Hybrid Proposal Refiner:
Revisiting DETR Series from the Faster R-CNN Perspective

Jinjing Zhao[*]   Fangyun Wei[*]   Chang Xu
The University of Sydney
jzha0100@uni.sydney.edu.au   fwei8714@uni.sydney.edu.au   c.xu@sydney.edu.au

Abstract

With the transformative impact of the Transformer, DETR pioneered the application of the encoder-decoder architecture to object detection. A collection of follow-up research, e.g., Deformable DETR, aims to enhance DETR while adhering to the encoder-decoder design. In this work, we revisit the DETR series through the lens of Faster R-CNN. We find that the DETR resonates with the underlying principles of Faster R-CNN's RPN-refiner design but benefits from end-to-end detection owing to the incorporation of Hungarian matching. We systematically adapt the Faster R-CNN towards the Deformable DETR, by integrating or repurposing each component of Deformable DETR, and note that Deformable DETR's improved performance over Faster R-CNN is attributed to the adoption of advanced modules such as a superior proposal refiner (e.g., deformable attention rather than RoI Align). When viewing the DETR through the RPN-refiner paradigm, we delve into various proposal refinement techniques such as deformable attention, cross attention, and dynamic convolution. These proposal refiners cooperate well with each other; thus, we synergistically combine them to establish a Hybrid Proposal Refiner (HPR). Our HPR is versatile and can be incorporated into various DETR detectors. For instance, by integrating HPR to a strong DETR detector, we achieve an AP of 54.9 on the COCO benchmark, utilizing a ResNet-50 backbone and a 36-epoch training schedule. Code and models are available at https://github.com/ZhaoJingjing713/HPR.

1. Introduction

Since its debut in 2017, the Transformer [46] has revolutionized a wide range of NLP tasks and has swiftly expanded its influence into the realm of computer vision, proving instrumental in tasks such as image recognition [5, 10, 12, 17, 34, 45] and object detection [4, 22, 31, 32, 37, 49, 55, 57, 59, 60]. The DEtection TRansformer (DETR) [4] stands at the forefront, being the first to adapt the Transformer's encoder-decoder architecture for the object detection task. DETR's innovation lies in its object queries, which engage with CNN-generated feature maps to concurrently predict an object's category and its spatial location. A notable feature of DETR is its ability to perform detection in an end-to-end manner, a function facilitated by the integration of Hungarian matching. Despite these advancements, DETR is hindered by suboptimal training efficiency and performance. To address these shortcomings, subsequent research has been geared toward enhancing DETR while maintaining the integrity of its original encoder-decoder architecture. Among these advancements, Deformable DETR [59] is a prominent example, promoting DETR's capabilities by incorporating a deformable encoder and a deformable attention mechanism.

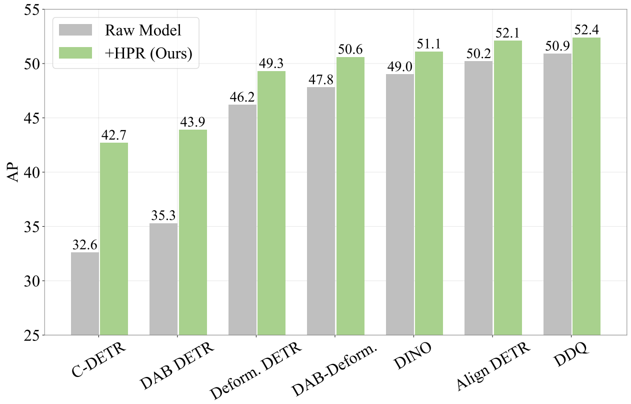

Figure 1. Applying Hybrid Proposal Refiner (HPR) to the DETR series including Conditional DETR [37], DAB DETR [32], Deformable DETR [59], DAB-Deformable DETR [32], DINO [55], Align DETR [3] and DDQ [57] on COCO dataset. All models use a ResNet-50 backbone and a 12-epoch training schedule. For efficiency, we use 300 queries for DDQ [57] and DDQ equipped with HPR.

Before the advent of DETR, Faster R-CNN [42] was commonly viewed as the seminal model for object detec-

[*] Equal contribution.

tion. It divides the detection framework into several distinct components including the backbone network, the neck network, the region proposal network (RPN), and a second-stage [19, 42] or multiple-stage [2, 6] proposal refiner. The architecture of Faster R-CNN can be described as an "RPN-refiner" setup. In this structure, the RPN initially generates a collection of object proposals. Subsequently, the proposal refiner, namely the R-CNN head, undertakes the task of categorizing each proposal and more accurately adjusting their spatial coordinates.

In this work, we revisit DETR series from the Faster R-CNN perspective. We posit that the *encoder-decoder* structure of the DETR series can be conceptualized as a refined version of the *RPN-refiner* paradigm utilized by the Faster R-CNN. we select Deformable DETR [59] with a ResNet-50 backbone as our primary model due to its widespread acclaim and exceptional performance. As shown in Table 1, we systematically adapt the Faster R-CNN towards the Deformable DETR, by integrating each component of the Deformable DETR to the Faster R-CNN. These adaptations span various aspects, including RPN modification (from a class-agnostic to a class-aware RPN), revisions to the neck network (from an FPN to a more capable deformable encoder), improvements to the proposal refiner (from an R-CNN to more advanced refiners like deformable attention), an increase in the stages of refinement (from a two-stage to a multi-stage process), and a transformation in the positive sample matching approach (from an IoU-based one-to-many strategy to a one-to-one Hungarian matching method).

Our research yields three primary insights: (1) The application of the Hungarian matching to the Faster R-CNN notably impedes its performance. This decrease in performance is primarily because Hungarian matching sharpens feature map activations, which causes the RoI Align operation to extract a regional feature map that includes an excess of non-essential information. (2) Using object features extracted by the neck network instead of the R-CNN features produced by RoI Align significantly mitigates the decline in performance when using Hungarian matching. Thus, a modified version of Faster R-CNN can also enjoy the advantage of end-to-end detection. (3) The performance enhancement of Deformable DETR over Faster R-CNN can be largely attributed to its integration of advanced components, notably the proposal refiner (employing deformable attention in place of RoI Align) and the enhanced neck network (utilizing a deformable encoder rather than a traditional FPN).

