Q-DETR: An Efficient Low-Bit Quantized Detection Transformer

Sheng Xu[1†], Yanjing Li[1†], Mingbao Lin[3], Peng Gao[4], Guodong Guo[5], Jinhu Lü[1,2], Baochang Zhang[1,2*]

[1] Beihang University   [2] Zhongguancun Laboratory   [3] Tencent

[4] Shanghai AI Laboratory   [5] UNIUBI Research, Universal Ubiquitous Co.

Abstract

The recent detection transformer (DETR) has advanced object detection, but its application on resource-constrained devices requires massive computation and memory resources. Quantization stands out as a solution by representing the network in low-bit parameters and operations. However, there is a significant performance drop when performing low-bit quantized DETR (Q-DETR) with existing quantization methods. We find that the bottlenecks of Q-DETR come from the query information distortion through our empirical analyses. This paper addresses this problem based on a distribution rectification distillation (DRD). We formulate our DRD as a bi-level optimization problem, which can be derived by generalizing the information bottleneck (IB) principle to the learning of Q-DETR. At the inner level, we conduct a distribution alignment for the queries to maximize the self-information entropy. At the upper level, we introduce a new foreground-aware query matching scheme to effectively transfer the teacher information to distillation-desired features to minimize the conditional information entropy. Extensive experimental results show that our method performs much better than prior arts. For example, the 4-bit Q-DETR can theoretically accelerate DETR with ResNet-50 backbone by 6.6× and achieve 39.4% AP, with only 2.6% performance gaps than its real-valued counterpart on the COCO dataset [1].

1. Introduction

Inspired by the success of natural language processing (NLP), object detection with transformers (DETR) has been introduced to train an end-to-end detector via a transformer encoder-decoder [4]. Unlike early works [22, 31] that often employ convolutional neural networks (CNNs) and require post-processing procedures, *e.g.*, non-maximum suppression (NMS), and hand-designed sample selection, DETR treats object detection as a direct set prediction problem.

---

† Equal contribution.

* Corresponding author: bczhang@buaa.edu.cn

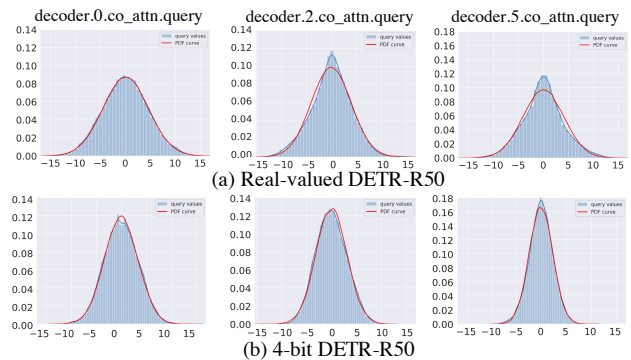[1] Code: `https://github.com/SteveTsui/Q-DETR`

Figure 1. The histogram of query values **q** (blue shadow) and corresponding PDF curves (red curve) of Gaussian distribution [17], *w.r.t* the cross attention of different decoder layers in (a) real-valued DETR-R50, and (b) 4-bit quantized DETR-R50 (baseline). Gaussian distribution is generated from the statistical mean and variance of the query values. The query in quantized DETR-R50 bears information distortion compared with the real-valued one. Experiments are performed on the VOC dataset [9].

Despite this attractiveness, DETR usually has a tremendous number of parameters and float-pointing operations (FLOPs). For instance, there are 39.8M parameters taking up 159MB memory usage and 86G FLOPs in the DETR model with ResNet-50 backbone [12] (DETR-R50). This leads to an unacceptable memory and computation consumption during inference, and challenges deployments on devices with limited supplies of resources.

Therefore, substantial efforts on network compression have been made towards efficient online inference [7, 33, 43, 44]. Quantization is particularly popular for deploying on AI chips by representing a network in low-bit formats. Yet prior post-training quantization (PTQ) for DETR [26] derives quantized parameters from pre-trained real-valued models, which often restricts the model performance in a sub-optimized state due to the lack of fine-tuning on the training data. In particular, the performance drastically drops when quantized to ultra-low bits (4-bits or less). Alternatively, quantization-aware training (QAT) [25, 42] performs quantization and fine-tuning on the training dataset
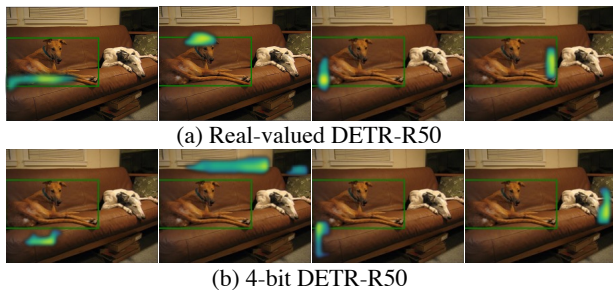
(a) Real-valued DETR-R50

(b) 4-bit DETR-R50

Figure 2. Spatial attention weight maps in the last decoder of (a) real-valued DETR-R50, and (b) 4-bit quantized DETR-R50. The green rectangle denotes the ground-truth bounding box. Following [29], the highlighted area denotes the large attention weights in the selected four heads in compliance with bound prediction. Compared to its real-valued counterpart that focuses on the ground-truth bounds, quantized DETR-R50 deviates significantly.

simultaneously, leading to trivial performance degradation even with significantly lower bits. Though QAT methods have been proven to be very effective in compressing CNNs [8, 27] for computer vision tasks, an exploration of low-bit DETR remains untouched.

In this paper, we first build a low-bit DETR baseline, a straightforward solution based on common QAT techniques [2]. Through an empirical study of this baseline, we observe significant performance drops on the VOC [9] dataset. For example, a 4-bit quantized DETR-R50 using LSQ [8] only achieves 76.9% AP_{50}, leaving a 6.4% performance gaps compared with the real-valued DETR-R50. We find that the incompatibility of existing QAT methods mainly stems from the unique attention mechanism in DETR, where the spatial dependencies are first constructed between the object queries and encoded features. Then the co-attended object queries are fed into box coordinates and class labels by a feed-forward network. A simple application of existing QAT methods on DETR leads to query information distortion, and therefore the performance severely degrades. Fig. 1 exhibits an example of information distortion in query features of 4-bit DETR-R50, where we can see significant distribution variation of the query modules in quantized DETR and real-valued version. The query information distortion causes the inaccurate focus of spatial attention, which can be verified by following [29] to visualize the spatial attention weight maps in 4-bit and real-valued DETR-R50 in Fig. 2. We can see that the quantized DETR-R50 bear's inaccurate object localization. Therefore, a more generic method for DETR quantization is necessary.

To tackle the issue above, we propose an efficient low-bit quantized DETR (Q-DETR) by rectifying the query information of the quantized DETR as that of the real-valued counterpart. Fig. 3 provides an overview of our Q-DETR, which is mainly accomplished by a distribution rectifica-

tion knowledge distillation method (DRD). We find ineffective knowledge transferring from the real-valued teacher to the quantized student primarily because of the information gap and distortion. Therefore, we formulate our DRD as a bi-level optimization framework established on the information bottleneck principle (IB). Generally, it includes an inner-level optimization to maximize the self-information entropy of student queries and an upper-level optimization to minimize the conditional information entropy between student and teacher queries. At the inner level, we conduct a distribution alignment for the query guided by its Gaussian-alike distribution, as shown in Fig. 1, leading to an explicit state in compliance with its maximum information entropy in the forward propagation. At the upper level, we introduce a new foreground-aware query matching that filters out low-qualified student queries for exact one-to-one query matching between student and teacher, providing valuable knowledge gradients to push minimum conditional information entropy in the backward propagation.

