Frontiers of Information Technology & Electronic Engineering
www.jzus.zju.edu.cn; engineering.cae.cn; www.springerlink.com
ISSN 2095-9184 (print); ISSN 2095-9230 (online)
E-mail: jzus@zju.edu.cn

End-to-end object detection using a query-selection encoder with hierarchical feature-aware attention*

Zuyi WANG[1], Zhimeng ZHENG[1], Jun MENG[‡1,2], Li XU[‡1,2]

[1]*College of Electrical Engineering, Zhejiang University, Hangzhou 310027, China*

[2]*Zhejiang University Robotics Institute, Yuyao 315400, China*

E-mail: zuyiwang@zju.edu.cn; zhengzhimengzju@foxmail.com; junmeng@zju.edu.cn; xupower@zju.edu.cn

Received Oct. 29, 2024; Revision accepted Feb. 9, 2025; Crosschecked

Unedited

Abstract: End-to-end object detection methods have attracted extensive interest recently since they alleviate the need for complicated human-designed components and simplify the detection pipeline. However, these methods suffer from slower training convergence and inferior detection performance compared to conventional detectors, as their feature fusion and selection processes are constrained by insufficient positive supervision. To address this issue, we introduce a novel query-selection encoder (QSE) designed for end-to-end object detectors to improve training convergence speed and detection accuracy. The QSE is composed of multiple encoder layers stacked on top of the backbone. A lightweight head network is added after each encoder layer to continuously optimize features in a cascading manner, providing more positive supervision for efficient training. Additionally, a hierarchical feature-aware attention (HFA) mechanism is incorporated in each encoder layer, including in-level feature attention and cross-level feature attention, to enhance the interaction between features from different levels. HFA can effectively suppress similar feature representations and highlight discriminative ones, thereby accelerating the feature selection process. Our method is highly versatile in accommodating both CNN-based and transformer-based detectors. Extensive experiments were conducted on the popular benchmark datasets MS COCO, CrowdHuman and PASCAL VOC to demonstrate the effectiveness of our method. The results showed that CNN-based and transformer-based detectors using QSE can achieve better end-to-end performance in a shorter training setting.

Key words: End-to-end object detection; Query-selection encoder; Hierarchical feature-aware attention;
https://doi.org/10.1631/FITEE.2400960 **CLC number:**

1 Introduction

Object detection is a crucial task in computer vision which aims to find targets of interest in images by circling bounding boxes and predicting categories (Pu et al., 2021; Qin et al., 2023; Wang et al., 2023). Traditional object detectors built by convolutional neural networks (CNN) adopt a dense prediction paradigm, which performs classification and localization tasks based on pre-defined densely tiled bounding boxes (Girshick, 2015; Ren et al., 2015; Lin et al., 2017) or grid points in the 2D image plane (Tian et al., 2019; Zhou et al., 2019). One-to-many label assignments are the core scheme of these methods, where each ground-truth box is assigned to multiple predictions of detectors as the supervised target. Despite their excellent performance, these detectors rely heavily on hand-designed components, i.e., Non-maximum Suppression (NMS), to remove duplicated predictions during inference, which introduces additional hyper-parameters to tune causing sub-optimal performance in dense scenes (Li et al., 2023b; Zhang et al., 2023).

To achieve a more flexible end-to-end detection, detection Transformer (DETR) (Carion et al., 2020)

‡ Corresponding authors
* Project supported by the Robotics Institute of Zhejiang University (No. K12106)
ORCID: Zuyi WANG, https://orcid.org/0000-0001-7652-268X
© Zhejiang University Press 2025

is proposed, viewing object detection as a set prediction problem and introducing a transformer encoder-decoder architecture. DETR adopts a sparse prediction paradigm, which reasons about the global image context and outputs the final predictions by using a small set of learnable object queries. The one-to-one label assignment plays a crucial role in DETR for conducting end-to-end detection, where each ground-truth box is assigned to only one prediction. Hence, DETR outputs only a single prediction for each object during inference and NMS is no longer necessary. This approach has encouraged many subsequent improvements (Yao et al. (2021); Wang et al. (2022); Li et al. (2023a)). In addition, POTO (Wang et al. (2021)) and OneNet (Sun et al. (2021a)) attempt to adopt the one-to-one label assignment in CNN-based detectors to realize end-to-end detection. However, these methods suffer from extremely slow training convergence and relatively low performance on small objects. The core reason for this problem is a conflict between the one-to-one label assignment and sufficient positive supervision (Jia et al. (2023); Hou et al. (2024)). During the training process, the one-to-one matching scheme assigns a single positive prediction to each ground-truth box, which leads to negative predictions dominating most of the loss function, causing insufficient positive supervision. Therefore, more training iterations are required for convergence. To alleviate this issue, previous studies introduced additional train-only architectures, such as query denoising (Li et al. (2022)), multiple groups of queries (Chen et al. (2023)) and auxiliary queries (Jia et al. (2023)) to provide more supervision. Despite these improvements, there is still a gap in training efficiency between end-to-end detectors and the traditional methods using one-to-many label assignments.

In this study, we aimed to eliminate this gap while maintaining the merit of end-to-end detectors. To address this challenge, we conducted a comparative analysis by visualizing the multi-scale classification feature maps ($P3, P4, P5, P6$) generated by two CNN-based detectors, FCOS, a conventional detector (Tian et al. (2019)) and POTO, an end-to-end detector (Wang et al. (2021)). As shown in Fig. 1, the activated feature of POTO is concentrated mainly within a more delimited area at a certain level ($P5$ and $P6$ feature maps) while FCOS enables a wider activation area on the feature maps at multiple levels. Fig. 1 illustrates that these sparsely activated features are required for end-to-end detection, and are more beneficial for detectors to output the only prediction for each object. Therefore, end-to-end detectors need to filter features, suppressing similar representations and highlighting the most discriminative. Additionally, for multiple levels of features with different scales, only a single level of features is activated for each object. However, in end-to-end detection, this feature filtering process is guided by sparse positive supervision, leading to long training iterations. Motivated by this observation, we propose a novel query-selection encoder (QSE) to improve the feature filtering process for end-to-end detection, which consists of multiple encoder layers, stacked on top of a backbone. The QSE takes multi-level features with different scales as input, continuously selecting and filtering them in a cascading manner to enhance feature representation. In each encoder layer, we incorporate a hierarchical feature-aware attention (HFA) mechanism, including in-level feature attention (ILFA) and cross-level feature attention (CLFA), to help feature filtering and selection within a single level and across multiple levels. In addition, the QSE can be flexibly applied to both transformer-based and CNN-based detectors, enabling them to achieve compa-

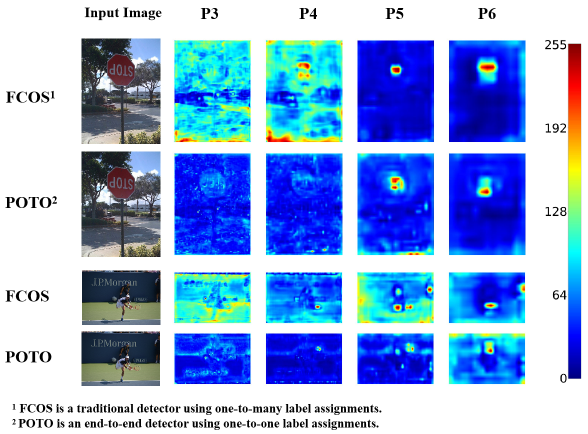

Fig. 1 Visualizations of classification feature maps produced by FCOS, a conventional method (Tian et al. (2019)) and POTO, an end-to-end detector (Wang et al. (2021)). $P3, P4, P5, P6$ **represent different levels of feature maps generated by FPN. The features of FCOS are activated on multiple levels (output high values). In contrast, the activated features of POTO are concentrated mainly within a more delimited area at certain levels. These results indicate that sparsely-activated feature maps are required for end-to-end detection**

Unedited

rable or better end-to-end detection performance in shorter training iterations. Specifically, the traditional CNN-based detectors achieve competitive end-to-end performance by using the QSE within shorter training epochs, which is superior to previous end-to-end methods. Transformer-based detectors with the QSE can also boost performance in the same training setting. The code will be made available at *https://github.com/ZuyiWang/QSE*.

