MS-DETR: Efficient DETR Training with Mixed Supervision

Chuyang Zhao[12], Yifan Sun[1], Wenhao Wang[3], Qiang Chen[1], Errui Ding[1], Yi Yang[4], Jingdong Wang[1]*
[1] Baidu VIS [2] Beihang University [3] University of Technology Sydney [4] Zhejiang University
{zhaochuyang, sunyifan01, chenqiang13}@baidu.com
wangwenhao0716@gmail.com, yangyics@zju.edu.cn
{dingerrui, wangjingdong}@baidu.com

Abstract

DETR accomplishes end-to-end object detection through iteratively generating multiple object candidates based on image features and promoting one candidate for each ground-truth object. The traditional training procedure using one-to-one supervision in the original DETR lacks direct supervision for the object detection candidates.

We aim at improving the DETR training efficiency by explicitly supervising the candidate generation procedure through mixing one-to-one supervision and one-to-many supervision. Our approach, namely MS-DETR, is simple, and places one-to-many supervision to the object queries of the primary decoder that is used for inference. In comparison to existing DETR variants with one-to-many supervision, such as Group DETR and Hybrid DETR, our approach does not need additional decoder branches or object queries; the object queries of the primary decoder in our approach directly benefit from one-to-many supervision and thus are superior in object candidate prediction. Experimental results show that our approach outperforms related DETR variants, such as DN-DETR, Hybrid DETR, and Group DETR, and the combination with related DETR variants further improves the performance. Code is available at: https://github.com/Atten4Vis/MS-DETR.

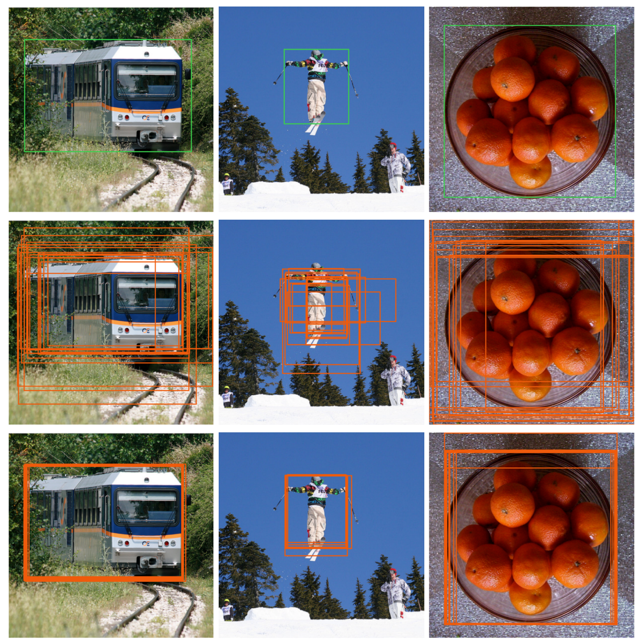

Figure 1. **Mixed supervision leads to better detection candidates.** Top: ground-truth box. Middle: candidate boxes from top-20 queries with the baseline. Bottom: candidate boxes from top-20 queries with our MS-DETR. One can see that MS-DETR generates better detection candidates than the baseline.

1. Introduction

Detection Transformer (DETR) [3], an end-to-end object detection approach, has been attracting a lot of research attention [17, 22, 23, 25, 40]. It is composed of a CNN backbone, a transformer encoder, and a transformer decoder. The decoder is a stack of decoder layers, and each layer consists of self-attention, cross-attention and FFNs, followed by class and box predictor.

The DETR decoder generates multiple object candidates that are represented in the form of object queries, and promotes one candidate and demotes other duplicate candidates for each ground-truth object in an end-to-end learning manner [3]. The duplicate candidates, which are close to the ground-truth object, are illustrated in Figure 1. The role of candidate generation is mainly taken by decoder cross-attention. The role of candidate de-duplication is mainly taken by decoder self-attention together with one-to-one supervision, ensuring the selection of a single candidate for each ground-truth object. Unlike NMS-based methods (e.g., Faster R-CNN [32]) which usually introduce a supervision for candidate generation, the DETR training procedure lacks explicit supervision for generating multiple object detection candidates.

*Corresponding author.

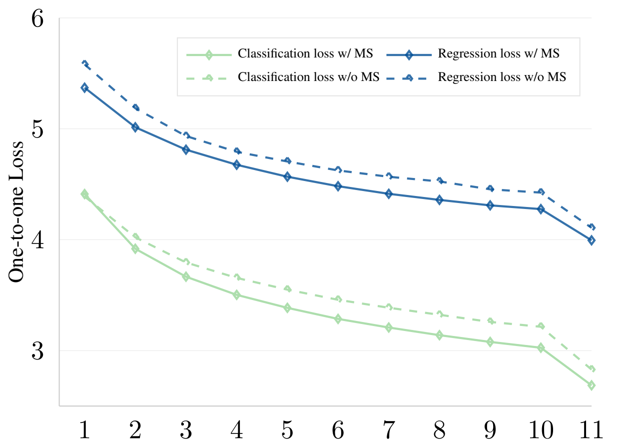

Figure 2. **Mixed supervision leads to lower one-to-one losses than the baseline**. The x-axis corresponds to #epochs, and the y-axis corresponds to the training loss from one-to-one supervision. Dashed and solid lines correspond to the loss curve of the Deformable DETR baseline and the MS-DETR, respectively. Best viewed in color.

We propose to supervise the queries of the primary decoder with the mixture of one-to-one supervision and additional one-to-many supervision to improve the training efficiency. The architecture is very simple. We add a module similar to the prediction head for one-to-one supervision, that consists of one box predictor, one class predictor for one-to-many supervision. The resulting approach, named MS-DETR, is illustrated in Figure 3. We want to point out that the additional modules only influence the training process and the inference process remains unchanged.

Figure 1 illustrates how the extra supervision influences the candidate detection. We observe that the DETR without one-to-many supervision also generates multiple candidates for each ground-truth object. After using the additional one-to-many supervision, one can see that the predicted boxes are better, implying that the candidates are better. We observe that the one-to-one classification and box regression losses are decreased when the additional one-to-many supervision is added (as illustrated in Figure 2). This provides evidence that one-to-many supervision is able to improve the candidates and thus is helpful for optimizing the one-to-one supervision loss.

