# OD-DETR: Online Distillation for Stabilizing Training of Detection Transformer

**Shengjian Wu**[1,2] , **Li Sun**[1,3*] , **Qingli Li**[1]
[1]Shanghai Key Laboratory of Multidimensional Information Processing, East China Normal University
[2]Finvolution Group
[3]Key Laboratory of Advanced Theory and Application in Statistics and Data Science, East China Normal University
sunli@ee.ecnu.edu.cn

## Abstract

DEtection TRansformer (DETR) becomes a dominant paradigm, mainly due to its common architecture with high accuracy and no post-processing. However, DETR suffers from unstable training dynamics. It consumes more data and epochs to converge compared with CNN-based detectors. This paper aims to stabilize DETR training through the online distillation. It utilizes a teacher model, accumulated by Exponential Moving Average (EMA), and distills its knowledge into the online model in following three aspects. *First*, the matching relation between object queries and ground truth (GT) boxes in the teacher is employed to guide the student, so queries within the student are not only assigned labels based on their own predictions, but also refer to the matching results from the teacher. *Second*, the teacher's initial query is given to the online student, and its prediction is directly constrained by the corresponding output from the teacher. *Finally*, the object queries from teacher's different decoder stages are used to build the auxiliary group to accelerate the convergence. For each GT, two queries with the least matching costs are selected into this extra group, and they predict the GT box and participate the optimization. Extensive experiments show that the proposed OD-DETR successfully stabilizes the training, and significantly increases the performance without bringing in more parameters.

## 1 Introduction

Object detection is a fundamental task in computer vision, and has been investigated by the community for decades. CNN-based detectors can be categorized into either anchor-based [Girshick, 2015; Liu *et al.*, 2016] or anchor-free [Tian *et al.*, 2019; Yang *et al.*, 2019] methods. The former builds upon the sliding anchor boxes, and can be designed into a single, two or multi-stage, while the latter only has grid point assumption and is generally of a single stage. Although, CNN-based detectors has achieved impressive performance, it needs to determine complex meta parameters like anchor shape and size, threshold for positive and negative samples, and non-maximum suppression (NMS) post-processing.

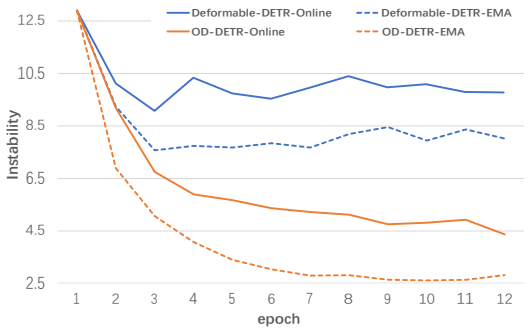


Figure 1: The matching instability curves for the first 12 training epochs. We compare our OD-DETR and its EMA version with Def-DETR. The metric, introduced by [Li *et al.*, 2022], is calculated on COCO VAL2017. Lower value means more stable matching. As is expected, EMA's instability is much lower than the online model. In OD-DETR, the online student learns from its EMA teacher, which greatly increases its stability. The improved online model also helps to stabilize the EMA's matching results.

DEtection TRansformer (DETR) greatly simplifies the cumbersome design. It employs the encoder attention to enhance image feature. At the same time, several decoder attention layers are also incorporated, which translate the initial parameters of learnable object queries into predicted boxes. DETR uses bipartite matching to setup the one-to-one relation between ground truths (GTs) and predictions from queries, therefore, one GT is assigned to a single query, and vice versa. The one-to-one matching scheme reduces the redundant predictions, and alleviates detector from NMS. However, DETR is often blamed for its unstable training and slow convergence. As is shown in Fig. 1, the GT for query is often switched during training. Many efforts intend to improve it, by either introducing local box prior [Meng *et al.*, 2021; Zhu *et al.*, 2021; Liu *et al.*, 2022], more query groups [Li *et al.*, 2022; Chen *et al.*, 2022a; Jia *et al.*, 2023], an initial stage [Yao *et al.*, 2021; Zhang *et al.*, 2023] or improved quality-aware loss functions [Liu *et al.*, 2023; Cai *et al.*, 2023].

This paper proposes a solution, named online distillation

*Corresponding Author

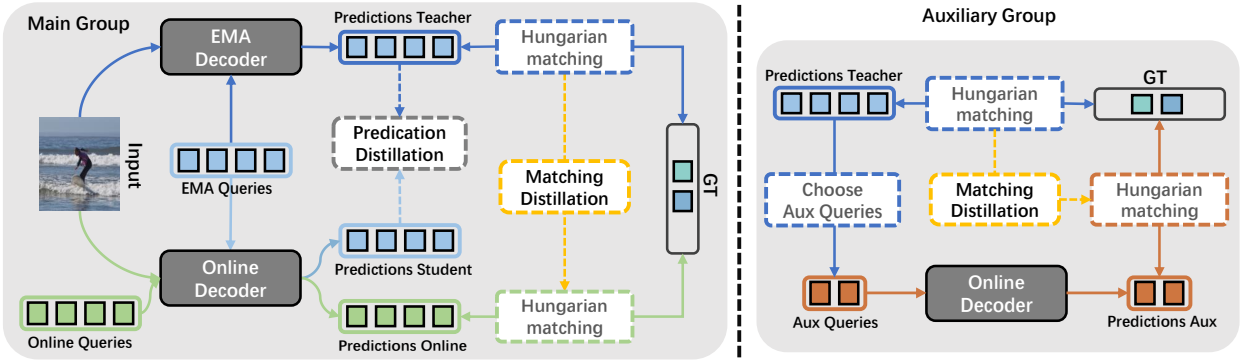


Figure 2: The overall architecture of OD-DETR. In the Main Group, the EMA model's queries are given to the decoder to create Predictions Teacher, and they are matched with the GT set using Hungarian matching. Simultaneously, these queries are also input into the Online Decoder to produce Predictions Student, and they directly learn from Predictions Teacher through prediction distillation. The Online model's own queries are decoded into Predictions Online, which are also matched with the GT set. Then, through matching distillation, they refer to matching result of Predictions Teacher. In the Auxiliary Group, we select two updated queries with the lowest matching cost for each GT from the Predictions Teacher. These selections are added as an extra group to the Online Decoder.