So far, we have effectively adapted the Faster R-CNN framework into the Deformable DETR, and we have identified the improvement of Deformable DETR over Faster R-CNN is attributable to the more sophisticated neck network and the more advanced proposal refiner. Typically, an object detector is equipped with a single neck network, but it may utilize numerous proposal refiners. Our study delves into an array of proposal refiners, each offering a distinct approach to processing and refining object proposals generated by the RPN. More precisely, our thorough examination includes RoI Align, dynamic convolution, cross attention, deformable attention, global attention, and object feature refinement. The empirical evidence from our experiments suggests that these object refinement mechanisms are mutually compatible and effective when used in conjunction. In light of these findings, we introduce a novel approach termed as the Hybrid Proposal Refiner (HPR), which incorporates various object refinement operators and facilitates feature interactions among them. As depicted in Figure 1, our HPR is versatile enough to be applied to a broad range of DETR models, yielding consistent improvements when compared to their vanilla versions.

Model	AP
Faster R-CNN (ResNet-50, FPN, 12-epoch)	36.5
+ Class-Agnostic RPN→Class-Aware RPN	36.1 (-0.4)
+ FPN→Deformable Encoder	44.0 (+7.9)
+ IoU Matching→Hungarian Matching (RPN)	32.7 (-11.3)
+ IoU Matching→Hungarian Matching (R-CNN)	32.2 (-0.5)
+ RoI Feature→Object Feature	41.2 (+9.0)
+ Object Feat.→Object Feat. + RoI Feat.	41.7 (+0.5)
+ Object Feat. + RoI Feat.→Deformable Attention	44.2 (+2.5)
+ 6× Deformable Attention	46.2 (+2.0)

Table 1. Step by step, we transform the Faster R-CNN [42] into the Deformable DETR [59]. We report AP on COCO benchmark. Object feature denotes RPN's point feature extracted by the neck network. Refer to Section 3.1 for more details.

The contributions of this work are threefold:

- We revisit the DETR series from the Faster R-CNN perspective, uncovering that the encoder-decoder structure in DETR series can be interpreted as analogous to the RPN-refiner paradigm of Faster R-CNN. We progressively transform the Faster R-CNN into the Deformable DETR (Table 1) and comprehensively study the key elements contributing to the improvement of Deformable DETR over Faster R-CNN.
- We conduct an extensive analysis of various proposal refinement strategies and introduce the Hybrid Proposal Refinement (HPR) technique. This innovation is compatible with many existing DETR models and consistently yields performance enhancements (Figure 1). Additionally, we introduce a novel data augmentation strategy termed data re-augmentation, which is particularly effective when used in conjunction with the proposed HPR.
- With a ResNet-50 backbone and a 36-epoch training schedule, our method attains an AP of 54.9 on COCO benchmark.

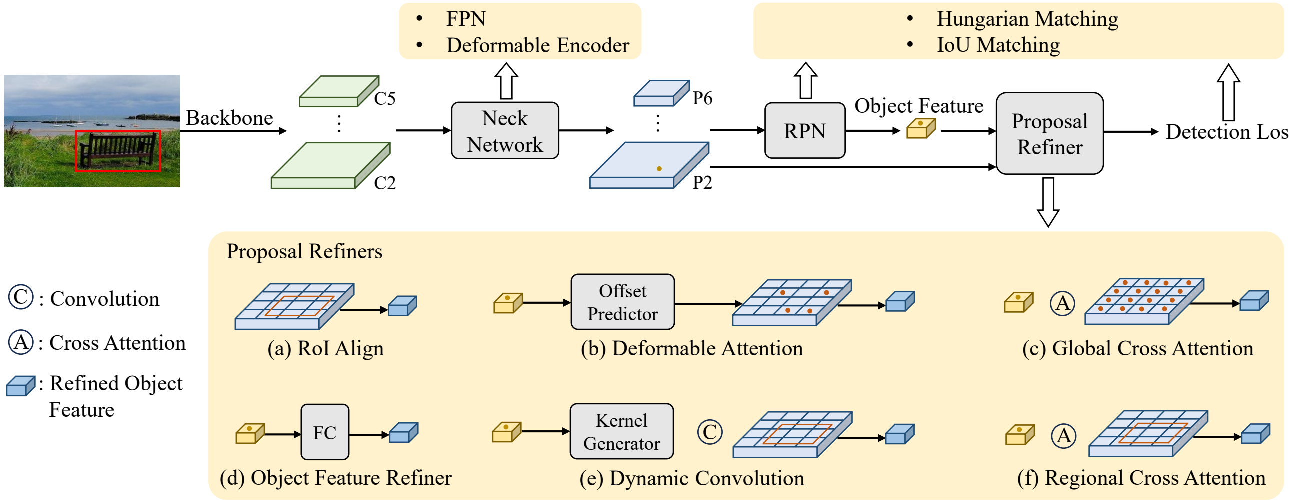

Figure 2. We regard the *encoder-decoder* structure employed by the DETR series as a refined version of the *RPN-refiner* paradigm utilized in Faster R-CNN. We investigate various elements (highlighted by yellow) that contribute to the transition from Faster R-CNN to Deformable DETR. Our hybrid proposal refiner (HPR) is predicated on exploring a multitude of proposal enhancement strategies that operate on different levels: regional (a, b, e, f), global (c), and point level (d).

2. Related work

Single-stage Detectors. Single-stage approaches have gained popularity for their simplicity and real-time performance. YOLO [1, 13, 26, 39–41, 47] stands as a seminal contribution in this domain, providing direct predictions for bounding boxes and class labels for each preset grid cell, eschewing a secondary stage for refining these proposals. Following YOLO, SSD [33] incorporates multi-level feature extraction to localize objects across various scales. Although early single-stage detectors are considered efficient, their performance is not on par with that of two-stage or multi-stage detectors. Recent advances in single-stage detectors include the development of RetinaNet, which addresses the imbalance between foreground and background samples through the application of Focal Loss [30]. This innovation enables RetinaNet to effectively learn from a large number of hard negative samples, enhancing the effectiveness of single-stage detectors. Building upon the success of Focal Loss, researchers have further explored alternative approaches to improve the performance of single-stage detectors. Anchor-free algorithms [21, 23, 25, 44, 50, 58] are proposed to make the detector simpler, offering a more straightforward and flexible architecture while still achieving competitive performance compared to anchor-based approaches. Subsequently, the introduction of ATSS [56] further unifies anchor-based and anchor-free models, and addresses the challenges of positive and negative sample selection during training, thus elevating the overall efficiency.