In summary, the main contributions of the paper are as follows:

- We introduce a query-selection encoder to improve the training efficiency for end-to-end detectors, which can help filter and select multi-level features to accelerate training convergence. The QSE can be applied to both transformer-based and CNN-based detectors to help them achieve comparable or better performance in shorter training iterations.
- In each QSE layer, hierarchical feature-aware attention is incorporated, comprising in-level feature attention and cross-level feature attention, to enhance the interaction between features within a single level and across multiple levels, helping suppress similar feature representations and highlight discriminative ones for end-to-end detection.
- Comprehensive experiments on MS COCO, PASCAL VOC and CrowdHuman datasets consistently demonstrated the superiority and generalization ability of our proposed QSE. In particular, our QSE-ATSS method outperformed previous end-to-end CNN-based methods. The transformer-based detectors incorporating QSE also achieved performance improvements.

The rest of the paper is organized as follows: related studies are briefly reviewed in Section 2. The proposed method is introduced in detail in Section 3. The experimental results are presented in Section 4. Finally, the conclusion is given in Section 5.

2 Related studies

2.1 CNN-based object detection

Object detection aims to find all the objects of interest in an image by predicting their location and categories. Early detectors were based on convolutional neural networks and can be divided into anchor-based and anchor-free methods. Anchor-based detectors (Ren et al. (2015); Redmon et al. (2016); Lin et al. (2017)) were proposed first, which use predefined anchor boxes as references to locate and classify objects. The size and scale of these anchor boxes are fixed in advance and subsequently adjusted to determine the actual object bounding boxes. As a milestone in object detection, Faster R-CNN (Ren et al. (2015)) uses a region proposal network to generate candidate regions and performs classification and localization on these regions through a detection network. SSD (Liu et al. (2016)) generates anchor boxes at multiple feature maps with different resolutions, thereby achieving multi-scale detection. Despite the good performance of anchor-based methods, the hyper-parameters of anchor shapes and sizes have to be carefully tuned across different datasets. To overcome this issue, anchor-free detectors (Tian et al. (2019); Zhou et al. (2019); Law and Deng (2018)) have been proposed to simplify the detection pipeline. These detectors classify the objects directly and regress the location from images by key points or center points detection. For example, FCOS (Tian et al. (2019)) defines anchor points in feature maps and directly regresses the distance from these points to the border of bounding boxes. CornerNet (Law and Deng (2018)) first predicts the top-left and bottom-right points of bounding boxes and then combines them to obtain the predicted boxes.

Although CNN-based detectors achieve good performance in different applications, these methods require a post-processing step, such as NMS, to remove duplicate predictions during inference. The post-processing step introduces additional hyper-parameters and may result in sub-optimal performance in dense scenes. Recently, Wang et al. (2021) and Sun et al. (2021a) introduced one-to-one label assignments into CNN-based detectors and conducted end-to-end detection without NMS. However, owing to the lack of sufficient positive supervision, these detectors need longer training iterations and may suffer a performance drop. To address this issue, we propose a novel query-selection encoder to improve the training efficiency for end-to-end detection. Our method can be adopted in CNN-based detectors to help them achieve performance comparable to that of traditional methods in shorter training iterations.

Unedited

2.2 Transformer-based object detection

Transformer-based detectors have been designed to realize end-to-end detection and have achieved satisfactory performance. DETR (Carion et al. (2020)) introduced a transformer encoder-decoder architecture in object detection and outputs a unique prediction for each object via a one-to-one bipartite matching. DETR uses the encoder to obtain global image context on the features from a backbone. Then a sparse set of learnable object queries is used as the training candidates and interacts with the image features through the decoder. However, DETR suffers from slow convergence and inferior performance on small objects. Owing to its novel paradigm for end-to-end detection, many subsequent studies attempted to improve it by designing new attention mechanisms (Zhu et al. (2021); Dai et al. (2021); Ye et al. (2023)), customized object queries (Wang et al. (2022); Li et al. (2022); Liu et al. (2022)) or additional network architectures (Jia et al. (2023); Chen et al. (2023); Zong et al. (2023)). For example, Deformable DETR (Zhu et al. (2021)) uses a deformable attention module to speed up training, using a small set of sampling locations as a pre-filter for prominent key elements out of all the feature maps. DAB-DETR (Liu et al. (2022)) uses box coordinates directly as queries in transformer decoders and dynamically updates them layer-by-layer for better performance. Group DETR (Chen et al. (2023)) uses multiple groups of object queries and conducts one-to-one assignments within each group to introduce more supervision, thereby improving DETR training. Despite the progress made, there is still a gap in training efficiency between transformer-based detectors and traditional methods. In this study, we bridged this gap from the perspective of feature filtering for end-to-end detection and devised a novel query-selection encoder, which can also be adopted in transformer-based detectors to obtain better performance.

3 Proposed method

3.1 Architecture overview

The network structures of CNN-based and transformer-based detectors can be divided roughly into three parts: feature extraction, feature fusion, and prediction output (Fig. 2). Specifically, CNN-based methods consist of a backbone, neck network and detection head. For transformer-based detectors, there are also three components: a backbone, encoder and decoder. The neck network and the encoder are responsible for further fusing features obtained from the backbone network. Sparse-activated features are needed for end-to-end detection (Fig. 1). However, this is challenging owing to insufficient positive supervision by one-to-one label assignments, resulting in longer training iterations and limited performance for end-to-end detection. To address this issue, we propose a query-selection encoder (QSE) to accelerate the process of feature fusion in end-to-end detectors. The QSE can replace the neck network and the encoder for efficient training and better performance. Especially for CNN-based detectors, the QSE can continuously select and filter the features in a cascading manner through multiple encoder layers. A lightweight head network is added after each encoder layer to provide more positive supervision. In addition, we designed hierarchical feature-aware attention (HFA) within each encoder layer to handle multi-level feature selection for more expressive features. The details of key components of the QSE are described below.

