

# DETRs with Collaborative Hybrid Assignments Training

Zhuofan Zong   Guanglu Song   Yu Liu  
SenseTime Research

{zongzhuofan, liuyuisanai}@gmail.com  
songguanglu@sensetime.com

## Abstract

In this paper, we provide the observation that too few queries assigned as positive samples in DETR with one-to-one set matching leads to sparse supervisions on the encoder’s output which considerably hurt the discriminative feature learning of the encoder and vice visa for attention learning in the decoder. To alleviate this, we present a novel collaborative hybrid assignments training scheme, namely Co-DETR, to learn more efficient and effective DETR-based detectors from versatile label assignment manners. This new training scheme can easily enhance the encoder’s learning ability in end-to-end detectors by training the multiple parallel auxiliary heads supervised by one-to-many label assignments such as ATSS, FCOS, and Faster RCNN. In addition, we conduct extra customized positive queries by extracting the positive coordinates from these auxiliary heads to improve the training efficiency of positive samples in the decoder. In inference, these auxiliary heads are discarded and thus our method introduces no additional parameters and computational cost to the original detector while requiring no hand-crafted non-maximum suppression (NMS). We conduct extensive experiments to evaluate the effectiveness of the proposed approach on DETR variants, including DAB-DETR, Deformable-DETR, and DINO-Deformable-DETR. Specifically, we improve the basic Deformable-DETR by 5.8% in 12-epoch training and 3.2% in 36-epoch training. The state-of-the-art DINO-Deformable-DETR with Swin-L can still be improved from 58.5% to 59.5%. Surprisingly, incorporated with the large-scale backbone MixMIM-g with 1-Billion parameters, we achieve the 64.5% mAP on MS COCO test-dev, achieving superior performance with much fewer extra data sizes. Codes will be available at <https://github.com/Sense-X/Co-DETR>.

## 1. Introduction

Object detection is a fundamental task in computer vision, which requires us to localize the object and classify

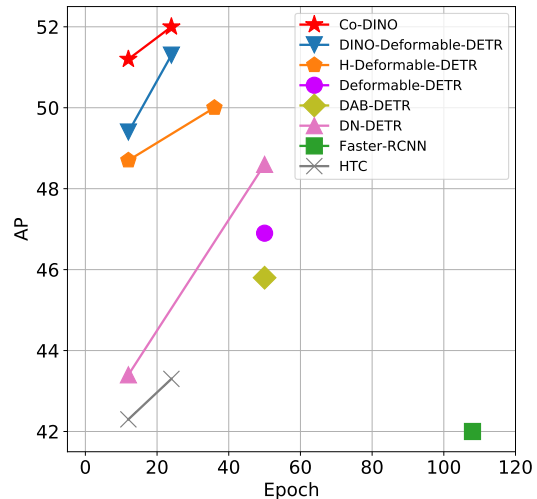


Figure 1. Performance of models with ResNet-50 backbone on COCO val.

its category. The seminal R-CNN families [9, 10, 25] and a series of variants such as ATSS [36], RetinaNet [18], FCOS [29] and PAA [14] lead to the significant breakthrough of object detection task [33, 38]. One-to-many label assignment is the core scheme of them, where each ground-truth box is assigned to multiple coordinates in the detector’s output as the supervised target cooperated with proposals [9, 25], anchors [18] or window centers [29]. Despite their promising performance, these detectors heavily rely on many hand-designed components like a non-maximum suppression procedure or anchor generation [1]. To conduct a more flexible end-to-end detector, DETECTION TRANSFORMER (DETR) [1] is proposed to view the object detection as a direct set prediction problem and introduce the one-to-one set matching scheme based on a transformer encoder-decoder architecture. In this manner, each ground-truth box will only be assigned to one specific query and multiple hand-designed components that encode prior knowledge are no longer needed. This approach introduces a flexible detection pipeline and encourages many DETR variants to further improve it. However, the performance of the vanilla end-to-end object detector is still inferior to the traditional

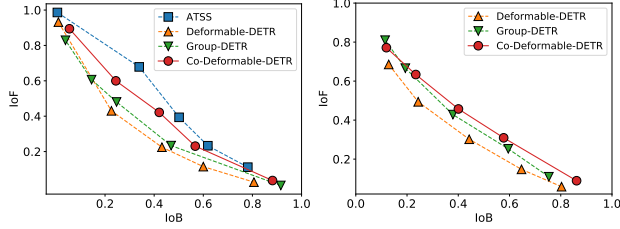


Figure 2. IoF-IoB curves for the feature discriminability score in the encoder and attention discriminability score in the decoder.

detectors with one-to-many label assignments.

In this paper, we try to make DETR-based detectors superior to conventional detectors while maintaining their end-to-end merit. To address this challenge, we focus on the intuitive drawback of one-to-one set matching that it explores less positive queries. This will lead to severe inefficient training issues. We detailedly analyze this from two aspects, the latent representation generated by the encoder and the attention learning in the decoder. We first compare the discriminability score of the latent features between the Deformable-DETR [37] and the one-to-many label assignment method where we simply replace the decoder with the ATSS head. Inspired by [12], the  $l^2$ -norm for the feature in each spatial coordinate is utilized to represent the discriminability score. Given the encoder’s output  $\mathcal{F} \in \mathbb{R}^{C \times H \times W}$ , we can obtain the discriminability score map  $\mathcal{S} \in \mathbb{R}^{1 \times H \times W}$ . The object can be better detected when the scores in the corresponding area are higher. As shown in Figure 2, we demonstrate the IoF-IoB curve (IoF: intersection over foreground, IoB: intersection over background) by applying different thresholds on the discriminability scores (details in Section 3.4). The higher IoF-IoB curve in ATSS indicates that it’s easier to distinguish the foreground and background. We further visualize the discriminability score map  $\mathcal{S} \in \mathbb{R}^{1 \times H \times W}$  in Figure 3. It’s obvious that the features in some salient areas are fully activated in the one-to-many label assignment method but less explored in one-to-one set matching. For the exploration of decoder training, we also demonstrate the IoF-IoB curve of the cross-attention score in the decoder based on the Deformable-DETR and the Group-DETR [4] which introduces more positive queries into the decoder. The illustration in Figure 2 shows that too few positive queries also influence the attention learning and increasing more positive queries in the decoder can slightly alleviate this.

This significant observation motivates us to present a simple but effective method, a collaborative hybrid assignment training scheme (Co-DETR). The key insight of Co-DETR is to use versatile one-to-many label assignments to improve the training efficiency and effectiveness of both the encoder and decoder. More specifically, we integrate the auxiliary heads with the output of the transformer encoder. These heads can be supervised by versatile one-to-many la-

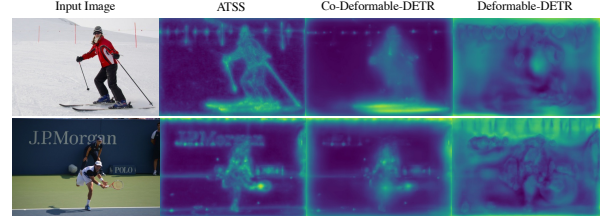


Figure 3. Visualizations of discriminability scores in the encoder.

bel assignments such as ATSS [36], FCOS [29], and Faster RCNN [25]. Different label assignments enrich the supervisions on the encoder’s output which forces it to be discriminative enough to support the training convergence of these heads. To further improve the training efficiency of the decoder, we elaborately encode the coordinates of positive samples in these auxiliary heads, including the positive anchors and positive proposals. They are sent to the original decoder as multiple groups of positive queries to predict the pre-assigned categories and bounding boxes. The foreground coordinates in each auxiliary head serve as an independent group that is isolated from the other groups. Versatile one-to-many label assignments can introduce lavish (positive query, ground-truth) pairs to improve the decoder’s training efficiency. Note that, only the original decoder is used during inference, thus the proposed training scheme only introduces extra overheads during training.

