Omni-DETR: Omni-Supervised Object Detection with Transformers

Pei Wang\textsuperscript{2,*} Zhaowei Cai\textsuperscript{1,†} Hao Yang\textsuperscript{1} Gurumurthy Swaminathan\textsuperscript{1}
Nuno Vasconcelos\textsuperscript{2} Bernt Schiele\textsuperscript{1} Stefano Soatto\textsuperscript{1}
AWS AI Labs\textsuperscript{1} UC San Diego\textsuperscript{2}
\{zhaoweic,haoyng,gurumurs,bschiel,soattos\}@amazon.com \{pew062,nuno\}@ucsd.edu

Abstract

We consider the problem of omni-supervised object detection, which can use unlabeled, fully labeled and weakly labeled annotations, such as image tags, counts, points, etc., for object detection. This is enabled by a unified architecture, Omni-DETR, based on the recent progress on student-teacher framework and end-to-end transformer based object detection. Under this unified architecture, different types of weak labels can be leveraged to generate accurate pseudo labels, by a bipartite matching based filtering mechanism, for the model to learn. In the experiments, Omni-DETR has achieved state-of-the-art results on multiple datasets and settings. And we have found that weak annotations can help to improve detection performance and a mixture of them can achieve a better trade-off between annotation cost and accuracy than the standard complete annotation. These findings could encourage larger object detection datasets with mixture annotations. The code is available at https://github.com/amazon-research/omni-detr.

1. Introduction

Most of the successes of recent object detection are attributed to the large-scale well-established object detection datasets [11, 12, 25, 28, 38], which have accurate and complete detection annotation (category and bounding box or segmentation mask) for every object of interest in an image. In general, complete and accurate detection annotation is very expensive. For example, complete annotation of a single image of MS-COCO [28] takes about $346$ seconds, $76.5$ seconds on each category elimination and $269.5$ seconds on accurate bounding box localization, on average [35]\textsuperscript{1}. Given this expensive cost, it is very difficult to scale up the data size. For example, OpenImages consisting of 9 million images [25] used a combination of machine annotation and human verification to reduce the annotation cost. The question is, do we need accurate and complete annotation which is expensive to achieve strong detection performances?

There are many weaker forms of object annotation as shown in Figure 1 (top), e.g., points, tags, counts, etc., but they are not well explored in the literature and the majority of the object detection frameworks are designed to be used with complete detection annotations. One of the main reasons for this is that using weaker forms of annotation has not shown promising results yet. For example, the performance of weakly supervised object detection (WSOD) [23, 32, 44] lags in performance compared to standard supervised detection using complete annotations. In addition, UFO\textsuperscript{2} [35], as the first work in omni-supervised object detection (OSOD), has shown that using additional weak annotations only has marginal gains. In this paper, however, we will show that, weak annotation can help to improve detection performance and achieve better cost-accuracy trade-off.

Towards this, we propose a unified architecture for OSOD, Omni-DETR, which can work with different types of weak annotations, such as image tags, counts, points, etc., for object detection. This is enabled by a unified architecture, Omni-DETR, based on the recent progress on student-teacher framework and end-to-end transformer based object detection. Under this unified architecture, different types of weak labels can be leveraged to generate accurate pseudo labels, by a bipartite matching based filtering mechanism, for the model to learn. In the experiments, Omni-DETR has achieved state-of-the-art results on multiple datasets and settings. And we have found that weak annotations can help to improve detection performance and a mixture of them can achieve a better trade-off between annotation cost and accuracy than the standard complete annotation. These findings could encourage larger object detection datasets with mixture annotations. The code is available at https://github.com/amazon-research/omni-detr.

\textsuperscript{*} Work done during internship at Amazon. \textsuperscript{†}Corresponding author.
\textsuperscript{1}Not accurate numbers from [28] but rough estimation from [35].
of weak annotations, including image tags, object counts, points, loose bounding boxes without tags, etc., or a mixture of them. It is built on recent progresses on student-teacher based semi-supervised object detection (SSOD) [30, 41, 45] to better leverage the data even if it is unlabeled, and the end-to-end detection architecture of [7, 52] with no heuristic detection procedures, like proposal detection, non-maximum suppression, thresholding, etc. The weak ground truth labels are used to filter the teacher predictions to generate pseudo labels for the student to learn. We formulate the pseudo-label filtering as a bipartite matching problem between the sets of predictions and available weak ground truths, and propose a unified pseudo-label filtering strategy to accommodate any form of weak annotations.

Omni-DETR provides a unified framework to explore different weak forms of object annotations. With this framework, we have found that 1) weak annotations can bring additional gains even on a strong baseline; and 2) a mixture of weak and complete annotations can achieve better accuracy-cost trade-offs than just using complete annotations. As seen in Figure 1 (bottom), our Omni-DETR achieves better results than a standard supervised and a stronger semi-supervised detection baseline. In addition, some annotation forms are more suited than the others depending on the dataset characteristics. For example, as shown in Figure 2, annotating accurate bounding boxes is difficult for Bees [3] and CrowdHuman [39] datasets, since objects are small and very crowded. However, it is easier to annotate with points for these datasets. Similarly, annotating accurate categories is difficult for Objects365 [38] as there are too many categories (365), but annotating just the bounding boxes is relatively easy and cheap. Omni-DETR can accommodate all these different cases and help to reduce the cost of annotating such datasets, encouraging a larger scale of object detection datasets with mixed annotations.

Our contributions are summarized as: 1) a unified framework, Omni-DETR, that can accommodate various forms of object annotations or a mixture of them. 2) a novel and unified pseudo label filtering strategy, based on bipartite matching; 3) experimental findings that show weak annotations can provide additional gains and achieve better accuracy-cost trade-off than the standard full detection annotations; 4) the empirical exploration of optimal annotation mixtures for a fixed annotation budget, showing that the optimal mixture depends on the dataset.