Our approach improves the quality of object queries by introducing additional supervision for collecting information from image features. It is distinct from and complements related training-efficient schemes [6, 20, 24, 34, 41, 48], such as conditional DETR [24] and Deformable DETR [48], which modify cross-attention architectures or change the query forms. Our approach is related to, yet clearly different from DETR variants employing one-to-many supervision [4, 13, 14]. Specifically, our approach directly imposes one-to-many supervision on queries of the primary decoder. In contrast, Group DETR and Hybrid DETR apply supervision to queries in additional decoders other than the primary decoder. The differences from closely-related methods are illustrated in Figure 3.

Experimental results show that our approach achieves consistent improvements over DETR-based methods, including DETR variants with modified cross-attention or query formulation (Deformable DETR [48], DAB-DETR [20]), as well as other training-efficient variants (DN-DETR [14], Group DETR [4], Hybrid-DETR [13]). Combining our approach with other DETR variants with one-to-many supervision, such as Group DETR and Hybrid DETR, is able to further improve the performance, indicating that our approach is complementary to these variants. In addition, it is observed that our approach is more computation and memory-efficient as our approach does not include additional decoder branches and object queries.

2. Related Work

Decoder cross-attention and query formulation modification. Cross-attention performs interactions between image features and current object queries to refine detection candidates that are represented in the form of object queries.

Deformable DETR [48] uses deformable attention, an extension of deformable convolution [7], that selects the highly informative regions, to replace the original cross-attention architecture. Conditional DETR [24] separates the spatial and content queries and computes the spatial attention to softly select the informative regions. SMCA [8] uses the Gaussian-like weight for spatial attention computation. DAB-DETR [20] and Conditional DETR v2 [6] use boxes to represent the position of queries. Anchor DETR [38] uses anchor boxes to serve as the predefined reference region to aid in detecting objects of varying scales.

One-to-many supervision with parallel decoders. One-to-many supervision assigns one ground-truth object to multiple object queries to speed up DETR training. Existing methods depend on the additional parallel weight-sharing decoder.

DN-DETR [14] introduces parallel weight-shared decoders with each decoder handling a group of of noisy queries that are formed by by adding noises to ground-truth objects[1]. Group DETR [4, 5] instead learns the object queries of the additional decoders. DN-DETR and Group DETR perform one-to-one supervision for each group of object queries, resulting in one-to-many supervision for all the groups of object queries. DINO [43] has a similar idea to DN-DETR, where contrastive denoising queries are introduced for group-wise one-to-one supervision. DQS [45]

[1] Initially DN-DETR is motivated by one-to-one assignment stabilization. We discuss it from another perspective: parallel decoders and one-to-many supervision.

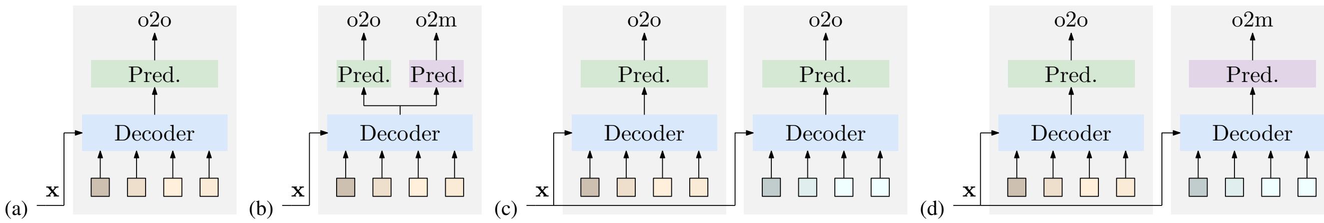

Figure 3. **Illustrating the architecture differences.** (a) Original DETR. It is trained with one-to-one supervision. (b) Our MS-DETR. It is trained by mixing one-to-one and one-to-many supervision. The two supervisions are both imposed on the primary decoder. (c) Group DETR and DN-DETR. Additional parallel decoders are introduced, and one-to-one supervision is imposed on the additional decoders. More additional decoders are possibly used in Group DETR and DN-DETR. (d) Hybrid DETR. An additional parallel decoder is added, and one-to-many supervision is imposed on the additional decoder.

adds a parallel dense query branch with one-to-many supervision alongside its distinct query branch. Co-DETR [49] and Hybrid DETR [13] adds additional parallel decoders with additional object queries, where one-to-many supervision is directly conducted on the additional decoder.

Our approach MS-DETR is related to those methods in that MS-DETR also introduces one-to-many supervision. MS-DETR clearly differs from those methods that do not modify the supervision for the original (primary) decoder. MS-DETR does not introduce additional decoders, additional queries, and performs one-to-many supervision merely on the original decoder.

DETA [28] directly performs one-to-many supervision on the same decoder with extra decoders and queries. Unfortunately, it removes one-to-one supervision and brings NMS back as post-processing. The mixture of one-to-one and one-to-many supervisions on the single decoder without using NMS is underexplored.

One-to-many supervision in traditional methods. One-to-many assignment is widely adopted in deep learning approach for object detection [9, 11, 15, 16, 19, 30, 31, 37, 44, 47]. For example, Faster R-CNN [32] and FCOS [35] forms the objective function by assigning multiple anchors and multiple center pixels for one ground-truth, followed by NMS postprocessing [26] for duplicate removal.

Our approach is partially inspired by the resemblance between DETR and traditional methods [1, 33, 36, 46]: the DETR decoder finds the candidates through cross-attention, interacting with image features, and filter out duplicate candidates through self-attention and one-to-one supervision. The latter part is similar to NMS postprocessing, and the former part is like most detectors. Thus, we introduce one-to-many supervision to the DETR decoder for improving the candidate quality.

3. MS-DETR

3.1. Preliminaries

DETR architecture. The initial DETR architecture consists of a CNN and transformer encoder, a transformer decoder, and object class and box position predictors.