This paper attempts to introduce a generic method for DETR quantization. The significant contributions in this paper are outlined as follows: (1) We develop the first QAT quantization framework for DETR, dubbed Q-DETR. (2) We use a bi-level optimization distillation framework, abbreviated as DRD. (3) We observe a significant performance increase compared to existing quantized baselines.

2. Related Work

Quantization. Quantized neural networks often possess low-bit (1∼4-bit) weights and activations to accelerate the model inference and save memory. For example, DoReFa-Net [45] exploits convolution kernels with low bitwidth parameters and gradients to accelerate training and inference. TTQ [46] uses two real-valued scaling coefficients to quantize the weights to ternary values. Zhuang *et al.* [48] present a $2 \sim 4$-bit quantization scheme using a two-stage approach to alternately quantize the weights and activations, providing an optimal tradeoff among memory, efficiency, and performance. In [14], the quantization intervals are parameterized, and optimal values are obtained by directly minimizing the task loss of the network. ZeroQ [3] supports uniform and mixed-precision quantization by optimizing for a distilled dataset which is engineered to match the statistics of the batch normalization across different network layers. Xie *et al.* [41] introduced transfer learning into network quantization to obtain an accurate low-precision model by utilizing Kullback-Leibler (KL) divergence. Fang *et al.* [10] enabled accurate approximation for tensor values that have bell-shaped distributions with long tails and found the entire range by minimizing the quantization error. Li *et al.* [17] proposed an information rectification module and distribution-guided distillation to push the bit-width in a quantized vision transformer. At
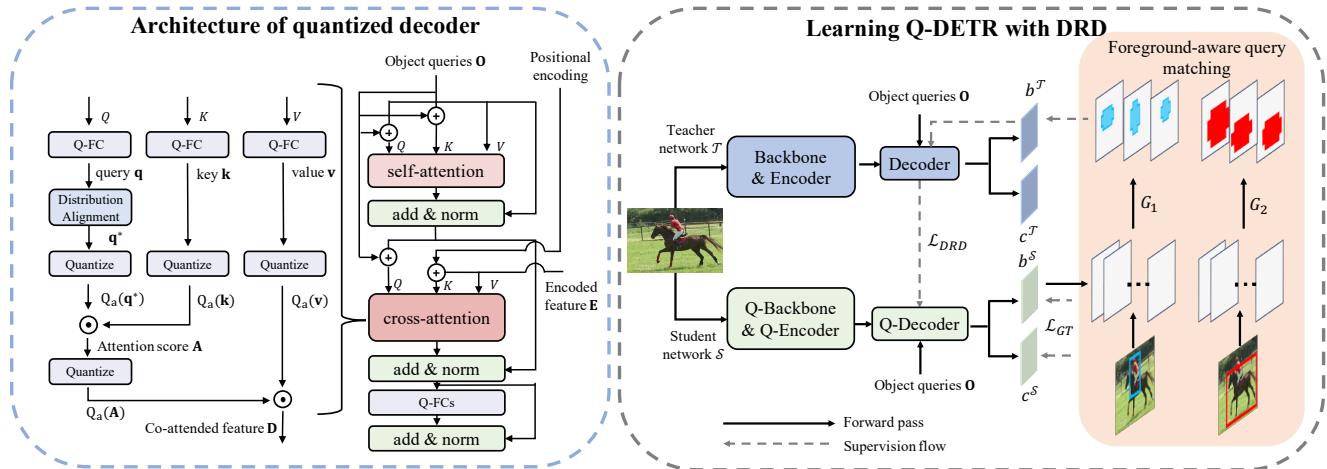
Figure 3. Overview of the proposed Q-DETR framework. We introduce the distribution rectification distillation method (DRD) to refine the performance of Q-DETR. From left to right, we respectively show the detailed decoder architecture of Q-DETR and the learning framework of Q-DETR. The Q-Backbone, Q-Encoder, and Q-Decoder denote quantized architectures, respectively.

the same time, we address the quantization in DETR from the IB principle. The architectural design has also drawn increasing attention using extra shortcut [27], and parallel parameter-free shortcuts [25] for example.

Detection Transformer. Driven by the success of transformers [37], several researchers have also explored transformer frameworks for vision tasks. The first DETR [4] work introduces the Transformer structure based on the attention mechanism for object detection. But the main drawback of DETR lies in the highly inefficient training process. The approachh of another work modifies the multi-head attention mechanism (MHA). Deformable-DETR [47] constructs a sparse and point-to-point MHA mechanism using a static point-wise query sampling method around the reference points. SMCA-DETR [11] introduces a Gaussian-distributed spatial function before formulating a spatially modulated co-attention. DAB-DETR [21] re-defines the query of DETR as dynamic anchor boxes and performs soft ROI pooling layer-by-layer in a cascade manner. DN-DETR [16] introduces query denoising into query generation, reducing the bipartite graph matching difficulty and leading to faster convergence. Another set of arts improves DETR methods using additional learning constraints. For example, UP-DETR [6] proposes a novel self-supervised loss to enhance the convergence speed and the performance of DETR.

However, prior arts mainly focus on the training efficiency of DETR, few of which have discussed the quantization of DETR. To this end, we first build a quantized DETR baseline and then address the query information distortion problem based on the IB principle. Finally, a new KD method based on a foreground-aware query matching scheme is achieved to solve Q-DETR effectively.

3. The Challenge of Quantizing DETR

3.1. Quantized DETR baseline

We first construct a baseline to study the low-bit DETR since no relevant work has been previously proposed. To this end, we follow LSQ+ [2] to introduce a general framework of asymmetric activation quantization and symmetric weight quantization:

$$\boldsymbol{x}_q = \lfloor \text{clip}\{\frac{(\boldsymbol{x}-z)}{\alpha_x}, Q_n^x, Q_p^x\}\rceil, \quad \mathbf{w}_q = \lfloor \text{clip}\{\frac{\mathbf{w}}{\alpha_{\mathbf{w}}}, Q_n^{\mathbf{w}}, Q_p^{\mathbf{w}}\}\rceil,$$
$$Q_a(x) = \alpha_x \circ \boldsymbol{x}_q + z, \qquad Q_w(x) = \alpha_{\mathbf{w}} \circ \mathbf{w}_q, \tag{1}$$

where $\text{clip}\{y, r_1, r_2\}$ clips the input y with value bounds r_1 and r_2; the $\lfloor y \rceil$ rounds y to its nearest integer; the $\circ$ denotes the channel-wise multiplication. And $Q_n^x = -2^{a-1}, Q_p^x = 2^{a-1}-1, Q_n^{\mathbf{w}} = -2^{b-1}, Q_p^{\mathbf{w}} = 2^{b-1}-1$ are the discrete bounds for a-bit activations and b-bit weights. x generally denotes the activation in this paper, including the input feature map of convolution and fully-connected layers and input of multi-head attention modules. Based on this, we first give the quantized fully-connected layer as:

$$\text{Q-FC}(\boldsymbol{x}) = Q_a(\boldsymbol{x}) \cdot Q_w(\mathbf{w}) = \alpha_x \alpha_{\mathbf{w}} \circ (\boldsymbol{x}_q \odot \mathbf{w}_q + z/\alpha_x \circ \mathbf{w}_q), \tag{2}$$

where $\cdot$ denotes the matrix multiplication and $\odot$ denotes the matrix multiplication with efficient bit-wise operations. The straight-through estimator (STE) [1] is used to retain the derivation of the gradient in backward propagation.