2. Related Work

Supervised Object Detection is a fundamental problem in computer vision [6, 15, 16, 27, 29, 33, 34, 47]. Most detection frameworks can be categorized into two groups: two-stage [6, 15, 16, 27, 34] v.s. single-stage detectors [29, 33, 47]. These detectors usually have some heuristic steps, e.g., proposal detection, thresholding, non-maximum suppression, etc. More recently, [7] proposed the DETR framework based on transformer [48] for end-to-end detection. It formulates detection as a set-to-set prediction problem, eliminating some of the previous heuristics and enabling a simpler detection pipeline. The subsequent Deformable DETR [52] improved on the slow training convergence of DETR and achieved better detection performances. Our Omni-DETR is also based on this end-to-end framework, which is now extended to support various forms of annotations.

Semi-Supervised Object Detection (SSOD) tries to improve detection performances by using additional unlabeled data [22, 30, 37, 41, 45, 50, 51]. A prevalent SSOD paradigm is to use a multi-stage self-training pipeline [37, 41, 49]: 1) train model on labeled data; 2) generate pseudo labels on unlabeled data; 3) retrain model on both labeled and pseudo-labeled data; 4) repeat this process if needed. Some recent works [22, 30, 45] have shown great progress by resorting to an online pipeline. [22] leverages the consistency regularization between two different augmented views of a single image. [30, 45] rely on a mean-teacher framework [46], where the teacher generates online pseudo labels for the student to learn. Omni-DETR is also based on this mean-teacher framework, but uses different weak annotations to generate accurate pseudo labels.

Weakly-Supervised Object Detection (WSOD) aims to reduce detection annotation efforts by leveraging cheaper weakly labeled data. Most works only use a single type of weak annotations, e.g. image-level tags [4, 17, 23, 42, 44] or instance-level points [8, 19, 32], and usually formulate WSOD as a multiple instance learning (MIL) problem. This, however, has very limited success so far. Some recent works study weakly semi-supervised object detection (WS-SSOD) [9, 13, 14], using a small amount of fully labeled data and a large amount of additional weakly labeled data. This has shown more promising results than WSOD. In general, different types of weak annotations require specific detection algorithms, e.g., [13, 14] for WSSOD with tags and [9] for WSSOD with points. The proposed Omni-DETR is closely related to WSSOD but can accommodate various
weak annotations instead of a single one.

Omni-Supervised Object Detection (OSOD) combines different forms of annotations to improve detection. It was first proposed in UFO\textsuperscript{2} [35], which is based on the Faster R-CNN [34] framework and formulates OSOD as a multitask learning problem. However, UFO\textsuperscript{2} has only shown very marginal improvements for the addition of weak annotations, and suggested that, for a fixed annotation budget, the best choice is still full annotation. However, our experiments of Omni-DETR have the opposite observations: weak annotations are helpful and a mixture of annotations is a better solution than full annotation given a fixed annotation budget. Table 1 summarizes the related works of object detection using different types of annotations, and our Omni-DETR is a more universal framework on annotation formats than the previous works.

Object Detection Data Annotation is known to be an expensive and tedious task [11, 12, 18, 25, 28, 38], that requires annotators to choose the right category and localize the accurate bounding box for each object. For example, [43] reports an average annotation time of 35 seconds for a high-quality bounding box. The total estimated time with associated categories per COCO image is 346 seconds [35]. This high annotation cost prevents the detection dataset from being scaled up, in terms of the number of images, classes and objects. Several strategies have been used to reduce the cost. For example, Caltech Pedestrian [11] interpolates annotations between two video frames, OpenImages [25] uses machine prediction first and then human verification next, LVIS [18] only annotates a few positive/negative classes for an image instead of complete category annotation, etc. Other approaches try to relax the accurate bounding box annotation, by proposing to use relatively loose bounding boxes [31] or near-center points [32]. In this work, using Omni-DETR, we empirically find that accurate and complete detection annotation is not the most economical, and a mixture of annotations can achieve a better trade-off between accuracy and cost.

3. Omni-DETR

We at first introduce the overall framework of Omni-DETR in this section and then the unified pseudo-label filtering for various weak annotations in the next section.

3.1. Omni-labels

Omni-DETR is a unified framework to combine fully and weakly labeled data. It assumes the availability of a fully labeled and a weakly labeled dataset. The fully labeled dataset \( D^f = \{ (x_i^f, y_i^f) \}_{i=1}^{N^f} \), where \( x_i^f \) is the \( i \)-th image and \( y_i^f = \{ (b_{i,j}, c_{i,j}) \in \mathbb{R}^4 \times \{1,2,...,C\} \}_{j=1}^{B_i} \), the corresponding label, composed by \( B_i \) pairs of 1) four coordinate bounding boxes \( b_{i,j} \) and 2) corresponding classes \( c_{i,j} \), assigns class label \( c_{i,j} \) to the object localized by bounding box \( b_{i,j} \). The weakly labeled data, \( D^w = \{ (x_i^w, y_i^w) \}_{i=1}^{N^w} \), \( y_i^w \) can consist of any of the annotations introduced in the following. We omit the image index \( i \) in subsequent notations, for notational simplicity, and term the fully labeled dataset as labeled and the weakly labeled dataset as Omni-labeled.

Omni-DETR supports any of the blow annotation forms \( y \), or a mixture of them, as weak annotations for image \( x \).