3.2 Query-selection encoder

The overall architecture of the QSE is depicted in Fig. 2. Similar to a vanilla encoder, the QSE consists of L sequentially connected encoder layers, where the output of each layer serves as the input to the next layer. Through this cascading manner, the features are continuously screened and optimized. To improve the detection performance for objects of various sizes, the QSE takes multi-level feature maps with different scales from the backbone network as input. Specifically, $S = \{s_i \in \mathbb{R}^{H_i \times W_i \times K}, \}, i \in \{1, 2, 3, 4\}$ represents a group of multi-level feature maps, where H_i indicates the height and W_i the width of feature maps. K is the number of channels. Before feeding into the QSE, these feature maps are first flattened into one dimension, producing 2D feature vectors $\{f_i \in \mathbb{R}^{H_i W_i \times K}\}, i \in \{1, 2, 3, 4\}$. Then we concatenate these feature vectors alone with the spatial dimension and obtain the input feature F^0 of the QSE:

$$F^0 = Concat(f_1, f_2, f_3, f_4) \in \mathbb{R}^{N_q \times K}, \quad (1)$$

Unedited

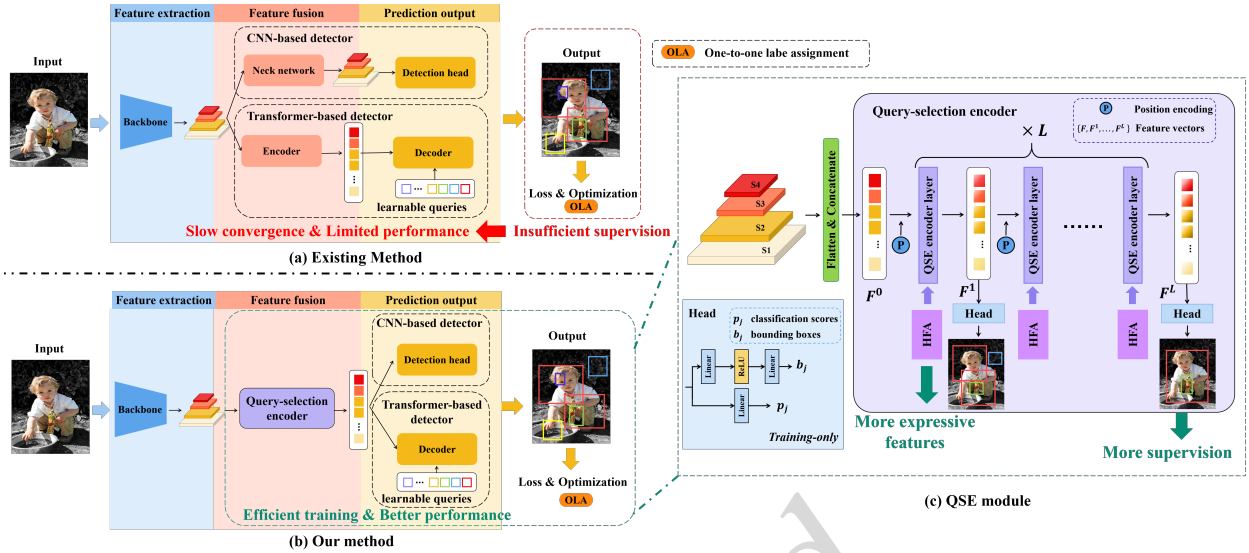

Fig. 2 The overall architecture of the proposed method. (a) Overview of existing methods. The structures of both CNN-based and transformer-based detectors can be divided roughly into three parts: feature extraction, feature fusion and prediction output. Due to insufficient positive supervision by one-to-one label assignment, these end-to-end detectors suffer from slow training convergence and limited performance. (b) Overview of our method. The proposed QSE module can work in the feature fusion part to replace the neck network and the encoder for efficient training and better performance. (c) The structure of the QSE module. The QSE takes multi-level features as input and consists of several encoder layers used to filter and select features in a sequential manner. HFA helps multi-level feature fusion for more expressive features, and a lightweight head network is added after each encoder layer to predict objects, which can provide more positive supervision

Unedited

where $Concat(\cdot)$ indicates a concatenation operation. $N_q = \sum_{i=1}^{4} H_i W_i$ is the length of the feature vector. As the feature maps are flattened from 3D matrices to 2D vectors, their spatial information is lost. Therefore, it is essential to introduce positional encoding, which is added to the feature F before being input into the QSE. The calculation process of the QSE can be formulated as:

$$F^j = QE_j(F^{j-1} + PE), j \in \{1, 2, ..., L\}. \quad (2)$$

Here, QE_j represents the calculation process of the jth QSE layer and PE is the position encoding. L is the number of QSE layers. The original encoder (Carion et al. (2020)) also uses a layer-stacked structure to extract the global relation of input features through the self-attention mechanism, which is indirectly trained by calculating the loss through the output of the decoder. This training scheme of the encoder is implicit and inefficient. In contrast, we introduce HFA to help fuse and select input features in each QSE layer. To train the QSE efficiently and obtain sparsely activated features for end-to-end detection, we designed a lightweight head network composed of several linear layers. The feature vector $F^j, j \in \{1, 2, ..., L\}$ from each QSE layer is fed into this head network to predict categories and bounding boxes. The classification scores $p_j \in \mathbb{R}^{N_q \times C}$ are output through one linear layer (Fig. 2). The bounding boxes $b_j \in \mathbb{R}^{N_q \times 4}$ are predicted by two linear layers and a ReLU function. C is the number of object categories. The one-to-one label assignment is used to calculate classification and localization losses based on these predictions. This structure can provide more positive supervision for efficient training. Besides, compared with a vanilla encoder, the QSE adopts a more direct training scheme, which can ensure that the feature fusion of each QSE layer is conducive to end-to-end object detection. Note that these head networks are used only in training and incur no computational cost during inference.

3.3 Hierarchical feature-aware attention

The traditional encoder layer in transformer-based detectors has a standard architecture including a multi-head self-attention (MHSA) module and a fully connected feed-forward network (FFN). The encoder layer models the global context relationships of the input feature vector and extracts significant in-

formation through MHSA. The attention mechanism is the core operation in the encoder layer. In particular, given three input vectors — a query, a key and a value — the similarity between the query and the key vectors is calculated as a weight assigned to the value vector. This calculation can be formulated as:

$$Att(Q,K,V) = softmax(\frac{QK^T}{\sqrt{d_k}}) \cdot V, \quad (3)$$

where $\sqrt{d_k}$ is a temperature parameter to avoid gradient disappearance in the $softmax$ function. Especially, Q, K, V are the same feature vector for self-attention and are different for cross-attention. Multi-head attention (MHA) is an extension of the attention mechanism, which enables the model to focus on different aspects of information by using multiple groups of learnable weights. The calculation of multi-head attention can be formulated as:

$$\begin{cases} MHA(Q,K,V) = Concat(H_1,\ldots,H_{n_h})W^O, \\ H_i = Att(QW_i^Q, KW_i^K, VW_i^V), \quad i \in \{1,\ldots,n_h\}, \end{cases} \quad (4)$$

where n_h is the number of heads. W^O, W_i^Q, W_i^K and W_i^V are learnable weights to project the query, the key and the value to different dimensions. The features processed by the attention mechanism are integrated in the form of a residual, defined as:

$$MHA_r(Q,K,V) = LN(Q + MHA(Q,K,V)). \quad (5)$$

Here LN denotes the layer normalization operation. Finally, an FFN module, comprising two linear layers and a ReLU activation function, is used to enhance feature representations. The overall calculation process of a traditional encoder layer Ec can be summarized as follows:

$$Ec = LN(MHA_r(Q,K,V) + FFN(MHA_r(Q,K,V))). \quad (6)$$

Since the encoder adopts self-attention, Q, K and V in Eq. 6 are all input features.