In DETR [4], the visual features generated by the backbone are augmented with position embedding and fed into the transformer encoder. Given an encoder output $\mathbf{E}$, DETR performs co-attention between object queries $\mathbf{O}$ and the vi-

(1) Quantizing backbone (2) Quantizing encoder (3) Quantizing MHA of decoder (4) Quantizing MLPs
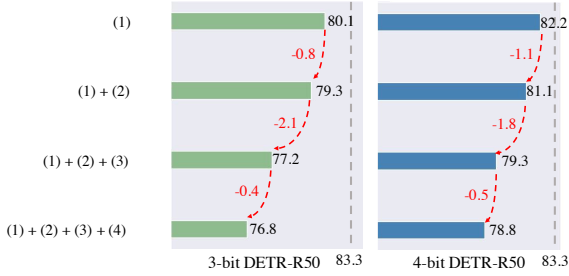

Figure 4. Performance of 3/4-bit quantized DETR-R50 on VOC with different quantized modules.

sual features $\mathbf{E}$, which are formulated as:

$$
\begin{aligned}
\mathbf{q} &= \text{Q-FC}(\mathbf{O}), \quad \mathbf{k}, \mathbf{v} = \text{Q-FC}(\mathbf{E}) \\
\mathbf{A}_i &= \text{softmax}(Q_a(\mathbf{q})_i \cdot Q_a(\mathbf{k})_i^{\top}/\sqrt{d}), \qquad (3) \\
\mathbf{D}_i &= Q_a(\mathbf{A})_i \cdot Q_a(\mathbf{v})_i,
\end{aligned}
$$

where $\mathbf{D}$ is the multi-head co-attention module, *i.e.*, the co-attended feature for the object query. The d denotes the feature dimension in each head. More FC layers transform the decoder's output features of each object query for the final output. Given box and class predictions, the Hungarian algorithm [4] is applied between predictions and ground-truth box annotations to identify the learning targets of each object query.

3.2. Challenge Analysis

Intuitively, the performance of the quantized DETR baseline largely depends on the information representation capability mainly reflected by the information in the multi-head attention module. Unfortunately, such information is severely degraded by the quantized weights and inputs in the forward pass. Also, the rounded and discrete quantization significantly affect the optimization during backpropagation.

We conduct the quantitively ablative experiments by progressively replacing each module of the real-valued DETR baseline with a quantized one and compare the average precision (AP) drop on the VOC dataset [9] as shown in Fig. 4. We find that quantizing the MHA decoder module to low bits, *i.e.*, (1)+(2)+(3), brings the most significant accuracy drops of accuracy among all parts of the DETR methods, up to 2.1% in the 3-bit DETR-R50. At the same time, other parts of DETR show comparative robustness to the quantization function. Consequently, the critical problem of improving the quantized DETR methods is restoring the information in MHA modules after quantization. Other qualitative results in Fig. 1 and Fig. 2 also indicate that the de-

graded information representation is the main obstacle to a better quantized DETR.

4. The Proposed Q-DETR

4.1. Information Bottleneck of Q-DETR

To address the information distortion of the quantized DETR, we aim to improve the representation capacity of the quantized networks in a knowledge distillation framework. Generally, we utilize a real-valued DETR as a teacher and a quantized DETR as a student, which are distinguished with superscripts $\mathcal{T}$ and $\mathcal{S}$, respectively.

Our Q-DETR pursues the best tradeoff between performance and compression, which is precisely the goal of the information bottleneck (IB) method through quantifying the mutual information that the intermediate layer contains about the input (less is better) and the desired output (more is better) [35, 36]. In our case, the intermediate layer comes from the student, while the desired output includes the ground-truth labels as well as the queries of the teacher for distillation. Thus, the objective target of our Q-DETR is:

$$
\min_{\theta^{\mathcal{S}}} I(X; \mathbf{E}^{\mathcal{S}}) - \beta I(\mathbf{E}^{\mathcal{S}}, \mathbf{q}^{\mathcal{S}}; \boldsymbol{y}^{GT}) - \gamma I(\mathbf{q}^{\mathcal{S}}; \mathbf{q}^{\mathcal{T}}), \quad (4)
$$

where $\mathbf{q}^{\mathcal{T}}$ and $\mathbf{q}^{\mathcal{S}}$ represent the queries in the teacher and student DETR methods as predefined in Eq. (3); β and γ are the Lagrange multipliers [35]; $\theta^{\mathcal{S}}$ is the parameters of the student; and $I(\cdot)$ returns the mutual information of two input variables. The first item $I(X; \mathbf{E}^{\mathcal{S}})$ minimizes information between input and visual features $\mathbf{E}^{\mathcal{S}}$ to extract task-oriented hints [40]. The second item $I(\mathbf{E}^{\mathcal{S}}, \mathbf{q}^{\mathcal{S}}; \boldsymbol{y}^{GT})$ maximizes information between extracted visual features and ground-truth labels for better object detection. These two items can be easily accomplished by common network training and detection loss constraints, such as proposal classification and coordinate regression.

The core issue of this paper is to solve the third item $I(\mathbf{q}^{\mathcal{S}}; \mathbf{q}^{\mathcal{T}})$, which attempts to address the information distortion in student query via introducing teacher query as a priori knowledge. To accomplish our goal, we first expand the third item and reformulate it as:

$$
I(\mathbf{q}^{\mathcal{S}}; \mathbf{q}^{\mathcal{T}}) = H(\mathbf{q}^{\mathcal{S}}) - H(\mathbf{q}^{\mathcal{S}}|\mathbf{q}^{\mathcal{T}}), \qquad (5)
$$

where $H(\mathbf{q}^{\mathcal{S}})$ returns the self information entropy expected to be maximized while $H(\mathbf{q}^{\mathcal{S}}|\mathbf{q}^{\mathcal{T}})$ is the conditional entropy expected to be minimized. It is challenging to optimize the above maximum & minimum items simultaneously. Instead, we make a compromise to reformulate Eq. (5) as a bi-level issue [5, 20] that alternately optimizes

the two items, which is explicitly defined as:

$$\min_{\theta} H(\mathbf{q}^{\mathcal{S}^*} | \mathbf{q}^{\mathcal{T}}),$$
$$\text{s.t.} \quad \mathbf{q}^{\mathcal{S}^*} = \arg\max_{\mathbf{q}^{\mathcal{S}}} H(\mathbf{q}^{\mathcal{S}}). \tag{6}$$

Such an objective involves two sub-problems, including an inner-level optimization to derive the current optimal query $\mathbf{q}^{\mathcal{S}^*}$ and an upper-level optimization to conduct knowledge transfer from the teacher to the student. Below, we show that the two sub-problems can be solved in the forward & backward network propagation's.

