OTA: Optimal Transport Assignment for Object Detection

Zheng Ge$^{1,2}$, Songtao Liu$^2$, Zeming Li$^2$, Osamu Yoshie$^1$, Jian Sun$^2$

$^1$Waseda University, $^2$Megvii Technology

jokerzz@fuji.waseda.jp; liusongtao@megvii.com; lizeming@megvii.com; yoshie@waseda.jp; sunjian@megvii.com

Abstract

Recent advances in label assignment in object detection mainly seek to independently define positive/negative training samples for each ground-truth (gt) object. In this paper, we innovatively revisit the label assignment from a global perspective and propose to formulate the assigning procedure as an Optimal Transport (OT) problem – a well-studied topic in Optimization Theory. Concretely, we define the unit transportation cost between each demander (anchor) and supplier (gt) pair as the weighted summation of their classification and regression losses. After formulation, finding the best assignment solution is converted to solve the optimal transport plan at minimal transportation costs, which can be solved via Sinkhorn-Knopp Iteration. On COCO, a single FCOS-ResNet-50 detector equipped with Optimal Transport Assignment (OTA) can reach 40.7% mAP under $1 \times$ scheduler, outperforming all other existing assigning methods. Extensive experiments conducted on COCO and CrowdHuman further validate the effectiveness of our proposed OTA, especially its superiority in crowd scenarios. The code is available at https://github.com/Megvii-BaseDetection/OTA.

1. Introduction

Current CNN-based object detectors [27, 30, 21, 47, 33, 29, 36] perform a dense prediction manner by predicting the classification (cls) labels and regression (reg) offsets for a set of pre-defined anchors$^1$. To train the detector, defining cls and reg targets for each anchor is a necessary procedure, which is called label assignment in object detection.

Classical label assigning strategies commonly adopt predefined rules to match the ground-truth (gt) object or background for each anchor. For example, RetinaNet [21] adopts Intersection-over-Union (IoU) as its thresholding criterion for pos/neg anchors division. Anchor-free detectors like FCOS [38] treat the anchors within the center/bbox region of any gt object as the corresponding positives. Such static strategies ignore a fact that for objects with different sizes, shapes or occlusion condition, the appropriate positive/negative (pos/neg) division boundaries may vary.

Motivated by this, many dynamic assignment strategies have been proposed. ATSS [47] proposes to set the division boundary for each gt based on statistical characteristics. Other recent advances [48, 19, 51, 16] suggest that the predicted confidence scores of each anchor could be a proper indicator to design dynamic assigning strategies, i.e., high confidence anchors can be easily learned by the networks and thus be assigned to the related gt, while anchors with uncertain predictions should be considered as negatives. Those strategies enable the detector to dynamically choose positive anchors for each individual gt object and achieve state-of-the-art performance.

However, independently assigning pos/neg samples for each gt without context could be sub-optimal, just like the lack of context may lead to improper prediction. When dealing with ambiguous anchors (i.e., anchors that are qualified as positive samples for multiple gts simultaneously as seen in Fig. 1.), existing assignment strategies are heavily based on hand-crafted rules (e.g., Min Area [38], Max IoU [16, 21, 47]). We argue that assigning ambiguous an-
anchors to any \(gt\) (or \textit{background}) may introduce harmful gradients \textit{w.r.t} other \(gt\). Hence the assignment for ambiguous anchors is non-trivial and requires further information beyond the local view. Thus a better assigning strategy should get rid of the convention of pursuing optimal assignment for each \(gt\) independently and turn to the ideology of global optimum, in other words, finding the global high confidence assignment for all \(gt\)s in an image.

DeTR [3] is the first work that attempts to consider \textit{label assignment} from global view. It replaces the detection head with transformer layers [39] and considers one-to-one assignment using the Hungarian algorithm that matches only one query for each \(gt\) with global minimum loss. However, for the CNN based detectors, as the networks often produce correlated scores to the neighboring regions around the object, each \(gt\) is assigned to many anchors (\textit{i.e.}, one-to-many), which also benefits to training efficiency. In this one-to-many manner, it remains intact to assign labels with a global view.

To achieve the global optimal assigning result under the one-to-many situation, we propose to formulate \textit{label assignment} as an Optimal Transport (OT) problem – a special form of Linear Programming (LP) in Optimization Theory. Specifically, we define each \(gt\) as a supplier who supplies a certain number of labels, and define each anchor as a demander who needs one unit label. If an anchor receives sufficient amount of positive label from a certain \(gt\), this anchor becomes one positive anchor for that \(gt\). In this context, the number of positive labels each \(gt\) supplies can be interpreted as “how many positive anchors that \(gt\) needs for better convergence during the training process”. The unit transportation cost between each anchor-\(gt\) pair is defined as the weighted summation of their pair-wise \textit{cls} and \textit{reg} losses. Furthermore, as being negative should also be considered for each anchor, we introduce another supplier - \textit{background} who supplies negative labels to make up the rest of labels in need. The cost between \textit{background} and a certain anchor is defined as their pair-wise classification loss only. After formulation, finding the best assignment solution is converted to solve the optimal transport plan, which can be quickly and efficiently solved by the off-the-shelf Sinkhorn-Knopp Iteration [5]. We name such an assigning strategy as Optimal Transport Assignment (OTA).

Comprehensive experiments are carried out on MS COCO [22] benchmark, and significant improvements from OTA demonstrate its advantage. OTA also achieves the SOTA performance among one-stage detectors on a crowded pedestrian detection dataset named CrowdHuman [35], showing OTA’s generalization ability on different detection benchmarks.