(OD-DETR), in another perspective to stabilizing the training of DETR. Inspired by the works in semi-supervised classification [Sohn *et al.*, 2020] and object detection [Liu *et al.*, 2021], we utilize Exponential Moving Average (EMA) as the teacher and distill its knowledge into the student model in an online manner. Unlike traditional distillation, the teacher also gets improved by the accumulation of the student. Particularly, we leverage the predicted bounding boxes, Hungarian matching result and updated object queries from the teacher model, and design schemes for prediction distillation, matching distillation and building auxiliary group within the student. For matching distillation, we assign another label for each object query through Hungarian matching according to cost matrix between teacher's predictions and GT boxes, and use it to guide the online student together with its original matching results. To employ two possible matched GTs, we propose a multi-target QFL loss to accommodate two labels from different classes, while keep only one regression target to avoid ambiguity. Meanwhile, two predictions associated with the same GT are given different regression loss weights, and the one with larger matching cost is down-weighted.

To take full advantage of the teacher, its initial queries are fed into the online student, giving predictions that can be directly constrained by the corresponding output from the teacher with no need to rematch. We name this simple constraint between two bounding boxes as prediction distillation. To further strengthen the link between teacher and student, we build independent augmented groups by object queries from each decoding stage of the teacher. Here, high quality queries with minimal matching costs for each GT are selected, and each group is given to the student's decoder to make predictions, re-match with GTs and compute losses. The augmented query groups are mainly used for speeding up the training convergence, hence it is abandoned during inference. The framework of OD-DETR is given in Fig. 2.

To verify the effectiveness of OD-DETR, we perform a large amount of experiments on MS-COCO [Lin *et al.*, 2014]. Particularly, OD-DETR is implemented to support different variations of DETR including Def-, DAB- and DINO. We find that our method is compatible with all of them and increases the performance obviously. In summary, the contributions of this paper lie in following aspects.

- We propose an EMA-teacher based online distillation scheme. Particularly, matching distillation between the teacher and student is proposed. It assigns each query to an extra GT box based on the matching results of teacher's prediction. Moreover, we adapt the loss functions and propose a multi-target QFL classification loss and a cost sensitive regression loss.
- We perform prediction distillation and build augmented query groups to fully utilize of the EMA teacher. Prediction distillation is carried out by feeding the EMA query into student's decoder, and constraining the prediction by teacher's output, while the augmented groups are built by high-quality queries from different decoding stages within the teacher.
- Intensive experiments are carried out, showing that the proposed OD-DETR is able to effectively increase the performance of different DETR's variations.

## 2 Related Work

### 2.1 Supervised Object Detection

Modern object detection models mainly use convolutional networks and achieved great success recently. These CNN-based detectors fall into two categories: anchor-based and anchor-free. [Girshick, 2015; Ren *et al.*, 2015; Lin *et al.*, 2017] are some well-known anchor-based model, while [Tian *et al.*, 2019; Yang *et al.*, 2019; Duan *et al.*, 2019] represent anchor-free models. Both both of them need manual components like non-maximum suppression (NMS) and heuristic label assignment rules.

DEtection TRansformer (DETR) [Carion *et al.*, 2020] changes this scenario. It's the first end-to-end, query-based object detector without handcrafted designs such as anchors

and NMS. However, it suffers from the slow training convergence. Several recent studies have focused on speeding up DETR's training process. Methods such as [Zhu *et al.*, 2021; Gao *et al.*, 2021; Gao *et al.*, 2021; Sun *et al.*, 2020; Zhao *et al.*, 2023b; Zhao *et al.*, 2023a; Liu *et al.*, 2022] employ local box priors to concentrate on local features, reducing the search space. Other methods like [Chen *et al.*, 2022a; Jia *et al.*, 2023; Li *et al.*, 2022; Zhang *et al.*, 2023] speed up training by adding more query groups, providing additional positive samples for ground truth boxes.

### 2.2 EMA in SSOD and Distillation

Using EMA model as teacher to distill knowledge into student has been consistently applied in semi-supervised learning (SSL) based classifications, examples of which include [Zhu, 2005; Laine and Aila, 2017; Tarvainen and Valpola, 2017; Berthelot *et al.*, 2019; Sohn *et al.*, 2020]. A critical challenge in SSL is to fully leverage unlabeled images, which is realized by either self-training or consistency regularization. Another area where EMA-based teacher approaches is applied is in self-supervised learning, as exemplified by methods like [Grill *et al.*, 2020; Gidaris *et al.*, 2020; Caron *et al.*, 2021]. The key to self-supervised learning also lies in the student model learning from the teacher model's outputs on unlabeled data.

Based on the similar idea in classification task, numerous SSOD (Semi-Supervised Object Detection) models have been developed. For instance, [Tang *et al.*, 2021; Xu *et al.*, 2021; Liu *et al.*, 2021] adopt the principles of FixMatch [Sohn *et al.*, 2020] and employs an EMA accumulated from the online student to provide pseudo GT boxes. However, for knowledge distillation (KD) task in object detection, current methods typically use a fixed, pre-trained model as the teacher to output labels, which are then provided to the online student for learning, as can be seen in [Chen *et al.*, 2017; Li *et al.*, 2017; Chang *et al.*, 2023; Chen *et al.*, 2022b; Wang *et al.*, 2022]. Different from works in SSOD or KD, this paper focuses on online supervised learning for detection, and we utilize the EMA model as the teacher to provide guidance in an online manner, without requiring a fixed teacher model optimized in advance.

## 3 Method

### 3.1 Preliminaries on DETR and Its Adaptation

DETR is composed of a backbone, an encoder of several self-attention layers, a set of learnable object queries, and a decoder followed by a detection head to translate updated queries into boxes with class predictions. We define $\mathcal{Q} = \{q_i | q_i \in \mathbb{R}^c\}$ as the query set. Each $q_i$ is a learnable parameters at the beginning stage. To make the model aware of the positions of bounding boxes, the enhanced versions of DETR [Liu *et al.*, 2022; Zhu *et al.*, 2021] explicitly encode the bounding box $(x, y, w, h)$ or only box center $(x, y)$ into positional embedding, specifying a set $\mathcal{P} = \{p_i | p_i \in \mathbb{R}^c\}$ corresponding to $\mathcal{Q}$. $\mathcal{P}$, $\mathcal{Q}$ and image feature $\mathcal{F}$ are given to the decoder $Dec$, resulting in an updated query set $\mathcal{Q}$ and box set $\mathcal{B}$ for the next stage.

$$[\mathcal{Q}^{t+1}, \mathcal{B}^{t+1}] = Dec^t(\mathcal{Q}^t, \mathcal{P}^t, \mathcal{F}; \theta) \quad (1)$$

In Eq. (1), $Dec$ indicates the decoder parameterized by $\theta$, and the superscript denotes the decoder stage. The predicted box set $\mathcal{B} = \{b_i | b_i = (x, y, w, h, c)\}$ not only has box coordinate but also a class score vector $c$. Apart from the detection head, the DETR's decoder consists of self attention layer computed by interactions among query elements $q_i$ in $\mathcal{Q}$, cross attention layer interacting between $\mathcal{Q}$ and $\mathcal{F}$, and feed-forward network between them. In Def-DETR [Zhu *et al.*, 2021], the cross attention can be improved by only sampling features around the reference point and giving a weighted average, and this local prior speeds up its convergence.