The input image $\mathbf{I}$ goes through the encoder, getting the image features:

$$\mathbf{X} = \texttt{Encoder}(\mathbf{I}).$$

The learnable object queries $\mathbf{Q}$ and the image features $\mathbf{X}$ are fed into the decoder, resulting in the final object queries:

$$\tilde{\mathbf{Q}} = \texttt{Decoder}(\mathbf{Q}).$$

The object queries are parsed to the boxes and the classification scores[2] through the predictors:

$$\mathbf{B} = \texttt{box}_{11}(\tilde{\mathbf{Q}}), \quad \mathbf{S} = \texttt{cls}_{11}(\tilde{\mathbf{Q}}). \tag{1}$$

For brevity, we use the subscript $\texttt{11}$ and $\texttt{1m}$ to indicate one-to-one and one-to-many respectively.

Decoder. The transformer decoder is a stack of decoder layers. There are two main layers: a self-attention layer, which collects the information of other queries (candidates) for each query for duplicate candidate removal, a cross-attention layer, which collects the object candidates from image features in the form of queries that are fed into an FFN layer and then the box and class predictors.

One-to-one supervision. The original DETR is trained with the one-to-one supervision. One candidate prediction corresponds to one ground-truth object, and vice versa,

$$(\mathbf{y}_{\sigma(1)}, \bar{\mathbf{y}}_1), (\mathbf{y}_{\sigma(2)}, \bar{\mathbf{y}}_2), \dots, (\mathbf{y}_{\sigma(N)}, \bar{\mathbf{y}}_N), \tag{2}$$

where $\sigma(\cdot)$ is the optimal permutation of N indices, and $[\bar{\mathbf{y}}_1\ \bar{\mathbf{y}}_2, \dots\ \bar{\mathbf{y}}_N] = \bar{\mathbf{Y}}$ corresponds to ground truth, and $\mathbf{y} = [\mathbf{s}^\top\ \mathbf{b}^\top]^\top$.

The one-to-one loss function is written as follows:

$$\mathcal{L}_{11} = \sum\nolimits_{n=1}^{N} (\ell_{c11}(s_{\sigma(n)}, \bar{s}_n) + \ell_{b11}(\mathbf{b}_{\sigma(n)}, \bar{\mathbf{b}}_n)), \tag{3}$$

[2]The classification score is a combination of the degree that the candidate belongs to one class and the degree that it is better than other (duplicate) candidates.

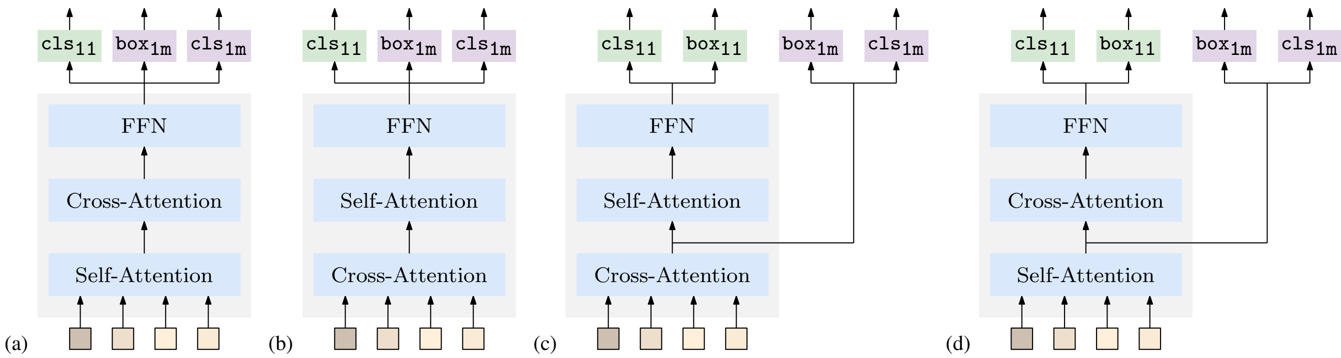

Figure 4. **MS-DETR implementations**. (a) One-to-one and one-to-many supervisions are conducted on the output object queries for each decoder layer. (b) The two supervisions are conducted on the output object queries for each decoder layer that is slightly modified: first perform cross-attention and then self-attention. (c) and (d) The one-to-many supervisions are conducted on the internal object queries. $\texttt{cls}_{11}$ and $\texttt{box}_{11}$ are class and box predictors for one-to-one supervision, and $\texttt{cls}_{1m}$, and $\texttt{box}_{1m}$ are class and box predictors for one-to-many supervision. The image features input to cross-attention are not depicted for clarity.

where $\ell_{c11}(\cdot)$ is the classification loss, and $\ell_{b11}(\cdot)$ is the box regression loss.

The one-to-one supervision helps suppress duplicate candidates and promote a single candidate per ground-truth object, by collecting information from other candidates through self-attention and comparing each candidate to the collected information. One-to-one supervision and self-attention performing interactions between object queries jointly take the role of NMS typically employed in traditional object detection methods.

3.2. Mixed Supervision

One-to-many supervision. One-to-many supervision is used in traditional detection methods to learn and provide better candidates for the NMS post-processing. For example, Faster R-CNN dynamically assigns the ground-truth object to predicted boxes if they have enough overlap with the ground-truth object. FCOS assigns the ground-truth object to the pixels at the object center.

In light of the resemblance between NMS and the joint role of self-attention and one-to-one supervision, we propose to use one-to-many assignment supervision to explicitly improve the quality of object queries and accordingly the detection candidates. We adopt an additional module for one-to-many prediction:

$$\mathbf{B} = \texttt{box}_{1m}(\tilde{\mathbf{Q}}), \quad \mathbf{S} = \texttt{cls}_{1m}(\tilde{\mathbf{Q}}) \tag{4}$$

The one-to-many loss function is given as:

$$\mathcal{L}_{1m} = \sum\nolimits_{n=1}^{N} \sum\nolimits_{i=1}^{K_n} (\ell_{c1m}(s_{n_i}, \bar{s}_n) + \ell_{b1m}(\mathbf{b}_{n_i}, \bar{\mathbf{b}}_n)),$$

where $\{(s_{n_1}, \mathbf{b}_{n_1}), (s_{n_2}, \mathbf{b}_{n_2}), \dots, (s_{n_{K_n}}, \mathbf{b}_{n_{K_n}})\}$ are assigned to the nth ground-truth object. K_n is the number of the matched predictions for the n-th ground-truth object.