2. Related Work

2.1. Fixed Label Assignment

Determining which \(gt\) (or \textit{background}) should each anchor been assigned to is a necessary procedure before training object detectors. Anchor-based detectors usually adopt IoU at a certain threshold as the assigning criterion. For example, RPN in Faster R-CNN [33] uses 0.7 and 0.3 as the positive and negative thresholds, respectively. When training the R-CNN module, the IoU threshold for \textit{posneg} division is changed to 0.5. IoU based \textit{label assignment} is proved effective and soon been adopted by many Faster R-CNN’s variants like [2, 12, 20, 42, 26, 49, 37], as well as many one-stage detectors like [31, 32, 25, 27, 23, 21].

Recently, anchor-free detectors have drawn much attention because of their concision and high computational efficiency. Without anchor box, FCOS [38], Foveabox [17] and their precursors [30, 14, 46] directly assign anchor points around the center of objects as positive samples, showing promising detection performance. Another stream of anchor-free detectors [18, 8, 50, 45, 4] view each object as a single or a set of key-points. They share distinct characteristics from other detectors, hence will not be further discussed in our paper.

Although detectors mentioned above are different in many aspects, as for \textit{label assignment}, they all adopt a single fixed assigning criterion (\textit{e.g.}, a fixed region of the center area or IoU threshold) for objects of various sizes, shapes, and categories, etc, which may lead to sub-optimal assigning results.

2.2. Dynamic Label Assignment

Many recent works try to make the label assigning procedure more adaptive, aiming to further improve the detection performance. Instead of using pre-defined anchors, GuidedAnchoring [40] generates anchors based on an anchor-free mechanism to better fit the distribution of various objects. MetaAnchor [44] proposes an anchor generation function to learn dynamic anchors from the arbitrary customized prior boxes. NoisyAnchors [19] proposes soft-label and anchor re-weighting mechanisms based on classification and localization losses. FreeAnchor [48] constructs top-k anchor candidates for each \(gt\) based on IoU and then proposes a detection-customized likelihood to perform \textit{posneg} division within each candidate set. ATSS [47] proposes an adaptive sample selection strategy that adopts mean+std of IoU values from a set of closest anchors for each \(gt\) as a \textit{posneg} threshold. PAA [16] assumes that the distribution of joint loss for positive and negative samples follows the Gaussian distribution. Hence it uses GMM to fit the distribution of positive and negative samples, and then use the center of positive sample distribution as the \textit{posneg} division boundary. AutoAssign [51] tackles label
3. Method

In this section, we first revisit the definition of the Optimal Transport problem and then demonstrate how we formulate the label assignment in object detection into an OT problem. We also introduce two advanced designs which we suggest adopting to make the best use of OTA.

3.1. Optimal Transport

The Optimal Transport (OT) describes the following problem: supposing there are \( m \) suppliers and \( n \) demanders in a certain area. The \( i \)-th supplier holds \( s_i \) units of goods while the \( j \)-th demander needs \( d_j \) units of goods. Transporting cost for each unit of good from supplier \( i \) to demander \( j \) is denoted by \( c_{ij} \). The goal of OT problem is to find a transportation plan \( \pi^* = \{\pi_{i,j}\}_{i=1,2,...m, j=1,2,...n} \), according to which all goods from suppliers can be transported to demanders at a minimal transportation cost:

\[
\min_{\pi} \sum_{i=1}^{m} \sum_{j=1}^{n} c_{ij}\pi_{ij}, \\
\text{s.t.} \quad \sum_{i=1}^{m} \pi_{ij} = d_j, \quad \sum_{j=1}^{n} \pi_{ij} = s_i, \\
\sum_{i=1}^{m} s_i = \sum_{j=1}^{n} d_j, \\
\pi_{ij} \geq 0, \quad i = 1, 2, ...m, j = 1, 2,...n.
\] (1)

This is a linear program which can be solved in polynomial time. In our case, however, the resulting linear program is large, involving the square of feature dimensions with anchors in all scales. We thus address this issue by a fast iterative solution, named Sinkhorn-Knopp [5] (described in Appendix.)

3.2. OT for Label Assignment

In the context of object detection, supposing there are \( m \) gt targets and \( n \) anchors (across all FPN [20] levels) for an input image \( I \), we view each gt as a supplier who holds \( k \) units of positive labels (i.e., \( s_i = k, i = 1, 2, ..., m \)), and each anchor as a demander who needs one unit of label (i.e., \( d_j = 1, j = 1, 2, ..., n \)). The cost \( c_{ij}^{l/b} \) for transporting one
unit of positive label from $g_t_i$ to anchor $a_j$ is defined as the weighted summation of their $cls$ and $reg$ losses:

$$
e^{fg}_{ij} = L_{cls}(P_j^{cls}(\theta), G_i^{cls}) + \alpha L_{reg}(P_j^{box}(\theta), G_i^{box}),$$

(2)

where $\theta$ stands for model’s parameters. $P_j^{cls}$ and $P_j^{box}$ denote predicted $cls$ score and bounding box for $a_j$. $G_i^{cls}$ and $G_i^{box}$ denote ground truth class and bounding box for $g_t_i$. $L_{cls}$ and $L_{reg}$ stand for cross entropy loss and IoU Loss [46]. One can also replace these two losses with Focal Loss [21] and GIoU [34]/SmoothL1 Loss [11]. $\alpha$ is the balanced coefficient.