We conduct extensive experiments to evaluate the efficiency and effectiveness of the proposed Co-DETR. As shown in Figure 1, Co-DETR achieves faster training convergence and even higher performance. Illustrated in Figure 3, Co-DETR greatly alleviates the poorly encoder’s feature learning in one-to-one set matching. As a plug-and-play approach, we easily combine it with different DETR variants, including Conditional-DETR [24], DAB-DETR [21], Deformable-DETR [37], and  $\mathcal{H}$ -Deformable-DETR [13]. Specifically, we improve the plain Deformable-DETR by 5.8% in 12-epoch training and 3.2% in 36-epoch training. The state-of-the-art DINO-Deformable-DETR with Swin-L [23] can still be improved from 58.5% to 59.5% on the MS COCO val. Surprisingly, incorporated with the large-scale backbone MixMIM-g [20] with 1-Billion parameters, we achieve 64.5% mAP on MS COCO test-dev, establishing the new state-of-the-art detector with much fewer data sizes.

## 2. Related Works

**One-to-many label assignment.** For one-to-many label assignment in object detection, multiple box candidates can be assigned to the same ground-truth box as positive samples in the training phase. In classic anchor-based detectors, such as Faster-RCNN [25] and RetinaNet [18], the sample selection is guided by the predefined IoU thresh and matching IoU between anchors and annotated boxes. The anchor-

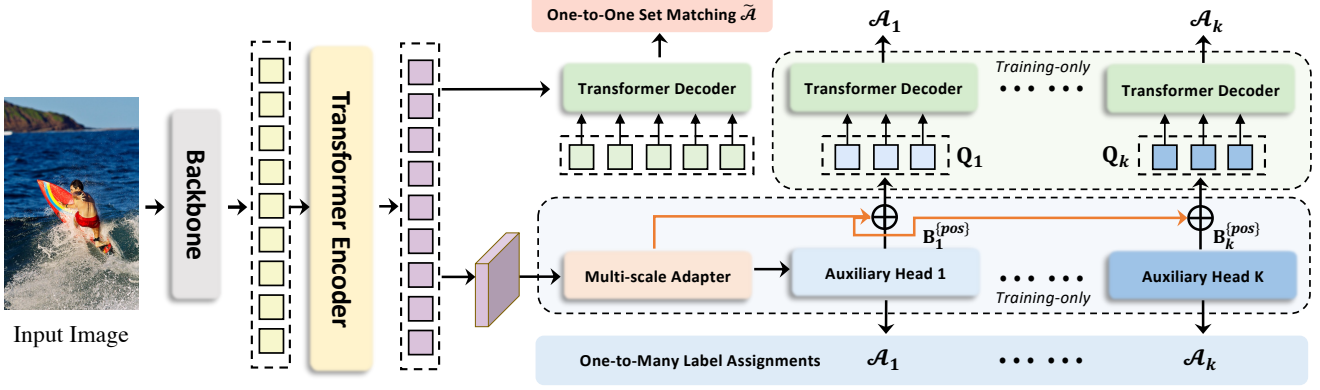


Figure 4. **Framework of our Collaborative Hybrid Assignment Training.** The auxiliary branches are discarded during evaluation.

free FCOS [29] leverages the center priors and assigns the spatial locations near the center of each bounding box as positives. Moreover, the adaptive mechanism is incorporated into one-to-many label assignment to overcome the limitation of fixed label assignment. ATSS [36] performs adaptive anchor selection by the statistical dynamic IoU values of top- $k$  closest anchors. PAA [14] adaptively separates anchors into positive and negative samples in a probabilistic manner. In this paper, we propose a collaborative hybrid assignment scheme to improve the representations of the encoder via auxiliary heads with one-to-many label assignment.

**One-to-one set matching.** The pioneering transformer-based detector, DETR [1], incorporates the one-to-one set matching scheme into object detection and performs fully end-to-end object detection. The one-to-one set matching strategy first calculates the global matching cost via Hungarian matching and assigns only one positive sample with the minimum matching cost for each ground-truth box. DN-DETR [15] demonstrates the slow convergence results from the instability of one-to-one set matching, thus introducing denoising training to eliminate this issue. DINO [35] inherits the advanced query formulation of DAB-DETR and incorporates an improved contrastive denoising technique to achieve state-of-the-art performance. Group-DETR [4] constructs group-wise one-to-many label assignment to exploit multiple positive object queries, which is similar to the hybrid matching scheme in  $\mathcal{H}$ -DETR [13]. In contrast with the above follow-up works, we present a new perspective of collaborative optimization for one-to-one set matching.

### 3. Method

#### 3.1. Overview

Following the standard DETR protocol, the input image is fed into the backbone and encoder to generate the latent features. Multiple predefined object queries interact with them in the decoder via cross-attention afterwards. We introduce the Co-DETR to improve the feature learning in the

encoder and the attention learning in the decoder via the collaborative hybrid assignments training scheme and the customized positive queries generation. We will detailedly describe these modules and give the insights why they can work well.

#### 3.2. Collaborative Hybrid Assignments Training

To alleviate the sparse supervision on the encoder’s output caused by the fewer positive queries in the decoder, we incorporate versatile auxiliary heads with different one-to-many label assignment paradigms, *e.g.*, ATSS, and Faster R-CNN. Different label assignments enrich the supervisions on the encoder’s output which forces it to be discriminative enough to support the training convergence of these heads. Specifically, given the encoder’s latent feature  $\mathcal{F}$ , we firstly transform it to the feature pyramid  $\{\mathcal{F}_1, \dots, \mathcal{F}_J\}$  via the multi-scale adapter where  $J$  indicates feature map with  $2^{2+J}$  downsampling stride. Similar to ViTDet [17], the feature pyramid is constructed by a single feature map in the single-scale encoder, while we use bilinear interpolation and  $3 \times 3$  convolution for upsampling. For instance, with the single-scale feature from the encoder, we successively apply downsampling ( $3 \times 3$  convolution with stride 2) or upsampling operations to produce a feature pyramid. As for the multi-scale encoder, we only downsample the coarsest feature in the multi-scale encoder features  $\mathcal{F}$  to build the feature pyramid. Defined  $K$  collaborative heads with corresponding label assignment manners  $\mathcal{A}_k$ , for the  $i$ -th collaborative head,  $\{\mathcal{F}_1, \dots, \mathcal{F}_J\}$  is sent to it to obtain the predictions  $\hat{\mathbf{P}}_i$ . At the  $i$ -th head,  $\mathcal{A}_i$  is used to compute the supervised targets for the positive and negative samples in  $\mathbf{P}_i$ . Denoted  $\mathbf{G}$  as the ground-truth set, this procedure can be formulated as:

$$\mathbf{P}_i^{\{pos\}}, \mathbf{B}_i^{\{pos\}}, \mathbf{P}_i^{\{neg\}} = \mathcal{A}_i(\hat{\mathbf{P}}_i, \mathbf{G}), \quad (1)$$

where  $\{pos\}$  and  $\{neg\}$  indicate the pair set of ( $j$ , positive coordinates or negative coordinates in  $\mathcal{F}_j$ ) determined by  $\mathcal{A}_i$ .  $j$  means the feature index in  $\{\mathcal{F}_1, \dots, \mathcal{F}_J\}$ .  $\mathbf{B}_i^{\{pos\}}$  is

Head $i$	Loss $\mathcal{L}_i$	Assignment $\mathcal{A}_i$		
		$\{pos\}, \{neg\}$ Generation	$\mathbf{P}_i$ Generation	$\mathbf{B}_i^{\{pos\}}$ Generation
Faster-RCNN [25]	cls: CE loss, reg: GIoU loss	$\{pos\}$ : IoU(proposal, gt)>0.5 $\{neg\}$ : IoU(proposal, gt)<0.5	$\{pos\}$ : gt labels, offset(proposal, gt) $\{neg\}$ : gt labels	positive proposals ( $x_1, y_1, x_2, y_2$ )
ATSS [36]	cls: Focal loss reg: GIoU, BCE loss	$\{pos\}$ : IoU(anchor, gt)>(mean+std) $\{neg\}$ : IoU(anchor, gt)<(mean+std)	$\{pos\}$ : gt labels, offset(anchor, gt), centerness $\{neg\}$ : gt labels	positive anchors ( $x_1, y_1, x_2, y_2$ )
RetinaNet [18]	cls: Focal loss reg: GIoU Loss	$\{pos\}$ : IoU(anchor, gt)>0.5 $\{neg\}$ : IoU(anchor, gt)<0.4	$\{pos\}$ : gt labels, offset(anchor, gt) $\{neg\}$ : gt labels	positive anchors ( $x_1, y_1, x_2, y_2$ )
FCOS [29]	cls: Focal Loss reg: GIoU, BCE loss	$\{pos\}$ : points inside gt center area $\{neg\}$ : points outside gt center area	$\{pos\}$ : gt labels, ltrb distance, centerness $\{neg\}$ : gt labels	FCOS point ( $cx, cy$ ) $w = h = 8 \times 2^{2+j}$