R-CNN Series. R-CNN [16] and Fast R-CNN [15] algorithms have been instrumental in advancing the field of object detection. They establish a two-stage framework, laying the groundwork for future innovations in this field. For instance, Faster R-CNN [42] presents the region proposal network, generating potential regions that are subsequently refined in the second stage. More recent advancements [2, 11, 36, 43, 51, 52, 54] have augmented the Faster R-CNN framework with novel architectures to boost detection capabilities. For example, Cascade R-CNN [2] extends a two-stage model by incorporating a multi-stage cascade of classifiers and regressors. Sparse R-CNN [43] replaces region proposals with a set of learnable queries, significantly diminishing computational complexity.

DETR Series. DETR [4] has emerged as a prominent approach in object detection research, introducing an innovative paradigm that leverages Transformer [46] and Hungarian algorithm [24]. Primarily because it eliminates the need for numerous manually engineered components, such as Non-Maximum Suppression (NMS), a number of follow-up studies [27, 31, 32, 37, 48, 49] develop various advanced extensions. With an aim to incorporate multi-level features into the DETR framework, Deformable DETR [59] utilizes a multi-scale deformable attention mechanism, that focuses on a small set of representative points around a reference point. It has been demonstrated that the Deformable DETR outperforms the original DETR, particularly in the detection of smaller objects. Subsequent research has contributed to advancing the field with more sophisticated designs [3, 7, 8, 22, 53, 57, 60]: DINO [55] improves accuracy by introducing a novel query denoising scheme; $\mathcal{H}$-DETR [22] and Group DETR [7] present the hybrid matching strategy, which combines the original one-to-one matching with an auxiliary one-to-many matching; $\mathcal{C}$o-DETR [60] introduces a collaborative hybrid assignment training scheme; DDQ [57] suggests that queries under the one-to-one assignment should exhibit both density and uniqueness; Align

DETR [3] incorporates a localization-precision-aware classification loss into its optimization process, and introduces a prime sample weighting mechanism to suppress the interference from unimportant samples.

3. Method

In Section 3.1, we discuss the evolution from Faster R-CNN to Deformable DETR. Building on the finding that the *encoder-decoder* structure of the DETR series can be conceptually viewed as a refined version of the *RPN-refiner* paradigm utilized by the Faster R-CNN, we introduce the Hybrid Proposal Refine (HPR) and elaborate its application to various DETR models in Section 3.2.

3.1. From Faster R-CNN to Deformable DETR

As illustrated in Figure 2, we study a number of factors that are involved in the evolution from Faster R-CNN to Deformable DETR, including the RPN, the neck network, the proposal refiners, the stages of refinement, and the positive sample matching strategy. The performance of each intermediate modification is reported in Table 1.

Faster R-CNN Baseline. Our baseline is established by employing Faster R-CNN with a ResNet-50 backbone and an FPN neck network, utilizing a 12-epoch training schedule. We adopt RoI Align to extract region features. This configuration achieves a 36.5 AP on the COCO `val` set.

Class-Agnostic RPN vs. Class-Aware RPN. We modify the class-agnostic RPN used in Faster-RCNN to be class-aware, in line with the approach taken by Deformable DETR. This results in a slight drop in performance, from 36.5 to 36.1.

Neck Network. Deformable DETR utilizes a powerful neck network known as the deformable encoder. Transformation from an FPN-style neck network to a deformable encoder enhances the AP from 36.1 to 44.0.

IoU Matching (One-to-Many) vs. Hungarian Matching (One-to-One). One of the most appealing advantages of the DETR series is its capability for end-to-end detection. This is attributed to the utilization of Hungarian matching, as opposed to the long-standing IoU-based matching strategy employed by the Faster R-CNN series. As shown in Table 1, transformation from IoU-based matching to Hungarian matching for RPN dramatically hinders the detector, resulting in a significant AP drop from 44.0 to 32.7 for this Faster R-CNN variant. In addition, the application of Hungarian matching in R-CNN yields an AP degradation of 0.5. We conjecture that the first performance drop (44.0→32.7) arises from the RoI Align operator. The use of Hungarian matching intensifies the feature map activations; however, the RoI Align operator extracts a regional feature map that includes an excess of non-essential information. To verify our hypothesis, we conduct two studies. First, we present visualizations of two activation maps in Figure 3: one map generated by a Faster R-CNN variant that uses Hungarian matching in the RPN, and another produced by a different Faster R-CNN variant that employs IoU matching in the RPN. It can be seen that the activation map of the former is much sharper than that of the latter. Next, we simply retrain a class-aware RPN with deformable encoder and the application of Hungarian matching, which can be viewed as a single-stage detector. The results presented in Table 2 show that Hungarian matching does not impede the performance of this enhanced RPN, suggesting that the object features (i.e., point features) utilized by the RPN are sufficient for object localization and classification.

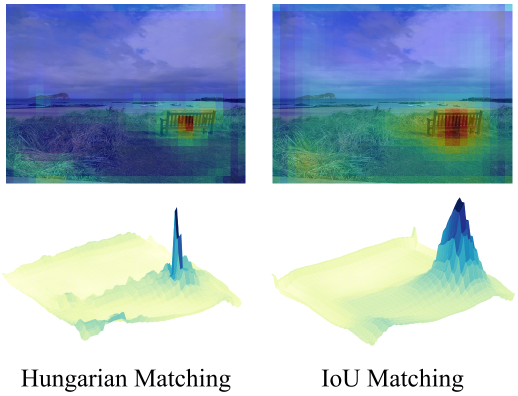

Figure 3. Visualization of two activation maps generated by variants of Faster R-CNN using either Hungarian matching or IoU matching.