- None (None) \( y = \emptyset \). No annotation for image \( x \).
- Tags (TagsU) \( y = \{c_j\}_{j=1}^M \), which is a list of image-level classes\(^2 \). \( M \) is the number of tags. In the examples of Figure 1 (top), \( M = 2 \), \( c_1 \) is “horses” and \( c_2 \) is “sheep”.
- Tags with counts (TagsK), \( y = \{ (c_j, n_j) \}_{j=1}^M \), where \( n_j \) is the count number of objects of class \( c_j \). In Figure 1 (top), \( c_1 \) is “horses” and \( n_1 = 1 \), while \( c_2 \) is “sheep” and \( n_2 = 3 \).
- Points without tags (PointsU), \( y = \{p_j \in \mathbb{R}^2 \}_{j=1}^p \), where \( p_j \) is a point annotation for an object, e.g., the geometric center of the object or a random point inside the region of support of the object in the image, and \( P \) the number of points. In Figure 1 (top), four point annotations identify four objects without class information.
- Points with tags (PointsK), \( y = \{ (p_j, c_j) \in \mathbb{R}^2 \times \{1,2,...,C\} \}_{j=1}^P \). In addition to PointsU, the label of each point is also known. In Figure 1 (top), the points and labels for three sheeps and a horse are annotated.
- Boxes without tags (BoxesU), \( y = \{b_j \in \mathbb{R}^4 \}_{j=1}^B \). The standard bounding box annotation but removing the class information. \( B \) is the number of boxes.
- Extreme Clicking Box (BoxesEC), \( y = \{b_j \in \mathbb{R}^4 \}_{j=1}^B \), where \( b_j \) is a box derived from the annotation of extreme points of the object. This was introduced in [31] and has much less annotation cost (5×) but only slightly worse quality than BoxesU annotation.

3.2. Unified Framework

Figure 3 gives an overview of the Omni-DETR framework. Motivated by the recent successes of student-teacher frameworks for semi-supervised learning [40] and SSOD [30, 45], our Omni-DETR is also composed of a student detection network \( F^s(x; \theta_s) \) and a teacher detection network \( F^t(x; \theta_t) \). We will also use “tag” to refer “class” interchangeably.

<table>
<thead>
<tr>
<th></th>
<th>None</th>
<th>Tags</th>
<th>Points with tags</th>
<th>Boxes with tags</th>
<th>Mixture</th>
</tr>
</thead>
<tbody>
<tr>
<td>SSOD [30, 45]</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>WSOD with tags [4, 23, 44]</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>WSOD with points [19, 32]</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>WSSOD with tags [13]</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>WSSOD with points [9]</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>UFO\textsuperscript{2} [35]</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>Our Omni-DETR</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
</tr>
</tbody>
</table>

Table 1. Summary of related works of object detection using different weak annotations.
For the omni-labeled data \((x^o, y^o) \in D^o\), two views of the image \(x^o\) are generated by a strong and a weak augmentation, \(x^{o,s}\) and \(x^{o,w}\), respectively. The weakly augmented view \(x^{o,w}\) is forwarded through the teacher to produce the detection prediction \(\hat{y}^t = \mathcal{F}^t(x^{o,w}; \theta_t)\), consisting of class prediction \(\hat{y}^{cls}\) and bounding box prediction \(\hat{y}^{box}\). The predictions \(\hat{y}^t\) are then passed to a pseudo label filter \(T\) together with the available omni-labels \(y^o\). The filter generates the pseudo-labels \(\hat{y}^o = T(\hat{y}^t; y^o)\), which are used to supervise the learning of the student on the strong augmentation \(x^{o,s}\). The pseudo-label filtering details will be discussed in Section 4. Here the weak/strong augmentation is only applied to teacher/student because the weak annotation can induce more accurate pseudo-labels for the teacher and the strong augmentation can make the learning of the student more challenging. For the labeled data \((x^l, y^l) \in D^l\), strong and weak augmentations are also generated, \((x^{l,s}, y^{l,s})\) and \((x^{l,w}, y^{l,w})\), and both are feed-forwarded to the student network for learning only.

Only the student \(\mathcal{F}^s\) is optimized by standard SGD with the overall loss,

\[
L^s = \sum_i \mathcal{L}(x^{l,s}_i, y^{l,s}_i) + \mathcal{L}(x^{l,w}_i, y^{l,w}_i) + \sum_i \mathcal{L}(x^{o,s}_i, \hat{y}^o_i),
\]

where

\[
\mathcal{L} = \alpha \mathcal{L}^{cls} + \beta \mathcal{L}^{box}
\]

(1)

is the weighted sum of classification loss \(\mathcal{L}^{cls}\) and bounding box regression loss \(\mathcal{L}^{box}\), and \(\alpha\) and \(\beta\) are the corresponding weights. The teacher \(\mathcal{F}^t\) is updated by the exponential moving average (EMA) from the student [46],

\[
\theta_t \leftarrow k\theta_t + (1 - k)\theta_s,
\]

(3)

where \(k\) is empirically set to a number close to 1, e.g., 0.9996. This EMA updated teacher can be seen as a temporal ensemble of student models along the training trajectories, which makes it more robust and able to generate more accurate pseudo-labels [1, 5, 21]. Note that this reduces to the Unbiased Teacher (UT) framework [30] proposed for SSOD when no omni-labels are available, and only unlabeled data is used. It follows that UT is a baseline for Omni-DETR and the addition of any weak annotations should improve the accuracy of this SSOD baseline. This establishes a much stronger baseline than any previous weakly supervised object detection (WSOD) and weakly semi-supervised object detection (WSSOD) work [13, 23, 32, 44].

3.3. Detection Architecture

Although there is no constraint on which detector to use, DETR is chosen here because it has removed many heuristic procedures in the traditional detection frameworks [27, 29, 33, 34]. This is necessary for Omni-DETR since it needs to accommodate many different kinds of annotations.