Table 1. **Detailed information of auxiliary heads.** The auxiliary heads include Faster-RCNN [25], ATSS [36], RetinaNet [18], and FCOS [29]. If not otherwise specified, we follow the original implementations, e.g., anchor generation.

the set of spatial positive coordinates.  $\mathbf{P}_i^{\{pos\}}$  and  $\mathbf{P}_i^{\{neg\}}$  are the supervised targets in the corresponding coordinates, including the categories and regressed offsets. To be specific, we describe the detailed information about each variable in Table 1. The loss functions can be defined as:

$$\mathcal{L}_i^{enc} = \mathcal{L}_i(\hat{\mathbf{P}}_i^{\{pos\}}, \mathbf{P}_i^{\{pos\}}) + \mathcal{L}_i(\hat{\mathbf{P}}_i^{\{neg\}}, \mathbf{P}_i^{\{neg\}}), \quad (2)$$

Note that the regression loss is discarded for negative samples. The training objective of the optimization for  $K$  auxiliary heads is formulated as follows:

$$\mathcal{L}^{enc} = \sum_{i=1}^K \mathcal{L}_i^{enc} \quad (3)$$

### 3.3. Customized Positive Queries Generation

In the one-to-one set matching paradigm, each ground-truth box will only be assigned to one specific query as the supervised target. Too few positive queries lead to inefficient cross-attention learning in the transformer decoder as shown in Figure 2. To alleviate this, we elaborately generate sufficient customized positive queries according to the label assignment  $\mathcal{A}_i$  in each auxiliary head. Specifically, given the positive coordinates set  $\mathbf{B}_i^{\{pos\}} \in \mathbb{R}^{M_i \times 4}$  in the  $i$ -th auxiliary head, where  $M_i$  is the number of positive samples, the extra customized positive queries  $\mathbf{Q}_i \in \mathbb{R}^{M_i \times C}$  can be generated by:

$$\mathbf{Q}_i = \text{Linear}(\text{PE}(\mathbf{B}_i^{\{pos\}})) + \text{Linear}(\text{E}(\{\mathcal{F}_*\}, \{pos\})). \quad (4)$$

where  $\text{PE}(\cdot)$  stands for positional encodings and we select the corresponding features from  $\text{E}(\cdot)$  according to the index pair ( $j$ , positive coordinates or negative coordinates in  $\mathcal{F}_j$ ).

As a result, there are  $K + 1$  groups of queries that contribute to a single one-to-one set matching branch and  $K$  branches with one-to-many label assignments during training. The auxiliary one-to-many label assignment branches share the same parameters with  $L$  decoders layers in the original main branch. All the queries in the auxiliary branch are regarded as positive queries, thus the matching process is discarded. To be specific, the loss of the  $l$ -th decoder layer

in the  $i$ -th auxiliary branch can be formulated as:

$$\mathcal{L}_{i,l}^{dec} = \tilde{\mathcal{L}}(\tilde{\mathbf{P}}_{i,l}, \mathbf{P}_i^{\{pos\}}). \quad (5)$$

$\tilde{\mathbf{P}}_{i,l}$  refers to the output predictions of the  $l$ -th decoder layer in the  $i$ -th auxiliary branch. Finally, the training objective for Co-DETR is:

$$\mathcal{L}^{global} = \sum_{l=1}^L (\tilde{\mathcal{L}}_l^{dec} + \lambda_1 \sum_{i=1}^K \mathcal{L}_{i,l}^{dec} + \lambda_2 \mathcal{L}^{enc}), \quad (6)$$

where  $\tilde{\mathcal{L}}_l^{dec}$  stands for the loss in the original one-to-one set matching branch [1],  $\lambda_1$  and  $\lambda_2$  are the coefficient balancing the losses.

### 3.4. Why Co-DETR works

Co-DETR leads to evident improvement to the DETR-based detectors. In the following, we try to investigate its effectiveness qualitatively and quantitatively. We conduct detailed analysis based on Deformable-DETR with ResNet-50 [11] backbone (trained for 36 epochs).

**Enrich the encoder’s supervisions.** Intuitively, too few positive queries lead to sparse supervisions as only one query is supervised by regression loss for each ground-truth. The positive samples in one-to-many label assignment manners receive more localization supervisions to help enhance the latent feature learning. To further explore how the sparse supervisions impede the model training, we detailedly investigate the latent features produced by the encoder. We introduce the IoF-IoB curve to quantize the discriminability score of the encoder’s output. Specifically, given the latent feature  $\mathcal{F}$  of the encoder, inspired by the feature visualization in Figure 3, we compute the IoF (intersection over foreground) and IoB (intersection over background). Given the encoder’s feature  $\mathcal{F}_j \in \mathbb{R}^{C \times H_j \times W_j}$  at level  $j$ , we first calculate the  $l^2$ -norm  $\hat{\mathcal{F}}_j \in \mathbb{R}^{1 \times H_j \times W_j}$  and resize it to the image size  $H \times W$ . The discriminability score  $\mathcal{D}(\mathcal{F})$  is computed by averaging the scores from all levels:

$$\mathcal{D}(\mathcal{F}) = \frac{1}{J} \sum_{j=1}^J \frac{\hat{\mathcal{F}}_j}{\max(\hat{\mathcal{F}}_j)}, \quad (7)$$



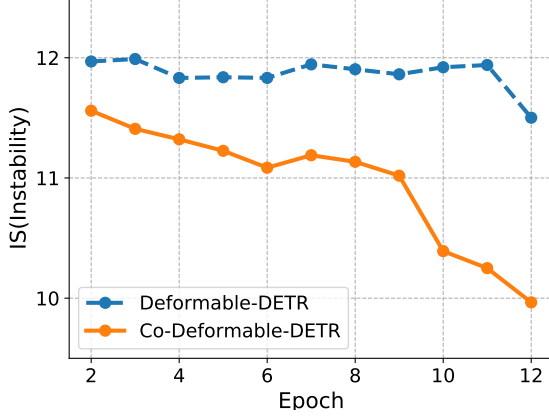


Figure 5. The instability (IS) [15] of Deformable-DETR and Co-Deformable-DETR on COCO dataset. These detectors are trained for 12 epochs with ResNet-50 backbone.

where the resize operation is omitted. We visualize the discriminability scores of ATSS, Deformable-DETR, and our Co-Deformable-DETR in Figure 3. Compared with Deformable-DETR, both ATSS and Co-Deformable-DETR own stronger ability to distinguish the areas of key objects, while Deformable-DETR is almost disturbed by the background. Consequently, we define the indicators for foreground and background as  $\mathbb{1}(\mathcal{D}(\mathcal{F}) > S) \in \mathbb{R}^{H \times W}$  and  $\mathbb{1}(\mathcal{D}(\mathcal{F}_j) < S) \in \mathbb{R}^{H \times W}$ , respectively.  $S$  is a predefined score thresh,  $\mathbb{1}(x)$  is 1 if  $x$  is true and 0 otherwise. As for the mask of foreground  $\mathcal{M}^{fg} \in \mathbb{R}^{H \times W}$ , the element  $\mathcal{M}_{h,w}^{fg}$  is 1 if the point  $(h, w)$  is inside the foreground and 0 otherwise. The area of intersection over foreground (IoF)  $\mathcal{I}^{fg}$  can be computed as:

$$\mathcal{I}^{fg} = \frac{\sum_{h=1}^H \sum_{w=1}^W (\mathbb{1}(\mathcal{D}(\mathcal{F}_{h,w}) > S) \cdot \mathcal{M}_{h,w}^{fg})}{\sum_{h=1}^H \sum_{w=1}^W \mathcal{M}_{h,w}^{fg}}. \quad (8)$$

Concretely, we compute the area of intersection over background areas (IoB) in a similar way and plot the curve IoF and IoB by varying  $S$  in Figure 2. Obviously, ATSS and Co-Deformable-DETR obtain higher IoF values than both Deformable-DETR and Group-DETR under the same IoB values, which demonstrates the encoder representations benefit from the one-to-many label assignment.