Matching Strategy	AP	AP_l	AP_m	AP_s
IoU	38.3	51.2	43.2	21.1
Hungarian	38.4	48.6	42.2	24.2

Table 2. The performance of the improved class-aware RPNs with different positive sample matching strategies.

Both quantitative and qualitative results verify our hypothesis—the RoI Align operator extracts a regional feature map that includes an excess of non-essential information when introducing Hungarian matching into the Faster R-CNN.

RoI Feature vs. Object Feature. Based on the aforementioned observation, we utilize the structure of “object feature→FC” as the second stage proposal refiner instead of the structure of “RoI Align→region feature→CNN→FC” (R-CNN), yielding a significant AP improvement from 32.2 to 41.2. Additionally, as shown in Table 1, integrating the RoI features into the object features further results in an AP enhancement of 0.5. These studies indicate that a proposal refinement module, more appropriate than RoI Align, aligns effectively with the application of Hungarian matching.

More Powerful Proposal Refiner. Deformable DETR introduces a deformable attention mechanism, which enhances object features by incorporating the features derived from a set of representative points. We replace the proposal refiner from "object feature + RoI feature→FC" to a deformable decoder, which introduces a sophisticated interaction between each object feature and the feature maps extracted by the neck network (i.e., deformable encoder). As shown in Table 1, this alteration leads to a +2.5 AP improvement. Finally, by adopting 6× deformable decoders, we achieve an AP of 46.2, marking the successful transition from Faster R-CNN to Deformable DETR.

3.2. Hybrid Proposal Refiner

We have identified that the improvement of Deformable DETR over Faster R-CNN can be credited to its sophisticated neck network and its advanced proposal refiner. In general, an object detector is equipped with a single neck network, yet it may utilize numerous proposal refiners. As illustrated in Figure 2, prior to presenting our hybrid proposal refiner (HPR), we first explore other potential proposal refiners besides RoI Align (Figure 2.a) and deformable attention (Figure 2.b).

Notations. Let H and W represent the height and width of the input image, respectively. We denote the feature maps extracted by a backbone network as $\{\mathcal{C}_l \in \mathbb{R}^{H/2^l \times W/2^l \times d_l}\}$, where d_l is the feature dimension and l denotes the stage number. The feature maps encoded by the neck network[1] are denoted as $\{\mathcal{P}_l \in \mathbb{R}^{H/2^l \times W/2^l \times D}\}$, where D is the feature dimension. We use $\boldsymbol{p}_i \in \mathbb{R}^D$ to denote the object feature (i.e., point feature used by RPN) of the i-th object proposal with bounding box $\boldsymbol{b}_i = (x_i, y_i, w_i, h_i)$. We represent the RoI feature of $\boldsymbol{b}_i$ as $\boldsymbol{r}_i \in \mathbb{R}^{7\times7\times D}$. $\boldsymbol{r}_i$ is generated by the RoI Align operator.

Global Cross Attention (Figure 2.c). This mechanism is adopted by the original DETR [4], where a set of learnable object queries are introduced to gather information from $\mathcal{P}_5$ via cross attention operation. Note that using global attention is computationally expensive.

Object Feature Refinement (Figure 2.d). In Section 3.1, this strategy has been discussed in the evolution from Faster R-CNN to Deformable DETR. The object feature refiner directly processes the object features $\{\boldsymbol{p}_i\}$ to refine the proposals generated by RPN.

Dynamic Convolution (Figure 2.e). Dynamic convolution [43] enhances object features by by facilitating the interaction between each object feature $\boldsymbol{p}_i$ and the corresponding RoI feature $\boldsymbol{r}_i$. Specifically, $\boldsymbol{p}_i$ first undergoes processing by FC layers to generate convolutional kernels. Subsequently, these kernels are applied to $\boldsymbol{r}_i$ through convolution layers followed by FC layers, resulting in an enhanced object feature of $\boldsymbol{p}_i$.

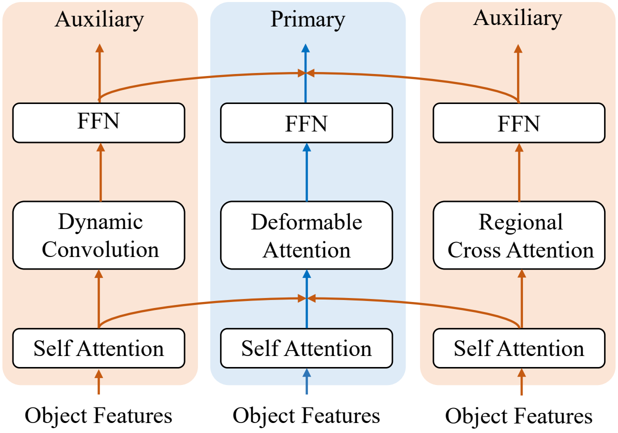

Figure 4. Illustration of the HPR module. The auxiliary refiners inject implicit information into the intermediate features of the primary refiner. We use 6× HPRs by default.

Regional Cross Attention (Figure 2.f). An alternative to perform the interaction between the object feature $\boldsymbol{p}_i$ and its RoI feature $\boldsymbol{r}_i$ is to adopt cross attention, where $\boldsymbol{p}_i$ serves as the query while the elements in $\boldsymbol{r}_i$ act as the keys and values. In this work, we refer to this object feature refinement strategy as regional cross attention.

Hybrid Proposal Refiner (HPR). Up to this point, we have explored various strategies aimed at refining proposals, which operate on different levels: global (global cross attention), regional (RoI Align, deformable attention, dynamic convolution and regional cross attention) and point level (object feature refinement). As described in Section 3.1, it has been noted that the RoI Align operation does not effectively coincide with Hungarian matching algorithm. In addition, we observe that the interaction between object features and the corresponding regional features is essential for the effectiveness of high-performance end-to-end detectors.