4.2. Distribution Rectification Distillation

Inner-level optimization. We first detail the maximization of self-information entropy. According to the definition of self information entropy, $H(\mathbf{q}^{\mathcal{S}})$ can be implicitly expanded as:

$$H(\mathbf{q}^{\mathcal{S}}) = -\int_{\mathbf{q}_i^{\mathcal{S}} \in \mathbf{q}^{\mathcal{S}}} p(\mathbf{q}_i^{\mathcal{S}}) \log p(\mathbf{q}_i^{\mathcal{S}}). \tag{7}$$

However, an explicit form of $H(\mathbf{q}^{\mathcal{S}})$ can only be parameterized with a regular distribution $p(\mathbf{q}_i^{\mathcal{S}})$. Luckily, the statistical results in Fig. 1 shows that the query distribution tends to follow a Gaussian distribution, which is also observed in [17]. This enables us to solve the inner-level optimization in a distribution alignment fashion. To this end, we first calculate the mean $\mu(\mathbf{q}^{\mathcal{S}})$ and variance $\sigma(\mathbf{q}^{\mathcal{S}})$ of query $\mathbf{q}^{\mathcal{S}}$ whose distribution is then modeled as $\mathbf{q}^{\mathcal{S}} \sim \mathcal{N}(\mu(\mathbf{q}^{\mathcal{S}}), \sigma(\mathbf{q}^{\mathcal{S}}))$. Then, the self-information entropy of the student query can be proceeded as:

$$H(\mathbf{q}^{\mathcal{S}}) = -\mathbb{E}[\log \mathcal{N}(\mu(\mathbf{q}^{\mathcal{S}}), \sigma(\mathbf{q}^{\mathcal{S}}))]$$
$$= -\mathbb{E}[\log[(2\pi\sigma(\mathbf{q}^{\mathcal{S}})^2)^{\frac{1}{2}} \exp(-\frac{(\mathbf{q}_i^{\mathcal{S}} - \mu(\mathbf{q}^{\mathcal{S}}))^2}{2\sigma(\mathbf{q}^{\mathcal{S}})^2})]]$$
$$= \frac{1}{2}\log 2\pi\sigma(\mathbf{q}^{\mathcal{S}})^2. \tag{8}$$

The above objective reaches its maximum of $H(\mathbf{q}^{\mathcal{S}^*}) = (1/2)\log 2\pi e[\sigma(\mathbf{q}^{\mathcal{S}})^2 + \epsilon_{\mathbf{q}^{\mathcal{S}}}]$ when $\mathbf{q}^{\mathcal{S}^*} = [\mathbf{q}^{\mathcal{S}} - \mu(\mathbf{q}^{\mathcal{S}})]/[\sqrt{\sigma(\mathbf{q}^{\mathcal{S}})^2 + \epsilon_{\mathbf{q}^{\mathcal{S}}}}]$ where $\epsilon_{\mathbf{q}^{\mathcal{S}}} = 1e^{-5}$ is a small constant added to prevent a zero denominator. In practice, the mean and variance might be inaccurate due to query data bias. To solve this we use the concepts in batch normalization (BN) [13,34] where a learnable shifting parameter $\beta_{\mathbf{q}^{\mathcal{S}}}$ is added to move the mean value. A learnable scaling parameter $\gamma_{\mathbf{q}^{\mathcal{S}}}$ is multiplied to move the query to the adaptive position. In this situation, we rectify the information entropy of the query in the student as follows:

$$\mathbf{q}^{\mathcal{S}^*} = \frac{\mathbf{q}^{\mathcal{S}} - \mu(\mathbf{q}^{\mathcal{S}})}{\sqrt{\sigma(\mathbf{q}^{\mathcal{S}})^2 + \epsilon_{\mathbf{q}^{\mathcal{S}}}}} \gamma_{\mathbf{q}^{\mathcal{S}}} + \beta_{\mathbf{q}^{\mathcal{S}}}, \tag{9}$$

in which case the maximum self-information entropy of student query becomes $H(\mathbf{q}^{\mathcal{S}^*}) = (1/2)\log 2\pi e[(\sigma_{\mathbf{q}^{\mathcal{S}}}^2 + \epsilon_{\mathbf{q}^{\mathcal{S}}})/\gamma_{\mathbf{q}^{\mathcal{S}}}^2]$. Therefore, in the forward propagation, we can obtain the current optimal query $\mathbf{q}^{\mathcal{S}^*}$ via Eq. (9), after which, the upper-level optimization is further executed as detailed in the following contents.

Upper-level optimization. We continue minimizing the conditional information entropy between the student and the teacher. Following DETR [4], we denote the ground-truth labels by $\boldsymbol{y}^{GT} = \{c_i^{GT}, b_i^{GT}\}_{i=1}^{N_{gt}}$ as a set of ground-truth objects where N_{gt} is the number of foregrounds, c_i^{GT} and b_i^{GT} respectively represent the class and coordinate (bounding box) for the i-th object. In DETR, each query is associated with an object. Therefore, we can obtain N objects for teacher and student as well, denoted as $\boldsymbol{y}^{\mathcal{S}} = \{c_j^{\mathcal{S}}, b_j^{\mathcal{S}}\}_{j=1}^{N}$ and $\boldsymbol{y}^{\mathcal{T}} = \{c_j^{\mathcal{T}}, b_j^{\mathcal{T}}\}_{j=1}^{N}$.

The minimization of the conditional information entropy requires the student and teacher objects to be in a one-to-one matching. However, it is problematic for DETR due primarily to the sparsity of prediction results and the instability of the query's predictions [16]. We propose a foreground-aware query matching to rectify "well-matched" queries to solve this. Concretely, we match the ground-truth bounding boxes with this student to find the maximum coincidence as:

$$G_i = \max_{1 \le j \le N} \text{GIoU}(b_i^{GT}, b_j^{\mathcal{S}}), \tag{10}$$

where $\text{GIoU}(\cdot)$ is the generalized intersection over union function [32]. Each G_i reflects the "closeness" of student proposals to the i-th ground-truth object. Then, we retain highly qualified student proposals around at least one ground truth to benefit object recognition [38] as:

$$b_j^{\mathcal{S}} = \begin{cases} b_j^{\mathcal{S}}, & \text{GIoU}(b_i^{GT}, b_j^{\mathcal{S}}) > \tau G_i, \ \forall \, i \\ \varnothing, & \text{otherwise}, \end{cases} \tag{11}$$

where τ is a threshold controlling the proportion of distilled queries. After removing object-empty ($\varnothing$) queries in $\tilde{\boldsymbol{q}}^{\mathcal{S}}$, we form a distillation-desired query set of students denoted as $\tilde{\boldsymbol{q}}^{\mathcal{S}}$ associated with its object set $\tilde{\boldsymbol{y}}^{\mathcal{S}} = \{\tilde{c}_j^{\mathcal{S}}, \tilde{b}_j^{\mathcal{S}}\}_{j=1}^{\tilde{N}}$. Correspondingly, we can obtain a teacher query set $\tilde{\boldsymbol{y}}^{\mathcal{T}} = \{\tilde{c}_j^{\mathcal{T}}, \tilde{b}_j^{\mathcal{T}}\}_{j=1}^{\tilde{N}}$. For the j-th student query, its corresponding teacher query is matched as:

$$\tilde{c}_j^{\mathcal{T}}, \tilde{b}_j^{\mathcal{T}} = \arg\max_{\tilde{c}_k^{\mathcal{T}}, \tilde{b}_k^{\mathcal{T}}} \sum_{k=1}^{N} \mu_1 \text{GIoU}(\tilde{b}_j^{\mathcal{S}}, b_k^{\mathcal{T}}) - \mu_2 \|\tilde{b}_j^{\mathcal{S}} - b_k^{\mathcal{T}}\|_1, \tag{12}$$

where $\mu_1 = 2$ and $\mu_2 = 5$ control the matching function, values of which is to follow [4].