**Improve the cross-attention learning by reducing the instability of Hungarian matching.** Hungarian matching is the core scheme in one-to-one set matching. Cross-attention is an important operation to help the positive queries encode abundant object information. It requires sufficient training to achieve this. We observe that the Hungarian matching introduces uncontrollable instability since the ground-truth assigned to a specific positive query in the same image is changing during the training process. Following [15], we present the comparison of instability in Figure 5, where we find our approach contributes to a more stable matching process. Furthermore, in order to quantify how well cross-

Method	$K$	#epochs	AP
Conditional DETR-C5 [24]	0	36	39.4
Conditional DETR-C5 [24]	1	36	41.5(+2.1)
Conditional DETR-C5 [24]	2	36	41.8(+2.4)
DAB-DETR-C5 [21]	0	36	41.2
DAB-DETR-C5 [21]	1	36	43.1(+1.9)
DAB-DETR-C5 [21]	2	36	43.5(+2.3)
Deformable-DETR [37]	0	12	37.1
Deformable-DETR [37]	1	12	42.3(+5.2)
Deformable-DETR [37]	2	12	42.9(+5.8)
Deformable-DETR [37]	0	36	43.3
Deformable-DETR [37]	1	36	46.8(+3.5)
Deformable-DETR [37]	2	36	46.5(+3.2)
Deformable-DETR++ [37]	0	12	47.1
Deformable-DETR++ [37]	1	12	48.7(+1.6)
Deformable-DETR++ [37]	2	12	49.5(+2.4)
$\mathcal{H}$ -Deformable-DETR [13]	0	12	48.4
$\mathcal{H}$ -Deformable-DETR [13]	1	12	49.2(+0.8)
$\mathcal{H}$ -Deformable-DETR [13]	2	12	49.7(+1.3)
DINO-Deformable-DETR* [35]	0	12	49.4
DINO-Deformable-DETR* [35]	1	12	51.0(+1.6)
DINO-Deformable-DETR* [35]	2	12	51.2(+1.8)

Table 2. All results are reproduced using mmdetection [3] on COCO val. Methods with \* use 5 feature levels.

Backbone	$K$	#epochs	AP	$AP_S$	$AP_M$	$AP_L$
R101	0	12	48.1	29.9	51.5	63.1
R101	1	12	49.6(+1.5)	31.7	53.2	64.3
R101	2	12	50.1(+2.0)	33.0	53.9	64.6
Swin-T	0	12	49.8	32.4	53.2	64.2
Swin-T	1	12	51.6(+1.8)	34.0	55.3	66.5
Swin-T	2	12	51.7(+1.9)	35.3	55.2	67.0
Swin-B	0	12	54.0	36.7	57.7	70.2
Swin-B	1	12	55.2(+1.2)	38.1	59.3	71.3
Swin-B	2	12	55.5(+1.5)	37.9	59.6	72.2
Swin-L	0	12	55.2	37.2	58.9	71.6
Swin-L	1	12	56.4(+1.2)	39.0	60.6	72.7
Swin-L	2	12	56.9(+1.7)	40.1	61.2	73.3

Table 3. Experimental results on Deformable-DETR++.

attention is being optimized, we also calculate the IoF-IoB curve for attention score. Similar to the feature discriminability score computation, we set different thresholds for attention score to get multiple IoF-IoB pairs. The comparisons between Deformable-DETR, Group-DETR, and Co-Deformable-DETR can be viewed in Figure 2. We find that the IoF-IoB curves of DETRs with more positive queries are generally above Deformable-DETR, which is consistent with our motivation.

## 4. Experiments

### 4.1. Setup

**Datasets and Evaluation Metrics.** Our experiments are conducted on the MS COCO 2017 dataset [19] that consists

Method	Backbone	Multi-scale	#query	#epochs	AP	AP <sub>50</sub>	AP <sub>75</sub>	AP <sub>S</sub>	AP <sub>M</sub>	AP <sub>L</sub>
Conditional-DETR [24]	R50	✗	300	108	43.0	64.0	45.7	22.7	46.7	61.5
Anchor-DETR [31]	R50	✗	300	50	42.1	63.1	44.9	22.3	46.2	60.0
DAB-DETR [21]	R50	✗	900	50	45.7	66.2	49.0	26.1	49.4	63.1
AdaMixer [8]	R50	✓	300	36	47.0	66.0	51.1	30.1	50.2	61.8
AdaMixer [8]	R101	✓	300	36	48.0	67.0	52.4	30.0	51.2	63.7
AdaMixer [8]	Swin-S	✓	300	36	51.3	71.2	55.7	34.2	54.6	67.3
Deformable-DETR [37]	R50	✓	300	50	46.9	65.6	51.0	29.6	50.1	61.6
DN-Deformable-DETR [15]	R50	✓	300	50	48.6	67.4	52.7	31.0	52.0	63.7
DINO-Deformable-DETR <sup>†*</sup> [35]	R50	✓	900	12	49.4	66.9	53.8	32.3	52.5	63.9
DINO-Deformable-DETR <sup>†*</sup> [35]	Swin-L (IN-22K)	✓	900	36	58.5	77.0	64.1	41.5	62.3	74.0
Group-DINO-Deformable-DETR <sup>†</sup> [4]	Swin-L (IN-22K)	✓	900	36	58.4	-	-	41.0	62.5	73.9
$\mathcal{H}$ -Deformable-DETR [13]	R50	✓	300	12	48.7	66.4	52.9	31.2	51.5	63.5
$\mathcal{H}$ -Deformable-DETR [13]	R101	✓	300	12	49.4	67.2	53.7	31.9	53.1	64.2
$\mathcal{H}$ -Deformable-DETR [13]	Swin-S	✓	300	36	54.4	72.9	59.4	36.9	58.3	69.5
$\mathcal{H}$ -Deformable-DETR <sup>†‡</sup> [13]	Swin-L (IN-22K)	✓	900	36	57.9	76.8	63.6	42.4	61.9	73.4
Co-Deformable-DETR	R50	✓	300	12	49.5	67.6	54.3	32.4	52.7	63.7
Co-Deformable-DETR	R101	✓	300	12	50.1	67.9	54.9	33.0	53.9	64.6
Co-DINO-Deformable-DETR <sup>†*</sup>	R50	✓	900	12	<b>51.2</b>	<b>68.3</b>	<b>56.1</b>	<b>34.0</b>	<b>54.6</b>	<b>64.8</b>
Co-Deformable-DETR	Swin-S	✓	300	36	55.3	73.6	60.9	39.0	59.0	70.1
Co-Deformable-DETR <sup>†</sup>	Swin-L (IN-22K)	✓	900	36	<b>58.5</b>	<b>77.1</b>	<b>64.5</b>	<b>42.4</b>	<b>62.4</b>	<b>74.0</b>
Co-DINO-Deformable-DETR <sup>†*</sup>	Swin-L (IN-22K)	✓	900	36	<b>59.5</b>	<b>77.6</b>	<b>65.4</b>	<b>43.7</b>	<b>62.9</b>	<b>74.5</b>

†: 300 predictions for evaluation. ‡: improved hyperparameters. \*: 5 feature levels.

Table 4. Comparison to the state-of-the-art DETR variants on COCO val.

of 115K labeled images for training. We report the detection results by default on 5K val images. The results of our largest model evaluated on the test-dev (20K images) are also reported. To verify the scalability and robustness of Co-DETR, we further apply our method to a large-scale object detection benchmark, namely Objects365 [28]. There are 1.7M labeled images used for training and 80K images for validation in the Objects365 dataset. All results follow the standard mean Average Precision (AP) under IoU thresholds ranging from 0.5 to 0.95 at different object scales.