DETR [7] is a transformer [48] based end-to-end object detection framework. In DETR, a standard CNN backbone is at first applied to a given image, and the output features are flattened and followed by an encoder transformer. In order to detect objects, the decoder transformer is applied by taking the object queries as input and cross-attending the encoded vision features, to generate the final object predictions with class and bounding box predictions \(\hat{y}^{cls}\) and \(\hat{y}^{box}\). Then a set-to-set alignment is enabled by using Hungarian matching [24] between the object predictions and the ground truth objects. After matching each hypothesis and ground truth, standard learning is used to optimize the classification task (with multi-class cross-entropy loss) and bounding box regression task (with generalized IoU and L1 loss). Due to the slow convergence of original DETR, we use Deformable DETR [52] for faster convergence speeds.

3.4. Training

The overall model is trained with two stages: 1) burn-in training of the student network alone on the labeled data; 2) student-teacher training on both labeled and omni-labeled data where the teacher model is initialized by duplicating the burn-in student model.

4. Pseudo-label Filtering

As shown in Figure 3, the pseudo-label filter is a key component to leverage weak annotations in our Omni-DETR. It takes in both detection predictions and available omni-labels of an omni-labeled image, and then generates the pseudo-labels to supervise the learning of the student.

4.1. Simple Pseudo-label Filtering

At first, we present some simple pseudo-label filtering approaches for different weak annotations. Object detectors usually output a vector of confidence scores \(s_j \in [0, 1]^C\) per detected bounding box \(b_j\). A popular approach to generate pseudo-labels is to simply threshold these scores. If only tag supervision (TagsU) is available, pseudo-labels can be generated by thresholding the confidence \(s_j^{c_j}\) of the ground truth class \(c_j\). For a ground truth class if there is no prediction greater than the confidence threshold, the top-1 prediction is retrieved as the pseudo-label for that class.

- **Weakly Semi-supervised Object Detection (WSSOD)**: WSSOD tries to accommodate many different kinds of annotations. It usually outputs a vector of confidence scores for each detected bounding box, which can be used to generate pseudo-labels. The pseudo-labeling process can be performed in two ways:
  - **Self-supervised filtering**: This approach leverages the confidence scores to generate pseudo-labels. For each detected bounding box, the maximum confidence score is used as the pseudo-label confidence.
  - **Weakly supervised filtering**: This approach uses the confidence scores to generate pseudo-labels. For each detected bounding box, a threshold is applied to the confidence scores to determine the pseudo-labels.

**Figure 3. The framework of Omni-DETR**, which is based on the student-teacher framework. The pseudo-label is used to filter the predictions of the teacher network, by a unified pseudo-label filter, to generate pseudo-labels for the student network to learn. The omni-label can be any annotation introduced in Section 3.1.
count supervision is available (TagsK), with \( n_j \) counts for each ground truth class \( c_j \), this can be extended to selecting the top \( n_j \) predictions for class \( c_j \). When point supervision is available (PointsU), a similar strategy is to choose the predicted bounding boxes that contain the ground truth points. If additional tag supervision is available (PointsK), this can be extended to choosing candidates whose class prediction matches the point tag [35]. However, these empirical filtering rules are specific to each type of weak supervision and do not provide a unified pseudo-labeling solution.

### 4.2. Unified Pseudo-label Filtering

Next, we introduce the proposed unified approach\(^3\). Formally, the filter is applied to \( \hat{y} = \mathcal{T}(\hat{y}; y^i) \) where \( \hat{y} = \{\hat{y}^{cls}, \hat{y}^{box}\} \) is the prediction of the teacher network, with \( \hat{y}^{cls} \) and \( \hat{y}^{box} \) being the class and box predictions, respectively. Here, we define \( \hat{y}^{cls} = [z_1, ..., z_K]^T \in \mathbb{R}^{K \times C} \) and \( \hat{y}^{box} = [\hat{b}_1, ..., \hat{b}_K]^T \in \mathbb{R}^{K \times 4} \), where \( z_k \) is a vector of logits (the network output vector before the softmax), \( \hat{b}_k \) the associated bounding box prediction, and \( K \) the number of object queries. \( y^i \) is the omni-label of section 3.1.

#### 4.2.1 No Annotation

When no annotation is available (None), pseudo-labels are derived from confidence scores, as used in SSOD [30]. Specifically, \( \hat{y}^{cls} \) is fed into a softmax layer, to produce \( [\hat{p}_1, ..., \hat{p}_K]^T \in \mathbb{R}^K \), where \( \hat{p}_k \) is the probability over \( C \) class for query \( k \). The predicted class of the \( k \)-th prediction is defined as \( \hat{c}_k = \arg\max_c \hat{p}_k^c \) and the confidence score as the associated probability \( s_k = \hat{p}_k^{\hat{c}_k} \). The threshold \( \tau \) is used to filter low-confidence predictions, by collecting the prediction index set \( I = \{k | \hat{c}_k > \tau, k \in [1, K]\} \). With the bounding box predictions \( \hat{y}^{box} = [\hat{b}_1, ..., \hat{b}_K]^T \in \mathbb{R}^{K \times 4} \), the pseudo-labels are then defined as \( \{\hat{b}_{\hat{c}_k}, \hat{c}_k\} \in I \) where \( \hat{b}_k \) is a pseudo box and \( \hat{c}_k \) its pseudo class.