One-to-many matching. One-to-many matching is based on the matching score between a prediction $(\mathbf{s}, \mathbf{b})$ (from the one-to-many predictors) and the ground-truth $(\bar{c}, \bar{\mathbf{b}})$, which is a combination of IoU and classification scores:

$$\texttt{MatchScore}(\mathbf{s}, \mathbf{b}, \bar{c}, \bar{\mathbf{b}}) = \alpha s_{\bar{c}} + (1-\alpha)\texttt{IoU}(\mathbf{b}, \bar{\mathbf{b}}),$$

Following [28], we select the top K queries in terms of the matching scores for each ground-truth object, and then filter out the queries if their matching scores are lower than a threshold τ, forming the matching query set. We also include the query obtained from one-to-one matching into the matching query set for each ground-truth, which brings a slight-better gain (+0.2 mAP).

3.3. Implementation

The additional module for one-to-many supervision consists of box and class predictors, which are identical to those used in one-to-one supervision. The box predictor is implemented as a three-layer MLP with ReLU activation, and the class predictor is implemented as a single linear layer.

A straightforward implementation (Figure 4 (a)) is to perform one-to-many prediction over the output object queries of each decoder layer, which is similar to one-to-one prediction. We merge the one-to-one box prediction into the one-to-many box prediction. The loss function for one ground-truth object consists of three parts: one-to-one classification loss, one-to-many box regression loss, and one-to-many classification loss:

$$\ell_{c11}(s_{\sigma(n)}, \bar{s}_n) + \sum\nolimits_{i=1}^{K_n} (\ell_{c1m}(s_{n_i}, \bar{s}_n) + \ell_{b1m}(\mathbf{b}_{n_i}, \bar{\mathbf{b}}_n))$$

Considering that the role of DETR cross-attention is to generate multiple candidates according to image features and the role of self-attention is to collect the information of other candidates for duplicate removal, we change the order of the components in the decoder layer from `self-attention` → `cross-attention` → `FFN` to `cross-attention` → `self-attention` → `FFN`.

This (illustrated in Figure 4 (b)) is similar to traditional methods, such as Faster R-CNN: first generate multiple candidates for each object and then remove duplicate candidates using NMS. This almost does not influence the performance[3].

We then place the one-to-many supervision over the internal object queries processed with an FFN output from cross-attention (illustrated in Figure 4 (c)). We assume that the internal object queries within the decoder layer (after cross-attention) contain much information about each individual candidate, and the output object query of the decoder layer (after self-attention) additionally contains the information about other candidates. Thus, imposing one-to-many supervision over internal object queries (output from cross-attention) potentially benefits the training, empirically verified in Table 5.

In contrast, placing one-to-many supervision over internal object queries without exchanging the order of cross-attention and self-attention (in Figure 4 (d)) leads to worse performance. The reason might be that the supervision placement is not consistent to the roles of cross-attention and self-attention: cross-attention is mainly about generating multiple candidates, and self-attention collects the information of other candidates mainly for promoting the winning candidate.

4. Experiments

4.1. Object Detection

Setting. We verify our approach on various representative DETR-based detectors, such as DAB-DETR [20], Deformable DETR [48] and its strong extension Deformable-DETR++ [13, 43] that is implemented with three additional tricks: mixed query selection, look forward twice, and zero dropout rate. We report the results in comparison to and in combination with representative DETR variants with one-to-many supervision, including DN-DETR [14], Hybrid DETR [13], Group DETR [4], and DINO [43]. We use ResNet-50 [10] as the CNN backbone. The models are trained mainly for 12 epochs and partially for 24 epochs. The models are trained on the COCO `train2017` and evaluated on the COCO `val2017`. Implementation details are in the Supplemental Material.

Comparison against DETR variants with one-to-many supervision. The results are reported in Table 1. MS-DETR brings consistent improvements on different DETR baselines. Specifically, the gains over DAB-Deformable-DETR, Deformable DETR, and Deformable DETR++ are 3.7, 3.7, and 1.8 in terms of mAP under 12 epochs.

In comparison to DETR variants with one-to-many supervision, the gains of our approach are greater than Group DETR and DN-DETR based on DAB-Deformable-DETR: +1.5 mAP vs +3.7 AP, +1.8 mAP vs +3.7 mAP. The gains are also greater than Hybrid DETR based on Deformable DETR and Deformable DETR++: +2.2 mAP vs +3.7 mAP, +1.7 mAP vs +1.8 mAP. In comparison to DINO, a strong method with one-to-many supervision with denoising queries, our improvement is also greater: +1.4 mAP vs +2.4 mAP and +0.8 mAP vs +1.1 mAP, for 12 epochs and 24 epochs, respectively.

The superiority over DETR variants with one-to-many supervision comes from that that our approach impose one-to-many supervision directly to the object queries in the primary decoder.

Combination with DETR variants with one-to-many supervision. Table 2 shows the results combining our MS-DETR with other methods with one-to-many supervision. Our method consistently improves the performance of these methods. It brings 2.0, 0.6, 1.0, 0.8 and 1.3 gains in mAP over DN-DETR(-DC5) [14], Group-DETR [4], DAC-DETR [12], Hybrid DETR [13] and DINO [43] under 12 epochs schedule, respectively. Our approach further improves the performance of DINO by 1.3 mAP under a longer training schedule (24 epochs).

These methods apply one-to-many supervision on extra queries in the extra decoder branch(es), while the queries in the primary decoder branch are still supervised in a one-to-one manner. Differently, our method directly applies one-to-many supervision on the queries in the primary decoder branch, thus achieving good complementary to these methods.

Computation and memory efficiency. Table 3 reports the computation cost and the memory cost of the baseline (Deformable DETR++ with 300 queries), Hybrid DETR, Group DETR, and our MS-DETR. The batch sizes are the same for all the methods. The training time of each epoch is obtained by averaging the time over 12 epochs.