Unlike previous DETR models that merely include one proposal refiner, our HPR integrates the strengths of various regional proposal refinement techniques such as deformable attention, dynamic convolution, and regional cross attention. Even though these regional proposal refiners are designed to capture the most essential features of foreground objects, the methods they use to encode local features vary significantly. Deformable attention adopts a sparse set of point features. In contrast, both dynamic convolution and regional cross attention employ RoI features, but they differ in their utilization of object features: the former kernelizes the object features, while the latter regards the object features as the queries of the cross attention.

As shown in Figure 4, to take full advantage of the potential of each proposal refiner, HPR designates one refiner to function as the primary refiner, while the others act as the

[1] In this work, all modules designed to enhance the features produced by the backbone network are collectively referred to as the neck network. This includes the FPN, the Transformer-encoder, and the deformable encoder.

Method	Backbone	#Queries	#Epochs	AP	AP_{50}	AP_{75}	AP_s	AP_m	AP_l
Conditional DETR [37]	ResNet-50	300	108	43.0	64.0	45.7	22.7	46.7	61.5
Anchor DETR [49]		300	50	42.1	63.1	44.9	22.3	46.2	60.0
Efficient DETR [53]		300	50	45.1	63.1	49.1	28.3	48.4	59.0
DAB DETR [32]		900	50	45.7	66.2	49.0	26.1	49.4	63.1
Deformable DETR [59]		300	50	46.9	65.6	51.0	29.6	50.1	61.6
DN-Deformable DETR [27]		900	50	48.6	67.4	52.7	31.0	52.0	63.7
$\mathcal{H}$-Deformable DETR [22]		300	12	48.7	66.4	52.9	31.2	51.5	63.5
$\mathcal{H}$-Deformable DETR [22]		300	36	50.0	-	-	32.9	52.7	65.3
DINO [55]		900	12	49.4	66.9	53.8	32.3	52.5	63.9
DINO [55]		900	36	51.2	69.0	55.8	35.0	54.3	65.3
Group DETR [7]		900	12	50.1	-	-	32.4	53.2	64.7
Align DETR [3]		900	12	50.2	67.8	54.4	32.9	53.3	65.0
Align DETR [3]		900	24	51.3	68.2	56.1	35.5	55.1	65.6
DETA [38]		900	12	50.5	67.6	55.3	33.1	54.7	65.2
DETA [38]		900	24	51.6	69.0	56.7	34.0	55.8	66.5
DDQ [57]		900	12	51.3	68.6	56.4	33.5	54.9	65.9
DDQ [57]		900	24	52.0	69.5	57.2	35.2	54.9	65.9
Deformable DETR with HPR		900	12	50.6	68.7	55.5	34.4	53.9	63.5
Deformable DETR with HPR		900	24	51.9	70.0	57.0	35.3	55.0	65.3
DINO with HPR		900	12	51.1	68.6	55.7	34.6	54.5	64.9
DINO with HPR		900	24	51.9	69.7	56.8	34.9	55.0	65.8
Align DETR with HPR		900	12	52.1	69.6	56.9	35.6	55.4	66.6
Align DETR with HPR		900	24	52.7	69.8	57.2	35.8	56.0	66.4
Align DETR with HPR†		900	12	52.4	70.3	57.2	35.9	56.3	68.5
Align DETR with HPR†		900	24	54.2	72.1	58.8	37.8	57.9	70.0
DDQ with HPR		300	12	52.4	69.9	57.5	35.9	55.5	66.7
DDQ with HPR		300	24	52.5	69.8	57.6	35.4	55.5	67.0
DDQ with HPR†		300	12	53.0	70.6	58.0	35.3	56.3	68.6
DDQ with HPR†		300	24	54.2	72.0	59.6	37.3	57.8	69.1
DDQ with HPR†		300	36	**54.9**	**72.4**	**60.3**	**37.7**	**58.9**	**69.6**

Table 3. Comparison with state-of-the-art DETR models on the COCO `val` set utilizing a ResNet-50 backbone. Considering GPU memory utilization, we use 300 queries when applying HPR to DDQ [57]. †: the application of large-scale jitter data augmentation.

auxiliary. The auxiliary refiners inject implicit information into the intermediate features of the primary refiner. Specifically, we integrate the self-attention and FFN features from the auxiliary proposal refiner into their counterparts in the primary proposal refiner using a simple addition operator with learnable weights. In Section 4.2, we also investigate other alternatives for information integration. Note that each proposal refiner is supervised by an independent detection loss. The loss weights of the primary proposal refiner and two auxiliary refiners are set to 1.0, 0.5 and 0.5, respectively. We use the same loss function as the one employed in Deformable DETR.

Application of HPR to DETR Series. Like most DETR detectors, stacking multiple HPRs is feasible to enhance overall performance. By default, we stack 6 HPRs. Our HPR can be incorporated into various DETR detectors that only have a single proposal refiner by appending the auxiliary refiners to the primary one. Figure 1 demonstrates the consistent performance improvement.

Data Re-Augmentation. We also introduce a novel data augmentation strategy termed "data re-augmentation", which first copies data that has been augmented by normal augmentation and then applies strong augmentations, including color jitter and geometric transformations, to the copies. This yields a new training batch that contains both normally augmented images and strongly augmented images. Our data re-augmentation technique differs from batch augmentation [20] in two aspects: (1) it copies weakly augmented images rather than the raw images; (2) it applies distinct and stronger augmentations to the copies. We experimentally find this novel augmentation works well with our HPR.