The updated $\mathcal{Q}$ are given to the detection head for classification and bounding box regression. But before loss computation, the predictions from queries are assigned to GTs according to matching costs. DETR adopts the Hungarian algorithm to set up the one-to-one relation between query set $\mathcal{Q}$ and all GTs, which is also the key reason that DETR can be free from NMS. However, this dynamic one-to-one matching strategy also leads to training instability. The work [Li *et al.*, 2022] finds that there are some queries assigned with different GTs between two training epochs, as is shown in the blue curve in Fig. 1, and it causes the slow convergence of DETR.

The training target of original DETR includes a focal loss $L_{cls}$ [Lin *et al.*, 2017] for classification, and $L_1$ and GIoU loss $L_{GIoU}$ for regression. Some works show that quality metric can improve performance. Notably, with quality focal loss $L_{QFL}$ [Li *et al.*, 2020] defined in Eq. (2), classification and regression tasks are bounded together. Here $t$ is the IoU target calculated between predicted box $b$ and its matched GT box. $s$ is a prediction score from sigmoid function.

$$L_{QFL}(s, t) = -|t - s|^\gamma((1 - t)\log(1 - s) + t\log(s)) \quad (2)$$

Our proposed OD-DETR is built on enhanced DETR with QFL classification loss. Next, we introduce three key components: matching distillation, prediction distillation, and auxiliary group, as is shown in Fig.2.

### 3.2 Matching Distillation

Inspired by the success of the EMA model, we take advantage of it to stabilize DETR training. We first validate its potential application in Def-DETR. Specifically, we train it in 12 epochs and make an EMA at the same time. We compare it with the online model based on the instability metrics defined in [Li *et al.*, 2022]. As is shown in Fig. 1, the behaviour of EMA is more stable than the online model, showing that it indeed prevents the label switch between two epochs, therefore giving a stable matching result. Considering that the EMA model has a more stable matching result, we intend to distill it into the online student.

Fig.3 presents an intuitive illustration of matching distillation. Matching result from the teacher and the student are combined into classification and regression losses for training the online model. Particularly, given teacher's query, PE and image feature $\mathcal{Q}'$, $\mathcal{P}'$ and $\mathcal{F}'$, the decoder of the teacher, parameterized by $\theta'$, can output predicted boxes $\mathcal{B}'$, as in Eq. (1). Hungarian matching result between $\mathcal{B}'$ and the GT set is used as a reference for constraining the prediction $\mathcal{B}$ from student model. Note that $\mathcal{B}$ also have a matching result,

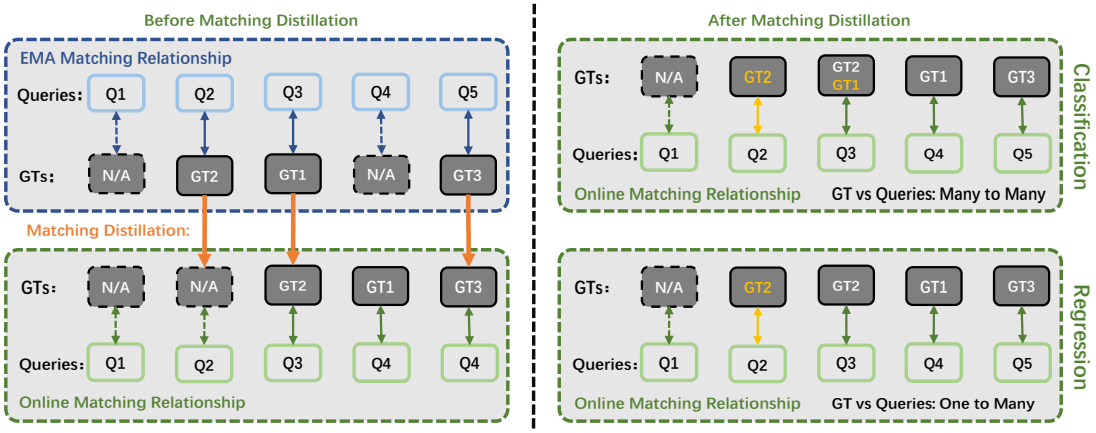


Figure 3: Matching Distillation injects GTs matched in the EMA teacher into the corresponding queries in the online student. On the left, two independent matching are first carried out, assigning GTs for queries in the EMA and online models, respectively. On the right, new matches for online model are shown in gold color. For classification, this creates many-to-many matches, where queries like Q3 can match with GT2 and GT1. But for regression, to avoid confusion, each query is assigned with one GT, making a one-to-many match between GTs and queries.

which means that $q_i$ and its EMA version $q_i'$ may have different GT to be matched. We now illustrate matching distillation for classification and regression in the following two sections.

**Matching distillation for classification.** Since the online query $q_i$ are tightly connected with $q_i'$, we design a **multi-target QFL** $L_{MQF}$ for class prediction in Eq. (3), which utilizes the matched target label of $q_i'$, if it is different from the original target. Here $s'$ is another predicted class score, and $t'$ is its corresponding IoU target computed based on the teacher's matching result.

$$L_{MQF}(s, s', t, t') = \begin{cases} L_{QFL}(s, t) & \text{one class} \\ L_{QFL}(s, t) + L_{QFL}(s', t') & \text{two classes} \end{cases} \tag{3}$$

Particularly, if two matched GT are from the same class, $L_{MQF}$ becomes the same with $L_{QFL}$ defined in Eq. (2), computed by its original IoU target $t$. Otherwise, if two GTs are in different classes, two IoU targets $t$ and $t'$ impose constraints on $s$ and $s'$, respectively. Fig.4 shows how the multi-target label is set for a query's prediction that matches two different categories of GTs. In dense scenes, one predicted box often contains multiple objects of different categories. Therefore, the single-category one-hot label is not appropriate. Our $L_{MQF}$ dynamically sets targets for different categories based on the IoU between the predicted box and two matched GTs, providing richer semantic information for training. It also increases training stability by preventing sudden label changes when online matching result changes.