**Implementation Details.** We incorporate our Co-DETR into the current DETR-like pipelines and keep the training setting consistent with the baselines. We adopt ATSS and Faster-RCNN as the auxiliary heads for  $K = 2$  and only remain ATSS for  $K = 1$ . More details about our auxiliary heads can be found in the supplementary materials. We choose the number of learnable object queries to 300 and set  $\{\lambda_1, \lambda_2\}$  to  $\{1.0, 2.0\}$  by default.

## 4.2. Main Results

In this section, we empirically analyze the effectiveness and generalization ability of our proposed co-training method on different DETR variants. Both the results of  $K = 1$  and  $K = 2$  are reported in Table 2 and Table 3.

**Results with ResNet-50.** We first apply the collaborative hybrid assignments training to single-scale DETRs with C5 features. Surprisingly, both Conditional-DETR and DAB-DETR obtain 2.4% and 2.3% AP gains over the baselines with a long training schedule. For Deformable-DETR with

multi-scale features, the detection performance is significantly boosted from 37.1% to 42.9%. The overall improvements (+3.2%) still hold when the training time is increased to 36 epochs. Moreover, we conduct experiments on the improved Deformable-DETR (denoted as Deformable-DETR++) following [13], where a +2.4% gain is observed. The state-of-the-art DINO-Deformable-DETR equipped with our method can achieve 51.2%, which is +1.8% higher than the competitive baseline and outperform all existing detectors in the same setting.

**Results with larger backbones.** We further scale up the backbone capacity from ResNet-50 to Swin Transformer [23] based on Deformable-DETR++. As presented in Table 3, our method achieves 56.9% and surpasses the baseline by a large margin (+1.7%) with Swin-L.

## 4.3. Comparisons with the state-of-the-art

We apply our method with  $K = 2$  to Deformable-DETR++ and report the comparisons on COCO val in Table 4. Note that the results of  $K = 1$  are released in the supplementary materials. Compared with other competitive counterparts, our method converges much faster. For example, Co-Deformable-DETR readily achieves 49.5% and 50.1% when using only 12 epochs with ResNet-50 and ResNet-101 backbone, respectively. We further use the Swin Transformer as backbone and increase the training time to a longer schedule of 36 epochs. We can see that our models consistently outperform the state-of-the-art  $\mathcal{H}$ -Deformable-DETR with the same backbone by  $\sim 1.0\%$  AP.

Method	Backbone	#Params	Image pre-training data	Detection pre-training data	val AP <sup>box</sup>	test-dev AP <sup>box</sup>
HTC++ [2]	SwinV2-G [22]	3.0B	IN-22K-ext (70M)	Objects365	62.5	63.1
DINO [35]	Swin-L [23]	218M	IN-22K (14M)	Objects365	63.2	63.3
BEiT3 [30]	ViT-g [7]	1.9B	IN-22K (14M) + Image-Text Pairs (35M) + Texts (160GB)	Objects365	—	63.7
FD [32]	SwinV2-G [22]	3.0B	IN-22K-ext (70M)	Objects365	—	64.2
DINO [35]	FocalNet-H [34]	746M	IN-22K (14M)	Objects365	64.2	64.3
Group DETR v2 [5]	ViT-Huge [7]	629M	IN-1K (1M)	Objects365	—	64.5
Co-Deformable-DETR	MixMIM-g	1.0B	IN-1K (1M)	Objects365	64.4	64.5

Table 5. Comparison to the state-of-the-art on COCO test-dev with fewer extra data sizes.

aux head	pos queries	#epochs	AP	AP <sub>50</sub>	AP <sub>75</sub>
$\times$	$\times$	12	37.1	55.5	40.0
		36	43.3	62.3	47.1
$\checkmark$	$\times$	12	41.6(+4.5)	59.8	45.6
		36	46.2(+2.9)	64.7	50.9
$\times$	$\checkmark$	12	40.5(+3.4)	58.8	44.4
		36	45.3(+2.0)	63.5	49.8
$\checkmark$	$\checkmark$	12	42.3(+5.2)	60.5	46.1
		36	46.8(+3.5)	65.1	51.5

Table 6. Component-wise ablations of our method. “aux head” denotes training with an auxiliary head and “pos queries” means the customized positive queries generation.

More importantly, the performance of DINO-Deformable-DETR can still be boosted from 58.5% to 59.5%, surpassing other state-of-the-art methods by a large margin.

To further explore the scalability potential of our method, we extend the backbone capacity to 1 billion parameters. This extremely large backbone MixMIM-g [20] is pre-trained on ImageNet-1K dataset [6] using self-supervised learning. More details about MixMIM-g are provided in the supplementary materials. We first pre-train this model on Objects365 for only 16 epochs, then fine-tune it on the COCO dataset for 12 epochs. In the fine-tuning stage, the input resolution is randomly selected between  $480 \times 2000$  and  $1472 \times 2000$ . Our results are evaluated with multi-scale testing and horizontal flip. Table 5 presents the state-of-the-art comparisons on the COCO benchmark with fewer extra data sizes. With only ImageNet-1K as pre-training data and fewer pre-training epochs (e.g., 16 epochs vs. 24 epochs of Group DETR v2 [5]), Co-Deformable-DETR achieves state-of-the-art results of 64.4% and 64.5% on COCO val and test-dev, respectively.

#### 4.4. Ablation Studies

Unless stated otherwise, all experiments for ablations are conducted on Deformable-DETR with a ResNet-50 backbone. We choose the number of auxiliary heads  $K$  to 1 by default and set the total batch size to 32. More ablations can be found in the supplementary materials.

**The effect of each component.** We perform a component-wise ablation to thoroughly analyze the effect of each component in Table 6. Simply incorporating an auxiliary head after the encoder yields significant gains, since the dense

Method	$K$	#epochs	GPU hours	AP
Deformable-DETR	1	36	288	46.8
Deformable-DETR	0	50	333	44.5
Deformable-DETR	0	100	667	46.0
Deformable-DETR	0	150	1000	45.9

Table 7. Comparison to baselines with longer schedule.

Auxiliary head	#epochs	AP	AP <sub>50</sub>	AP <sub>75</sub>
Baseline	36	43.3	62.3	47.1
RetinaNet [18]	36	46.1	64.2	50.1
Faster-RCNN [25]	36	46.3	64.7	50.5
Mask-RCNN [10]	36	46.5	65.0	50.6
FCOS [29]	36	46.5	64.8	50.7
PAA [14]	36	46.5	64.6	50.7
GFL [16]	36	46.5	65.0	51.0
ATSS [36]	36	<b>46.8</b>	<b>65.1</b>	<b>51.5</b>

Table 8. Detection performance of our approach with various one-to-many heads on COCO val.

spatial supervision enables the encoder features more discriminative. Alternatively, introducing the customized positive queries into the decoder contributes remarkably to the final results, while improving the training efficiency of the one-to-one set matching. In summary, we observe the overall improvements stem from two aspects, which include more discriminative features for the encoder and more efficient attention learning for the decoder.

**Comparisons to the longer training schedule.** As presented in Table 7, we find Deformable-DETR can not benefit from longer training as the performance saturates. On the contrary, Co-DETR greatly accelerates the convergence as well as increasing the peak performance.

**Influence of various auxiliary heads.** To explore the influence of different auxiliary heads, we conduct experiments with various detection heads. The results in Table 8 reveal that both one-stage and two-stage auxiliary heads consistently improve the baseline and ATSS achieves the best performance. Accordingly, we observe additional instance segmentation annotations slightly increase the accuracy while introducing an extra mask branch during training. As a result, we choose ATSS as the auxiliary head when  $K$  is 1.

**The number of auxiliary heads.** To further delve into the

Method	$K$	Auxiliary head	Memory (MB)	GPU hours	AP
Deformable-DETR++	0	—	12808	70	47.1
$\mathcal{H}$ -Deformable-DETR	0	—	15307	104	48.4
Deformable-DETR++	1	ATSS	13947	86	48.7
Deformable-DETR++	2	ATSS + PAA	14629	124	49.0
Deformable-DETR++	2	ATSS + Faster-RCNN	14387	120	<b>49.5</b>
Deformable-DETR++	3	ATSS + Faster-RCNN + PAA	15263	150	<b>49.5</b>
Deformable-DETR++	6	ATSS + Faster-RCNN + PAA + GFL + FCOS + RetinaNet	19385	280	48.9

Table 9. Experimental results of  $K$  varying from 1 to 6.