4. Experiments

Dataset and Evaluation Metric. We conduct experiments on COCO [29] benchmark. It offers 118,287 labeled images

Method	Backbone	#Queries	#Epochs	AP	AP$_{50}$	AP$_{75}$	AP$_s$	AP$_m$	AP$_l$
HTC [6]	Swin-L (IN-22K)	900	36	57.1	75.6	62.5	42.4	60.7	71.1
Group-DINO [7]		900	36	58.4	-	-	41.0	62.5	73.9
DETA [38]		900	24	58.5	76.5	64.4	38.5	62.6	73.8
DINO [55]		900	12	57.5	-	-	-	-	-
DINO [55]		900	36	58.5	77.0	64.1	41.5	62.3	74.0
DDQ [57]		900	36	58.7	76.8	64.5	41.6	62.9	74.3
Mask DINO [28]		300	50	59.0	-	-	-	-	-
$\mathcal{H}$-Deformable DETR [22]		900	12	55.9	-	-	39.1	59.9	72.2
$\mathcal{H}$-Deformable DETR [22]		900	36	57.1	-	-	39.7	61.4	73.4
$\mathcal{H}$-DINO [22]		900	36	59.4	77.8	65.4	43.1	63.1	74.2
DDQ with HPR		300	12	58.7	76.7	64.5	41.5	62.5	74.6
DDQ with HPR†		300	12	58.4	76.8	64.3	41.2	62.5	75.1
DDQ with HPR†		300	24	59.3	77.6	65.0	43.1	63.4	75.5
AlignDETR with HPR		900	12	58.6	76.8	64.0	40.9	62.7	75.4
AlignDETR with HPR		900	24	59.3	77.5	64.7	41.9	63.7	75.2
AlignDETR with HPR†		900	12	58.5	76.7	63.7	41.6	62.8	76.6
AlignDETR with HPR†		900	24	59.6	77.9	64.5	42.6	64.0	**76.9**
AlignDETR with HPR†		900	36	**60.0**	**78.0**	**65.5**	**43.8**	**64.5**	76.6

Table 4. Comparison with other DETR models on the COCO `val` set utilizing a Swin-L backbone pre-trained on ImageNet-22K. Considering GPU memory utilization, we use 300 queries when applying HPR to DDQ [57]. †: the utilization of large-scale jitter.

Proposal Refiner	AP	AP$_l$	AP$_m$	AP$_s$
Global Cross Attention	42.3	56.3	45.4	26.8
RoI Align	32.2	37.3	36.9	23.6
Deformable Attention	47.8	62.0	51.2	30.6
Dynamic Convolution	48.3	62.7	51.2	32.3
Regional Cross Attention	47.6	61.5	50.6	31.7
Object Feature Refiner	41.2	51.7	44.8	26.9

Table 5. Performance of each proposal refiner.

across 80 object categories in its `train` set. The `val` set consists of 5,000 images. Following common practice, we report average precision (AP) on COCO `val` split.

Implementation Details. Our code base is built upon MMDetection [4]. Unless otherwise specified, we adopt ResNet-50 [18] pre-trained on ImageNet-1K [9] as the backbone under a 12-epoch training schedule. By default, 900 object queries are adopted. We use the AdamW [35] optimizer with a learning rate of $1e{-}4$. We adopt DETR-style normal data augmentation following [3, 7, 22, 55, 60] and the proposed data re-augmentation technique. When compared with other approaches, we utilize a larger backbone (Swin-L [34] pre-trained on ImageNet-22K [9]) and longer training schedules (24 or 36 epochs), and incorporate large-scale jitter with copy-paste technique [14] into the normal data augmentation.

4.1. Main Results

As shown in Figure 1, our HPR can be applied to various DETR detectors, including Conditional DETR [37], DAB DETR [32], Deformable DETR [59], DAB-Deformable DETR [32], DINO [55], Align DETR [3] and DDQ [57]. Models equipped with our HPR technique consistently outperform their counterparts without HPR, showing improvements ranging from +1.5 to +10.1 AP.

The comparison with state-of-the-art methods utilizing a ResNet-50 backbone is presented in Table 3. Notably, by applying HPR to a strong DETR, namely DDQ [57], we achieve an AP of 54.9 under a 36-epoch training schedule. In Table 4, we compare our method with other approaches using a Swin-L backbone. When applied to DDQ [57], and AlignDETR [3], our approach achieves AP scores of 59.3 and 60.0, respectively.

4.2. Ablation Studies

Unless otherwise specified, for all ablation studies, an enhanced Deformable DETR introduced by DINO [55] serves as our base model. It achieves 47.8 AP, using a ResNet-50 backbone, normal data augmentation and 300 object queries under a 12-epoch training schedule.

Various Proposal Refiners. In Section 3.2, we introduce a variety of proposal refiners that function at distinct levels: global (global cross attention), regional (RoI Align, deformable attention, dynamic convolution and regional cross attention) and point level (object feature refiner). The performance of each refiner is detailed in Table 5. With the exception of RoI Align and the object feature refiner, all refiners utilize a six-stage refinement process. As explored in Section 3.1, the RoI Align technique does not effectively integrate with Hungarian matching, leading to suboptimal results. The global cross attention mechanism, as proposed by the original DETR, incurs significant compu-

SA	Dedicated Module	FFN	AP
✓			48.0
	✓		46.5
		✓	49.2
✓	✓		48.6
	✓	✓	48.6
✓		✓	**49.3**
✓	✓	✓	49.1

Table 6. Ablation study on the integration of various features.

Weight	Type	Initialization	AP
Fixed	Scalar	1:1:1	48.9
Fixed	Scalar	2:1:1	49.1
Learnable	Scalar	1:1:1	48.9
Learnable	Scalar	2:1:1	48.8
Learnable	Vector	1:1:1	**49.3**
Learnable	Vector	2:1:1	49.0

Table 7. Ablation study on the integration weights.

tational overhead and poses challenges for the integration of multi-level feature maps that modern DETR detectors typically need. In contrast to the simple object feature refiner which adopts a single FC layer for object feature enhancement, deformable attention (DA), dynamic convolution (DC) and regional cross attention (RCA) exhibit superior performance. This improvement stems from their intricate architectures, which facilitate interactions between object and regional features. Thus, DA, DC and RCA are adopted in our HPR.

Feature Integration. As shown in Figure 4, the self-attention (SA) and feed-forward network (FFN) features from the auxiliary proposal refiners are integrated into their counterparts within the primary proposal refiner. Each refiner is composed of a SA layer, a dedicated module (deformable attention, dynamic convolution or regional cross attention), and a FFN layer. In Table 6, we study the effectiveness of different features for information injection, including SA features, FFN features, and features from the dedicated module. Experimentally, we find that injecting SA features and FFN features into the primary proposal refiner yields the best performance.