Besides positive assigning, a large set of anchors are treated as negative samples during training. As the optimal transportation involves all anchors, we introduce another supplier – background, who only provides negative labels. In a standard OT problem, the total supply must be equal to the total demand. We thus set the number of negative labels that background can supply as $n - m \times k$. The cost for transporting one unit of negative label from background to $a_j$ is defined as:

$$
e^{bg}_{ij} = L_{cls}(P_j^{cls}(\theta), \emptyset),$$

(3)

where $\emptyset$ means the background class. Concatenating this $e^{bg} \in \mathbb{R}^{1 \times n}$ to the last row of $e^{fg} \in \mathbb{R}^{m \times n}$, we can get the complete form of the cost matrix $c \in \mathbb{R}^{(m+1) \times n}$. The supplying vector $s$ should be correspondingly updated as:

$$s_i = \begin{cases} k, & \text{if } i \leq m \\ n - m \times k, & \text{if } i = m + 1. \end{cases}$$

As we already have the cost matrix $c$, supplying vector $s \in \mathbb{R}^{m+1}$ and demanding vector $d \in \mathbb{R}^n$, the optimal transportation plan $\pi^* \in \mathbb{R}^{(m+1) \times n}$ can be obtained by solving this OT problem via the off-the-shelf Sinkhorn-Knopp Iteration [5]. After getting $\pi^*$, one can decode the corresponding label assigning solution by assigning each anchor to the supplier who transports the largest amount of labels to them. The subsequent processes (e.g., calculating losses based on assigning result, back-propagation) are exactly the same as in FCOS [38] and ATSS [47]. Noted that the optimization process of OT problem only contains some matrix multiplications which can be accelerated by GPU devices, hence OT only increases the total training time by less than 20% and is totally cost-free in testing phase.

3.3. Advanced Designs

Center Prior. Previous works [47, 16, 48] only select positive anchors from the center region of objects with limited areas, called Center Prior. This is because they suffer from either a large number of ambiguous anchors or poor statistics in the subsequent process. Instead of relying on statistical characteristics, our OTA is based on global optimization methodology and thus is naturally resistant to these two issues. Theoretically, OTA can assign any anchor within the region of $gts$’ boxes as a positive sample. However, for general detection datasets like COCO, we find the Center Prior still benefit the training of OTA. Forcing detectors focus on potential positive areas (i.e., center areas) can help stabilize the training process, especially in the early stage of training, which will lead to a better final performance. Hence, we impose a Center Prior to the cost matrix. For each $g_t$, we select $r^2$ closest anchors from each FPN level according to the center distance between anchors and $gts$. As for anchors not in the $r^2$ closest list, their corresponding entries in the cost matrix $c$ will be subject to an additional constant cost to reduce the possibility they are assigned as positive samples during the training stage. In Sec. 4, we will demonstrate that although OTA adopts a certain degree of Center Prior like other works [38, 47, 48] do, OTA consistently outperforms counterparts by a large margin when $r$ is set to a large value (i.e., large number of

---

Algorithm 1 Optimal Transport Assignment (OTA)

**Input:**

$I$ is an input image

$A$ is a set of anchors

$G$ is the $g_t$ annotations for objects in image $I$

$\gamma$ is the regularization intensity in Sinkhorn-Knopp Iter.

$T$ is the number of iterations in Sinkhorn-Knopp Iter.

$\alpha$ is the balanced coefficient in Eq. 2

**Output:**

$\pi^*$ is the optimal assigning plan

1: $m \leftarrow |G|$, $n \leftarrow |A|
2: P^{cls}, P^{box} \leftarrow Forward(I,A)
3: s_i(i = 1, 2, ..., m) \leftarrow Dynamic k Estimation
4: $s_{m+1} = n - \sum_{i=1}^{m} s_i$
5: $d_j(j = 1, 2, ..., n) \leftarrow OnesInit$
6: pairwise $cls$ cost: $c^{ij}_{cls} = FocalLoss(P_j^{cls}, G_i^{cls})$
7: pairwise $reg$ cost: $c^{ij}_{reg} = IoULoss(P_j^{box}, G_i^{box})$
8: pairwise Center Prior cost: $c^{ij}_{c} = (A_j, G_i^{box})$
9: bg $cls$ cost: $c^{bg}_{cls} = FocalLoss(P_j^{cls}, \emptyset)$
10: $fg$ cost: $c^{fg} = c^{cls} + \alpha c^{reg} + c_{cp}$
11: compute final cost matrix $c$ via concatenating $c^{cls}$ to the last row of $c^{fg}$
12: $v^0, v^0 \leftarrow OnesInit$
13: for $t=0$ to $T$ do:
14: $u^{t+1}, v^{t+1} \leftarrow SinkhornIter(c, u^t, v^t, s, d)$
15: compute optimal assigning plan $\pi^*$ according to Eq. ??
16: return $\pi^*$

---

2 For anchor-based methods, the distances are measured between the geometric center of anchors and $gts$
potential positive anchors as well as more ambiguous anchors).