A traditional encoder layer processes the input feature vector as a whole, extracting relationships within the feature sequence. However, in the QSE, the input features are obtained by flattening and concatenating multi-level feature maps. According to the analysis in Fig. 1, sparsely activated feature maps are more conducive to end-to-end detection, which can help the model output a unique prediction for each object. Therefore, in addition to extracting

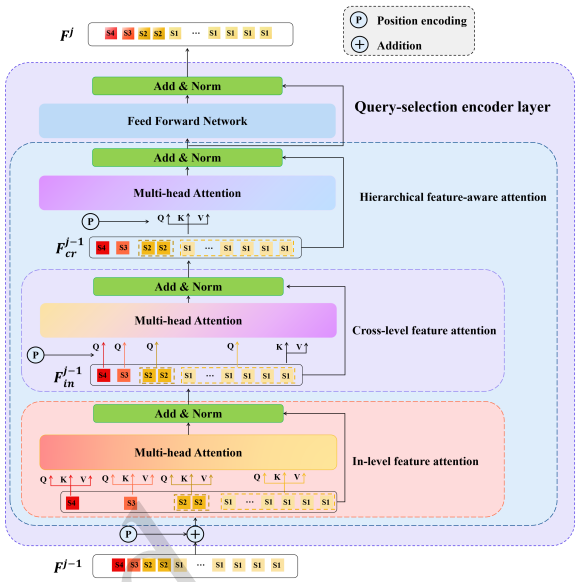

Fig. 3 Detailed illustration of the query-selection encoder layer. $S1, S2, S3, S4$ are the features from different levels. HFA enhances the interaction between different levels of features to stress the discriminative representations for end-to-end detection. The interaction is implemented in a single level of features by ILFA and across multiple levels of features by CLFA. Finally, the feature vector is output through a feed-forward network

global context relationships, the QSE should also filter and select features, suppressing similar feature representations and highlighting significant ones. We argue that the features from different levels should be processed separately, as each level of feature maps contains information about the entire image, and the feature maps at different levels reflect information on distinct scales and depths of the image. Hence, first, the features of the same level interact with each other to fuse and filter similar representations. Then the features of different levels interact to retain the most significant representations. We use hierarchical feature-aware attention (HFA) to implement the above operation, which consists of an in-level feature attention (ILFA) and a cross-level feature attention (CLFA) mechanism. The implementation of the QSE layer is shown in Fig. 3. Specifically, given the input feature vector F^{j-1} of the jth QSE layer, where $j \in \{1, 2, ..., L\}$, it contains the features from four levels

$$F^{j-1} = Concat(f_1^{j-1}, f_2^{j-1}, f_3^{j-1}, f_4^{j-1}). \quad (7)$$

Here, f_1^{j-1}, f_2^{j-1}, f_3^{j-1} and f_4^{j-1} indicate the vectors from feature maps of the four levels. The ILFA performs multi-head self-attention calculation on the

Unedited

features of each level and outputs a new feature vector F_{in}^{j-1}, which can be formulated as:

$$\begin{cases} F_{in}^{j-1} = Concat(f_{1,in}^{j-1}, f_{2,in}^{j-1}, f_{3,in}^{j-1}, f_{4,in}^{j-1}), \\ f_{i,in}^{j-1} = MHA_r(Q_i^{j-1}, Q_i^{j-1}, Q_i^{j-1}), i \in \{1,2,3,4\}, \\ Q_i^{j-1} = f_i^{j-1} + PE, i \in \{1,2,3,4\}. \end{cases} \tag{8}$$

ILFA captures the feature relationships within the same level and fuses similar feature representations. Then, the output feature vector F_{in}^{j-1} is processed through CLFA, which uses features at different levels as query vectors and performs interactive calculations with multi-level features. The calculation of CLFA can be defined as:

$$\begin{cases} F_{cr}^{j-1} = Concat(f_{1,cr}^{j-1}, f_{2,cr}^{j-1}, f_{3,cr}^{j-1}, f_{4,cr}^{j-1}), \\ f_{i,cr}^{j-1} = MHA_r(Q_{i,in}^{j-1}, V_{i,in}^{j-1}, V_{i,in}^{j-1}), i \in \{1,2,3,4\}, \\ Q_{i,in}^{j-1} = f_{i,in}^{j-1} + PE, i \in \{1,2,3,4\}, \\ V_{i,in}^{j-1} = F_{in}^{j-1} + PE, i \in \{1,2,3,4\}. \end{cases} \tag{9}$$

F_{cr}^{j-1} is the output feature vector of CLFA. CLFA captures the feature relationships across different levels to highlight the most discriminative feature representations and effectively filter out similar features at other levels. Subsequently, a multi-head self-attention module and an FFN module are used in the QSE layer to further enhance the feature representations, which can be denoted as:

$$\begin{cases} F^j = LN(F_{se}^{j-1} + FFN(F_{se}^{j-1})), \\ F_{se}^{j-1} = MHA_r(Q_{cr}^{j-1}, Q_{cr}^{j-1}, Q_{cr}^{j-1}), \\ Q_{cr}^{j-1} = F_{cr}^{j-1} + PE, \end{cases} \tag{10}$$

where $F^j \in \mathbb{R}^{N_q \times K}$ is the output feature vector of the jth QSE layer comprising features from four levels, which is also the input feature vector of next layer. The deformable attention (Zhu et al. (2021)) is used in the multi-head attention module in our QSE to reduce computational complexity. Accordingly, the QSE refines the features gradually via multiple encoder layers. Within each layer, HFA, including ILFA and CLFA, makes the features interact with each other at the same level and between different levels, thereby enhancing core feature representations and filtering out irrelevant ones. Such a design explicitly makes the model learn powerful representations of each object and promotes exploration for sparse-activated features for end-to-end detection. In addition, the QSE is highly versatile in accommodating CNN-based and transformer-based detectors. In particular, the output feature vector F^L can be used as the key and value vectors in the transformer decoder, which can also be restored to multi-level feature maps for detection head networks in CNN-based detectors.

3.4 Optimization

We apply a multi-stage enhancement strategy to train the proposed QSE. L query-selection encoder layers are cascaded and trained in an end-to-end manner. A lightweight head network is used to process the feature vector $F^j, j \in \{1,2,...,L\}$ after the jth QSE layer and predict classification scores p_j and bounding boxes b_j. For each group of predictions, a one-to-one label assignment is used to calculate the losses. Therefore, given the ground-truth bounding boxes b_{gt} and their corresponding labels y_{gt}, the loss function L_{QSE} can be described as:

$$L_{QSE} = \sum_{j=1}^{L} L_{cls}(y_{gt}, p_j) + L_{loc}(b_{gt}, b_j), \tag{11}$$

where L_{cls} is classification Focal Loss (Lin et al. (2017)) and L_{loc} is localization GIoU loss (Rezatofighi et al. (2019)). This training scheme can provide more positive supervision to improve training efficiency. Besides, L_{QSE} can explicitly ensure that the optimization of each QSE layer facilitates end-to-end object detection, enabling the detector to learn robust feature representations. The QSE can be applied to CNN-based and transformer-based methods and the detectors are trained in an end-to-end approach using one-to-one label assignments. The overall loss function is formulated as follows:

$$L_{all} = L_{det} + \lambda L_{QSE}, \tag{12}$$

where λ is a weighting coefficient. L_{det} is the original loss function in the detector. For the CNN-based methods, L_{det} consists of classification and localization losses, and may also include IoU loss (Kim and Lee (2020)) or centerness loss (Tian et al. (2019)) for certain detectors. For the transformer-based methods, L_{det} comprises the classification and localization losses from the decoders, and may also include auxiliary losses for some detectors (Li et al. (2022); Zhang et al. (2023)).