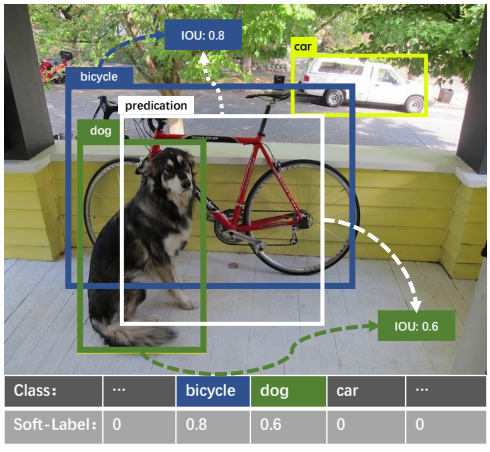


Figure 4: Illustration for setting the multi-target label. The prediction shown here matches two objects of different classes, the dog and the bicycle, with IOUs of 0.6 and 0.8, respectively. In its label vector, two entries for dog and bicycle are set accordingly. Other category elements stay at 0.

**Matching distillation for regression.** For bounding box regression, the matching result from the teacher are also referred to. But we avoid the ambiguity due to the case in which the two different GT boxes are matched with one query, and only use the online matched result as the target to compute $L_1$ and $L_{GIoU}$. Moreover, since one GT may still be matched by two queries, we simply **down-weight** the regression loss for the query with a bigger matching cost. The regression loss $L_r$ is defined in Eq. (4). Here $b$ and $b_{gt}$ are the predicted and matched GT boxes. $w_d$ is a hyper-parameter with value of 0.51 following the idea in [Cai *et al.*, 2023].

$$L_r(b, b_{gt}) = \begin{cases} L_1(b, b_{gt}) + L_{GIoU}(b, b_{gt}) & b \text{ with lower cost} \\ w_d[L_1(b, b_{gt}) + L_{GIoU}(b, b_{gt})] & b \text{ with higher cost} \end{cases} \tag{4}$$

### 3.3 Prediction Distillation

Besides the matching distillation, the output from the teacher model $\mathcal{B}'$ can also be exploited for training, and we name it prediction distillation. However, since the online predictions are already constrained by GT boxes, it is ambiguous to require them to approach another set of targets. To take full advantage of $\mathcal{B}'$, we feed the EMA queries $\mathcal{Q}'$ together with the PE $\mathcal{P}$ and image feature $\mathcal{F}$ into the student decoder, and obtain output $\hat{\mathcal{B}}$ as is in Eq. (1). Note that $\mathcal{B}'$ is different from $\hat{\mathcal{B}}$. The former are totally from the teacher and work

as a target set, while the latter are predictions which need to be constrained. Moreover, $\mathcal{B}'$ and $\hat{\mathcal{B}}$ are explicitly associated since they all begin from the same $\mathcal{Q}'$. Therefore, prediction distillation loss $L_{pd}$ can be directly computed between them.

Here a naive way is to adopt $L_{QFL}$ defined in Eq. (2) but replace the IoU target $t$ by the class score $c'$ from the teacher, which is a strategy for distillation under a fixed teacher [Chen *et al.*, 2022b]. However, the predictions from the EMA teacher are not very accurate in our case, especially during the early training stages. Applying it in the naive way can introduce many errors from the teacher itself, which lead to worse results. We adapt it so it becomes suitable for online distillation. First, since $\mathcal{B}'$ have a matching result with GT boxes, an implicit class label index $c_g$ for each box in $\mathcal{B}'$ can be inferred. Then we modify the predicted score vector $c'$ at the corresponding entry according to Eq. (5).

$$c'[c_g] \leftarrow (c'[c_g])^{\alpha} \cdot (IoU')^{\beta} \tag{5}$$

Here $IoU'$ is computed between bounding box in $\mathcal{B}'$ and its matched GT. $\alpha$ and $\beta$ are two hyper-parameters. Following [Feng *et al.*, 2021], we set $\alpha = 0.25$ and $\beta = 0.75$. After update on $c'$, we then replace $t$ in Eq. (2) by it, therefore, giving $L^{cls}_{distill}$ for prediction distillation. For regression task, $c'[c_g]$ is also used as a weight. Using $c'[c_g]$ in $L^{r}_{distill}$ and $L^{cls}_{distill}$ is named as **TOOD-weight**, which is first proposed in [Feng *et al.*, 2021] for one-stage detector. Note that $c'[c_g]$ is computed from the teacher, which gives the key difference with [Feng *et al.*, 2021]. Consequently, $L^{r}_{distill} = c'[c_g](L_1 + L_{GIoU})$. $L^{cls}_{distill}$ and $L^{r}_{distill}$ are combined into $L_{pd}$ in Eq. (6).

$$L_{pd} = \mathbb{1}(IoU' > IoU) \cdot L^{r}_{distill} + L^{cls}_{distill} \tag{6}$$

Here $\mathbb{1}$ denotes an indicator function, and it only equals to one when the predictions from the teacher have a larger IoU with GT box than the student, otherwise, it returns 0, meaning that $L^{r}_{distill}$ is not considered in this case. This approach is named as **Listen2stu**.

### 3.4 Auxiliary Group

To better utilize the EMA teacher and enhance training stability, we choose some updated queries $\tilde{q}$ and corresponding predicted boxes $\tilde{b}$ from $(\mathcal{Q}')^t$ and $(\mathcal{B}')^t$ at $t$-th stage of the teacher's decoder. Then we use them as independent initial queries and anchors for PE. They are fed into the first decoding stage of online model, providing more positive examples for learning. To reduce the computational load of the auxiliary group, we select only the top two queries with the lowest matching cost to each GT from each decoder stage. This method ensures that the selected queries include both positive examples and challenging negative ones. Note that queries from the same decoding stage forms an independent group. Predictions from each group is matched with GT set in one-to-one manner.

Matching distillation is also used in each auxiliary group, just as we do in main group. This method combines the original matching from the teacher with the new ones, which enhances the training stability for auxiliary group. So the loss in auxiliary group can be denoted by $L_{aux} = \tilde{L}_{MQF} + \tilde{L}_r$. In summary, the total training loss is shown in Eq. (7).

$$L_{total} = L_{MQF} + L_r + L_{pd} + L_{aux} \tag{7}$$

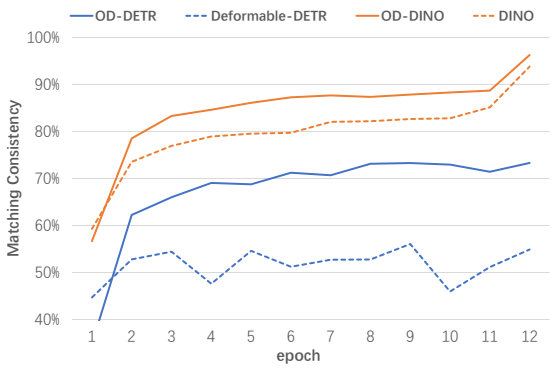


Figure 5: The matching consistency between the EMA and online model. The curve shows the percentage of queries matched with the same GT during training. In both our OD-DETR and OD-DINO models, this ratio is significantly higher compared to Def-DETR and DINO.