One can see that for our approach MS-DETR, the additional time from the one-to-many supervision is minor: increase 2 minutes from the time cost of the baseline. In contrast, the additional time costs of Group DETR and Hybrid DETR are +36 and +28 minutes, much larger than our approach.

Our approach is also more memory-efficient. For instance, our method incurs only a minor increase of 127M memory (∼2%) relative to the baseline, while Hybrid DETR and Group DETR lead to substantial memory increases of nearly 60% and 40% respectively, in relation to the baseline. The reason is that Hybrid DETR and Group DETR introduce more queries and thus more computation overhead.

Convergence curves for longer training schedules. In Figure 6, we present the convergence curve of our MS-

[3] The change makes a large influence if there are fewer decoder layers.

Table 1. **Comparison of MS-DETR against other methods with one-to-many (O2M) supervision on various baselines.** MS-DETR consistently improves various popular DETR baselines. Compared with other O2M methods, our improvement is comparable (usually larger). Baseline denotes the results of the baselines without any O2M methods applied. * denotes using auxiliary denoising queries, where the number of queries is a rough approximation.

Baseline	O2M method	#epochs	#queries (primary)	#queries (extra)	extra decoder branch	mAP	AP_{50}	AP_{75}	AP_s	AP_m	AP_l
DAB-Deformable-DETR	Baseline	12	300	0	✗	44.2	62.5	47.3	27.5	47.1	58.6
	Group-DETR	12	300	3000	✓	45.7	-	-	28.1	49.0	60.6
	DN-DETR	12	300	70*	✓	46.0	63.8	49.9	27.7	49.1	62.3
	MS-DETR	12	300	0	✗	**47.9** (+**3.7**)	65.1	51.6	30.1	51.2	63.2
Deformable DETR	Baseline	12	300	0	✗	43.7	62.2	46.9	26.4	46.4	57.9
	H-DETR	12	300	1500	✓	45.9	-	-	-	-	-
	MS-DETR	12	300	0	✗	**47.6** (+**3.7**)	64.9	51.7	29.6	50.9	63.3
Deformable DETR++	Baseline	12	300	0	✗	47.0	65.3	51.0	30.1	50.5	60.7
	H-DETR	12	300	1500	✓	48.7	66.4	52.9	31.2	51.5	63.5
	MS-DETR	12	300	0	✗	**48.8** (+**1.8**)	66.2	53.2	31.5	52.3	63.7
Deformable DETR++	Baseline	12	900	0	✗	47.6	65.8	51.8	31.2	50.6	62.6
	DINO	12	900	200*	✓	49.0	66.6	53.5	32.0	52.3	63.0
	MS-DETR	12	900	0	✗	**50.0** (+**2.4**)	67.3	54.4	31.6	53.2	64.0
Deformable DETR++	Baseline	24	900	0	✗	49.8	67.0	54.2	31.4	52.8	64.1
	DINO	24	900	200*	✓	50.4	68.3	54.8	33.3	53.7	64.8
	MS-DETR	24	900	0	✗	**50.9** (+**1.1**)	68.4	56.1	34.7	54.3	65.1

Table 2. **Combination with other methods with one-to-many supervision.** MS-DETR is a complementary approach to existing O2M methods and consistently improves performance.

Model	w/ MS-DETR	#epochs	mAP	AP_{50}	AP_{75}	AP_s	AP_m	AP_l
DN-DETR		12	41.7	61.4	44.1	21.2	45.0	60.2
DN-DETR	✓	12	43.7 (+**2.0**)	62.5	46.6	22.3	47.9	62.0
DAC-DETR		12	47.1	64.8	51.1	29.2	50.6	62.4
DAC-DETR	✓	12	48.1 (+**1.0**)	65.6	52.3	30.5	51.2	63.0
Group-DETR		12	48.0	66.4	52.2	30.9	51.1	63.2
Group-DETR	✓	12	48.6 (+**0.6**)	66.0	53.1	30.3	52.2	64.1
H-DETR		12	48.7	66.4	52.9	31.2	51.5	63.5
H-DETR	✓	12	49.5 (+**0.8**)	67.0	53.8	31.2	52.7	64.0
DINO		12	49.0	66.6	53.5	32.0	52.3	63.0
DINO	✓	12	50.3 (+**1.3**)	67.4	55.1	32.7	54.0	64.6
DINO		24	50.4	68.3	54.8	33.3	53.7	64.8
DINO	✓	24	51.7 (+**1.3**)	68.7	56.5	34.0	55.4	65.5

DETR against its corresponding baselines under 12 epochs and a longer 50 epochs training schedule. We observe the introduction of mixed supervision accelerates the training convergence in both schedules. Notably, it brings a non-trivial improvement, increasing the mAP by +2.1 (from 46.9 to 49.0), even under the 50 epochs training schedule.

Combination with IoU-aware loss. We study the combination of MS-DETR with another line of work improving DETR with IoU-aware loss [2, 21, 29, 39, 42]. We apply our approach over Align-DETR [2] based on DINO baseline. Table 4 shows that MS-DETR improves the performance of Align-DETR by 0.5 AP and 0.6 AP under 12 and 24 epochs training schedules respectively. This shows that MS-DETR is also complementary to IoU-aware loss.

4.2. Ablation Study

Hyperparameters in one-to-many matching. We illustrate the influence of the three hyperparameters in one-to-many matching.

Figure 5 (a) illustrates the impact of the hyperparameter K on the selection of top-K queries. We empirically find that our approach achieves optimal performance when $K = 6$. A small value of K decreases the number of positive queries. A large value of K causes the object imbalance problem [27, 28].

Figure 5 (b) visualizes the influence of the threshold τ

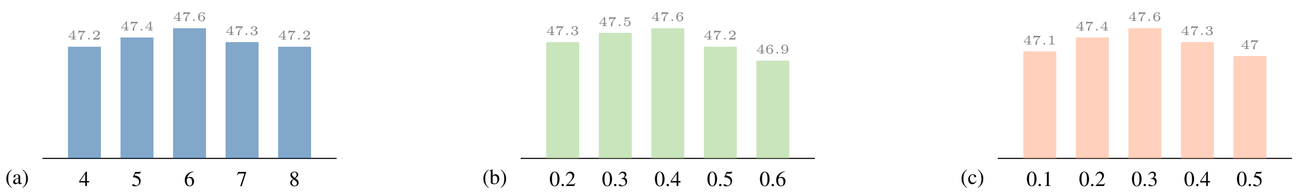

Figure 5. **Influence of the hyper-parameters for one-to-many assignment**. (a) Influence of K for selecting top-K positive queries, (b) Influence of the threshold τ to filter out low-quality queries, and (c) Influence of the matching score weight α.