Unedited

4 Experiments

In this section, we first introduce the datasets and the experimental settings. Then, we present the results of our proposed method on multiple benchmarks, with comparisons with other end-to-end algorithms. Finally, we describe ablation studies conducted to show the contribution of each component and the detailed designs.

4.1 Datasets

To verify the effectiveness of our method, we conducted comprehensive experiments on three public benchmark datasets: MS COCO 2017 (Lin et al. (2014)),PASCAL VOC (Everingham et al. (2010)) and CrowdHuman (Shao et al. (2018)). MS COCO, encompassing 80 classes, contains 118k images for training and 5k images for validation. The standard COCO metric, Average Precision (AP) was used as the evaluation metric. PASCAL VOC consists of 20 classes, including 5k images from VOC 2007 and 11k images from VOC 2012 for training. Moreover, it has another 5k images from VOC 2007 for testing. The mean AP (mAP) was used as the evaluation metric. CrowdHuman is a widely used dataset for human detection in crowded scenes, and has 15k images for training and 4k images for validation. As suggested by the official paper (Shao et al. (2018)), the average log miss rate over false positives per-image (mMR) was used as the main metric. In addition, AP and recall results are reported for reference.

Notably, the selection of these three datasets followed previous work (Wang et al. (2021); Sun et al. (2021a); Zhang et al. (2023); Chen et al. (2022)) and facilitated comparison with other methods. Since MS COCO 2017 and PASCAL VOC cover a wide range of categories and have sufficient training data in various scenes, the experiments on these two datasets can illustrate the effectiveness and generalization of our method. In addition, to verify the effectiveness of the QSE in dense and challenging scenes, we conducted experiments on the CrowdHuman dataset, which is a high-quality dataset for pedestrian detection and crowd detection. It is specially designed for crowded scenes and contains a large number of pedestrian annotations useful for improving the performance of object detection in crowded environments. Hence, experiments on CrowdHuman can demonstrate QSE's performance in dense scenes.

4.2 Implementation Details

Our model was implemented by using the open-source toolbox MMDetection (Chen et al. (2019)), and ResNet-50 (He et al. (2016)) pre-trained on the ImageNet (Deng et al. (2009)) was used as the backbone network. The loss coefficient λ was set to 1 and the number of QSE layers L was set to 6. To illustrate the robustness of our method, we conducted experiments on both CNN-based and transformer-based detectors, including FCOS (Tian et al. (2019)), ATSS (Zhang et al. (2020)), Deformable-DETR (Zhu et al. (2021)) and DINO (Zhang et al. (2023)). The standard Adam was used as the optimizer, with a learning rate of 0.0002 and weight decay of 0.0001. Most experiments had a $1\times$ training schedule (Chen et al. (2019)), in which there were 12 epochs and the learning rate was reduced by a factor of ten after 11 epochs. For a fair comparison, we also conducted experiments with 36 epochs and the learning rate reduced tenfold after 30 epochs. All the experiments were implemented on a server with two Intel(R) Xeon(R) E5-2670 V3 CPUs, 128 GB RAM and 8 NVIDIA GeForce RTX 3090 GPUs. The batch size was set to 16 images for all experiments (two images per GPU).

4.3 Comparison With State-of-the-Art systems

In this section, we compare the performance of our QSE on three widely used benchmarks with that of state-of-the-art end-to-end detectors, including CNN-based and transformer-based methods. FCOS (Tian et al. (2019)) and ATSS (Zhang et al. (2020)) were selected as the baseline for CNN-based detectors, and Deformable-DETR (Zhu et al. (2021)) and DINO (Zhang et al. (2023)) for the transformer-based methods. The results show that our QSE performed better and improved the training efficiency on these datasets.

4.3.1 MS COCO Dataset

To demonstrate the effectiveness of our approach, we applied our QSE to CNN-based and transformer-based detectors and compared them with various end-to-end methods on the MS COCO dataset. The results are presented in Table 1. For

Unedited

Table 1 Comparison of performance of the QSE with that of current mainstream end-to-end methods on the MS COCO Dataset. The evaluation metric is AP. * represents the baseline of our method

Method	Epoch	NMS	Query	Label Assignment	AP(%)	AP_{50}(%)	AP_{75}(%)	AP_S(%)	AP_M(%)	AP_L(%)
CNN-based detectors:										
FCOS* (Tian et al. (2019))	12	✗	-	One-to-one	27.3	42.9	29.6	16.2	30.7	35.5
ATSS* (Zhang et al. (2020))	12	✗	-	One-to-one	29.2	43.7	31.3	17.6	32.8	37.4
POTO (Wang et al. (2021))	12	✗	-	One-to-one	37.8	55.6	41.8	22.1	41.3	48.7
OneNet (Sun et al. (2021a))	12	✗	-	One-to-one	35.7	54.3	38.4	17.9	39.3	48.6
DATE (Chen et al. (2022))	12	✗	-	One-to-one	37.3	55.3	40.7	21.2	40.3	48.8
O2f (Li et al. (2023b))	12	✗	-	One-to-few	38.9	56.7	42.3	23.4	41.7	49.3
YOLOv10-m (Wang et al. (2024))	20	✗	-	One-to-one	35.1	49.1	38.4	19.0	39.3	46.3
QSE-FCOS (Ours)	12	✗	-	One-to-one	38.9	**57.3**	42.1	23.2	42.0	50.1
QSE-ATSS (Ours)	12	✗	-	One-to-one	**39.5**	56.0	**43.1**	**25.7**	**43.4**	**52.6**
POTO (Wang et al. (2021))	36	✗	-	One-to-one	41.4	59.5	45.6	26.1	44.9	52.0
OneNet (Sun et al. (2021a))	36	✗	-	One-to-one	38.9	57.3	42.3	23.9	41.9	49.5
DATE (Chen et al. (2022))	36	✗	-	One-to-one	40.6	58.9	44.4	25.6	44.1	50.9
O2f (Li et al. (2023b))	36	✗	-	One-to-few	42.2	60.2	46.4	26.9	45.7	53.0
QSE-FCOS (Ours)	36	✗	-	One-to-one	42.4	60.3	46.8	27.2	45.8	53.2
QSE-ATSS (Ours)	36	✗	-	One-to-one	**42.8**	**61.0**	**47.1**	**27.3**	**46.2**	**53.7**
Transformer-based detectors:										
DETR (Carion et al. (2020))	500	✗	100	One-to-one	42.0	62.4	44.2	20.5	45.8	61.1
DAB-DETR (Liu et al. (2022))	50	✗	300	One-to-one	42.6	63.2	45.6	21.8	46.2	61.1
DN-DETR (Li et al. (2022))	50	✗	300	One-to-one	44.1	64.4	46.7	22.9	48.0	63.4
Deformable-DETR* (Zhu et al. (2021))	12	✗	300	One-to-one	43.3	62.2	46.7	27.1	46.5	57.2
DINO* (Zhang et al. (2023))	12	✗	300	One-to-one	47.9	65.6	52.0	30.5	50.7	63.0
QSE-Deformable-DETR (Ours)	12	✗	300	One-to-one	44.5	63.3	48.3	27.7	47.7	59.4
QSE-DINO (Ours)	12	✗	300	One-to-one	**48.6**	**65.7**	**52.9**	**30.8**	**52.3**	**63.3**