Method	$K$	GPU hours	AP	$AP_S$	$AP_M$	$AP_L$
Deformable-DETR	0	534	50.8	34.7	55.3	62.7
Deformable-DETR	1	560	<b>52.8(+2.0)</b>	35.8	57.3	66.7
Deformable-DETR	2	587	<b>54.1(+3.3)</b>	37.4	59.1	68.4
Deformable-DETR++	0	547	57.4	41.8	61.2	70.7
Deformable-DETR++	1	569	<b>58.4(+1.0)</b>	42.8	62.9	72.0
Deformable-DETR++	2	592	<b>58.7(+1.3)</b>	43.0	62.9	72.3

Table 10. Results with the MixMIM-g under 12 training epochs.

influence of  $K$ , we report the performance and training efficiency by controlling the number of auxiliary heads. In Table 9, we find Co-DETR with ATSS performs better than hybrid matching scheme [13] while training faster and requiring less GPU memory. It is worth noting that the accuracy continues to increase as the training costs of the model increase when choosing  $K$  smaller than 3. We speculate the optimization conflicts among these heads lead to performance degradation when  $K \geq 4$ . Overall, we choose  $K \leq 2$  and Faster-RCNN as the second auxiliary head since it achieves the best trade-offs between training efficiency and accuracy.

**Effectiveness of collaborative one-to-many label assignments.** To verify the effectiveness of collaborative one-to-many label assignments, we compare our approach with Group-DETR (3 groups), which can be viewed as  $K = 2$  with two auxiliary one-to-one set matching branches. As shown in Table 6, our method ( $K = 1$ ) without the customized positive queries generation improves the AP score from 43.3% to 46.2%, surpassing Group-DETR (44.6%) by a large margin. More importantly, the IoF-IoB curve in Figure 2 demonstrates Group-DETR fails to enhance the feature representations in the encoder, while our method alleviates the poorly encoder’s feature learning.

**Scalability to large vision model.** We apply our approach to the extremely large-scale backbone using 12-epoch settings to verify the capacity scalability. As shown in Table 10, the 1-billion-parameter MixMiM-g can be boosted to achieve 3.3% and 1.3% gains on Deformable-DETR and its improved variant, respectively. The results reveals our approach also shows the ability to enhance training efficiency of large vision models while introducing less than 10% training overheads.

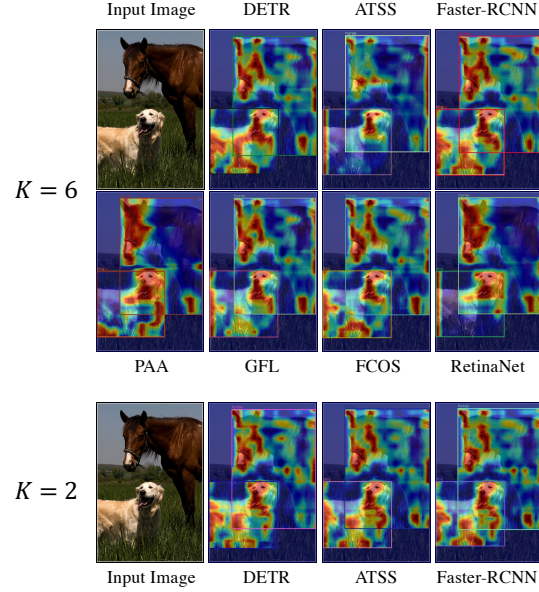


Figure 6. CAM visualizations of multiple heads. Stronger CAM areas are covered by warmer colors. Best viewed in color.

**CAM visualizations of multiple heads.** We find that the performance degrades when  $K = 6$  in Table 9. To better understand the effects of collaborative one-to-many label assignments, we visualize the class activation maps (CAM) [27] of DETR branch and auxiliary heads of our model ( $K = 6$  and  $K = 2$  in Table 9) using Grad CAM. As presented in Figure 6, we observe the concentrated regions of multiple heads are inconsistent when  $K = 6$  as different heads fully cover different parts of the objects. This indicates that the optimization conflicts emerge due to inconsistent optimization targets. Besides, We also notice the CAM results become similar when  $K = 2$ , which implies consistent optimization improves training efficiency.

## 5. Conclusions

In this paper, we present a novel collaborative hybrid assignments training scheme, namely Co-DETR, to learn more efficient and effective DETR-based detectors from versatile label assignment manners. This new training scheme can easily enhance the encoder’s learning ability in end-to-end detectors by training the multiple parallel auxiliary heads supervised by one-to-many label assignments. In addition, we conduct extra customized positive queries by extracting the positive coordinates from these auxiliary heads to improve the training efficiency of positive samples in decoder. Extensive experiments on MS COCO dataset demonstrate the efficiency and effectiveness of our Co-DETR. Surprisingly, incorporated with the large-scale backbone MixMIM-g with 1-Billion parameters, we achieve the 64.5% mAP on MS COCO test-dev, achieving superior performance with much fewer extra data sizes.



## References

- [1] Nicolas Carion, Francisco Massa, Gabriel Synnaeve, Nicolas Usunier, Alexander Kirillov, and Sergey Zagoruyko. End-to-end object detection with transformers. *ArXiv*, abs/2005.12872, 2020. 1, 3, 4
- [2] Kai Chen, Jiangmiao Pang, Jiaqi Wang, Yu Xiong, Xiaoxiao Li, Shuyang Sun, Wansen Feng, Ziwei Liu, Jianping Shi, Wanli Ouyang, et al. Hybrid task cascade for instance segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4974–4983, 2019. 7
- [3] Kai Chen, Jiaqi Wang, Jiangmiao Pang, Yuhang Cao, Yu Xiong, Xiaoxiao Li, Shuyang Sun, Wansen Feng, Ziwei Liu, Jiarui Xu, et al. Mmdetection: Open mmlab detection toolbox and benchmark. *arXiv preprint arXiv:1906.07155*, 2019. 5
- [4] Qiang Chen, Xiaokang Chen, Gang Zeng, and Jingdong Wang. Group detr: Fast training convergence with decoupled one-to-many label assignment. *arXiv preprint arXiv:2207.13085*, 2022. 2, 3, 6
- [5] Qiang Chen, Jian Wang, Chuchu Han, Shan Zhang, Zexian Li, Xiaokang Chen, Jiahui Chen, Xiaodi Wang, Shuming Han, Gang Zhang, et al. Group detr v2: Strong object detector with encoder-decoder pretraining. *arXiv preprint arXiv:2211.03594*, 2022. 7
- [6] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *2009 IEEE conference on computer vision and pattern recognition*, pages 248–255. Ieee, 2009. 7
- [7] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, Jakob Uszkoreit, and Neil Houlsby. An image is worth 16x16 words: Transformers for image recognition at scale. *ArXiv*, abs/2010.11929, 2021. 7
- [8] Ziteng Gao, Limin Wang, Bing Han, and Sheng Guo. Adamixer: A fast-converging query-based object detector. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5364–5373, 2022. 6, 12
- [9] Ross Girshick. Fast r-cnn. In *Proceedings of the IEEE international conference on computer vision*, pages 1440–1448, 2015. 1
- [10] Kaiming He, Georgia Gkioxari, Piotr Dollár, and Ross Girshick. Mask r-cnn. In *Proceedings of the IEEE international conference on computer vision*, pages 2961–2969, 2017. 1, 7, 12
- [11] Kaiming He, X. Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 770–778, 2016. 4
- [12] Syed Sameed Husain, Eng-Jon Ong, and Miroslaw Bober. Actnet: end-to-end learning of feature activations and multi-stream aggregation for effective instance image retrieval. *International Journal of Computer Vision*, 129(5):1432–1450, 2021. 2
- [13] Ding Jia, Yuhui Yuan, Haodi He, Xiaopei Wu, Haojun Yu, Weihong Lin, Lei Sun, Chao Zhang, and Han Hu. Detsr with hybrid matching. *arXiv preprint arXiv:2207.13080*, 2022. 2, 3, 5, 6, 8, 12
- [14] Kang Kim and Hee Seok Lee. Probabilistic anchor assignment with iou prediction for object detection. In *European Conference on Computer Vision*, pages 355–371. Springer, 2020. 1, 3, 7, 12
- [15] 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. 3, 5, 6, 12
- [16] 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. 7, 12
- [17] Yanghao Li, Hanzi Mao, Ross Girshick, and Kaiming He. Exploring plain vision transformer backbones for object detection. *arXiv preprint arXiv:2203.16527*, 2022. 3
- [18] 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. 1, 2, 4, 7, 12
- [19] 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 *European conference on computer vision*, pages 740–755. Springer, 2014. 5
- [20] Jihao Liu, Xin Huang, Yu Liu, and Hongsheng Li. Mixmim: Mixed and masked image modeling for efficient visual representation learning. *arXiv preprint arXiv:2205.13137*, 2022. 2, 7, 12
- [21] 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. *arXiv preprint arXiv:2201.12329*, 2022. 2, 5, 6, 12
- [22] Ze Liu, Han Hu, Yutong Lin, Zhuliang Yao, Zhenda Xie, Yixuan Wei, Jia Ning, Yue Cao, Zheng Zhang, Li Dong, et al. Swin transformer v2: Scaling up capacity and resolution. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 12009–12019, 2022. 7
- [23] Ze Liu, Yutong Lin, Yue Cao, Han Hu, Yixuan Wei, Zheng Zhang, S. Lin, and B. Guo. Swin transformer: Hierarchical vision transformer using shifted windows. *ArXiv*, abs/2103.14030, 2021. 2, 6, 7
- [24] 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 3651–3660, 2021. 2, 5, 6, 12
- [25] Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. Faster r-cnn: Towards real-time object detection with region proposal networks. *Advances in neural information processing systems*, 28, 2015. 1, 2, 4, 7, 12