#### 4.2.2 Weak Annotations

Motivated by DETR [7], we formulate the pseudo-label filtering problem as a bipartite matching problem, between the \( K \) teacher predictions and the available \( G \) ground truth omni-labels \( \{g_i\}_{i=1}^G \). Specifically, we search for a permutation \( \hat{\sigma} \in \mathcal{S}_K \) of \( K \) elements such that

\[
\hat{\sigma} = \arg\min_{\sigma \in \mathcal{S}_K} \sum_{\sigma = 1}^{K} \mathcal{L}_{match}(g_i, \hat{y}_{\sigma(i)}),
\]

where \( \mathcal{L}_{match}(g_i, \hat{y}_{\sigma(i)}) \) is an annotation-specific pair-wise matching cost between ground truth omni-label \( g_i \) and teacher prediction \( \hat{y} \) of index \( \sigma(i) \). The optimal assignment is enabled with Hungarian matching [7, 24], assigning pseudo-labels \( \{(b_1^*, c_1), ..., (b_G^*, c_G)\} \). Here, \( b_1^* \in \mathbb{R}^4 \) is the pseudo bounding box and \( c_2 \) the pseudo class, depending on the weak annotation type. Next, we present the specific \( \mathcal{L}_{match}(g_i, \hat{y}_{\sigma(i)})(i) \) and \( (b_1^*, c_1^*) \) for different annotations.

#### TagsU

When the image-level ground truth tags are available, \( y^e = \{c_j\}_{j=1}^M \), where \( M \) is the number of tags and \( c_j \) is the \( j \)-th class, but the exact number of objects per class is not known, the matching of (4) is not directly applicable. To address this problem, the count \( n_j \) of tag \( c_j \) is first predicted with,

\[
n_j = \max(1, |\{k | \hat{p}_k^{c_j} > \tau, k \in [1, K]\}|),
\]

where \( \hat{p}_k^{c_j} \) is the probability of assigning the \( k \)-th prediction to class \( c_j \), and \(|·|\) is the set cardinality. The predicted count is the number of predictions that pass the confidence threshold, if any, and set to one otherwise. This is because there is at least one object per ground truth tag. In order to accommodate the matching of (4) regarding \( G \) ground truth, we re-write the ground truth set as \( \{g_i\} = \{c_j\}_{j=1}^M \), with \( G = \sum_{j=1}^{M} n_j \) and \( n_j \) repetitions of each tag \( c_j \). Note that, for different \( i \), \( c_i \) could be the same if there are multiple objects per class. \( \mathcal{L}_{match}(g_i, \hat{y}_{\sigma(i)}) \) in (4) is defined as

\[
\mathcal{L}_{match}(g_i, \hat{y}_{\sigma(i)}) = 1 - \hat{p}_{\sigma(i)}.
\]

After bipartite matching of (6), the pseudo-labels are \( \{(\hat{b}_{\hat{\sigma}(i)}, c_i)\}_{i=1}^G \), where \( \hat{\sigma}(i) \in \{1, ..., K\} \) is the matched index to the \( i \)-th ground truth omni-label. \( \hat{b}_{\hat{\sigma}(i)} \) is the predicted box and \( c_i \) the available ground truth class.

#### TagsK

When the tags and their counts are known, \( y^k = \{c_j, n_j\}_{j=1}^M \), where \( n_j \) is the number of objects of class \( c_j \). There is no need to predict the counts anymore. The optimal matching can be computed with (6), to obtain the pseudo-labels \( \{(\hat{b}_{\hat{\sigma}(i)}, c_i)\}_{i=1}^G \).

#### PointsU

When points of objects are known, \( y^e = \{g_i\} = \{p_i \in \mathbb{R}^2\}_{i=1}^G \), where \( p_i \) is a point and \( G \) points in total. The matching cost is defined as

\[
\mathcal{L}_{match}(g_i, \hat{y}_{\sigma(i)}) = (d_{i,\sigma(i)} + e_{i,\sigma(i)}) \ast \eta_{i,\sigma(i)},
\]

where \( d_{i,\sigma(i)} \) is the L2 normalized distance, between the center of the predicted box and ground truth point, normalized to \([0, 1]\) across \( K \times G \) distances by min-max normalization, and \( e_{i,\sigma(i)} = 1 - s_{\sigma(i)} \), where \( s_{\sigma(i)} \) is the confidence score of \( \sigma(i) \)-th prediction. Finally, \( \eta_{i,\sigma(i)} \) is an indicator: \( \eta_{i,\sigma(i)} = 1 \) if the \( i \)-th ground truth point is inside the \( \sigma(i) \)-th predicted box, otherwise +∞. This cost encourages the selected predicted box to cover the ground truth point with small geometric distance and high confidence. The pseudo-labels \( \{(\hat{b}_{\hat{\sigma}(i)}, c_i)\}_{i=1}^G \) are obtained by optimizing (7) via Hungarian matching.

---

\(^3\)Although specific design is still needed for each weak annotation, here we use “unified” because the filtering of different weak annotations can be interpreted by a unified bipartite matching mechanism.
PointsK When both the point and tag of an object are known, the ground truth is \( y^o = \{ g_i \} = \{ (p_i, c_i) \in \mathcal{R}^2 \times \{1, 2, ..., C\} \}_{i=1}^C \). We combine (6) and (7) linearly as the overall matching cost,

\[
L_{\text{match}}(g_i, \hat{y}_{\sigma(i)}) = \gamma L_{\text{match}}^o + (1 - \gamma) L_{\text{match}}^p,
\]

where \( \gamma \) is the trade-off coefficient, to obtain the pseudo-labels \( \{ (\hat{b}_i, \hat{c}_i) \}_{i=1}^C \).

Boxes When the bounding boxes are known but without classes, \( y^o = \{ g_i \} = \{ b_i \in \mathcal{R}^4 \}_{i=1}^C \), we follow the bounding box cost definition of [7],

\[
L_{\text{match}}^b(g_i, \hat{y}_{\sigma(i)}) = \lambda_{\text{iou}} L_{\text{iou}}(g_i, \hat{b}_{\sigma(i)}) + \lambda_1 \| g_i - \hat{b}_{\sigma(i)} \|_1,
\]

where \( L_{\text{iou}} \) is the generalized IoU loss [36], to obtain the pseudo-labels \( \{ (b_i, c_{\sigma(i)}) \}_{i=1}^C \). Although BoxesEC and BoxesU have different box qualities, they are not differentiated by their matching costs here.