Unedited

CNN-based detectors, FCOS and ATSS trained with the one-to-one label assignment were used as the baseline. The detectors using our QSE achieved the best results. Compared with CNN-based end-to-end detectors, both QSE-FCOS and QSE-ATSS obtained better performance after training for 12 epochs and 36 epochs. In particular, QSE-ATSS achieved 39.5 AP and 42.8 AP after 12 and 36 epochs respectively, which were the best results for CNN-based end-to-end methods. Since FCOS and ATSS are NMS-based methods, they performed poorly when adopting one-to-one label assignment directly without NMS as a post-processing step (as shown in the first two rows in Table 1). In comparison, QSE-FCOS and QSE-ATSS surpassed the baseline by 11.6 AP and 10.3 AP respectively, demonstrating that our QSE can improve training efficiency and help traditional detectors implement end-to-end detection. Notably, as a representative model for object detection, YOLO series detectors achieve cutting-edge performance in terms of accuracy and speed for efficient real-time object detection. There are several custom designs in YOLO-based detectors to balance computational cost and detection performance, including efficient architectures, strong data augmentation strategies and longer training iterations. In this experiment, YOLOv10-m was trained for 20 epochs. Despite the longer training iterations than other methods, its end-to-end performance did not show a significant advantage over other detectors. Hence, YOLO detectors need a longer training time to show their advantages in real-time end-to-end object detection. Besides, the results of transformer-based detectors also show that QSE-Deformable-DETR and QSE-DINO achieved superior performance than others. Compared with DETR, DAB-DETR and DN-DETR, our methods performed better with fewer training epochs. Compared with the baseline models, QSE-Deformable-DETR and QSE-DINO also showed improvements.

4.3.2 CrowdHuman Dataset

We performed experiments to compare our methods with mainstream end-to-end detectors on the CrowdHuman dataset. The results are reported in Table 2. We applied the QSE to FCOS and ATSS to evaluate our method. The baseline models were trained with one-to-one label assignment and tested without NMS. As can be seen, QSE-FCOS and QSE-ATSS performed better than the competitors. The results of FCOS and ATSS without NMS were extremely poor, due to directly using one-to-one la-

Table 2 Comparison of performance of the QSE with that of end-to-end methods on the CrowdHuman dataset. The evaluation metric is mMR. AP and recall are also reported. * represents the baseline of our method

Method	Epoch	NMS	Label Assignment	$mMR(\%)\downarrow$[1]	$AP(\%)\uparrow$	$Recall(\%)\uparrow$
FCOS * (Tian et al. (2019))	50	✗	One-to-one	64.0	82.9	89.8
ATSS * (Zhang et al. (2020))	50	✗	One-to-one	60.6	85.6	90.1
DETR (Carion et al. (2020))	300	✗	One-to-one	80.6	66.1	-
Deformable-DETR (Zhu et al. (2021))	50	✗	One-to-one	50.0	89.1	95.3
POTO (Wang et al. (2021))	50	✗	One-to-one	52.0	88.7	96.6
OneNet[2] (Sun et al. (2021a))	-	✗	One-to-one	50.0	90.1	**97.9**
DATE[2] (Chen et al. (2022))	-	✗	One-to-one	49.0	90.5	**97.9**
O2f (Li et al. (2023b))	50	✗	One-to-few	45.2	90.9	**97.9**
QSE-FCOS (Ours)	50	✗	One-to-one	45.8	91.3	96.3
QSE-ATSS (Ours)	50	✗	One-to-one	**44.9**	**91.4**	96.5

[1] The arrows indicate that the larger or smaller is better.
[2] OneNet and DATE were trained for 30k iterations as reported in the official paper (Chen et al. (2022))

Unedited

bel assignment, especially for the large number of crowded scenes in the CrowdHuman dataset. In detail, QSE-FCOS attained an mMR of 45.8, outperforming FCOS (Tian et al. (2019)) by 18.2 points, and QSE-ATSS obtained an mMR of 44.9, surpassing the baseline by 15.7 points. Compared with end-to-end detectors, our methods showed superior results in terms of mMR, AP and Recall. In particular, QSE-FCOS and QSE-ATSS performed better than CNN-based end-to-end detectors, including POTO, OneNet and O2f. Our methods were superior to transformer-based Deformable-DETR by about 5 points in the mMR metric. Note that the results of Deformable-DETR were slightly lower than those of CNN-based methods, which may be because that there are more dense scenes in the CrowdHuman dataset and these scenes are not conducive to detection paradigms using sparse queries. In summary, these results on the CrowdHuman dataset demonstrate the leading performance of our method.

4.3.3 PASCAL VOC Dataset

To make the experiments more comprehensive, we tested our method on the PASCAL VOC dataset. All the detectors were trained under the same setting: 12 epochs and an input size of 1000×600. We compared QSE-FCOS and QSE-ATSS with other end-to-end detectors to evaluate our proposed method. FCOS and ATSS trained with the one-to-one label assignment were used as the baseline. The results are reported in Table 3. Our approache achieved excellent results compared with the other detectors. Specially, QSE-ATSS obtained an mAP of 78.0, the best performance among the results of all the end-to-end methods. Compared with the transformer-based Deformable-DETR, QSE-ATSS showed an improvement of 1.6 mAP. Although QSE-FCOS did not perform better than others, it surpassed the baseline by 12.1 mAP, which indicates the effectiveness of the QSE. The performance of POTO (Wang et al. (2021)) and OneNet (Sun et al. (2021a)) was poor, since they were trained on 12 epochs with input images of a single size for fair comparison with the competitors, which differs from the 36 epochs with multi-scale input sizes stated in their official paper. In contrast, our method performed better performance within 12 epochs, illustrating that the QSE can improve training efficiency effectively for end-to-end detection.

Table 3 Comparison of performance of the QSE with that of end-to-end methods on the PASCAL VOC dataset. The evaluation metric is mAP. * represents the baseline of our method

Method	Label assignment	NMS	$mAP(\%)\uparrow$[1]
FCOS* (Tian et al. (2019))	One-to-one	✗	60.1
ATSS* (Zhang et al. (2020))	One-to-one	✗	63.9
Deformable DETR (Zhu et al. (2021))	One-to-one	✗	76.4 [3]
Sparse R-CNN (Sun et al. (2021b))	One-to-one	✗	73.0 [2]
POTO (Wang et al. (2021))	One-to-one	✗	41.2
OneNet (Sun et al. (2021a))	One-to-one	✗	37.3
DATE (Chen et al. (2022))	One-to-one	✗	68.4
O2f (Li et al. (2023b))	One-to-one	✗	75.8
QSE-FCOS (Ours)	One-to-one	✗	72.2
QSE-ATSS (Ours)	One-to-one	✗	**78.0**

[1] The arrows indicate that the larger or smaller is better.
[2, 3] Sparse R-CNN and Deformable DETR were trained on PASCAL VOC with 100 queries.