**Dynamic $k$ Estimation.** Intuitively, the appropriate number of positive anchors for each $gt$ (i.e., $s_i$ in Sec. 3.1) should be different and based on many factors like objects’ sizes, scales, and occlusion conditions, etc. As it is hard to directly model a mapping function from these factors to the positive anchor’s number, we propose a simple but effective method to roughly estimate the appropriate number of positive anchors for each $gt$ based on the IoU values between predicted bounding boxes and $gts$. Specifically, for each $gt$, we select the top $q$ predictions according to IoU values. These IoU values are summed up to represent this $gt$’s estimated number of positive anchors. We name this method as Dynamic $k$ Estimation. Such an estimation method is based on the following intuition: The appropriate number of positive anchors for a certain $gt$ should be positively correlated with the number of anchors that well-regress this $gt$. In Sec. 4, we present a detailed comparison between the fixed $k$ and Dynamic $k$ Estimation strategies.

A toy visualization of OTA is shown in Fig. 2. We also describe the OTA’s completed procedure including Center Prior and Dynamic $k$ Estimation in Algorithm 1.

### 4. Experiments

In this section, we conduct extensive experiments on MS COCO 2017 [22] which contains about 118k, 5k and 20k images for train, val, and test-dev sets, respectively. For ablation studies, we train detectors on train set and report the performance on val set. Comparisons with other methods are conducted on test-dev set. We also compare OTA with other methods on CrowdHuman [35] validation set to demonstrate the superiority of OTA in crowd scenarios.

#### 4.1. Implementation Details

If not specified, we use ResNet-50 [13] pre-trained on ImageNet [6] with FPN [20] as our default backbone. Most of experiments are trained with 90k iterations which is denoted as “1x”. The initial learning rate is 0.01 and is decayed by a factor of 10 after 60k and 80k iterations. Mini-batch size is set to 16. Following the common practice, the model is trained with SGD [1] on 8 GPUs.

OTA can be adopted in both anchor-based and anchor-free detectors, the following experiments are mainly conducted on FCOS [38] because of its simplicity. We adopt Focal Loss and IoU Loss as $L_{cls}$ and $L_{reg}$ that make up the cost matrix. $\alpha$ in Eq. 2 is set to 1.5. For back-propagation, the regression loss is replaced by GIoU Loss and is re-weighted by a factor of 2. IoU Branch is first introduced in YOLOv1 [30] and proved effective in modern one-stage object detectors by PAA [16]. We also adopt IoU Branch as a default component in our experiments. The top $q$ in Sec. 3.3 is directly set to 20, as we find this set of parameter values can consistently yield stable results in various situations.

### 4.2. Ablation Studies and Analysis

#### Effects of Individual Components.

We verify the effectiveness of each component in our proposed methods. For fair comparisons, all detectors’ regression losses are multiplied by 2, which is known as a useful trick to boost the AP at high IoU thresholds [28]. As seen in Table 1, when no auxiliary branch is adopted, OTA outperforms FCOS by 0.9% AP (39.2% v.s.38.3%). This gap almost remains the same after adding IoU branch to both of them (39.5% v.s. 40.3% and 38.8% v.s. 39.6% with or without center prior, respectively). Finally, dynamic $k$ pushes AP to a new state-of-the-art 40.7%. In the whole paper, we emphasize that OTA can be applied to both anchor-based and anchor-free detectors. Hence we also adopt OTA on RetinaNet [21] with only one square anchor per-location across feature maps. As shown in Table 1, the AP values of OTA-FCOS and OTA-RetinaNet are exactly the same, demonstrating OTA’s applicability on both anchor-based and anchor-free detectors.

#### Effects of $r$.

The values of radius $r$ for Center Prior serve to control the number of candidate anchors for each $gt$. If adopting a small $r$, only anchors near objects’ centers could be assigned as positives, helping the optimization process focus on regions that are more likely to be informative. As $r$ increases, the number of candidates also quadratically increases, leading to potential instability in the optimization process. For example, when $r$ is set to 3, 5 or 7, their corresponding numbers of candidate anchors are 45, 125 and 245\(^3\), respectively. We study behaviors of ATSS [47],

<table>
<thead>
<tr>
<th>Method</th>
<th>Aux. Branch</th>
<th>Center Dyn. $k$</th>
<th>AP</th>
<th>AP$_{50}$</th>
<th>AP$_{75}$</th>
</tr>
</thead>
<tbody>
<tr>
<td>FCOS</td>
<td>-</td>
<td>✓</td>
<td>38.3</td>
<td>57.1</td>
<td>41.3</td>
</tr>
<tr>
<td></td>
<td>CenterNess</td>
<td>✓</td>
<td>38.9</td>
<td>57.5</td>
<td>42.0</td>
</tr>
<tr>
<td></td>
<td>IoU</td>
<td>✓</td>
<td>38.8</td>
<td>57.7</td>
<td>41.8</td>
</tr>
<tr>
<td></td>
<td>IoU</td>
<td>✓</td>
<td>39.5</td>
<td>57.6</td>
<td>42.9</td>
</tr>
<tr>
<td>OTA (FCOS)</td>
<td>-</td>
<td>✓</td>
<td>39.2</td>
<td>58.3</td>
<td>42.2</td>
</tr>
<tr>
<td></td>
<td>IoU</td>
<td>✓</td>
<td>39.6</td>
<td>58.1</td>
<td>42.5</td>
</tr>
<tr>
<td></td>
<td>IoU</td>
<td>✓ ✓</td>
<td>40.3</td>
<td>58.6</td>
<td>43.7</td>
</tr>
<tr>
<td></td>
<td>IoU</td>
<td>✓ ✓</td>
<td>40.7</td>
<td>58.4</td>
<td>44.3</td>
</tr>
<tr>
<td>OTA (RetinaNet)</td>
<td>IoU</td>
<td>✓ ✓</td>
<td>40.7</td>
<td>58.6</td>
<td>44.1</td>
</tr>
</tbody>
</table>