## 4 Experiments

### 4.1 Settings

**Datasets.** We conduct all our experiments on MS-COCO [Lin *et al.*, 2014] 2017 dataset and evaluate the performance of our models on validation dataset by using mean average precision (mAP) metric. The COCO dataset contains 117K training images and 5K validation images.

**Implementation details.** We use Def-DETR (with iterative bounding box refinement), DAB-Def-DETR, and DINO as our baseline methods. Our experiments are conducted over 12 (1x) and 24 (2x) epochs on 8 GPUs. Learning rate settings for OD-DETR are identical to those of Def-DETR, with a learning rate of $2 \times 10^{-5}$ for the backbone and $2 \times 10^{-4}$ for the Transformer encoder-decoder framework, coupled with a weight decay of $2 \times 10^{-5}$. The learning rates and batch sizes for OD-DAB-DETR and OD-DINO follow their respective baselines. We set the EMA decay value at 0.9996.

### 4.2 Main Results

As shown in Table 1, we firstly use Def-DETR [Zhu *et al.*, 2021] as baseline, and compare our OD-DETR on COCO val2017 dataset with other competitive DETR variants. Our OD-DETR achieved a significant improvement, reaching 47.7 AP. It notably surpassed the baseline (Def-DETR iterative bounding box refinement) trained for 50 epochs, with a substantial increase of 2.3 in AP after just 2x training.

The OD-DAB-DETR model, improving upon the DAB-Def-DETR [Liu *et al.*, 2022], achieved 50.2 AP. It exceeds the baseline by 1.6 AP after just 2x training and outperforms DN-Def-DETR [Li *et al.*, 2022] with 50 epochs of training.

Under both 1x and 2x schedulers, our OD-DINO consistently outperforms the baseline DINO [Zhang *et al.*, 2023], achieving 50.4 AP with 1x and 51.8 AP with 2x scheduling, marking a 1.4 increase in AP. Our OD-DINO series outperform other DINO-based approaches, including Stable-DINO [Liu *et al.*, 2023], Align-DETR [Cai *et al.*, 2023] and so on.

| Method | Backbone | epochs | AP | $AP_{50}$ | $AP_{75}$ | $AP_S$ | $AP_M$ | $AP_L$ |
|---|---|---|---|---|---|---|---|---|
| Def-DETR+ | R50 | 50 | 45.4 | 64.7 | 49.0 | 26.8 | 48.3 | 61.7 |
| Align-DETR | R50 | 50 | 46.0 | 64.9 | 49.5 | 25.2 | 50.5 | 64.7 |
| OD-DETR (ours) | R50 | 24 | 47.7 (+2.3) | 65.2 | 51.4 | 29.8 | 51.5 | 63.0 |
| DAB-Def-DETR* | R50 | 50 | 48.6 | 67.1 | 52.8 | 31.8 | 51.5 | 64.1 |
| DN-DAB-Def-DETR | R50 | 50 | 49.4 | 67.5 | 53.8 | 31.2 | 52.5 | 65.1 |
| OD-DAB-DETR (ours) | R50 | 24 | 50.2 (+1.6) | 67.6 | 54.8 | 33.9 | 54.0 | 65.0 |
| H-Def-DETR | R50 | 12 | 48.7 | 66.4 | 52.9 | 31.2 | 51.5 | 65.0 |
| DINO-4scale | R50 | 12 | 49.0 | 66.6 | 53.5 | 32.0 | 52.3 | 63.0 |
| DINO-4scale | R50 | 24 | 50.4 | 68.3 | 54.8 | 33.3 | 53.7 | 64.8 |
| DINO-5scale | R50 | 12 | 49.4 | 66.9 | 53.8 | 32.3 | 52.5 | 63.9 |
| Align-DETR | R50 | 12 | 50.2 | 67.8 | 54.4 | 32.9 | 53.3 | 65.0 |
| Align-DETR | R50 | 24 | 51.3 | 68.2 | 56.1 | 35.5 | 55.1 | 65.6 |
| Group-DETR-DINO-4scale | R50 | 36 | 51.3 | - | - | 34.7 | 54.5 | 65.3 |
| Stable-DINO-4scale | R50 | 12 | 50.4 | 67.4 | 55.0 | 32.9 | 54.0 | 65.5 |
| Stable-DINO-4scale | R50 | 24 | 51.5 | 68.5 | 56.3 | 35.2 | 54.7 | 66.5 |
| Stable-DINO-5scale | R50 | 12 | 50.5 | 66.8 | 55.3 | 32.6 | 54.0 | 65.3 |
| OD-DINO-4scale (ours) | R50 | 12 | 50.4 (+1.4) | 67.4 | 55.2 | 32.9 | 54.1 | 65.6 |
| OD-DINO-4scale (ours) | R50 | 24 | 51.8 (+1.4) | **69.3** | 56.6 | 34.5 | **55.1** | **66.9** |
| OD-DINO-5scale (ours) | R50 | 12 | **52.0** (+2.6) | 68.8 | **56.9** | **35.4** | 54.8 | 66.8 |

Table 1: Results for three versions of Our OD-DETR and other detection models using ResNet50 backbone on COCO val2017 dataset. Def-DETR+ indicates Def-DETR with iterative bounding box refinement. DAB-Def-DETR* is the optimized implementation with deformable attention in both encoder and decoder, which is better than the version in paper. H-Def-DETR is the model with deformable attention in Hybrid Matching [Jia *et al.*, 2023].

| Method | MD | PD | AG | AP | $AP_{50}$ | $AP_{75}$ |
|---|---|---|---|---|---|---|
| 0 (baseline) | | | | 45.4 | 64.7 | 49.0 |
| 1 | ✓ | | | 46.8 | 64.6 | 50.7 |
| 2 | ✓ | ✓ | | 47.3 | 65.0 | 51.4 |
| 3 | ✓ | | ✓ | 47.2 | 65.0 | 51.2 |
| 4 | ✓ | ✓ | ✓ | **47.7** | **65.2** | **51.4** |

Table 2: Ablations on all designed components. MD stands for matching distillation. PD refers to prediction distillation. AG means auxiliary group built by updated queries and boxes.

| Method | AP | $AP_{50}$ | $AP_{75}$ |
|---|---|---|---|
| QFL w/o MD | 45.8 | 63.6 | 49.8 |
| Conditional MD | 45.9 | 64.1 | 49.7 |
| QFL MD | **46.4** | **64.1** | **50.5** |
| QFL + Conditional MD | 46.3 | 64.1 | 49.9 |

Table 3: Comparison on various matching distillation methods for classification. QFL w/o MD means only applying QFL without MD. QFL MD refers to our MD combined with $L_{MQF}$. Condition MD is introduced in [Teng *et al.*, 2023].