In addition, we examine the integration weights. Let $\boldsymbol{f}_p$, $\boldsymbol{f}_{a1}$ and $\boldsymbol{f}_{a2}$ denote the features extracted by the FFN or SA layer of the primary proposal refiner, the first auxiliary proposal refiner, and the second auxiliary proposal refiner, respectively. The corresponding refined feature $\boldsymbol{f}'_p$ is computed as $\boldsymbol{f}'_p = w_p\boldsymbol{f}_p + w_{a1}\boldsymbol{f}_{a1} + w_{a2}\boldsymbol{f}_{a2}$. In Table 7, we study several factors including: (1) whether these weights $\{w_p, w_{a1}, w_{a2}\}$ are fixed or learnable; (2) the data type of $\{w_p, w_{a1}, w_{a2}\}$ as either scalar or vector of the same dimension of $\boldsymbol{f}_p/\boldsymbol{f}_{a1}/\boldsymbol{f}_{a2}$; (3) the initial values of $\{w_p, w_{a1}, w_{a2}\}$.

HPR	Data Re-Augmentation	900 Queries	AP
			47.8
✓			49.3
✓		✓	49.8
✓	✓		50.3
✓	✓	✓	**50.6**

Table 8. Study on data re-augmentation and more object queries.

Augmentation Strategy	AP
Normal Augmentation	49.3
Strong Augmentation	48.4
Batch Augmentation [20]	49.6
Data Re-Augmentation	**50.3**

Table 9. Comparison among standard data augmentation (the first and second rows), batch augmentation [20] (the third row), and data re-augmentation (the last row).

Performance Enhancement. We introduce data re-augmentation in the end of Section 3.2. Table 8 shows the effects of incorporating data re-augmentation and increasing the number of object queries from 300 to 900. Our data re-augmentation involves first duplicating data that has undergone normal augmentation, and then applying strong augmentations to these duplicates to create a new batch. The training is conducted on the combination of the original batch (augmented by normal augmentation) and the new batch (augmented by data re-augmentation). To evaluate the effectiveness of our data re-augmentation, we compare it with the standard normal and strong augmentation, and the batch augmentation [20] in Table 9.

5. Conclusion

In this work, we revisit the DETR series from the perspective of Faster R-CNN, uncovering that the encoder-decoder structure in the DETR series can be interpreted as analogous to the RPN-refiner paradigm of Faster R-CNN. We progressively transform Faster R-CNN into Deformable DETR and identify the key elements that contribute to the improvement of Deformable DETR over Faster R-CNN. Inspired by these findings, we explore various object proposal refiners and introduce HPR. Our HPR can be applied to a number of DETR detectors and shows consistent improvements over the original versions. We also introduce a novel data augmentation technique that synergizes well with the proposed HPR. Experimentally, we achieve an AP of 54.9 on the COCO benchmark by using a ResNet-50 backbone under a 36-epoch training schedule.

Acknowledgements

This work was supported in part by the Australian Research Council under Projects DP240101848 and FT230100549.

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] Zhaowei Cai and Nuno Vasconcelos. Cascade r-cnn: Delving into high quality object detection. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 6154–6162, 2018. 2, 3

[3] 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. 1, 3, 4, 6, 7

[4] Nicolas Carion, Francisco Massa, Gabriel Synnaeve, Nicolas Usunier, Alexander Kirillov, and Sergey Zagoruyko. End-to-end object detection with transformers. In *European conference on computer vision*, pages 213–229. Springer, 2020. 1, 3, 5, 7

[5] Chun-Fu Richard Chen, Quanfu Fan, and Rameswar Panda. Crossvit: Cross-attention multi-scale vision transformer for image classification. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 357–366, 2021. 1

[6] 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. 2, 7

[7] Qiang Chen, Xiaokang Chen, Jian Wang, Shan Zhang, Kun Yao, Haocheng Feng, Junyu Han, Errui Ding, Gang Zeng, and Jingdong Wang. Group detr: Fast detr training with group-wise one-to-many assignment. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 6633–6642, 2023. 3, 6, 7

[8] Xiaokang Chen, Fangyun Wei, Gang Zeng, and Jingdong Wang. Conditional detr v2: Efficient detection transformer with box queries. *arXiv preprint arXiv:2207.08914*, 2022. 3

[9] 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

[10] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, et al. An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv preprint arXiv:2010.11929*, 2020. 1

[11] Yu Du, Fangyun Wei, Zihe Zhang, Miaojing Shi, Yue Gao, and Guoqi Li. Learning to prompt for open-vocabulary object detection with vision-language model. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 14084–14093, 2022. 3

[12] Stéphane d'Ascoli, Hugo Touvron, Matthew L Leavitt, Ari S Morcos, Giulio Biroli, and Levent Sagun. Convit: Improving vision transformers with soft convolutional inductive biases. In *International Conference on Machine Learning*, pages 2286–2296. PMLR, 2021. 1

[13] Glenn Jocher et. al. ultralytics/yolov5: v6.0 - YOLOv5n 'Nano' models, Roboflow integration, TensorFlow export, OpenCV DNN support, 2021. 3

[14] Golnaz Ghiasi, Yin Cui, Aravind Srinivas, Rui Qian, Tsung-Yi Lin, Ekin D Cubuk, Quoc V Le, and Barret Zoph. Simple copy-paste is a strong data augmentation method for instance segmentation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 2918–2928, 2021. 7

[15] Ross Girshick. Fast r-cnn. In *Proceedings of the IEEE international conference on computer vision*, pages 1440–1448, 2015. 3

[16] Ross Girshick, Jeff Donahue, Trevor Darrell, and Jitendra Malik. Rich feature hierarchies for accurate object detection and semantic segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 580–587, 2014. 3

[17] Kai Han, An Xiao, Enhua Wu, Jianyuan Guo, Chunjing Xu, and Yunhe Wang. Transformer in transformer. *Advances in Neural Information Processing Systems*, 34:15908–15919, 2021. 1

[18] 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. 7

[19] 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. 2

[20] Elad Hoffer, Tal Ben-Nun, Itay Hubara, Niv Giladi, Torsten Hoefler, and Daniel Soudry. Augment your batch: Improving generalization through instance repetition. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8129–8138, 2020. 6, 8