Table 4 Ablation results of the QSE on the MS COCO dataset. The evaluation metric is AP

Method	Epoch	NMS	Label Assignment	AP(%)	AP_{50}(%)	AP_{75}(%)	AP_S(%)	AP_M(%)	AP_L(%)
FCOS (Tian et al. (2019))	12	✓	One-to-many	38.7	57.4	41.8	22.9	42.5	50.1
		✗	One-to-many	17.8	23.9	19.8	13.9	22.8	24.1
		✓	One-to-one	28.0	44.2	30.2	16.2	30.9	36.7
		✗	One-to-one	27.3	42.9	29.6	16.2	30.7	35.5
QSE-FCOS	12	✓	One-to-one	**39.0**	**57.6**	**42.1**	23.1	41.9	**50.1**
QSE-FCOS	12	✗	One-to-one	38.9	57.3	**42.1**	**23.2**	**42.0**	**50.1**
ATSS (Zhang et al. (2020))	12	✓	One-to-many	39.4	**57.0**	42.8	23.6	42.9	50.3
		✗	One-to-many	19.6	25.8	21.7	14.7	24.0	25.7
		✓	One-to-one	29.9	45.9	31.7	17.7	32.9	38.7
		✗	One-to-one	29.2	43.7	31.3	17.6	32.8	37.4
QSE-ATSS	12	✓	One-to-one	**39.6**	**57.0**	42.9	25.6	43.3	**52.6**
QSE-ATSS	12	✗	One-to-one	39.5	56.6	**43.1**	**25.7**	**43.4**	**52.6**
YOLOv8-m (Jocher et al. (2023))	20	✓	One-to-many	33.9	48.2	36.8	16.4	37.2	48.1
		✗	One-to-many	9.9	12.6	10.7	11.3	16.6	14.7
		✓	One-to-one	28.9	43.5	30.7	15.0	30.4	36.0
		✗	One-to-one	28.4	42.0	30.9	15.0	30.9	35.6
QSE-YOLOv8-m	20	✓	One-to-one	**34.2**	**48.5**	37.3	**16.5**	**37.3**	**48.3**
QSE-YOLOv8-m	20	✗	One-to-one	34.0	48.3	37.1	16.4	**37.4**	48.0
Deformable-DETR (Zhu et al. (2021))	12	✗	One-to-one	43.3	62.2	46.7	27.1	46.5	57.2
QSE-Deformable-DETR	12	✗	One-to-one	**44.5**	**63.3**	**48.3**	**27.7**	**47.7**	**59.4**

Table 5 Ablation results of core components in the QSE on the MS COCO dataset. The evaluation metric is AP

Method	ILFA	CLFA	LH [1]	AP(%)	AP_{50}(%)	AP_{75}(%)	AP_S(%)	AP_M(%)	AP_L(%)
ATSS	-	-	-	29.2	43.7	31.3	17.6	32.8	37.4
QSE-ATSS	✗	✗	✗	37.7	53.6	41.2	**25.9**	41.9	49.9
	✗	✗	✓	38.6	55.3	41.8	25.2	42.2	50.9
	✓	✗	✓	39.4	55.8	43.0	25.7	43.3	52.3
	✗	✓	✓	39.3	55.6	42.8	26.0	43.2	51.4
	✓	✓	✓	**39.5**	**56.0**	**43.1**	25.7	**43.4**	**52.6**

[1] LH represents the lightweight head network after each QSE layer.

Unedited

4.4 Ablation Study

We conducted ablation studies on the MS COCO dataset to further verify the effectiveness of the QSE and to investigate the contribution of its core components: in-level feature attention (ILFA), cross-level feature attention (CLFA) and the lightweight head network (LH).

4.4.1 Effect of the QSE

We propose using the QSE to alleviate the slow training convergence of end-to-end detectors caused by the one-to-one label assignment. To illustrate the effectiveness of the QSE, we conducted experiments on the MS COCO benchmark using FCOS, ATSS, YOLOv8-m and Deformable-DETR as the baseline models. The results are shown in Table 4. FCOS, ATSS and YOLOv8-m are conventional CNN-based detectors, which achieve excellent performance when adopting one-to-many label assignments and NMS. Their performance drops significantly when the models are trained with one-to-many label assignments evaluated without NMS, indicating that the post-processing step of NMS is essential for these traditional methods. When FCOS, ATSS and YOLOv8-m are trained for the same epochs using the one-to-one label assignments directly, performance also decreases substantially. Notably, the results are improved by around 0.7 AP if the models are tested with NMS (as shown in the third and fourth rows of the results for FCOS and ATSS), demonstrating that there are still redundant predictions and the models need further training to converge. In comparison, QSE-FCOS and QSE-ATSS achieved end-to-end detection results of 38.9 AP and 39.5 AP, respectively, which are better than the original re-

sults of FCOS and ATSS. In particular, there was no remarkable performance improvement when QSE-FCOS and QSE-ATSS used NMS during inference, which illustrates the absence of duplicate predictions and successful convergence of the model training. In addition, the results of QSE-YOLOv8-m were similar to those of QSE-FCOS and QSE-ATSS, which further prove the effectiveness of our method. Furthermore, we applied our QSE in transformer-based Deformable-DETR, and our method achieved 44.5 AP, an improvement of 1.2 AP performance. These results show that the QSE can help improve training efficiency and performance for end-to-end detection and it can be flexibly applied to CNN-based and transformer-based detectors.

Note that Focal Loss Lin et al. (2017) was devised to alleviate the imbalance of positive and negative supervision. However, the imbalance addressed by Focal Loss differs from the challenges encountered in end-to-end detection. Focal Loss mitigates the issue where negative supervision dominates despite sufficient positive supervision, due mainly to the large number of negative samples. The imbalance issue in end-to-end detection our method attempts to address arises from insufficient positive supervision due to the one-to-one label assignment, which assigns only a single positive sample for each object. Consequently, Focal Loss may not be effective in this context. During the experiments, Focal Loss was used in FCOS, ATSS and YOLOv8-m, but their end-to-end performance remained suboptimal, further supporting this point.

4.4.2 Effect of core components in the QSE

Unlike in vanilla encoders, we devised a lightweight head network after each QSE layer to output predictions separately, which can provide more positive supervision for training. In addition, hierarchical feature-aware attention, including ILFA and CLFA, was designed in the QSE to enhance the interaction within a single level of features and across multiple levels of features, effectively improving end-to-end detection by suppressing similar feature representations and highlighting discriminative ones. To further analyze the effectiveness of these core components in the QSE, we conducted ablation experiments on the MS COCO dataset. All the models were trained for 12 epochs using the one-to-one label assignment and tested without NMS. The results are reported in Table 5. The baseline version of QSE-ATSS without any special components achieved an AP of 37.7, 8.5 AP higher than ATSS, which indicates that the standard encoder facilitates end-to-end detection. However, this result was inferior to that of the traditional detectors. When the lightweight head network was added to the QSE, QSE-ATSS achieved 38.6 AP, implying that LH helps improve performance by providing more positive supervision. Further, when ILFA and CLFA were used in the model, QSE-ATSS obtained improvements of 0.8 AP and 0.7 AP, respectively. Finally, QSE-ATSS achieved the best result of 39.5 AP when all components were integrated, showing that HFL contributes to effective feature selection and fusion, thereby further enhancing end-to-end detection performance. In addition, when ILFA and CLFA were applied at the same time, the performance improvement was limited compared to applying only one of them. We infer that parts of their roles in feature selection and fusion are overlapping. Overall, these results show the contributions of each component in the QSE.