- [26] Hamid Rezatofighi, Nathan Tsoi, JunYoung Gwak, Amir Sadeghian, Ian Reid, and Silvio Savarese. Generalized intersection over union: A metric and a loss for bounding box regression. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 658–666, 2019. 12
- [27] Ramprasaath R Selvaraju, Michael Cogswell, Abhishek Das, Ramakrishna Vedantam, Devi Parikh, and Dhruv Batra. Grad-cam: Visual explanations from deep networks via gradient-based localization. In *Proceedings of the IEEE international conference on computer vision*, pages 618–626, 2017. 8, 11
- [28] 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 *Proceedings of the IEEE/CVF international conference on computer vision*, pages 8430–8439, 2019. 6
- [29] 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. 1, 2, 3, 4, 7, 12
- [30] Wenhui Wang, Hangbo Bao, Li Dong, Johan Bjorck, Zhiliang Peng, Qiang Liu, Kriti Aggarwal, Owais Khan Mohammed, Saksham Singhal, Subhojit Som, et al. Image as a foreign language: Beit pretraining for all vision and vision-language tasks. *arXiv preprint arXiv:2208.10442*, 2022. 7
- [31] Yingming Wang, Xiangyu Zhang, Tong Yang, and Jian Sun. Anchor detr: Query design for transformer-based detector. In *Proceedings of the AAAI conference on artificial intelligence*, volume 36, pages 2567–2575, 2022. 6, 12
- [32] Yixuan Wei, Han Hu, Zhenda Xie, Zheng Zhang, Yue Cao, Jianmin Bao, Dong Chen, and Baining Guo. Contrastive learning rivals masked image modeling in fine-tuning via feature distillation. *arXiv preprint arXiv:2205.14141*, 2022. 7
- [33] Zeyue Xue, Jianming Liang, Guanglu Song, Zhuofan Zong, Liang Chen, Yu Liu, and Ping Luo. Large-batch optimization for dense visual predictions. In *Advances in Neural Information Processing Systems*, 2022. 1
- [34] Jianwei Yang, Chunyuan Li, and Jianfeng Gao. Focal modulation networks. *arXiv preprint arXiv:2203.11926*, 2022. 7
- [35] 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. *arXiv preprint arXiv:2203.03605*, 2022. 3, 5, 6, 7, 12
- [36] Shifeng Zhang, Cheng Chi, Yongqiang Yao, Zhen Lei, and Stan Z Li. Bridging the gap between anchor-based and anchor-free detection via adaptive training sample selection. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 9759–9768, 2020. 1, 2, 3, 4, 7, 12
- [37] Xizhou Zhu, Weijie Su, Lewei Lu, Bin Li, Xiaogang Wang, and Jifeng Dai. Deformable detr: Deformable transformers for end-to-end object detection. *arXiv preprint arXiv:2010.04159*, 2020. 2, 5, 6, 12
- [38] Zhuofan Zong, Qianggang Cao, and Biao Leng. Rcnet: Reverse feature pyramid and cross-scale shift network for object detection. In *Proceedings of the 29th ACM International Conference on Multimedia*, pages 5637–5645, 2021. 1

#convs	0	1	2	3	4	5
AP	41.8	<b>42.3</b>	41.9	42.1	<b>42.3</b>	42.0

Table 11. Influence of number of convolutions in auxiliary head.

$\lambda_1$	$\lambda_2$	#epochs	AP	AP <sub>S</sub>	AP <sub>M</sub>	AP <sub>L</sub>
0.25	2.0	36	46.2	28.3	49.7	60.4
0.5	2.0	36	46.6	29.0	50.5	61.2
1.0	2.0	36	<b>46.8</b>	<b>28.1</b>	<b>50.6</b>	<b>61.3</b>
2.0	2.0	36	46.1	27.4	49.7	61.4
1.0	1.0	36	46.1	27.9	49.7	60.9
1.0	2.0	36	<b>46.8</b>	<b>28.1</b>	<b>50.6</b>	<b>61.3</b>
1.0	3.0	36	46.5	29.3	50.4	61.4
1.0	4.0	36	46.3	29.0	50.1	61.0

Table 12. Results of hyper-parameter tuning for  $\lambda_1$  and  $\lambda_2$ .

Branch	NMS	#epochs	$K=0$	$K=1$	$K=2$
Deformable-DETR++	✗	12	47.1	48.7	49.5
ATSS	✓	12	46.8	47.4	48.0
Faster-RCNN	✓	12	45.9	—	46.7

Table 13. Collaborative training consistently improves performances of all branches on Deformable-DETR++ with ResNet-50.

## A. More ablation studies

**The number of stacked convolutions.** Table 11 reveals our method is robust for the number of stacked convolutions in the auxiliary head (trained for 12 epochs). Concretely, we simply choose only 1 shared convolution to enable lightweight while achieving higher performance.

**Loss weights of collaborative training.** Experimental results related to weighting the coefficient  $\lambda_1$  and  $\lambda_2$  are presented in Table 12. We find the proposed method is quite insensitive to the variations of  $\{\lambda_1, \lambda_2\}$ , since the performance slightly fluctuates when varying the loss coefficients. In summary, the coefficients  $\{\lambda_1, \lambda_2\}$  are robust and we set  $\{\lambda_1, \lambda_2\}$  to  $\{1.0, 2.0\}$  by default.

**The number of customized positive queries.** We compute the average ratio of positive samples in one-to-many label assignment to the ground-truth boxes. For instance, the ratio is 18.7 for Faster-RCNN and 8.8 for ATSS on COCO dataset, indicating more than  $8\times$  extra positive queries are introduced when  $K=1$ .

**Performances of collaborative branches.** Surprisingly, we observe the collaborative training also brings consistent gains for the auxiliary heads in Table 13. These results imply our training paradigm contributes to the consistent optimization targets, which improves the training efficiency of both decoder and auxiliary heads.