Finally, the upper-level optimization after rectification in Eq. (6) becomes:

$$\min_{\theta} H(\tilde{\mathbf{q}}^{\mathcal{S}^*} | \tilde{\mathbf{q}}^{\mathcal{T}}). \tag{13}$$

Optimizing Eq. (13) is challenging. Alternatively, we minimize the norm distance between $\tilde{\mathbf{q}}^{\mathcal{S}^*}$ and $\tilde{\mathbf{q}}^{\mathcal{T}}$, optima of which, *i.e.*, $\tilde{\mathbf{q}}^{\mathcal{S}^*} = \tilde{\mathbf{q}}^{\mathcal{T}}$, is exactly the same with that in Eq. (13). Thus, the final loss for our distribution rectification distillation loss becomes:

$$\mathcal{L}_{DRD}(\tilde{\mathbf{q}}^{\mathcal{S}^*}, \tilde{\mathbf{q}}^{\mathcal{T}}) = \mathbb{E}[\|\tilde{\mathbf{D}}^{\mathcal{S}^*} - \tilde{\mathbf{D}}^{\mathcal{T}}\|_2], \quad (14)$$

where we use the Euclidean distance of co-attented feature $\tilde{\mathbf{D}}$ (see Eq. 3) containing the information query $\tilde{\mathbf{q}}$ for optimization.

In backward propagation, the gradient updating drives the student queries toward their teacher hints. Therefore we accomplish our distillation. The overall training losses for our Q-DETR model are:

$$\mathcal{L} = \mathcal{L}_{GT}(\boldsymbol{y}^{GT}, \boldsymbol{y}^{\mathcal{S}}) + \lambda\mathcal{L}_{DRD}(\tilde{\mathbf{q}}^{\mathcal{S}^*}, \tilde{\mathbf{q}}^{\mathcal{T}}), \quad (15)$$

where L_{GT} is the common detection loss for missions such as proposal classification and coordinate regression [4], and λ is a tradeoff hyper-parameter.

5. Experiments

In this section, we evaluate the performance of the proposed Q-DETR mode using popular DETR [4] and SMCA-DETR [11] models. To the best of our knowledge, there is no publicly available source code on quantization-aware training of DETR methods at this point, so we implement the baseline and LSQ [8] methods ourselves.

5.1. Datasets and Implementation Details

Datasets. We first conduct the ablative study and hyper-parameter selection on the PASCAL VOC dataset [9], which contains natural images from 20 different classes. We use the VOC `trainval2012`, and VOC `trainval2007` sets to train our model, which contains approximately 16k images, and the VOC `test2007` set to evaluate our Q-DETR, which contains 4952 images. We report COCO-style metrics for the VOC dataset: AP, AP_{50} (default VOC metric), and AP_{75}. We further conduct the experiments on the COCO `2017` [18] object detection tracking. Specifically, we train the models on COCO `train2017` and evaluate the models on COCO `val2017`. We list the average precision (AP) for IoUs$\in$ $[0.5 : 0.05 : 0.95]$, designated as AP, using COCO's standard evaluation metric. For further analyzing our method, we also list AP_{50}, AP_{75}, AP_s, AP_m, and AP_l.

Implementation Details. Our Q-DETR is trained with the DETR [4] and SMCA-DETR [11] framework. We select the ResNet-50 [12] and modify it with Pre-Activation structures and RPReLU [25] function following [23]. PyTorch [30] is used for implementing Q-DETR. We run the experiments on 8 NVIDIA Tesla A100 GPUs with 80 GB memory. We use ImageNet ILSVRC12 [15] to pre-train
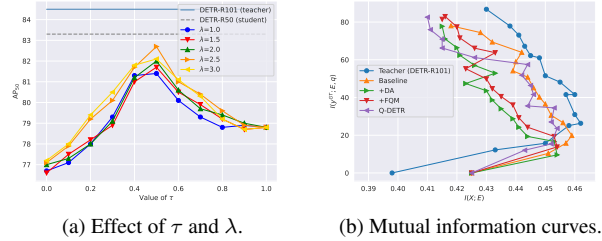

(a) Effect of τ and λ. (b) Mutual information curves.

Figure 5. (a) We select τ and λ using 4-bit Q-DETR-R50 on VOC. (b) The mutual information curves of $I(X; \mathbf{E})$ and $I(\boldsymbol{y}^{GT}; \mathbf{E}, \mathbf{q})$ (Eq. 4) on the information plane. The red curves represent the teacher model (DETR-R101). The orange, green, red, and purple lines represent the 4-bit baseline, 4-bit baseline + DA, 4-bit baseline + FQM, and 4-bit baseline + DA + FQM (4-bit Q-DETR).

the backbone of a quantized student. The training protocol is the same as the employed frameworks [4, 11]. Specifically, we use a batch size of 16. AdamW [28] is used to optimize the Q-DETR, with the initial learning rate of $1e^{-4}$. We train for 300/500 epochs for the Q-DETR on VOC/COCO dataset, and the learning rate is multiplied by 0.1 at the 200/400-th epoch, respectively. Following the SMCA-DETR, we train the Q-SMCA-DETR for 50 epochs, and the learning rate is multiplied by 0.1 at the 40-th epoch on both the VOC and COCO datasets. We utilize a multi-distillation strategy, where we save the encoder and decoder network as real-valued at the first stage. Then we train the fully quantized DETR at the second stage, where we load the weight from the checkpoint of first stage. We select real-valued DETR-R101 (84.5% AP_{50} on VOC and 43.5% AP on COCO) and SMCA-DETR-R101 (85.3% AP_{50} on VOC and 44.4% AP on COCO) as teacher network.

5.2. Ablation Study

Hyper-parameter selection. As mentioned above, we select hyper-parameters τ and λ in this part using the 4-bit Q-DETR model. We show the model performance (AP_{50}) with different setups of hyper-parameters $\{\tau, \lambda\}$ in Fig. 5 (a), where we conduct ablative experiments on the baseline + DA (AP_{50}=78.8%). As can be seen, the performances increase first and then decrease with the increase of τ from left to right. Since τ controls the proportion of selected distillation-desired queries, we show that the full-imitation ($\tau = 0$) performs worse than the vanilla baseline with no distillation ($\tau = 1$), showing query selection is necessary. The figure also shows that the performances increase first and then decrease with the increase of τ from left to right. The Q-DETR obtains better performances with τ set as 0.5 and 0.6. With the varying value of λ, we find $\{\lambda, \tau\} = \{2.5, 0.6\}$ boost the performance of Q-DETR most, achieving 82.7% AP on VOC `test2007`. Based on the ablative study above, we set hyper-parameters τ and λ as 0.6 and 2.5

Table 1. Evaluating the components of Q-DETR-R50 on the VOC dataset. #Bits (W-A-Attention) denotes the bit-width of weights, activations, and attention activations. DA denotes the distribution alignment module. FQM denotes foreground-aware query matching.