Table 3. **Comparison of training time cost and memory cost.** The costs of DETR variants with one-to-many supervision, Hybrid DETR and Group DETR are reported. The baseline method is Deformable-DETR++ with ResNet50 backbone. All the methods are trained with the same batch size and on the same machine with $8\times$ V100 GPUs. Training time is the average training time of one epoch.

Method	#queries (primary)	#queries (extra)	training time	GPU Memory
Baseline	300	0	67min	5116M
Hybrid DETR	300	1500	103min	8680M
Group DETR	300	1500	95min	7128M
MS-DETR	300	0	69min	5243M

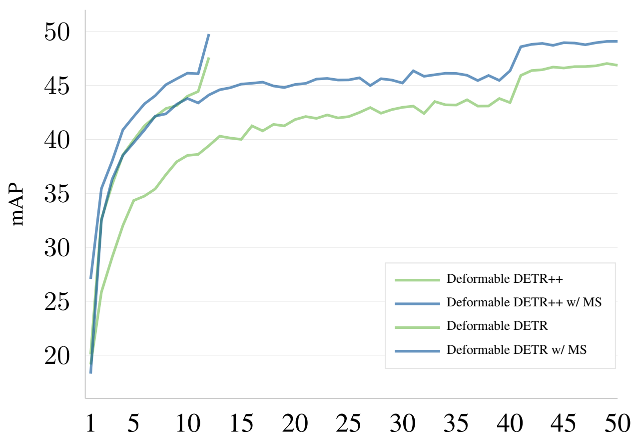

Figure 6. **Convergence curves.** MS-DETR accelerates the training process for DETR variants. Green and blue lines correspond to the baseline method and our MS counterparts. The x-axis corresponds to #epochs and the y-axis corresponds to the mAP evaluated on COCO `val2017`.

Table 4. **Combination of MS-DETR with Align-DETR.** Our approach further improves the Align-DETR performance. The Align-DETR includes the one-to-many loss like Hybrid DETR.

#epochs	w/ Align Loss	w/ Hybrid Loss	w/ MS-DETR	mAP
12	✓	✓		50.2
12	✓		✓	50.7 (**+0.5**)
24	✓	✓		51.3
24	✓		✓	51.9 (**+0.6**)

Table 5. **The effect of one-to-many supervision placement.** Deformable DETR is used as the baseline. The four ways for placing one-to-many supervision are illustrated in Figure 4. Output of `FFN` = the output object queries of each decoder layer. Internal output = the internal object queries output from the first attention of each decoder layer.

	decoder configuration	queries for supervision	mAP
Baseline	–	–	43.7
(a)	`SA` → `CA` → `FFN`	Output of `FFN`	47.0
(b)	`CA` → `SA` → `FFN`	Output of `FFN`	47.1
(c)	`CA` → `SA` → `FFN`	Internal output of `CA`	**47.6**
(d)	`SA` → `CA` → `FFN`	Internal output of `SA`	46.1

that is used to filter out low-quality queries for one-to-many supervision. Our approach achieves its best results when $\tau = 0.4$. Lowering the value of τ increases the inclusion of low-quality queries, while raising it reduces the number of positive queries eligible for one-to-many supervision.

In Figure 5 (c), we present the impact of the score weight α in the one-to-many match score. A higher value of α will increase the importance of the classification score and a lower value of α will increase the importance of the IoU score. We empirically find our method achieves the best performance when α is set to 0.4.

One-to-many supervision placement. We report the empirical results for placing one-to-many supervision over internal and output object queries in the decoder layer, as well as the two order configurations of cross-attention and self-attention in the layer. Figure 4 illustrates the four variants.

As the results shown in Table 5, the four MS-DETR variants achieve large gains over the baseline. The simple variant, directly placing the one-to-many supervision over the output object queries of each decoder layer, gets a gain of 3.3 mAP, and exchanging the order of cross-attention and self-attention does not influence the result. If placing the one-to-many supervision on the internal object queries output from cross-attention for the configuration of `cross-attention` → `self-attention` → `FFN`, a further gain 0.6 is obtained, confirming analysis in Sec. 3.3.

Weight sharing for predictors of one-to-many and one-to-one supervision. We perform empirical analysis for sharing weights of box and class predictors between one-to-many and one-to-one supervision.

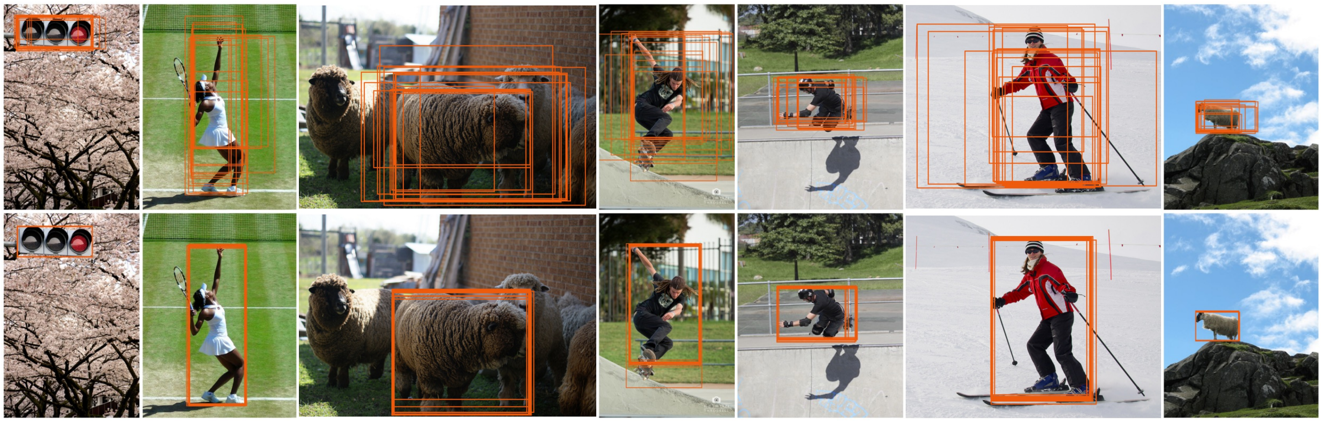

Figure 7. **More Examples illustrating better candidates from one-to-many supervision.** We visualized the detection results of the top-20 candidates for one ground-truth object. Top: the Deformable DETR++ baseline trained with one-to-one supervision. Bottom: our MS-DETR trained with mixed one-to-one and one-to-many supervision. One can see that the quality of the candidates is better with mixed supervision.