Table 1. Ablation studies on each components in OTA. “Center” stands for Center Prior and Center Sampling for OTA and FCOS, respectively. Dyn.$k$ is the abbreviation of our proposed Dynamic $k$ Estimation strategy.
OTA, and evaluate their corresponding performance under different \( r \) in Table 2. Noted that the optimal assigning plan in OTA is continuous, hence we define anchor \( a_j \) as an ambiguous anchor if \( \max \pi_j^* < 0.9 \). Table 2 shows that for ATSS, the number of ambiguous anchors greatly increases as \( r \) varies from 3 to 7. Its performance correspondingly drops from 39.4% to 37.2%. For PAA, the number of ambiguous anchors is less sensitive to \( r \), but its performance still drops 0.8%, indicating that Max IoU adopted by PAA is not an ideal prior to ambiguous anchors. In OTA, when multiple \( gts \) tend to transport positive labels to the same anchor, the OT algorithm will automatically resolve their conflicts based on the principle of minimum global costs. Hence the number of ambiguous anchor for OTA remains low and barely increases as \( r \) increases from 3 to 7. The corresponding performance is also stable.

Further, we manually assign the ambiguous anchors based on hand-crafted rules before performing OTA. In this case, OTA is only in charge of post/prev samples division. Table 3 shows that such a combination of hand-crafted rules and OTA decreases the AP by 0.7% and 0.4%, respectively. Finally, we visualize some assigning results in Fig. 3. Red arrows and dashed ovals highlight the ambiguous regions (i.e., overlaps between different \( fg_s \) or junctions between \( fg_s \) and \( bg \)). Suffering from the lack of context and global information, ATSS and PAA perform poorly in such regions, leading to sub-optimal detection performances. Conversely, OTA assigns much less positive anchors in such regions, which we believe is a desired behavior.

Effects of \( k \).

Before performing Sinkhorn-Knopp Iteration, we need to define how many positive labels can each

---

**Table 2. Performances of different label assigning strategies under different number of anchor candidates.**

<table>
<thead>
<tr>
<th>Method</th>
<th>ATSS [47]</th>
<th>PAA [16]</th>
<th>OTA</th>
</tr>
</thead>
<tbody>
<tr>
<td>( r )</td>
<td>( N_{amb} )</td>
<td>( AP )</td>
<td>( AP_{50} )</td>
</tr>
<tr>
<td>3</td>
<td>2.1</td>
<td>39.4</td>
<td>57.5</td>
</tr>
<tr>
<td>5</td>
<td>15.9</td>
<td>38.0</td>
<td>55.8</td>
</tr>
<tr>
<td>7</td>
<td>36.3</td>
<td>37.2</td>
<td>58.9</td>
</tr>
</tbody>
</table>

---

**Table 3.**

<table>
<thead>
<tr>
<th>Method</th>
<th>AP</th>
<th>( AP_{50} )</th>
<th>( AP_{75} )</th>
</tr>
</thead>
<tbody>
<tr>
<td>Min Area</td>
<td>40.0</td>
<td>57.8</td>
<td>43.6</td>
</tr>
<tr>
<td>Max IoU</td>
<td>40.3</td>
<td>58.1</td>
<td>43.7</td>
</tr>
<tr>
<td>Min Loss</td>
<td>40.3</td>
<td>57.9</td>
<td>43.6</td>
</tr>
<tr>
<td>OTA</td>
<td>40.7</td>
<td>58.4</td>
<td>44.3</td>
</tr>
</tbody>
</table>

**Ambiguous Anchors Handling.** Most existing dynamic label assigning methods [47, 16, 48] only conduct a small candidate set for each \( gt \), because a large number of candidates brings trouble—when occlusion happens or several objects are close enough, an anchor may simultaneously be a qualified candidate for multiple \( gts \). We define such anchors as ambiguous anchors. Previous methods mainly handle this ambiguity by introducing hand-crafted rules e.g., Min Area [38], Max IoU [47, 16, 21] and Min Loss\(^4\). To illustrate OTA’s superiority on ambiguous handling, We count the number of ambiguous anchors in ATSS, PAA and

---

\(^{4}\)Assigning ambiguous anchor to the \( gt \) with the minimal loss.

---

**Table 3. Performance comparisons on ambiguity handling between OTA and other human-designed strategies on the COCO val set.**

\( f.b. \) denotes “followed by”.

---

**Figure 3.** Visualizations of assigning results. For PAA, the dots stand for geometric centers of positive anchor boxes. For ATSS and OTA, the dots stand for positive anchor points. Rectangles represent the \( gt \) bounding boxes. To clearly illustrate the differences between different assigning strategies, we set \( r \) to 5 for all methods. Only FPN layers with the largest number of positive anchors are shown for better visualization.

---

**Table 2.** Performances of different label assigning strategies under different number of anchor candidates. \( N_{amb} \) denotes the average number of ambiguous anchors per-image calculated on COCO train set.
shows that dy-
and compare its performance
AP
Table 4. Analysis of different values of $k$ and Dynamic $k$ Estimation strategy on the COCO val set.