## 4.3 Ablation Study

We conduct a series of ablation studies to assess the effectiveness of the components. All experiments in this section are based on the Def-DETR model with an R50 backbone and follow a standard 2x training schedule.

### Ablation Study on Components

In this section, we perform a set of ablation studies on key components: matching distillation (MD), prediction distillation (PD), and auxiliary group (AG), to assess their individual effectiveness. The findings are detailed in Table 2. These studies reveal that each component plays a significant role in enhancing performance. Specifically, MD contributes an increase of 1.4 AP, PD adds 0.5 AP, and AG contributes 0.4 AP. Cumulatively, these components lead to an overall improvement of 2.3 AP over the baseline model.

Fig.5 shows that our methods improves the consistency of matching GTs between the online model's queries and the EMA model's queries. As the online model's matching improves with training, it also boosts the EMA model's stability. This two-way improvement creates a strong combined effect.

### Comparisons of Different Matching Distillation Methods

Experiments in Table 3 focuses on different matching distillation methods for only classification. The first row shows that employing only QFL [Li *et al.*, 2020] without MD results in a significantly lower performance, achieving 45.8 AP, compared to our MD method combined with $L_{MQF}$ (QFL MD), which reaches 46.4 AP. This difference highlights the significant contribution of MD beyond the impact of QFL alone. Conditional MD, introduced in the StageInteractor [Teng *et al.*, 2023], is a technique that allows a query to match with multiple ground truths. This method involves verifying if the additional query-GT matches have an IoU above 0.5, discarding those that don't meet this criterion. The performance of it is 45.9 AP. Integrating $L_{MQF}$ with Conditional MD under the label QFL + Conditional MD doesn't enhance performance. The performance is 46.3 AP. This observation implies that $L_{MQF}$ and MD are inherently synergistic. $L_{MQF}$ eliminates the necessity of manually setting thresholds like in Conditional MD for Hungarian matching.

| Method | Cls | Reg | AP | $AP_{50}$ | $AP_{75}$ |
|---|---|---|---|---|---|
| 1 | w/o $w_d$ | N/A | 46.4 | 64.1 | 50.5 |
| 2 | w/o $w_d$ | w/o $w_d$ | 46.6 | 64.5 | 50.6 |
| 3 | w/o $w_d$ | w/ $w_d$ | **46.8** | **64.6** | **50.7** |
| 4 | w/ $w_d$ | w/ $w_d$ | 46.7 | 64.3 | 50.5 |

Table 4: Analyzing the impact of using down-weight ($w_d$) in MD. N/A in the first line indicates that MD was not applied for regression.

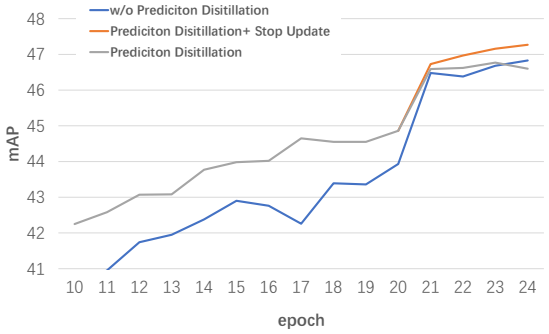


Figure 6: The Stop Update strategy extends the advantage of PD from before the learning rate decay to after it. This ensures that PD ultimately achieves better performance than methods without PD.

**Analyzing Down-Weight Impact in Matching Distillation**

In Method 1 from Table 4, MD is only used for classification, keeping a one-to-one match for regression, resulting in a lower AP of 46.4 (QFL MD in Table 3). This implies that MD is beneficial for regression. Method 3, which applies down-weight ($w_d$) only for regression, achieves the highest AP of 46.8. This confirms the effectiveness of down-weighting regression loss. It also demonstrates QFL MD's effectiveness in handling many-to-many matching in classification. It eliminates the need to adjust loss for GTs that match multiple queries.

**Comparisons of Various Prediction Distillation Methods**

In Table 5, w/o PD means using MD alone without PD, and it's also referred to Method 1 in Table 2. Naive distillation applies distillation evenly across all queries, but this method only achieves 44.2 AP, lower than w/o PD (46.8 AP). This suggests that simply adding online distillation doesn't work. Using TOOD-weight and Listen2stu methods led to a notable increase of 2.3 AP, reaching 46.6 AP. As Fig.6 shows, PD notably enhances performance before learning rate decay, but tends to have a negative impact after the learning rate decay. Thus, we adopt a **Stop Update** strategy, halting updates to the EMA teacher's weights after learning rate decay. This method eventually achieves 47.3 AP. We also test Query-prior Assignment Distillation from DETRDistill [Chang *et al.*, 2023]. It feeds the teacher's query into the student's decoder as an extra group to learn GT. The outcome, at 46.9 AP, is less effective than our PD approach.

**Experiments on Applying MD to the Auxiliary Group**

Table 6 shows results for different matching methods in auxiliary group. Using either the EMA model's original matching or re-matching gives 47.0 AP and 46.7 AP, respectively. The highest AP, 47.2, comes from using MD for auxiliary group (Method 3 in Table 2), just like in the main group. This proves MD works well in both main and auxiliary group.

| Method | AP | $AP_{50}$ | $AP_{75}$ |
|---|---|---|---|
| w/o PD | 46.8 | 64.6 | 50.7 |
| Naive Distillation | 44.2 | 62.6 | 47.7 |
| TOOD-weight | 46.5 | 64.4 | 50.4 |
| TOOD-weight + Listen2stu | 46.6 | 64.4 | 50.6 |
| TOOD-weight + Listen2stu + Stop Update | **47.3** | **65.0** | **51.4** |
| Query-prior Assignment Distillation | 46.9 | 64.5 | 50.9 |

Table 5: Comparisons of various methods of PD. w/o PD means that only MD is used, and PD is not utilized. Stop Update refers to ceasing the update of the teacher model after learning rate decay.

| Method | AP | $AP_{50}$ | $AP_{75}$ |
|---|---|---|---|
| Original Matching | 46.7 | 64.1 | 50.8 |
| Re-Matching | 47.0 | 64.5 | 50.9 |
| MD | **47.2** | **65.0** | **51.2** |

Table 6: Experiments on applying MD to the auxiliary group. Original Matching uses the original matching results from the EMA model. Re-Matching involves utilizing a new Hungarian matching results during the training of the auxiliary group. MD combines the two match results.