**Relations among different heads.** To better understand the diverse representations learned by multiple heads, we define the relation between head  $H_i$  and head  $H_j$  as:

$$\mathcal{S}_{i,j} = \text{KL}(\text{CAM}(H_i), \text{CAM}(H_j)), \quad (9)$$

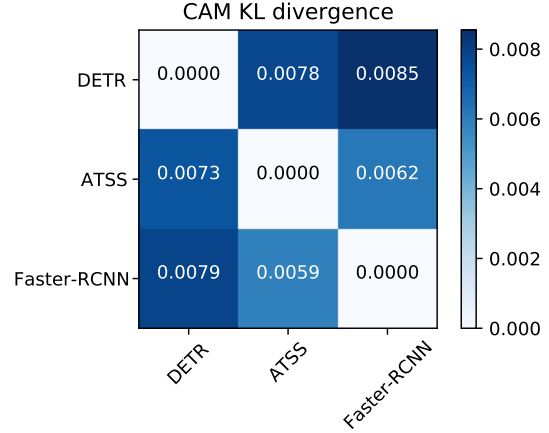


Figure 7. The relation matrix for the DETR head, ATSS head, and Faster-RCNN head. The detector is Co-Deformable-DETR ( $K=2$ ) with ResNet-50.

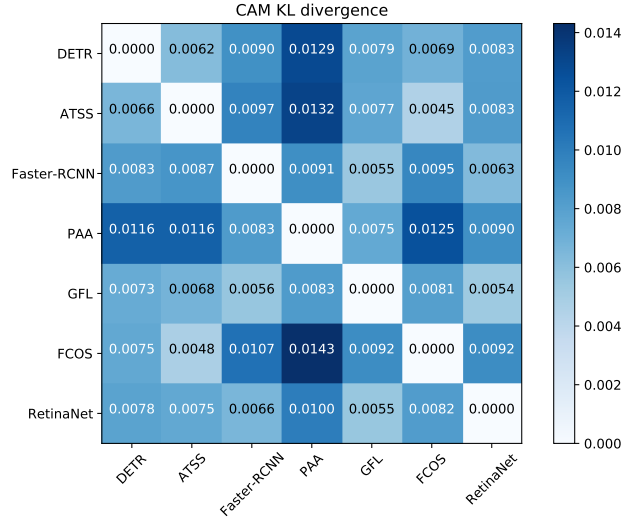


Figure 8. Distances among 7 various heads in our model with  $K=6$ .

where KL denotes Kullback–Leibler divergence and CAM stands for the class activation maps obtained by Grad-CAM [27]. We evaluate the relations for each image on COCO val and obtain the average values that are presented in Figure 7 and Figure 8. First, we find the overall distance (average value of the relation matrix that excludes diagonal values) of  $K=2$  (0.0073) is lower than the one of  $K=6$  (0.0084). This indicates the CAM areas of different heads are more similar when  $K=2$ , which is consistent with our results in Table 13. Then, the distance of our model with  $K=6$  fluctuates significantly (from 0.0045 to 0.0143). Such head diversity leads to incompatible optimization targets and hurts the detection performance.

Method	Backbone	Multi-scale	#query	#epochs	AP	AP <sub>50</sub>	AP <sub>75</sub>	AP <sub>S</sub>	AP <sub>M</sub>	AP <sub>L</sub>
Conditional-DETR [24]	R50	✗	300	108	43.0	64.0	45.7	22.7	46.7	61.5
Anchor-DETR [31]	R50	✗	300	50	42.1	63.1	44.9	22.3	46.2	60.0
DAB-DETR [21]	R50	✗	900	50	45.7	66.2	49.0	26.1	49.4	63.1
AdaMixer [8]	R50	✓	300	36	47.0	66.0	51.1	30.1	50.2	61.8
AdaMixer [8]	R101	✓	300	36	48.0	67.0	52.4	30.0	51.2	63.7
AdaMixer [8]	Swin-S	✓	300	36	51.3	71.2	55.7	34.2	54.6	67.3
Deformable-DETR [37]	R50	✓	300	50	46.9	65.6	51.0	29.6	50.1	61.6
DN-Deformable-DETR [15]	R50	✓	300	50	48.6	67.4	52.7	31.0	52.0	63.7
DINO-Deformable-DETR <sup>†</sup> [35]	R50	✓	900	12	47.9	65.3	52.1	31.2	50.9	61.9
DINO-Deformable-DETR <sup>†</sup> [35]	Swin-L (IN-22K)	✓	900	36	58.0	76.7	63.4	41.3	61.9	73.7
$\mathcal{H}$ -Deformable-DETR [13]	R50	✓	300	12	48.7	66.4	52.9	31.2	51.5	63.5
$\mathcal{H}$ -Deformable-DETR [13]	R101	✓	300	12	49.4	67.2	53.7	31.9	53.1	64.2
$\mathcal{H}$ -Deformable-DETR [13]	Swin-T	✓	300	36	53.2	71.5	58.2	35.9	56.4	68.2
$\mathcal{H}$ -Deformable-DETR [13]	Swin-S	✓	300	36	54.4	72.9	59.4	36.9	58.3	69.5
$\mathcal{H}$ -Deformable-DETR [13]	Swin-L (IN-22K)	✓	300	36	57.1	76.2	62.5	39.7	61.4	73.4
$\mathcal{H}$ -Deformable-DETR <sup>†‡</sup> [13]	Swin-L (IN-22K)	✓	900	36	57.9	76.8	63.6	42.4	61.9	73.4
Co-Deformable-DETR	R50	✓	300	12	48.7	66.6	53.2	30.5	52.2	62.8
Co-Deformable-DETR	R101	✓	300	12	49.6	67.6	54.0	31.7	53.2	64.3
Co-Deformable-DETR	Swin-B (IN-22K)	✓	300	12	55.2	74.0	60.6	38.1	59.3	71.3
Co-Deformable-DETR	Swin-L (IN-22K)	✓	300	12	56.4	75.4	62.0	38.9	60.5	72.9
Co-Deformable-DETR	Swin-T	✓	300	36	53.9	72.0	59.2	37.8	57.3	68.6
Co-Deformable-DETR	Swin-S	✓	300	36	55.0	73.5	60.5	38.5	58.8	70.2
Co-Deformable-DETR	Swin-B (IN-22K)	✓	300	36	57.0	75.7	62.3	40.3	60.9	73.2
Co-Deformable-DETR	Swin-L (IN-22K)	✓	900	36	58.1	76.6	63.8	41.3	62.1	74.0
Co-Deformable-DETR <sup>†</sup>	Swin-L (IN-22K)	✓	900	36	<b>58.3</b>	<b>77.0</b>	<b>64.0</b>	<b>42.1</b>	<b>62.3</b>	<b>74.0</b>

†: 300 predictions for evaluation. ‡: improved hyperparameters.

Table 14. Comparison to the state-of-the-art DETR variants on COCO val. We report the results of Co-Deformable-DETR with  $K = 1$ .

## B. More implementation details

**One-stage auxiliary heads.** Based on the conventional one-stage detectors, we experiment with various first-stage designs [14, 16, 18, 29, 36] for the auxiliary heads. First, we use the GIoU [26] loss for the one-stage heads. Then, the number of stacked convolutions is reduced from 4 to 1. Such modification improves the training efficiency without any accuracy drop. For anchor-free detectors, *e.g.*, FCOS [29], we assign the width of  $8 \times 2^j$  and height of  $8 \times 2^j$  for the positive coordinates with stride  $2^j$ .

**Two-stage auxiliary heads.** We adopt the RPN and RCNN as our two-stage auxiliary heads based on the popular Faster-RCNN [25] and Mask-RCNN [10] detectors. To make Co-DETR compatible with various detection heads, we adopt the same multi-scale features (stride 8 to stride 128) as the one-stage paradigm for two-stage auxiliary heads. Moreover, we adopt the GIoU loss for regression in the RCNN stage.

**Details of MixMIM-g.** Following the previous practice, we scale up the MixMIM [20] to 1-billion parameters with configurations listed below:

- channel numbers:  $C = (384, 768, 1536, 3072)$ ,
- numbers of the attention heads:  $H = (6, 12, 24, 48)$ ,

- numbers of blocks for each stage:  $B = (2, 6, 24, 2)$ .

## C. More results

As described in Table 14, we also compare Co-Deformable-DETR ( $K = 1$ ) with other state-of-the-art on COCO val. Our best model achieves a comparable performance of 58.3%, is able to outperform both DINO-Deformable-DETR (58.0%) and  $\mathcal{H}$ -Deformable-DETR (57.9%), further demonstrating the effectiveness of collaborative hybrid assignments training.