Method	#Bits	AP_{50}	#Bits	AP_{50}	#Bits	AP_{50}
Real-valued	32-32-32	83.3	-	-	-	-
Baseline	4-4-8	78.0	3-3-8	76.8	2-2-8	69.7
+DA	4-4-8	78.8	3-3-8	78.0	2-2-8	71.6
+FQM	4-4-8	81.5	3-3-8	80.9	2-2-8	74.9
+DA+FQM (Q-DETR)	4-4-8	**82.7**	3-3-8	**82.1**	2-2-8	**76.4**

for the experiments in this paper.

Effectiveness of components. We show quantitative improvements of components in Q-DETR in Tab. 1. As shown in Tab. 1, the quantized DETR baseline suffers a severe performance drop on AP_{50} (13.6%, 6.5%, and 5.3% with 2/3/4-bit, respectively). DA and FQM improve the performance when used alone, and the two techniques further boost the performance considerably when combined. For example, the DA improves the 2-bit baseline by 1.9%, and the FQM achieves a 5.2% performance improvement. While combining the DA and FQM, the performance improvement achieves 6.7%.

Information analysis. We further show the information plane following [39] in Fig. 5(b). We adopt the test AP_{50} to quantify $I(\boldsymbol{y}^{GT}; \mathbf{E}, \mathbf{q})$. We employ a reconstruction decoder to decode the encoded feature $\mathbf{E}$ to reconstruct the input and quantify $I(X; \mathbf{E})$ using the ℓ_1 loss. As shown in Fig. 5(b), the curve of the larger teacher DETR-R101 is usually on the right of the curve of small student models, which indicates a greater ability of information representation. Likewise, the purple line (Q-DETR-R50) is usually on the right of the three left curves, showing the information representation improvements with the proposed methods.

5.3. Results on PASCAL VOC

We first compare our method with the 2/3/4-bit baseline and LSQ [8] based on the same frameworks for object detection task with the VOC dataset. We also report the detection performance of the 8-bit post-training quantization networks, such as percentile [19], VT-PTQ [26]. We use the input resolution following [4], *i.e.* 1333×800. We mainly discuss the AP_{50} (default VOC metric) in the following.

We evaluate the proposed Q-DETR on DETR-R50 models in Tab. 2. For the DETR-R50 model, compared with the 8-bit PTQ method, our 4-bit Q-DETR achieves a much larger compression ratio than 8-bit VT-PTQ, but with a bit of performance improvement (82.7% *vs.* 82.3%). Also, the proposed method boosts the performance of 2/3/4-bit baseline by 6.7%, 5.3%, and 4.7% with the same architecture

Table 2. We report AP, AP_{50}, and AP_{75} (%) with state-of-the-art quantization methods on DETR and SMCA-DETR using VOC test2007. #Bits (W-A-Attention) denotes the bit-width of weights, activations, and attention activations.

Model	Method	#Bits	AP	AP_{50}	AP_{75}
DETR-R50	Real-valued	32-32-32	59.5	83.3	64.7
	Percentile	8-8-8	54.7	79.2	60.1
	VT-PTQ		57.6	82.3	63.1
	LSQ	4-4-8	49.7	76.9	53.0
	Baseline		51.3	78.0	54.1
	Q-DETR		**57.1**	**82.7**	**61.5**
	LSQ	3-3-8	47.0	75.3	49.1
	Baseline		49.2	76.8	51.8
	Q-DETR		**56.8**	**82.1**	**61.2**
	LSQ	2-2-8	42.6	68.2	44.8
	Baseline		44.0	69.7	45.8
	Q-DETR		**50.7**	**76.4**	**54.1**
SMCA-DETR -R50	Real-valued	32-32-32	56.7	83.7	62.0
	Percentile	8-8-8	54.7	79.2	60.1
	VT-PTQ		55.9	83.0	61.3
	LSQ	4-4-8	49.6	78.6	53.4
	Baseline		50.7	79.5	55.4
	Q-DETR		**56.2**	**83.3**	**61.6**
	LSQ	3-3-8	47.7	76.5	51.7
	Baseline		49.9	77.5	53.6
	Q-DETR		**54.3**	**82.6**	**59.5**
	LSQ	2-2-8	42.3	69.7	44.8
	Baseline		43.9	70.4	46.1
	Q-DETR		**50.2**	**76.7**	**52.6**

and bit-width, which significantly validates the effectiveness of our method.

Besides, our method generates convincing results on SMCA-DETR. As shown in Tab. 2, the performance of the proposed Q-DETR with SMCA-DETR-R50 outperforms the 2/3/4-bit Baseline method by 6.3% , 5.1% and 3.8% on AP_{50}, a large margin. Compared with 8-bit post-training quantization methods, our method achieves a significantly higher compression rate and comparable performance.

5.4. Results on COCO

We further show comparison on the large-scale COCO [18] dataset. We compare our method with the 2/3/4-bit baseline and LSQ [8] based on the same frameworks. We also report the detection performance of the 8-bit post-training quantization networks, such as percentile [19] , VT-PTQ [26]. The AP with different IoU thresholds, and AP of objects with varying scales are all reported in Tab. 3.

Tab. 3 lists the comparison of several quantization approaches and detection frameworks in computing complexity, storage cost. Our Q-DETR significantly accelerates computation and reduces storage requirements for various

Table 3. Comparison with state-of-the-art quantization methods using DETR and SMCA-DETR on COCO `val2017`. #Bits (W-A-Attention) denotes bit-width of weights, activations, and attention activations.

Model	Method	#Bits	Size$_{(MB)}$	OPs$_{(G)}$	AP	AP$_{50}$	AP$_{75}$	AP$_s$	AP$_m$	AP$_l$
	Real-valued	32-32-32	159.32	85.51	42.0	62.4	44.2	20.5	45.8	61.1
	Percentile	8-8-8	39.83	23.01	38.6	-	-	-	-	-
	VT-PTQ				41.2	-	-	-	-	-
	LSQ	4-4-8	19.92	13.02	33.3	53.7	33.9	12.8	37.0	51.6
	Baseline				34.1	55.3	35.4	14.3	38.0	53.8
	Q-DETR				**39.4**	**60.2**	**41.4**	**17.7**	**43.4**	**59.9**
DETR-R50	LSQ	3-3-8	15.03	7.61	31.0	52.3	32.1	11.3	33.9	48.5
	Baseline				32.3	52.2	32.9	12.3	35.4	50.3
	Q-DETR				**36.1**	**55.9**	**37.5**	**14.6**	**39.4**	**55.2**
	LSQ	2-2-8	10.03	5.32	24.7	44.6	26.5	6.3	25.3	42.7
	Baseline				26.6	46.6	26.5	8.4	28.2	44.4
	Q-DETR				**31.4**	**51.3**	**31.6**	**11.6**	**34.3**	**49.6**
	Real-valued	32-32-32	164.75	86.65	41.0	62.2	43.6	21.9	44.3	59.1
	Percentile	8-8-8	41.19	23.66	37.5	58.5	40.1	17.6	39.1	55.9
	VT-PTQ				40.2	61.0	42.6	20.3	42.9	57.7
	LSQ	4-4-8	20.59	13.48	33.9	55.0	35.0	13.2	37.2	51.4
	Baseline				35.0	56.4	36.4	15.6	38.3	52.5
	Q-DETR				**38.3**	**59.7**	**39.8**	**17.7**	**41.7**	**56.8**
SMCA-DETR-R50	LSQ	3-3-8	15.68	8.05	30.1	52.6	31.4	11.9	33.4	46.6
	Baseline				31.8	53.7	32.6	12.6	35.2	49.8
	Q-DETR				**35.0**	**56.3**	**36.9**	**15.0**	**39.0**	**53.1**
	LSQ	2-2-8	10.84	4.54	23.9	42.2	24.2	9.4	26.2	37.5
	Baseline				25.4	44.3	25.2	8.4	27.2	40.3
	Q-DETR				**30.5**	**51.8**	**31.8**	**12.0**	**33.2**	**48.0**