The discussion above unifies pseudo-label filtering for all weak annotations as a bipartite matching problem, performed by global optimization on a set-to-set matching problem. This will be shown, in experiments, to outperform the heuristic choice of Section 4.1.

5. Experiments

Omni-DETR is extensively evaluated on different datasets and settings.

5.1. Experimental Settings

Datasets: MS-COCO [28], PASCAL VOC [12], Bees [3], CrowdHuman [39] and Objects365 [38] are used for evaluation. To evaluate and compare Omni-DETR to methods addressing different problems, we use the multiple experimental settings of [13, 30, 35]. (I) COCO-standard: we randomly sample \{1, 2, 5, 10, 20, 30\}% of data from COCO train2017 as the fully labeled training data and use the rest as the semi-labeled training data. (II) COCO-35to80: we use the COCO-35 (a.k.a. valminusminival), a subset of 35K images of COCO train2017 as the fully labeled data and COCO-80, the COCO train2014 of 80K images, as the semi-labeled data. (III) VOC-07to12: we use the VOC07 trainval as the fully labeled set and the VOC12 trainval as the semi-labeled set. On COCO, model performance is evaluated on the COCO val2017 and VOC07 test on VOC.

Implementation details: For fair comparison, ResNet-50 pretrained on ImageNet [10, 20] is used as the backbone. The confidence threshold \( \tau = 0.7 \). For strong augmentation, following [30, 52], we apply random horizontal flipping, random resizing, random size cropping, color jittering, grayscale, Gaussian blur, and cutout patches. For weak augmentation, only random horizontal flipping is used. To mimic point annotations, we follow [9, 35] and randomly sample a point from the instance mask if the dataset has instance segmentation, otherwise, we randomly sample a point inside each bounding box. For Extreme Clicking boxes, since [31] only has partial annotations on VOC, we simulate similar annotations on other datasets by adding noise to their ground truth bounding box annotations, so that the resulting boxes have close distribution to that of Extreme Clicking on VOC. More details can be found in the supplementary. The detection performance is evaluated with the teacher model for all experiments. We use \( \text{AP}_{50:95} \), denoted as mAP, as the evaluation metric unless otherwise noted. The minimum size of image height and width is set to 600-pixels for faster experiments, except in experiments involving comparisons with other methods that use the standard 800-pixel size.

5.2. Evaluation on Single Annotation

Under the setting of COCO-standard-10%, we first evaluate Omni-DETR for individual weak annotations in Table 2, to study the effect of each weak annotation. The baseline is the standard supervised learning on 10% labeled data. A few observations are available. First, the additional 90% unlabeled data improves the baseline by 4.4% when using semi-supervised learning. Annotating extra weak labels always enhances the performances by 1.7 – 4.4%. Second, among all annotation formats, PointsU has the smallest benefit and BoxesU the largest. Third, Extreme Clicking boxes (BoxesEC) is economical: only 0.3% worse than the high-quality boxes of BoxesU but 5 times less costly. Fourth, count annotation provides a gain of 0.5% over tag annotation (TagsU v.s. TagsK). Fifth, adding tag information to points (PointsU v.s. PointsK), leads to 1.6% improvement.

5.3. Comparison with the State-of-the-art

Omni-DETR is compared with previous works under different settings. In this section, “Supervised” is the supervised Deformable DETR baseline trained on the available fully-labeled data only.

SSOD When no annotation is available, the Omni-DETR becomes a standard semi-supervised detector, which is compared to other SSOD methods in Table 3. We implemented the supervised Faster R-CNN and Deformable
In addition to Table 4, we also compare with UFO\textsuperscript{2} on COCO-standard-10\% and VOC-07to12. * indicates our implementation.

<table>
<thead>
<tr>
<th>Model</th>
<th>COCO-standard-10%</th>
<th>COCO-standard-1%</th>
<th>VOC-07to12</th>
</tr>
</thead>
<tbody>
<tr>
<td>Faster R-CNN\textsuperscript{2} [30,34]</td>
<td>9.1</td>
<td>12.7</td>
<td>18.5</td>
</tr>
<tr>
<td>Faster R-CNN\textsuperscript{*}</td>
<td>11.7</td>
<td>14.9</td>
<td>20.7</td>
</tr>
<tr>
<td>Deformable DETR\textsuperscript{*}</td>
<td>11.0</td>
<td>14.7</td>
<td>23.7</td>
</tr>
<tr>
<td>STAC [41]</td>
<td>14.0</td>
<td>18.3</td>
<td>24.4</td>
</tr>
<tr>
<td>Unbiased Teacher [30]</td>
<td>20.8</td>
<td>24.3</td>
<td>28.3</td>
</tr>
<tr>
<td>Coarse teacher [42]</td>
<td>17.0</td>
<td>21.7</td>
<td>27.7</td>
</tr>
<tr>
<td>Omni-DETR (ours)</td>
<td>18.6</td>
<td>23.2</td>
<td>30.2</td>
</tr>
</tbody>
</table>

Table 3. SSOD result comparison on COCO-standard and VOC-07to12. * indicates our implementation.

