22, no. 3, pp. 1375–1387, 2021.
[18] M. Hassaballah, M. A. Kenk, K. Muhammad, and S. Minneci, “Vehicle detection and tracking in adverse weather using a deep learning framework,” IEEE Trans. Intell. Transp. Syst., vol. 22, no. 7, pp. 4230–4242, 2021.
Qixiang Ye received the B.S. and M.S. degrees from Harbin Institute of Technology, China, in 1999 and 2001, respectively, and the Ph.D. degree from the Institute of Computing Technology, Chinese Academy of Sciences in 2006. He has been a professor with the University of Chinese Academy of Sciences since 2009, and was a visiting assistant professor with the Institute of Advanced Computer Studies (UMIACS), University of Maryland, College Park until 2013. His research interests include image processing, visual object detection and machine learning. He has published more than 100 papers in refereed conferences and journals including IEEE CVPR, ICCV, ECCV and PAMI. He is on the editorial boards of IEEE Transactions on Intelligent Transportation System and IEEE Transactions on Circuit and System on Video Technology.

Jianbin Jiao received the B.S., M.S., and Ph.D. degrees in mechanical and electronic engineering from Harbin Institute of Technology (HIT), Harbin, China, in 1989, 1992, and 1995, respectively. From 1997 to 2005, he was an Associate Professor with HIT. Since 2006, he has been a Professor with the School of Electronic, Electrical, and Communication Engineering, University of the Chinese Academy of Sciences, Beijing, China. His current research interests include image processing, pattern recognition, and intelligent surveillance.

Zhenjun Han received the B.S. degree in software engineering from Tianjin University, Tianjin, China, in 2006 and the M.S. and Ph.D. degrees from University of Chinese Academy of Sciences, Beijing, China, in 2009 and 2012, respectively. Since 2013, he has been an Associate Professor with the School of Electronic, Electrical, and Communication Engineering, University of Chinese Academy of Sciences. His research interests include object tracking and detection.