detectors. We follow [40] to calculate memory usage, by adding $32\times$ the number of real-valued weights and $a\times$ the number of quantized weights in the a-bit networks. The number of operations (OPs) is calculated in the same way as [40]. Current CPUs can handle both bit-wise XNOR and bit-count operations in parallel. The respective number of FLOPs adds $\{\frac{1}{32}, \frac{1}{16}, \frac{1}{8}\}$ of the number of $\{2,3,4\}$-bit multiplications equals the OPs following [24].

We summarize the experimental results on COCO `val2017` of Q-DETR-R50 from lines 2 to 17 in Tab. 3. For the DETR-R50 model, compared with the 8-bit PTQ method, our 4-bit Q-DETR achieves a much larger acceleration than the 8-bit VT-PTQ but with an acceptable performance gap. Also, the proposed method boosts the performance of 2/3/4-bit baseline by 4.8%, 3.8% and 5.1% AP with the same architecture and bit-width, which is significant on the large-scale COCO dataset. Compared with the real-valued counterparts, the proposed 2/3/4-bit Q-DETR achieves computation acceleration and storage savings by $16.07\times/11.23\times/6.57\times$ and $15.88\times/10.60\times/7.99\times$. The above results are of great significance in the real-time inference of object detection. All of the improvements have impacts on object detection.

For the SMCA-DETR-R50 model, we observe similar performance improvements and compression ratios. For example, the 4-bit Q-SMCA-DETR-R50 theoretically accelerates $6.42\times$ with only a 2.7% performance gap compared with the real-valued counterpart, which is significant for real-time DETR methods.

6. Conclusion

This paper introduces a novel method for training quantized DETR (Q-DETR) with knowledge distillation to rectify the query distribution. Q-DETR generalizes the information bottleneck (IB) principle and leads a bi-level distribution rectification distillation. We effectively employ a distribution alignment module to solve inner-level and a foreground-aware query matching scheme to solve upper level. As a result, Q-DETR significantly boosts performance of low-bit DETR. Extensive experiments show that Q-DETR surpasses state-of-the-arts in DETR quantization.

7. Acknowledgements

References

[1] Yoshua Bengio, Nicholas Léonard, and Aaron Courville. Estimating or propagating gradients through stochastic neurons for conditional computation. *arXiv preprint arXiv:1308.3432*, 2013. 3

[2] Yash Bhalgat, Jinwon Lee, Markus Nagel, Tijmen Blankevoort, and Nojun Kwak. Lsq+: Improving low-bit quantization through learnable offsets and better initialization. In *Proc. of CVPR Workshops*, pages 696–697, 2020. 2, 3

[3] Yaohui Cai, Zhewei Yao, Zhen Dong, Amir Gholami, Michael W Mahoney, and Kurt Keutzer. Zeroq: A novel zero shot quantization framework. In *Proc. of CVPR*, pages 13169–13178, 2020. 2

[4] Nicolas Carion, Francisco Massa, Gabriel Synnaeve, Nicolas Usunier, Alexander Kirillov, and Sergey Zagoruyko. End-to-end object detection with transformers. In *Proc. of ECCV*, pages 213–229, 2020. 1, 3, 4, 5, 6, 7

[5] Benoît Colson, Patrice Marcotte, and Gilles Savard. An overview of bilevel optimization. *Annals of operations research*, 153(1):235–256, 2007. 4

[6] Zhigang Dai, Bolun Cai, Yugeng Lin, and Junying Chen. Up-detr: Unsupervised pre-training for object detection with transformers. In *Proc. of CVPR*, pages 1601–1610, 2021. 3

[7] Misha Denil, Babak Shakibi, Laurent Dinh, Marc'Aurelio Ranzato, and Nando De Freitas. Predicting parameters in deep learning. In *Proc. of NeurIPS*, pages 2148–2156, 2013. 1

[8] Steven K Esser, Jeffrey L McKinstry, Deepika Bablani, Rathinakumar Appuswamy, and Dharmendra S Modha. Learned step size quantization. *arXiv preprint arXiv:1902.08153*, 2019. 2, 6, 7

[9] Mark Everingham, Luc Van Gool, Christopher KI Williams, John Winn, and Andrew Zisserman. The pascal visual object classes (voc) challenge. *International Journal of Computer Vision*, 88(2):303–338, 2010. 1, 2, 4, 6

[10] Jun Fang, Ali Shafiee, Hamzah Abdel-Aziz, David Thorsley, Georgios Georgiadis, and Joseph H Hassoun. Post-training piecewise linear quantization for deep neural networks. In *Proc. of ECCV*, pages 69–86, 2020. 2

[11] Peng Gao, Minghang Zheng, Xiaogang Wang, Jifeng Dai, and Hongsheng Li. Fast convergence of detr with spatially modulated co-attention. In *Proc. of ICCV*, pages 3621–3630, 2021. 3, 6

[12] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proc. of CVPR*, pages 770–778, 2016. 1, 6

[13] Sergey Ioffe and Christian Szegedy. Batch normalization: Accelerating deep network training by reducing internal covariate shift. In *Proc. of ICML*, pages 448–456, 2015. 5

[14] Sangil Jung, Changyong Son, Seohyung Lee, Jinwoo Son, Jae-Joon Han, Youngjun Kwak, Sung Ju Hwang, and Changkyu Choi. Learning to quantize deep networks by optimizing quantization intervals with task loss. In *Proc. of CVPR*, pages 4350–4359, 2019. 2

[15] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. Imagenet classification with deep convolutional neural networks. In *Proc. of NeurIPS*, pages 1097–1105, 2012. 6

[16] Feng Li, Hao Zhang, Shilong Liu, Jian Guo, Lionel M Ni, and Lei Zhang. Dn-detr: Accelerate detr training by introducing query denoising. In *Proc. of CVPR*, pages 13619–13627, 2022. 3, 5

[17] Yanjing Li, Sheng Xu, Baochang Zhang, Xianbin Cao, Peng Gao, and Guodong Guo. Q-vit: Accurate and fully quantized low-bit vision transformer. In *Proc. of NeurIPS*, pages 1–12, 2022. 1, 2, 5