<table>
<thead>
<tr>
<th>$k$</th>
<th>AP</th>
<th>AP$_{50}$</th>
<th>AP$_{75}$</th>
<th>AP$_s$</th>
<th>AP$_m$</th>
<th>AP$_l$</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>36.5</td>
<td>55.4</td>
<td>38.8</td>
<td>21.4</td>
<td>39.7</td>
<td>46.2</td>
</tr>
<tr>
<td>5</td>
<td>39.5</td>
<td>58.1</td>
<td>42.7</td>
<td>23.1</td>
<td>43.0</td>
<td>50.6</td>
</tr>
<tr>
<td>8</td>
<td>39.8</td>
<td>58.4</td>
<td>42.9</td>
<td>22.7</td>
<td>43.6</td>
<td>51.5</td>
</tr>
<tr>
<td>10</td>
<td>40.3</td>
<td>58.6</td>
<td>43.7</td>
<td>23.4</td>
<td>44.2</td>
<td>52.1</td>
</tr>
<tr>
<td>12</td>
<td>40.3</td>
<td>58.6</td>
<td>43.6</td>
<td>23.2</td>
<td>44.2</td>
<td>51.9</td>
</tr>
<tr>
<td>15</td>
<td>40.2</td>
<td>58.4</td>
<td>43.6</td>
<td>23.2</td>
<td>44.1</td>
<td>51.9</td>
</tr>
<tr>
<td>20</td>
<td>40.1</td>
<td>58.2</td>
<td>43.6</td>
<td>23.5</td>
<td>44.0</td>
<td>52.8</td>
</tr>
</tbody>
</table>

Dyn. $k$ | 40.7 | 58.4 | 44.3 | 23.2 | 45.0 | 53.6 |

Table 4: Analysis of different values of $k$ and Dynamic $k$ Estimation strategy on the COCO val set.

Fixing $k$ strategy assumes every $gt$ has the same number of appropriate positive anchors. However, we believe that this number for each $gt$ should vary and may be affected by many factors like objects’ sizes, spatial attitudes, and occlusion conditions, etc. Hence we adopt the Dynamic $k$ Estimation proposed in Sec 3.3 and compare its performance to the fixed $k$ strategy. Results in Table 4 shows that dynamic $k$ surpasses the best performance of fixed $k$ by 0.4% AP, validating our point and the effectiveness of Dynamic $k$ Estimation strategy.

4.3. Comparison with State-of-the-art Methods.

We compare our final models with other state-of-the-art one-stage detectors on MS Coco test-dev. Following previous works [21, 38], we randomly scale the shorter side of images in the range from 640 to 800. Besides, we double the total number of iterations to 180K with the learning rate change points scaled proportionally. Other settings are consistent with [21, 38].

As shown in Table 5, our method with ResNet-101-FPN achieves 45.3% AP, outperforms all other methods with the same backbone including ATSS (43.6% AP), AutoAssign (44.5% AP) and PAA (44.6% AP). Noted that for PAA, we remove the score voting procedure for fair comparisons between different label assigning strategies. With ResNetx-64x4d-101-FPN [43], the performance of OTA can be further improved to 47.0% AP. To demonstrate the compatibility of our method with other advanced technologies in object detection, we adopt Deformable Convolutional Networks (DCN) [54] to ResNetX backbones as well as the last convolution layer in the detection head. This improves our model’s performance from 47.0% AP to 49.2% AP. Finally, with the multi-scale testing technique, our best model achieves 51.5% AP.

4.4. Experiments on CrowdHuman

Object detection in crowded scenarios has raised more and more attention [24, 15, 9, 10]. Compared to dataset designed for general object detection like COCO, ambiguity happens more frequently in crowded dataset. Hence to demonstrate OTA’s advantage on handling ambiguous anchors, it is necessary to conduct experiments on a crowded dataset – CrowdHuman [35]. CrowdHuman contains 15000, 4370, and 5000 images in training, validation, and test set, respectively, with the average number of persons in an image 22.6. For all experiments, we train the detectors for 30 epochs (i.e., 2.5x) for better convergence. NMS threshold is set to 0.6. We adopt ResNet-50 [13] as the default backbone in our experiments. Other settings are the same as our experiments on COCO. For evaluation, we follow the standard Caltech [7] evaluation metric – MR, which stands for the Log-Average Missing Rate over false positives per image (FPPI) ranging in $[10^{-2}, 10^{0}]$. AP and Recall are also reported for reference. All evaluation results are reported on the CrowdHuman val subset.

As shown in Table 6, RetinaNet and FCOS only achieve 58.8% and 55.0% MR respectively, which are far worse than two stage detectors like Faster R-CNN (with FPN), revealing the dilemma of one-stage detectors in crowd scenarios. Starting from FreeAnchor, the performances of one-stage detectors gradually get improved by the dynamic labeling assigning strategies. ATSS achieves 49.5% MR, which is very close to the performance of Faster R-CNN (48.7% AP). Recent proposed LLA [10] leverages loss-aware label assignment, which is similar to OTA and achieves 47.9% MR. However, our OTA takes a step forward by introducing global information into the label assignment, boosting MR to 46.6%. The AP and Recall of OTA also surpass other existing one-stage detectors by a clear margin.

Although PAA achieves competitive performance with OTA on COCO, it performs struggling on CrowdHuman. We conjecture that PAA needs clear positive/negative decision boundaries to help GMM learn better clusters. But in crowded scenarios, such clear boundaries may not exist because potential negative samples usually cover a sufficient amount of foreground areas, resulting in PAA’s poor performance. Also, PAA performs per-gr’s clustering, which heavily increases the training time on crowded datasets like CrowdHuman. Compared to PAA, OTA still shows promis-
Table 5. Performance comparison with state-of-the-art one-stage detectors on MS COCO 2017 test-dev set. * indicates the specific form of multi-scale testing that adopted in ATSS [47].