**Comparative Results of EMA Models**

Table 7 compares the EMA model performances of DINO-4scale and our OD-DINO-4scale, both trained using a 2x setting. Our OD-DETR-EMA outperforms DINO-EMA by 1.3 AP. DINO's EMA version achieves 50.7 AP, which is lower than the the online version of OD-DINO (51.8 AP). This indicates that our methods not only improves the performance of the online student but also enhances the EMA teacher itself. A better teacher then further improves the student, creating a beneficial cycle.

| Method | AP | $AP_{50}$ | $AP_{75}$ |
|---|---|---|---|
| DINO-EMA | 50.7 | 68.4 | 55.2 |
| OD-DINO-EMA | **52.0** (+1.3) | **69.4** | **56.8** |

Table 7: Performance comparison of EMA models.

## 5 Conclusion

This paper proposes an OD-DETR, which is an online distillation method to stabilize the training of DETR. We find that the EMA model accumulated during training provides not only high-quality predicted boxes, but also the query-GT matching result and extra query groups. Under the help of EMA model, we improve the online training through prediction distillation, matching distillation and auxiliary query group. We show that the proposed OD-DETR can steadily increase the performance across different DETR variations.

## Acknowledgments

This work is supported by the Science and Technology Commission of Shanghai Municipality under Grant No. 22511105800, 19511120800 and 22DZ2229004.

## References

[Berthelot *et al.*, 2019] David Berthelot, Nicholas Carlini, Ian Goodfellow, Nicolas Papernot, Avital Oliver, and Colin A Raffel. Mixmatch: A holistic approach to semi-supervised learning. *Advances in neural information processing systems*, 32:5050–5060, 2019.

[Cai *et al.*, 2023] Zhi Cai, Songtao Liu, Guodong Wang, Zheng Ge, Xiangyu Zhang, and Di Huang. Align-detr: Improving DETR with simple iou-aware BCE loss. *CoRR*, abs/2304.07527, 2023.

[Carion *et al.*, 2020] Nicolas Carion, Francisco Massa, Gabriel Synnaeve, Nicolas Usunier, Alexander Kirillov, and Sergey Zagoruyko. End-to-end object detection with transformers. In *Proceedings of the European conference on computer vision (ECCV)*, pages 213–229. Springer, 2020.

[Caron *et al.*, 2021] Mathilde Caron, Hugo Touvron, Ishan Misra, Herv'e J'egou, Julien Mairal, Piotr Bojanowski, and Armand Joulin. Emerging properties in self-supervised vision transformers. *2021 IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 9630–9640, 2021.

[Chang *et al.*, 2023] Jiahao Chang, Shuo Wang, Hai-Ming Xu, Zehui Chen, Chenhongyi Yang, and Feng Zhao. Detrdistill: A universal knowledge distillation framework for detr-families. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 6875–6885. IEEE, 2023.

[Chen *et al.*, 2017] Guobin Chen, Wongun Choi, Xiang Yu, Tony X. Han, and Manmohan Chandraker. Learning efficient object detection models with knowledge distillation. In *Advances in neural information processing systems*, pages 742–751, 2017.

[Chen *et al.*, 2022a] Qiang Chen, Xiaokang Chen, Gang Zeng, and Jingdong Wang. Group DETR: fast training convergence with decoupled one-to-many label assignment. *CoRR*, abs/2207.13085, 2022.

[Chen *et al.*, 2022b] Xiaokang Chen, Jiahui Chen, Y. Liu, and Gang Zeng. D3etr: Decoder distillation for detection transformer. *ArXiv*, abs/2211.09768, 2022.

[Duan *et al.*, 2019] Kaiwen Duan, Song Bai, Lingxi Xie, Honggang Qi, Qingming Huang, and Qi Tian. Centernet: Keypoint triplets for object detection. *2019 IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 6568–6577, 2019.

[Feng *et al.*, 2021] Chengjian Feng, Yujie Zhong, Yu Gao, Matthew R. Scott, and Weilin Huang. TOOD: task-aligned one-stage object detection. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 3490–3499. IEEE, 2021.

[Gao *et al.*, 2021] Peng Gao, Minghang Zheng, Xiaogang Wang, Jifeng Dai, and Hongsheng Li. Fast convergence of DETR with spatially modulated co-attention. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 3601–3610. IEEE, 2021.

[Gidaris *et al.*, 2020] Spyros Gidaris, Andrei Bursuc, Gilles Puy, Nikos Komodakis, Matthieu Cord, and Patrick Pérez. Obow: Online bag-of-visual-words generation for self-supervised learning. *2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 6826–6836, 2020.

[Girshick, 2015] Ross Girshick. Fast r-cnn. In *Proceedings of the IEEE international conference on computer vision*, pages 1440–1448, 2015.

[Grill *et al.*, 2020] Jean-Bastien Grill, Florian Strub, Florent Altché, Corentin Tallec, Pierre Richemond, Elena Buchatskaya, Carl Doersch, Bernardo Avila Pires, Zhaohan Guo, Mohammad Gheshlaghi Azar, et al. Bootstrap your own latent-a new approach to self-supervised learning. *Advances in neural information processing systems*, 33:21271–21284, 2020.

[Jia *et al.*, 2023] Ding Jia, Yuhui Yuan, Haodi He, Xiaopei Wu, Haojun Yu, Weihong Lin, Lei Sun, Chao Zhang, and Han Hu. Detrs with hybrid matching. In *Proceedings of the IEEE/CVF CVPR*, pages 19702–19712. IEEE, 2023.

[Laine and Aila, 2017] Samuli Laine and Timo Aila. Temporal ensembling for semi-supervised learning. In *5th International Conference on Learning Representations, ICLR 2017, Toulon, France, April, 2017, Conference Track Proceedings*. OpenReview.net, 2017.

[Li *et al.*, 2017] Quanquan Li, Shengying Jin, and Junjie Yan. Mimicking very efficient network for object detection. In *Proceedings of the IEEE/CVF CVPR*, pages 7341–7349, 2017.

[Li *et al.*, 2020] Xiang Li, Wenhai Wang, Lijun Wu, Shuo Chen, Xiaolin Hu, Jun Li, Jinhui Tang, and Jian Yang. Generalized focal loss: Learning qualified and distributed bounding boxes for dense object detection. *Advances in Neural Information Processing Systems*, 33:21002–21012, 2020.

[Li *et al.*, 2022] Feng Li, Hao Zhang, Shilong Liu, Jian Guo, Lionel M Ni, and Lei Zhang. Dn-detr: Accelerate detr training by introducing query denoising. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 13619–13627, 2022.