[18] 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 *Proc. of ECCV*, pages 740–755, 2014. 6, 7

[19] Yang Lin, Tianyu Zhang, Peiqin Sun, Zheng Li, and Shuchang Zhou. Fq-vit: Post-training quantization for fully quantized vision transformer. In *Proc. of IJCAI*, pages 1173–1179, 2021. 7

[20] Risheng Liu, Jiaxin Gao, Jin Zhang, Deyu Meng, and Zhouchen Lin. Investigating bi-level optimization for learning and vision from a unified perspective: A survey and beyond. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2021. 4

[21] Shilong Liu, Feng Li, Hao Zhang, Xiao Yang, Xianbiao Qi, Hang Su, Jun Zhu, and Lei Zhang. Dab-detr: Dynamic anchor boxes are better queries for detr. pages 1–19, 2022. 3

[22] Wei Liu, Dragomir Anguelov, Dumitru Erhan, Christian Szegedy, Scott Reed, Cheng-Yang Fu, and Alexander C Berg. Ssd: Single shot multibox detector. In *Proc. of ECCV*, pages 21–37, 2016. 1

[23] Zechun Liu, Kwang-Ting Cheng, Dong Huang, Eric P Xing, and Zhiqiang Shen. Nonuniform-to-uniform quantization: Towards accurate quantization via generalized straight-through estimation. In *Proc. of CVPR*, pages 4942–4952, 2022. 6

[24] Zechun Liu, Wenhan Luo, Baoyuan Wu, Xin Yang, Wei Liu, and Kwang-Ting Cheng. Bi-real net: Binarizing deep network towards real-network performance. *International Journal of Computer Vision*, 128(1):202–219, 2020. 8

[25] Zechun Liu, Zhiqiang Shen, Marios Savvides, and Kwang-Ting Cheng. Reactnet: Towards precise binary neural network with generalized activation functions. In *Proc. of ECCV*, pages 143–159, 2020. 1, 3, 6

[26] Zhenhua Liu, Yunhe Wang, Kai Han, Wei Zhang, Siwei Ma, and Wen Gao. Post-training quantization for vision transformer. *Proc. of NeurIPS*, pages 1–12, 2021. 1, 7

[27] Zechun Liu, Baoyuan Wu, Wenhan Luo, Xin Yang, Wei Liu, and Kwang-Ting Cheng. Bi-real net: Enhancing the performance of 1-bit cnns with improved representational capability and advanced training algorithm. In *Proc. of ECCV*, pages 722–737, 2018. 2, 3

[28] Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. In *Proc. of ICLR*, pages 1–18, 2017. 6

[29] Depu Meng, Xiaokang Chen, Zejia Fan, Gang Zeng, Houqiang Li, Yuhui Yuan, Lei Sun, and Jingdong Wang. Conditional detr for fast training convergence. In *Proc. of ICCV*, pages 3651–3660, 2021. 2

[30] Adam Paszke, Sam Gross, Soumith Chintala, Gregory Chanan, Edward Yang, Zachary DeVito, Zeming Lin, Alban Desmaison, Luca Antiga, and Adam Lerer. Automatic differentiation in pytorch. In *Proc. of NeurIPS Workshops*, pages 1–4, 2017. 6

[31] Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. Faster r-cnn: Towards real-time object detection with region proposal networks. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 39(6):1137–1149, 2016. 1

[32] Hamid Rezatofighi, Nathan Tsoi, JunYoung Gwak, Amir Sadeghian, Ian Reid, and Silvio Savarese. Generalized intersection over union: A metric and a loss for bounding box regression. In *Proc. of CVPR*, pages 658–666, 2019. 5

[33] Adriana Romero, Nicolas Ballas, Samira Ebrahimi Kahou, Antoine Chassang, Carlo Gatta, and Yoshua Bengio. Fitnets: Hints for thin deep nets. In *Proc. of ICLR*, pages 1–13, 2015. 1

[34] Shibani Santurkar, Dimitris Tsipras, Andrew Ilyas, and Aleksander Madry. How does batch normalization help optimization? In *Proc. of NeurIPS*, pages 1–11, 2018. 5

[35] Ravid Shwartz-Ziv and Naftali Tishby. Opening the black box of deep neural networks via information. *arXiv:1703.00810*, 2017. 4

[36] Naftali Tishby, Fernando C Pereira, and William Bialek. The information bottleneck method. *arXiv preprint physics/0004057*, 2000. 4

[37] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *Proc. of NeurIPS*, pages 1–11, 2017. 3

[38] Tao Wang, Li Yuan, Xiaopeng Zhang, and Jiashi Feng. Distilling object detectors with fine-grained feature imitation. In *Proc. of CVPR*, pages 4933–4942, 2019. 5

[39] Yulin Wang, Zanlin Ni, Shiji Song, Le Yang, and Gao Huang. Revisiting locally supervised learning: an alternative to end-to-end training. In *Proc. of ICLR*, pages 1–21, 2021. 7

[40] Ziwei Wang, Ziyi Wu, Jiwen Lu, and Jie Zhou. Bidet: An efficient binarized object detector. In *Proc. of CVPR*, pages 2049–2058, 2020. 4, 8

[41] Zheng Xie, Zhiquan Wen, Jing Liu, Zhiqiang Liu, Xixian Wu, and Mingkui Tan. Deep transferring quantization. In *Proc. of ECCV*, pages 625–642, 2020. 2

[42] Sheng Xu, Yanjing Li, Tiancheng Wang, Teli Ma, Baochang Zhang, Peng Gao, Yu Qiao, Jinhu Lü, and Guodong Guo. Recurrent bilinear optimization for binary neural networks. In *Proc. of ECCV*, pages 19–35, 2022. 1

[43] Sheng Xu, Yanjing Li, Bohan Zeng, Teli Ma, Baochang Zhang, Xianbin Cao, Peng Gao, and Jinhu Lü. Ida-det: An information discrepancy-aware distillation for 1-bit detectors. In *Proc. of ECCV*, pages 346–361, 2022. 1

[44] Sheng Xu, Junhe Zhao, Jinhu Lu, Baochang Zhang, Shumin Han, and David Doermann. Layer-wise searching for 1-bit detectors. In *Proc. of CVPR*, pages 5682–5691, 2021. 1

[45] Shuchang Zhou, Yuxin Wu, Zekun Ni, Xinyu Zhou, He Wen, and Yuheng Zou. Dorefa-net: Training low bitwidth convolutional neural networks with low bitwidth gradients. *arXiv preprint arXiv:1606.06160*, 2016. 2

[46] Chenzhuo Zhu, Song Han, Huizi Mao, and William J Dally. Trained ternary quantization. In *Proc. of ICLR*, pages 1–10, 2017. 2

[47] Xizhou Zhu, Weijie Su, Lewei Lu, Bin Li, Xiaogang Wang, and Jifeng Dai. Deformable detr: Deformable transformers for end-to-end object detection. In *Proc. of ICLR*, pages 1–16, 2020. 3

[48] Bohan Zhuang, Chunhua Shen, Mingkui Tan, Lingqiao Liu, and Ian Reid. Towards effective low-bitwidth convolutional neural networks. In *Proc. of CVPR*, pages 7920–7928, 2018. 2