<table>
<thead>
<tr>
<th>Method</th>
<th>Iteration</th>
<th>Backbone</th>
<th>AP</th>
<th>AP50</th>
<th>AP75</th>
<th>APs</th>
<th>APm</th>
<th>API</th>
</tr>
</thead>
<tbody>
<tr>
<td>RetinaNet</td>
<td>135k</td>
<td>ResNet-101</td>
<td>39.1</td>
<td>59.1</td>
<td>42.3</td>
<td>21.8</td>
<td>42.7</td>
<td>50.2</td>
</tr>
<tr>
<td>FCOS</td>
<td>180k</td>
<td>ResNet-101</td>
<td>41.5</td>
<td>60.7</td>
<td>45.0</td>
<td>24.4</td>
<td>44.8</td>
<td>51.6</td>
</tr>
<tr>
<td>NoisyAnchor</td>
<td>180k</td>
<td>ResNet-101</td>
<td>41.8</td>
<td>61.1</td>
<td>44.9</td>
<td>23.4</td>
<td>44.9</td>
<td>52.9</td>
</tr>
<tr>
<td>FreeAnchor</td>
<td>180k</td>
<td>ResNet-101</td>
<td>43.1</td>
<td>62.2</td>
<td>46.4</td>
<td>24.5</td>
<td>46.1</td>
<td>54.8</td>
</tr>
<tr>
<td>SAPD</td>
<td>180k</td>
<td>ResNet-101</td>
<td>43.5</td>
<td>63.6</td>
<td>46.5</td>
<td>24.9</td>
<td>46.8</td>
<td>54.6</td>
</tr>
<tr>
<td>MAL</td>
<td>180k</td>
<td>ResNet-101</td>
<td>43.6</td>
<td>61.8</td>
<td>47.1</td>
<td>25.0</td>
<td>46.9</td>
<td>55.8</td>
</tr>
<tr>
<td>ATSS</td>
<td>180k</td>
<td>ResNet-101</td>
<td>43.6</td>
<td>62.1</td>
<td>47.4</td>
<td>26.1</td>
<td>47.0</td>
<td>53.6</td>
</tr>
<tr>
<td>AutoAssign</td>
<td>180k</td>
<td>ResNet-101</td>
<td>44.5</td>
<td>64.3</td>
<td>48.4</td>
<td>25.9</td>
<td>47.4</td>
<td>55.0</td>
</tr>
<tr>
<td>PAA</td>
<td>180k</td>
<td>ResNet-101</td>
<td>44.6</td>
<td>63.3</td>
<td>48.4</td>
<td>26.4</td>
<td>48.5</td>
<td>56.0</td>
</tr>
<tr>
<td>OTA (Ours)</td>
<td>180k</td>
<td>ResNet-101</td>
<td>45.3</td>
<td>63.5</td>
<td>49.3</td>
<td>26.9</td>
<td>48.8</td>
<td>56.1</td>
</tr>
<tr>
<td>FoveaBox</td>
<td>180k</td>
<td>ResNeXt-101</td>
<td>42.1</td>
<td>61.9</td>
<td>45.2</td>
<td>24.9</td>
<td>46.8</td>
<td>55.6</td>
</tr>
<tr>
<td>FSAF</td>
<td>180k</td>
<td>ResNeXt-64x4d-101</td>
<td>42.9</td>
<td>63.8</td>
<td>46.3</td>
<td>26.6</td>
<td>46.2</td>
<td>52.7</td>
</tr>
<tr>
<td>FCOS</td>
<td>180k</td>
<td>ResNeXt-64x4d-101</td>
<td>43.2</td>
<td>62.8</td>
<td>46.6</td>
<td>26.5</td>
<td>46.2</td>
<td>53.3</td>
</tr>
<tr>
<td>NoisyAnchor</td>
<td>180k</td>
<td>ResNeXt-64x4d-101</td>
<td>44.1</td>
<td>63.8</td>
<td>47.5</td>
<td>26.0</td>
<td>47.4</td>
<td>55.0</td>
</tr>
<tr>
<td>FreeAnchor</td>
<td>180k</td>
<td>ResNeXt-64x4d-101</td>
<td>44.9</td>
<td>64.3</td>
<td>48.5</td>
<td>26.8</td>
<td>48.3</td>
<td>55.9</td>
</tr>
<tr>
<td>SAPD</td>
<td>180k</td>
<td>ResNeXt-64x4d-101</td>
<td>45.4</td>
<td>65.6</td>
<td>49.7</td>
<td>27.3</td>
<td>48.7</td>
<td>55.6</td>
</tr>
<tr>
<td>ATSS</td>
<td>180k</td>
<td>ResNeXt-64x4d-101</td>
<td>45.6</td>
<td>64.6</td>
<td>49.7</td>
<td>28.5</td>
<td>48.9</td>
<td>55.6</td>
</tr>
<tr>
<td>MAL</td>
<td>180k</td>
<td>ResNeXt-101</td>
<td>45.9</td>
<td>65.4</td>
<td>49.7</td>
<td>27.8</td>
<td>49.1</td>
<td>57.8</td>
</tr>
<tr>
<td>AutoAssign</td>
<td>180k</td>
<td>ResNeXt-64x4d-101</td>
<td>46.5</td>
<td>66.5</td>
<td>50.7</td>
<td>28.3</td>
<td>49.7</td>
<td>56.6</td>
</tr>
<tr>
<td>PAA</td>
<td>180k</td>
<td>ResNeXt-64x4d-101</td>
<td>46.6</td>
<td>65.6</td>
<td>50.7</td>
<td>28.7</td>
<td>50.5</td>
<td>58.1</td>
</tr>
<tr>
<td>OTA (Ours)</td>
<td>180k</td>
<td>ResNeXt-64x4d-101</td>
<td>47.0</td>
<td>65.8</td>
<td>51.1</td>
<td>29.2</td>
<td>50.4</td>
<td>57.9</td>
</tr>
<tr>
<td>SAPD</td>
<td>180k</td>
<td>ResNeXt-64x4d-101-DCN</td>
<td>47.4</td>
<td>67.4</td>
<td>51.1</td>
<td>28.1</td>
<td>50.3</td>
<td>61.5</td>
</tr>
<tr>
<td>ATSS</td>
<td>180k</td>
<td>ResNeXt-64x4d-101-DCN</td>
<td>47.7</td>
<td>66.5</td>
<td>51.9</td>
<td>29.7</td>
<td>50.8</td>
<td>59.4</td>
</tr>
<tr>
<td>AutoAssign</td>
<td>180k</td>
<td>ResNeXt-64x4d-101-DCN</td>
<td>48.3</td>
<td>67.4</td>
<td>52.7</td>
<td>29.2</td>
<td>51.0</td>
<td>60.3</td>
</tr>
<tr>
<td>PAA</td>
<td>180k</td>
<td>ResNeXt-64x4d-101-DCN</td>
<td>48.6</td>
<td>67.5</td>
<td>52.7</td>
<td>29.9</td>
<td>52.2</td>
<td>61.5</td>
</tr>
<tr>
<td>OTA (Ours)</td>
<td>180k</td>
<td>ResNeXt-64x4d-101-DCN</td>
<td>49.2</td>
<td>67.6</td>
<td>53.5</td>
<td>30.0</td>
<td>52.5</td>
<td>62.3</td>
</tr>
<tr>
<td>ATSS</td>
<td>180k</td>
<td>ResNeXt-64x4d-101-DCN</td>
<td>49.7</td>
<td>68.9</td>
<td>56.3</td>
<td>33.2</td>
<td>52.9</td>
<td>62.2</td>
</tr>
<tr>
<td>PAA</td>
<td>180k</td>
<td>ResNeXt-64x4d-101-DCN</td>
<td>51.3</td>
<td>68.8</td>
<td>56.6</td>
<td>34.3</td>
<td>53.5</td>
<td>63.6</td>
</tr>
<tr>
<td>OTA (Ours)*</td>
<td>180k</td>
<td>ResNeXt-64x4d-101-DCN</td>
<td>51.5</td>
<td>68.6</td>
<td>57.1</td>
<td>34.1</td>
<td>53.7</td>
<td>64.1</td>
</tr>
</tbody>
</table>