[Lin *et al.*, 2014] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. Microsoft coco: Common objects in context. In *Proceedings of the European conference on computer vision (ECCV*, pages 740–755. Springer, 2014.

[Lin *et al.*, 2017] Tsung-Yi Lin, Priya Goyal, Ross Girshick, Kaiming He, and Piotr Dollár. Focal loss for dense object detection. In *Proceedings of the IEEE international conference on computer vision*, pages 2980–2988, 2017.

[Liu *et al.*, 2016] Wei Liu, Dragomir Anguelov, Dumitru Erhan, Christian Szegedy, Scott Reed, Cheng-Yang Fu, and Alexander C Berg. Ssd: Single shot multibox detector. In *Proceedings of the European conference on computer vision (ECCV)*, pages 21–37. Springer, 2016.

[Liu *et al.*, 2021] Yen-Cheng Liu, Chih-Yao Ma, Zijian He, Chia-Wen Kuo, Kan Chen, Peizhao Zhang, Bichen Wu, Zsolt Kira, and Peter Vajda. Unbiased teacher for semi-supervised object detection. In *9th International Conference on Learning Representations, ICLR 2021, Virtual Event, Austria, May, 2021*. OpenReview.net, 2021.

[Liu *et al.*, 2022] Shilong Liu, Feng Li, Hao Zhang, Xiao Yang, Xianbiao Qi, Hang Su, Jun Zhu, and Lei Zhang. DAB-DETR: dynamic anchor boxes are better queries for DETR. In *The Tenth International Conference on Learning Representations, ICLR 2022, Virtual Event, April, 2022*. OpenReview.net, 2022.

[Liu *et al.*, 2023] Shilong Liu, Tianhe Ren, Jiayu Chen, Zhaoyang Zeng, Hao Zhang, Feng Li, Hongyang Li, Jun Huang, Hang Su, Jun Zhu, and Lei Zhang. Detection transformer with stable matching. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 6468–6477. IEEE, 2023.

[Meng *et al.*, 2021] Depu Meng, Xiaokang Chen, Zejia Fan, Gang Zeng, Houqiang Li, Yuhui Yuan, Lei Sun, and Jingdong Wang. Conditional DETR for fast training convergence. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 3631–3640. IEEE, 2021.

[Ren *et al.*, 2015] Shaoqing Ren, Kaiming He, Ross B. Girshick, and Jian Sun. Faster r-cnn: Towards real-time object detection with region proposal networks. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 39:1137–1149, 2015.

[Sohn *et al.*, 2020] Kihyuk Sohn, David Berthelot, Nicholas Carlini, Zizhao Zhang, Han Zhang, Colin A Raffel, Ekin Dogus Cubuk, Alexey Kurakin, and Chun-Liang Li. Fixmatch: Simplifying semi-supervised learning with consistency and confidence. *Advances in neural information processing systems*, 33:596–608, 2020.

[Sun *et al.*, 2020] Pei Sun, Rufeng Zhang, Yi Jiang, Tao Kong, Chenfeng Xu, Wei Zhan, Masayoshi Tomizuka, Lei Li, Zehuan Yuan, Changhu Wang, and Ping Luo. Sparse r-cnn: End-to-end object detection with learnable proposals. *2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 14449–14458, 2020.

[Tang *et al.*, 2021] Yihe Tang, Weifeng Chen, Yijun Luo, and Yuting Zhang. Humble teachers teach better students for semi-supervised object detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3132–3141, 2021.

[Tarvainen and Valpola, 2017] Antti Tarvainen and Harri Valpola. Mean teachers are better role models: Weight-averaged consistency targets improve semi-supervised deep learning results. *Advances in neural information processing systems*, 30:1195–1204, 2017.

[Teng *et al.*, 2023] Yao Teng, Haisong Liu, Sheng Guo, and Limin Wang. Stageinteractor: Query-based object detector with cross-stage interaction. In *IEEE/CVF International Conference on Computer Vision, ICCV 2023, Paris, France, October, 2023*, pages 6554–6565. IEEE, 2023.

[Tian *et al.*, 2019] Zhi Tian, Chunhua Shen, Hao Chen, and Tong He. Fcos: Fully convolutional one-stage object detection. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 9627–9636, 2019.

[Wang *et al.*, 2022] Yu Wang, Xin Li, Shengzhao Wen, Fu-En Yang, Wanping Zhang, Gang Zhang, Haocheng Feng, Junyu Han, and Errui Ding. Knowledge distillation for detection transformer with consistent distillation points sampling. *ArXiv*, abs/2211.08071, 2022.

[Xu *et al.*, 2021] Mengde Xu, Zheng Zhang, Han Hu, Jianfeng Wang, Lijuan Wang, Fangyun Wei, Xiang Bai, and Zicheng Liu. End-to-end semi-supervised object detection with soft teacher. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 3060–3069, 2021.

[Yang *et al.*, 2019] Ze Yang, Shaohui Liu, Han Hu, Liwei Wang, and Stephen Lin. Reppoints: Point set representation for object detection. *2019 IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 9656–9665, 2019.

[Yao *et al.*, 2021] Zhuyu Yao, Jiangbo Ai, Boxun Li, and Chi Zhang. Efficient DETR: improving end-to-end object detector with dense prior. *CoRR*, abs/2104.01318, 2021.

[Zhang *et al.*, 2023] Hao Zhang, Feng Li, Shilong Liu, Lei Zhang, Hang Su, Jun Zhu, Lionel M. Ni, and Heung-Yeung Shum. DINO: DETR with improved denoising anchor boxes for end-to-end object detection. In *The Eleventh International Conference on Learning Representations, ICLR 2023, Kigali, Rwanda, May, 2023*. OpenReview.net, 2023.

[Zhao *et al.*, 2023a] Jing Zhao, Li Sun, and Qingli Li. Recursivedet: End-to-end region-based recursive object detection. In *IEEE/CVF International Conference on Computer Vision, ICCV 2023, Paris, France, October, 2023*, pages 6284–6293, 2023.

[Zhao *et al.*, 2023b] Jing Zhao, Shengjian Wu, Li Sun, and Qingli Li. Iou-enhanced attention for end-to-end task specific object detection. In *Computer Vision – ACCV 2022*, pages 124–141, Cham, 2023. Springer Nature Switzerland.

[Zhu *et al.*, 2021] Xizhou Zhu, Weijie Su, Lewei Lu, Bin Li, Xiaogang Wang, and Jifeng Dai. Deformable DETR: deformable transformers for end-to-end object detection. In *9th International Conference on Learning Representations, ICLR 2021, Virtual Event, Austria, May, 2021*, 2021.

[Zhu, 2005] Xiaojin Zhu. Semi-supervised learning literature survey. *world*, 10:10, 2005.