[21] Lichao Huang, Yi Yang, Yafeng Deng, and Yinan Yu. Densebox: Unifying landmark localization with end to end object detection. *arXiv preprint arXiv:1509.04874*, 2015. 3

[22] Ding Jia, Yuhui Yuan, Haodi He, Xiaopei Wu, Haojun Yu, Weihong Lin, Lei Sun, Chao Zhang, and Han Hu. Detrs with hybrid matching. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 19702–19712, 2023. 1, 3, 6, 7

[23] Tao Kong, Fuchun Sun, Huaping Liu, Yuning Jiang, Lei Li, and Jianbo Shi. Foveabox: Beyound anchor-based object detection. *IEEE Transactions on Image Processing*, 29:7389–7398, 2020. 3

[24] Harold W Kuhn. The hungarian method for the assignment problem. *Naval research logistics quarterly*, 2(1-2):83–97, 1955. 3

[25] Hei Law and Jia Deng. Cornernet: Detecting objects as paired keypoints. In *Proceedings of the European conference on computer vision (ECCV)*, pages 734–750, 2018. 3

[26] Chuyi Li, Lulu Li, Hongliang Jiang, Kaiheng Weng, Yifei Geng, Liang Li, Zaidan Ke, Qingyuan Li, Meng Cheng, Weiqiang Nie, et al. Yolov6: A single-stage object detection framework for industrial applications. *arXiv preprint arXiv:2209.02976*, 2022. 3

[27] 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, 6

[28] Feng Li, Hao Zhang, Huaizhe Xu, Shilong Liu, Lei Zhang, Lionel M Ni, and Heung-Yeung Shum. Mask dino: Towards a unified transformer-based framework for object detection and segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3041–3050, 2023. 7

[29] 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 *Computer Vision–ECCV 2014: 13th European Conference, Zurich, Switzerland, September 6-12, 2014, Proceedings, Part V 13*, pages 740–755. Springer, 2014. 6

[30] 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

[31] Yutong Lin, Yuhui Yuan, Zheng Zhang, Chen Li, Nanning Zheng, and Han Hu. Detr does not need multi-scale or locality design. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 6545–6554, 2023. 1, 3

[32] 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. 1, 3, 6, 7

[33] Wei Liu, Dragomir Anguelov, Dumitru Erhan, Christian Szegedy, Scott Reed, Cheng-Yang Fu, and Alexander C Berg. Ssd: Single shot multibox detector. In *Computer Vision–ECCV 2016: 14th European Conference, Amsterdam, The Netherlands, October 11–14, 2016, Proceedings, Part I 14*, pages 21–37. Springer, 2016. 3

[34] Ze Liu, Yutong Lin, Yue Cao, Han Hu, Yixuan Wei, Zheng Zhang, Stephen Lin, and Baining Guo. Swin transformer: Hierarchical vision transformer using shifted windows. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 10012–10022, 2021. 1, 7

[35] Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. *arXiv preprint arXiv:1711.05101*, 2017. 7

[36] Xin Lu, Buyu Li, Yuxin Yue, Quanquan Li, and Junjie Yan. Grid r-cnn. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7363–7372, 2019. 3

[37] 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. 1, 3, 6, 7

[38] Jeffrey Ouyang-Zhang, Jang Hyun Cho, Xingyi Zhou, and Philipp Krähenbühl. Nms strikes back. *arXiv preprint arXiv:2212.06137*, 2022. 6, 7

[39] Joseph Redmon and Ali Farhadi. Yolo9000: better, faster, stronger. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 7263–7271, 2017. 3

[40] Joseph Redmon and Ali Farhadi. Yolov3: An incremental improvement. *arXiv preprint arXiv:1804.02767*, 2018.

[41] 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

[42] 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, 3

[43] 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, 5

[44] 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. 3

[45] Hugo Touvron, Matthieu Cord, Alexandre Sablayrolles, Gabriel Synnaeve, and Hervé Jégou. Going deeper with image transformers. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 32–42, 2021. 1

[46] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. *Advances in neural information processing systems*, 30, 2017. 1, 3

[47] 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

[48] Tao Wang, Li Yuan, Yunpeng Chen, Jiashi Feng, and Shuicheng Yan. Pnp-detr: Towards efficient visual analysis with transformers. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 4661–4670, 2021. 3

[49] 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*, pages 2567–2575, 2022. 1, 3, 6

[50] Fangyun Wei, Xiao Sun, Hongyang Li, Jingdong Wang, and Stephen Lin. Point-set anchors for object detection, instance segmentation and pose estimation. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part X 16*, pages 527–544. Springer, 2020. 3

[51] Fangyun Wei, Yue Gao, Zhirong Wu, Han Hu, and Stephen Lin. Aligning pretraining for detection via object-level contrastive learning. *Advances in Neural Information Processing Systems*, 34:22682–22694, 2021. 3

[52] Xingyi Yang, Jingwen Ye, and Xinchao Wang. Factorizing knowledge in neural networks. In *European Conference on Computer Vision*, pages 73–91. Springer, 2022. 3

[53] 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. 3, 6

[54] Hongkai Zhang, Hong Chang, Bingpeng Ma, Naiyan Wang, and Xilin Chen. Dynamic r-cnn: Towards high quality object detection via dynamic training. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XV 16*, pages 260–275. Springer, 2020. 3

[55] Hao Zhang, Feng Li, Shilong Liu, Lei Zhang, Hang Su, Jun Zhu, Lionel Ni, and Heung-Yeung Shum. Dino: Detr with improved denoising anchor boxes for end-to-end object detection. In *The Eleventh International Conference on Learning Representations*, 2022. 1, 3, 6, 7

[56] 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

[57] 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. 1, 3, 6, 7

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

[59] Xizhou Zhu, Weijie Su, Lewei Lu, Bin Li, Xiaogang Wang, and Jifeng Dai. Deformable detr: Deformable transformers for end-to-end object detection. In *International Conference on Learning Representations*, 2020. 1, 2, 3, 6, 7

[60] 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. 1, 3, 7