Table 6. Performance comparison on the CrowdHuman validation set. All experiments are conducted under 2.5x scheduler.

<table>
<thead>
<tr>
<th>Method</th>
<th>MR</th>
<th>AP</th>
<th>Recall</th>
</tr>
</thead>
<tbody>
<tr>
<td>Faster R-CNN with FPN [20]</td>
<td>48.7</td>
<td>86.1</td>
<td>90.4</td>
</tr>
<tr>
<td>RetinaNet</td>
<td>58.8</td>
<td>81.0</td>
<td>88.2</td>
</tr>
<tr>
<td>FCOS</td>
<td>55.0</td>
<td>86.4</td>
<td>94.1</td>
</tr>
<tr>
<td>FreeAnchor</td>
<td>51.3</td>
<td>83.9</td>
<td>89.8</td>
</tr>
<tr>
<td>ATSS</td>
<td>49.5</td>
<td>87.4</td>
<td>94.2</td>
</tr>
<tr>
<td>PAA</td>
<td>52.2</td>
<td>86.0</td>
<td>92.0</td>
</tr>
<tr>
<td>LLA</td>
<td>47.9</td>
<td>88.0</td>
<td>94.0</td>
</tr>
<tr>
<td>OTA (Ours)</td>
<td>46.6</td>
<td>88.4</td>
<td>95.1</td>
</tr>
</tbody>
</table>

Table 5. Performance comparison with state-of-the-art one-stage detectors on MS COCO 2017 test-dev set. * indicates the specific form of multi-scale testing that adopted in ATSS [47].

5. Conclusion

In this paper, we propose Optimal Transport Assignment (OTA) – an optimization theory based label assigning strategy. OTA formulates the label assigning procedure in object detection into an Optimal Transport problem, which aims to transport labels from ground-truth objects and backgrounds to anchors at minimal transporting costs. To determine the number of positive labels needed by each $g_t$, we further propose a simple estimation strategy based on the IoU values between predicted bounding boxes and each $g_t$. As shown in experiments, OTA achieves the new SOTA performance on MS COCO. Because OTA can well-handle the assignment of ambiguous anchors, it also outperforms all other one-stage detectors on CrowdHuman dataset by a large margin, demonstrating its strong generalization ability.

Acknowledgements

This research was partially supported by National Key R&D Program of China (No. 2017YFA0700800), and Beijing Academy of Artificial Intelligence (BAAI).
References


[29] Han Qiu, Yuchen Ma, Zeming Li, Songtao Liu, and Jian Sun. Borderdet: Border feature for dense object detection. In