<table>
<thead>
<tr>
<th>Model</th>
<th>SSOD standard-5%</th>
<th>SSOD standard-1%</th>
<th>SSOD standard-2%</th>
<th>SSOD standard-5%</th>
</tr>
</thead>
<tbody>
<tr>
<td>Faster R-CNN\textsuperscript{2} [30,34]</td>
<td>9.1</td>
<td>12.7</td>
<td>18.5</td>
<td>23.9</td>
</tr>
<tr>
<td>Faster R-CNN\textsuperscript{*}</td>
<td>11.7</td>
<td>14.9</td>
<td>20.7</td>
<td>25.6</td>
</tr>
<tr>
<td>Deformable DETR\textsuperscript{*}</td>
<td>11.0</td>
<td>14.7</td>
<td>23.7</td>
<td>29.2</td>
</tr>
<tr>
<td>STAC [41]</td>
<td>14.0</td>
<td>18.3</td>
<td>24.4</td>
<td>28.6</td>
</tr>
<tr>
<td>Unbiased Teacher [30]</td>
<td>20.8</td>
<td>24.3</td>
<td>28.3</td>
<td>31.5</td>
</tr>
<tr>
<td>Coarse teacher [42]</td>
<td>17.0</td>
<td>21.7</td>
<td>27.7</td>
<td>31.6</td>
</tr>
<tr>
<td>Omni-DETR (ours)</td>
<td>18.6</td>
<td>23.2</td>
<td>30.2</td>
<td>34.1</td>
</tr>
</tbody>
</table>

Table 4. WSSOD comparison with UFO\textsuperscript{2} on COCO-standard. The numbers in parentheses are gains over the supervised baseline (Faster R-CNN for UFO\textsuperscript{2} but Deformable DETR for ours).

<table>
<thead>
<tr>
<th>Model</th>
<th>SSOD standard-5%</th>
<th>SSOD standard-1%</th>
<th>SSOD standard-2%</th>
<th>SSOD standard-5%</th>
</tr>
</thead>
<tbody>
<tr>
<td>Faster R-CNN\textsuperscript{2} [30,34]</td>
<td>9.1</td>
<td>12.7</td>
<td>18.5</td>
<td>23.9</td>
</tr>
<tr>
<td>Faster R-CNN\textsuperscript{*}</td>
<td>11.7</td>
<td>14.9</td>
<td>20.7</td>
<td>25.6</td>
</tr>
<tr>
<td>Deformable DETR\textsuperscript{*}</td>
<td>11.0</td>
<td>14.7</td>
<td>23.7</td>
<td>29.2</td>
</tr>
<tr>
<td>STAC [41]</td>
<td>14.0</td>
<td>18.3</td>
<td>24.4</td>
<td>28.6</td>
</tr>
<tr>
<td>Unbiased Teacher [30]</td>
<td>20.8</td>
<td>24.3</td>
<td>28.3</td>
<td>31.5</td>
</tr>
<tr>
<td>Coarse teacher [42]</td>
<td>17.0</td>
<td>21.7</td>
<td>27.7</td>
<td>31.6</td>
</tr>
<tr>
<td>Omni-DETR (ours)</td>
<td>18.6</td>
<td>23.2</td>
<td>30.2</td>
<td>34.1</td>
</tr>
</tbody>
</table>

Table 5. WSSOD with tags result comparison on COCO-standard. DETR trained only on the labeled data as the baselines. Omni-DETR achieves the best results on 5% and 10% of COCO, and VOC, and comparable results with the state-of-the-art on 1% and 2% of COCO\textsuperscript{4}. Note that our Omni-DETR is not designed specifically for SSOD, but it still achieves competitive results.

**WSSOD with tags** When additional tag annotation is available, SSOD becomes WSSOD with tags. We compare with the state-of-the-art methods, UFO\textsuperscript{2} [35] and Fang et al. [13], on their settings. The results are reported in Table 4 and 5, showing that our model consistently outperforms [13,35]. It is worth noting, in Table 5, that our model trained on 5% (10\%) labeled data achieves 31.7 (35.9) mAP, which is higher than [13] trained on 10% (20\%) labeled data. In Table 4, we improve the supervised baseline by 5.1\% by using tags, whereas the improvement is 0.3\% for UFO\textsuperscript{2}. Our absolute improvement over UFO\textsuperscript{2} is 10\%.

**WSSOD with points** When additional point with tag annotation is available for SSOD, the problem becomes WSSOD with points [9]. Omni-DETR is compared with Point DETR and UFO\textsuperscript{2}, in Table 6 and 4 respectively. It can be observed in Table 6 that we outperform Point DETR by a significant margin (5 – 7\%). In Table 4, when using points, Omni-DETR improves over the supervised baseline by 5.9\% where UFO\textsuperscript{2} improves by 1\%, and our absolute gain over UFO\textsuperscript{2} is 10.1\%.

**OSOD** In addition to Table 4, we also compare with UFO\textsuperscript{2} on the X%B settings of [35], where X%B are different annotation policies using 10K images of COCO. Under a fixed budget, X% budget is spent on fully labeled annotations, and the rest on PointsK. As shown in Table 7, Omni-DETR still has consistently significant gains over UFO\textsuperscript{2}.

### 5.4. Ablation Study

We ablate some key components of our Omni-DETR on COCO-standard-10\% setting.

**Comparison with simple filters** We compare the proposed unified pseudo-label filter with the simple filter of Section 4.1. Table 8 shows that the proposed unified filter is better than the simple and heuristic filter under various settings. This is because the matching in the unified filter is a global solution by Hungarian algorithm, instead of a heuristic one as in the simple filter.

**Confidence threshold** The confidence threshold $\tau$ used in Section 4.2.1 and (5) determines the trade-off between the quality and quantity of the pseudo-labels. A larger $\tau$ leads to fewer examples passing the threshold but with high quality, but a smaller $\tau$ allows more examples passing but more likely false positives. The results of different values of $\tau$ (0.5 to 0.9) are reported in Table 9. $\tau = 0.7$ is the best.

The effect of $\gamma$ The hyperparameter $\gamma$ of (8) balances the importance of positions and tag labels during the matching for point annotation. Its effect is evaluated in Table 10, and $\gamma = 0.5$ is the best.