Table 6. **The effect of sharing the weights** of box and class predictors between one-to-many supervision and one-to-one supervision.

box predictor	✗	✗	✓	✓
cls predictor	✗	✓	✗	✓
mAP	47.2	47.0	47.4	**47.6**

As demonstrated in Table 6, one can see that sharing the weights for both box and class predictors gets the best performance. It is relatively easily understandable for sharing the weights of the box predictor: both the box predictors for one-to-one and one-to-many supervision need to extract the same features for box prediction, and sharing weights in some sense adds more supervision.

We assume that sharing the weights of the class predictor leads to (1) adding more supervision for training some weights in the predictor that are useful for both one-to-one and one-to-many classification, (2) leaving the weights for scoring the duplicate candidates learned from one-to-one supervision which do not influence the prediction for one-to-many supervision.

Illustration of better candidate prediction from one-to-many supervision. In Figure 7, we present more examples to illustrate the improvement of candidate predictions achieved through one-to-many supervision. The predictions are obtained from the final object queries. The detection results of the top-20 candidate queries with respect to the IoU scores are visualized. In the top row, we showcase the detection results obtained by the Deformable DETR baseline, which is trained only with one-to-one supervision. The bottom row displays detection results obtained by our MS-DETR. One can see that the candidates produced under mixed supervision exhibit superior quality, demonstrating the effectiveness of our approach in enhancing the quality of the candidates.

Table 7. **Instance segmentation** on the COCO-2017 *val* set [18]. The results are obtained with ResNet50 [10] and 12 and 50 epochs.

Epochs	w/ MS-DETR	Mask mAP	Box mAP
12		28.3	43.8
12	✓	31.5 (+**3.2**)	47.1 (+**3.3**)
50		32.2	45.6
50	✓	34.7 (+**2.5**)	48.3 (+**2.7**)

4.3. Application to Instance Segmentation

We report the results for the problem of instance segmentation, to further demonstrate the effectiveness. We report the instance segmentation results over Mask-Deformable-DETR [13] baseline on the COCO-2017 `val` set. We run experiments for 12 and 50 epochs based on ResNet50 [10] backbone. Table 7 shows that MS-DETR significantly improves the mask mAP of the baseline by 3.2 mAP under 12 epochs training schedule. It can still improve the mask mAP of the baseline by 2.5 mAP under a longer 50 epochs training schedule.

5. Conclusion

Our approach mixes an additional one-to-many supervision with the original one-to-one supervision for DETR training. The improvement implies that the additional one-to-many supervision benefits the optimization for one-to-one supervision. One main characteristic is that our approach explicitly supervises the object queries. Our approach is complementary to related methods that mainly modify the cross-attention architecture or learn the decoder weights with additional queries or additional decoders.

References

[1] Alexey Bochkovskiy, Chien-Yao Wang, and Hong-Yuan Mark Liao. Yolov4: Optimal speed and accuracy of object detection. *arXiv preprint arXiv:2004.10934*, 2020. 3

[2] Zhi Cai, Songtao Liu, Guodong Wang, Zheng Ge, Xiangyu Zhang, and Di Huang. Align-detr: Improving detr with simple iou-aware bce loss. *arXiv preprint arXiv:2304.07527*, 2023. 6

[3] Nicolas Carion, Francisco Massa, Gabriel Synnaeve, Nicolas Usunier, Alexander Kirillov, and Sergey Zagoruyko. End-to-end object detection with transformers. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part I 16*, pages 213–229. Springer, 2020. 1

[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, 5

[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. 2

[6] Xiaokang Chen, Fangyun Wei, Gang Zeng, and Jingdong Wang. Conditional detr v2: Efficient detection transformer with box queries. *arXiv preprint arXiv:2207.08914*, 2022. 2

[7] Jifeng Dai, Haozhi Qi, Yuwen Xiong, Yi Li, Guodong Zhang, Han Hu, and Yichen Wei. Deformable convolutional networks. In *Proceedings of the IEEE international conference on computer vision*, pages 764–773, 2017. 2

[8] Peng Gao, Minghang Zheng, Xiaogang Wang, Jifeng Dai, and Hongsheng Li. Fast convergence of detr with spatially modulated co-attention. In *ICCV*, pages 3621–3630, 2021. 2

[9] Zheng Ge, Songtao Liu, Feng Wang, Zeming Li, and Jian Sun. Yolox: Exceeding yolo series in 2021. *arXiv preprint arXiv:2107.08430*, 2021. 3

[10] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016. 5, 8

[11] Kaiming He, Georgia Gkioxari, Piotr Dollár, and Ross Girshick. Mask r-cnn, 2018. 3

[12] Zhengdong Hu, Yifan Sun, Jingdong Wang, and Yi Yang. Dac-detr: Divide the attention layers and conquer. 2023. 5

[13] Ding Jia, Yuhui Yuan, Haodi He, Xiaopei Wu, Haojun Yu, Weihong Lin, Lei Sun, Chao Zhang, and Han Hu. Detrs with hybrid matching. *arXiv preprint arXiv:2207.13080*, 2022. 2, 3, 5, 8

[14] 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. 2, 5

[15] 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. 3

[16] Xiang Li, Wenhai Wang, Xiaolin Hu, Jun Li, Jinhui Tang, and Jian Yang. Generalized focal loss v2: Learning reliable localization quality estimation for dense object detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 11632–11641, 2021. 3