**Pseudo bounding box** In Unbiased Teacher [30], pseudo bounding boxes are not used for learning from unlabeled data since the class confidence score does not reflect the goodness of the bounding box. However, we found that pseudo bounding boxes are useful and provide consistent improvement of 0.5 – 1\% in our experiments. One possible
reason for the improvement is the higher quality of OmniDETR pseudo bounding boxes.

5.5. Budget-Aware Omni-Supervised Detection

We also empirically study the trade-off between annotation cost and accuracy of several annotation policies. Here, annotation policy refers to the strategy for mixing different annotation formats. Five diverse datasets with different characteristics are tested. The annotation cost, in seconds per image, for each type of annotation is shown in Table 11, following [2, 31, 35, 43]. For each dataset, we attempt to identify the best annotation policy, given different budgets. SSOD is used as the baseline because, when the entire budget is used on standard full object annotation (bounding boxes and tags), the remaining data is considered unlabeled, which is the standard SSOD setup and widely adopted in practice. Next, we consider different weakly semi-supervised settings with 10% data fully labeled and the remaining 90% labeled with different weak annotations. Finally, two choices of mixture annotation are tested under three budgets per dataset, to show the omni-supervised results. We only tested the combination of weak annotations that are better than the SSOD baseline, and the combination ratio was decided manually so that the mixture and full annotations had similar costs, for a fair comparison. Please see more details in the supplementary.

The results are summarized in Figure 4. The grey lines and green dots are the SSOD and WSSOD results, respectively, whereas red/blue/purple lines are for OSOD variants. It can be found that the OSOD results are higher than the strong SSOD baseline in general. For a target accuracy, the mixture annotation strategy can help reduce cost, and for a given cost, SSOD can improve accuracy. For example, on Bees at the accuracy of 40% mAP, using mixture annotations of TagsK and PointsU can save the cost of about 15 hours from the standard detection annotation (25 hours v.s. 40 hours). On CrowdHuman, for the cost of about 330 hours SSOD improves the strong SSOD baseline mAP by ∼4%. These findings support our claim that weak annotations are useful and can achieve a better cost-accuracy trade-off than standard detection annotation.

\[\text{Datasets}\]

<table>
<thead>
<tr>
<th>Datasets</th>
<th>TagsU</th>
<th>TagsK</th>
<th>PointsU</th>
<th>PointsK</th>
<th>BoxesEC</th>
<th>BoxesU</th>
<th>Fully</th>
</tr>
</thead>
<tbody>
<tr>
<td>Bees</td>
<td>-</td>
<td>6.1</td>
<td>6.4</td>
<td>6.4</td>
<td>50</td>
<td>249.9</td>
<td>249.9</td>
</tr>
<tr>
<td>CrowdHuman</td>
<td>-</td>
<td>19.4</td>
<td>20.4</td>
<td>20.4</td>
<td>158.5</td>
<td>792.4</td>
<td>792.4</td>
</tr>
<tr>
<td>VOC</td>
<td>20</td>
<td>21</td>
<td>2.2</td>
<td>22.9</td>
<td>16.8</td>
<td>84</td>
<td>102.6</td>
</tr>
<tr>
<td>COCO</td>
<td>80</td>
<td>84.2</td>
<td>6.9</td>
<td>88.7</td>
<td>53.9</td>
<td>269.5</td>
<td>346</td>
</tr>
<tr>
<td>Objects365</td>
<td>365</td>
<td>375.8</td>
<td>14.2</td>
<td>381.7</td>
<td>110.6</td>
<td>553</td>
<td>913</td>
</tr>
</tbody>
</table>

Table 11. Labeling cost estimation for different annotations (seconds per image).

Some additional interesting observations are also available in Figure 4. First, the green upside-down triangles, rectangles and red/blue lines are all higher than the strong SSOD baseline. This suggests that annotating points (PointsU) and/or Extreme Clicking boxes (BoxesEC) is a better choice than standard complete annotation. Second, the green plus is above the reference on Bees and CrowdHuman, but not for the other three datasets, indicating that count annotation (TagsK) is useful for datasets with dense objects. Finally, weak annotations such as TagsU (green cross), TagsK (green plus) and PointsK (green triangle), are far below the SSOD baseline on datasets like VOC, COCO and Objects365. This suggests that tags are not a good annotation format for datasets with large number of classes, where annotating tags is expensive. In general, the optimal annotation choice is quite dataset-specific, depending on characteristics such as number of categories, number of objects per image, object size, etc.

6. Conclusion

We have proposed a unified framework for omni-supervised object detection, which can use different types of weak annotations. With this unified framework, we have found weak annotations are helpful and a mixture of them can achieve a better cost-accuracy trade-off. Limitations and Potential Negative Social Impact: It is unclear whether these findings of this paper are still consistent on larger datasets, since we have not explored dataset size beyond COCO (~120K images) yet. In addition, Omni-DETR could potentially increase the risk of improper use of detection systems, because it makes good detectors more accessible with fewer annotation efforts.

3Only human annotation costs are considered, and other costs are ignored if there is any.
References

[1] Ben Athiwaratkun, Marc Finzi, Pavel Izmailov, and Andrew Gordon Wilson. There are many consistent explanations of unlabeled data: Why you should average. In ICLR, 2019. 4


[32] Dim P Papadopoulos, Jasper RR Uijlings, Frank Keller, and Vittorio Ferrari. Training object class detectors with click supervision. In CVPR, pages 6374–6383, 2017. 1, 2, 3, 4


[38] Shuai Shao, Zeming Li, Tianyuan Zhang, Chao Peng, Gang Yu, Xiangyu Zhang, Jing Li, and Jian Sun. Objects365: A large-scale, high-quality dataset for object detection. In CVPR, pages 8430–8439, 2019. 1, 2, 3, 6