[17] Yanghao Li, Hanzi Mao, Ross Girshick, and Kaiming He. Exploring plain vision transformer backbones for object detection. In *European Conference on Computer Vision*, pages 280–296. Springer, 2022. 1

[18] Tsung-Yi Lin, Michael Maire, Serge Belongie, Lubomir Bourdev, Ross Girshick, James Hays, Pietro Perona, Deva Ramanan, C. Lawrence Zitnick, and Piotr Dollár. Microsoft coco: Common objects in context, 2015. 8

[19] 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. 3

[20] 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

[21] Shilong Liu, Tianhe Ren, Jiayu Chen, Zhaoyang Zeng, Hao Zhang, Feng Li, Hongyang Li, Jun Huang, Hang Su, Jun Zhu, et al. Detection transformer with stable matching. *arXiv preprint arXiv:2304.04742*, 2023. 6

[22] Yingfei Liu, Tiancai Wang, Xiangyu Zhang, and Jian Sun. Petr: Position embedding transformation for multi-view 3d object detection. In *European Conference on Computer Vision*, pages 531–548. Springer, 2022. 1

[23] Yingfei Liu, Junjie Yan, Fan Jia, Shuailin Li, Aqi Gao, Tiancai Wang, and Xiangyu Zhang. Petrv2: A unified framework for 3d perception from multi-camera images. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 3262–3272, 2023. 1

[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

[25] Ishan Misra, Rohit Girdhar, and Armand Joulin. An end-to-end transformer model for 3d object detection. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 2906–2917, 2021. 1

[26] Alexander Neubeck and Luc Van Gool. Efficient non-maximum suppression. In *18th international conference on pattern recognition (ICPR'06)*, pages 850–855. IEEE, 2006. 3

[27] Kemal Oksuz, Baris Can Cam, Sinan Kalkan, and Emre Akbas. Imbalance problems in object detection: A review. *IEEE transactions on pattern analysis and machine intelligence*, 43(10):3388–3415, 2020. 6

[28] Jeffrey Ouyang-Zhang, Jang Hyun Cho, Xingyi Zhou, and Philipp Krähenbühl. Nms strikes back. *arXiv preprint arXiv:2212.06137*, 2022. 3, 4, 6

[29] Yifan Pu, Weicong Liang, Yiduo Hao, Yuhui Yuan, Yukang Yang, Chao Zhang, Han Hu, and Gao Huang. Rank-detr for high quality object detection. *arXiv preprint arXiv:2310.08854*, 2023. 6

[30] Joseph Redmon, Santosh Divvala, Ross Girshick, and Ali Farhadi. You only look once: Unified, real-time object detection. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 779–788, 2016. 3

[31] Joseph Redmon, Santosh Divvala, Ross Girshick, and Ali Farhadi. You only look once: Unified, real-time object detection. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 779–788, 2016. 3

[32] 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, 3

[33] Peize Sun, Rufeng Zhang, Yi Jiang, Tao Kong, Chenfeng Xu, Wei Zhan, Masayoshi Tomizuka, Lei Li, Zehuan Yuan, Changhu Wang, et al. Sparse r-cnn: End-to-end object detection with learnable proposals. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 14454–14463, 2021. 3

[34] Zhiqing Sun, Shengcao Cao, Yiming Yang, and Kris M Kitani. Rethinking transformer-based set prediction for object detection. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 3611–3620, 2021. 2

[35] Zhi Tian, Chunhua Shen, Hao Chen, and Tong He. Fcos: Fully convolutional one-stage object detection, 2019. 3

[36] Chien-Yao Wang, Alexey Bochkovskiy, and Hong-Yuan Mark Liao. Scaled-yolov4: Scaling cross stage partial network. In *Proceedings of the IEEE/cvf conference on computer vision and pattern recognition*, pages 13029–13038, 2021. 3

[37] Chien-Yao Wang, Alexey Bochkovskiy, and Hong-Yuan Mark Liao. Yolov7: Trainable bag-of-freebies sets new state-of-the-art for real-time object detectors. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7464–7475, 2023. 3

[38] Yingming Wang, Xiangyu Zhang, Tong Yang, and Jian Sun. Anchor detr: Query design for transformer-based detector. *national conference on artificial intelligence*, 2021. 2

[39] Shengkai Wu, Jinrong Yang, Xinggang Wang, and Xiaoping Li. Iou-balanced loss functions for single-stage object detection. *Pattern Recognition Letters*, 156:96–103, 2022. 6

[40] Chenyu Yang, Yuntao Chen, Hao Tian, Chenxin Tao, Xizhou Zhu, Zhaoxiang Zhang, Gao Huang, Hongyang Li, Yu Qiao, Lewei Lu, et al. Bevformer v2: Adapting modern image backbones to bird's-eye-view recognition via perspective supervision. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 17830–17839, 2023. 1

[41] Zhuyu Yao, Jiangbo Ai, Boxun Li, and Chi Zhang. Efficient detr: Improving end-to-end object detector with dense prior. *arXiv preprint arXiv:2104.01318*, 2021. 2

[42] Haoyang Zhang, Ying Wang, Feras Dayoub, and Niko Sunderhauf. Varifocalnet: An iou-aware dense object detector. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 8514–8523, 2021. 6

[43] 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. 2, 5

[44] 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. 3

[45] Shilong Zhang, Xinjiang Wang, Jiaqi Wang, Jiangmiao Pang, Chengqi Lyu, Wenwei Zhang, Ping Luo, and Kai Chen. Dense distinct query for end-to-end object detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7329–7338, 2023. 2

[46] Zhaohui Zheng, Ping Wang, Wei Liu, Jinze Li, Rongguang Ye, and Dongwei Ren. Distance-iou loss: Faster and better learning for bounding box regression. In *Proceedings of the AAAI conference on artificial intelligence*, pages 12993–13000, 2020. 3

[47] Xingyi Zhou, Dequan Wang, and Philipp Krähenbühl. Objects as points. *arXiv preprint arXiv:1904.07850*, 2019. 3

[48] 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

[49] Zhuofan Zong, Guanglu Song, and Yu Liu. Detrs with collaborative hybrid assignments training. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 6748–6758, 2023. 3