4.4.3 Effect of the head networks in the QSE

In this section, we further explored the effect of the lightweight head networks on performance. In our method, a lightweight head network is used after each QSE layer to output predictions and calculate losses, which can provide more positive supervision for training. To investigate the effects of adopting different label assignments in the head networks, ablation experiments were conducted on the MS COCO dataset using QSE-FCOS and QSE-ATSS. The results are shown in Table 6. QSE-ATSS and QSE-FCOS achieved satisfactory performance when the one-to-one label assignment was used in the head networks. However, when the one-to-many label assignment was adopted, performance dropped substantially. The reason may be the conflict between one-to-many label assignments and the final objective for end-to-end detection. Specifically, although one-to-many strategies can assign more positive samples for each object thereby providing additional supervision, they fail to assist the QSE in effectively fusing and selecting discriminative features. Therefore, one-to-many label assignments will lead to redundant predictions, hindering performance improvements in end-to-end object detection. In contrast, multiple QSE layers can fuse and filter key fea-

Unedited

Fig. 4 Visualization of the heatmap. The activated areas of ATSS are regions of objects, illustrating that ATSS focuses on similar features of objects and there is a high probability of redundant predictions. In contrast, the activated areas of QSE-ATSS are several points of objects, demonstrating that our method can focus on the discriminative features of the objects and is conducive to achieving end-to-end detection.

Unedited

tures for end-to-end detection in a cascade manner when the one-to-one label assignment is used. The features obtained after each QSE layer are refined and more conducive to end-to-end prediction than those of the previous layers. In addition, we evaluated the performance of each head network after the QSE layers. The results are shown in Table 7. The index of head networks i represents the lightweight head network located after the ith QSE layer and NMS was not used during the test. As presented in Table 7, the end-to-end performance of the corresponding head network improved gradually with an increase in the number of QSE layers, which further demonstrates that each QSE layer plays a role in fusing and filtering discriminative features, which are progressively refined. Besides, we observe that the performance of the last head network was close to the final result of the detector. A potential direction for future research is to explore the feasibility of eliminating the final detection head and achieving end-to-end detection solely through the QSE and a simplified head network.

4.5 Visual analysis

In this section, we report the use of a variety of visualization results to verify the effectiveness of the QSE, including visual analysis of the heatmap, training process and detection examples.

4.5.1 Visual analysis of heatmaps

To verify that our method can effectively suppress similar features and obtain discriminative ones, we compared QSE-ATSS and the baseline. The classification features of the last output layer were used for visualization. Specifically, given the detector's classification output feature $F_{cls} \in \mathbb{R}^{C\times H\times W}$, we can obtain the classification heatmaps $F_{hm} \in \mathbb{R}^{1\times H\times W}$ by outputting the maximum value along the feature channel. Here, C, H and W are the channel, height and width of the feature, respectively. The classification heatmaps of these two methods are shown in Fig. 4. Objects can be better detected when the values in the corresponding area are higher. It can be seen that our method can focus on the effective features of the object in images compared to the ATSS method. Since the feature F_{cls} was used to output the final classification scores, the activated areas of our method are narrowly focused on several key points to eliminate redundant predictions. As a result, our method achieved superior performance in end-to-end detection. In contrast, the activated heatmap areas of ATSS are regions, indicating that there will be redundant predictions, hence, NMS is needed as the post-process step. For the dense and

Table 6 Ablation results of different label assignments in the head networks on the MS COCO dataset. The evaluation metric is AP

Method	Epoch	Label Assignment	AP(%)	AP_{50}(%)	AP_{75}(%)	AP_S(%)	AP_M(%)	AP_L(%)
QSE-FCOS	12	One-to-many	31.1	45.9	33.2	18.7	33.7	41.3
		One-to-one	**38.9**	**57.3**	**42.1**	**23.2**	**42.0**	**50.1**
QSE-ATSS	12	One-to-many	32.0	46.4	34.4	19.5	33.2	43.1
		One-to-one	**39.5**	**56.0**	**43.1**	**25.7**	**43.4**	**52.6**

Table 7 Performance results of the head networks in the QSE on the MS COCO dataset. The evaluation metric is AP

Method	Index of Head Networks [1]	AP(%)	AP_{50}(%)	AP_{75}(%)
QSE-ATSS	1	19.2	31.4	20.5
	2	22.1	35.4	21.8
	3	25.1	38.7	27.1
	4	28.1	42.4	30.9
	5	32.1	50.5	35.3
	6	**36.3**	**54.5**	**40.3**

[1] The index of head networks i represents the lightweight head network located after the ith QSE layer.

small object scenes (the cars and the cattle in the images), QSE-ATSS could still focus on the discriminative features of the objects. This illustrates that our method pays more attention to the discriminative features of the object and achieves end-to-end detection.

4.5.2 Visual analysis of training process

To illustrate the effect of our method on the training process, we show the convergence curves of different detectors on the MS COCO dataset. The figure on the right of Fig. 5 displays the results of transformer-based methods. QSE-DINO and QSE-Deformable-DETR converged faster and achieved better performance after 12 epochs than DAB-DETR (Liu et al. (2022)) and Anchor-DETR (Wang et al. (2022)), which need 50 training epochs. In addition, our method can further improve the performance of the baseline, DINO and Deformable-DETR, within the same training epochs. The figure on the left of Fig. 5 presents the results for CNN-based detectors. It is clear that the curve of QSE-ATSS is significantly higher than that of the baseline (ATSS without NMS), indicating that the QSE effectively enhances the training efficiency of these CNN-base methods for end-to-end detection. Compared with other detectors, our method can also ensure a faster convergence curve and ultimately achieve better performance. Since the learning rate of QSE-ATSS dropped only once during training (after the 11th epoch for training 12 epochs and the 30th epoch for training 36 epochs) whereas the learning rate for other methods dropped twice (after the 8th and 11th epochs for the 12-epoch training, the 24th and 33rd epochs for the 36-epoch training), the convergence curve for our method may be lower than that of others for several epochs. However, this did not impact the final performance of QSE-ATSS.

4.5.3 Visual analysis of detection results

To validate the effectiveness of our QSE, we show several detection examples of MS COCO datasets using FCOS (Tian et al. (2019)), O2f (Li et al. (2023b)) and our QSE-FCOS. As shown in Fig. 6, although O2f also achieves end-to-end detection, it output redundant bounding boxes when detecting some larger objects, such as the person in the first column of images, the traffic lights in the second column and the trains in the third column. In comparison, QSE-FCOS performed better in these situations, since the QSE can enhance the interaction between the features of different levels and reduce redundant feature representations. Besides, in scenes with dense and small objects, FCOS and O2f may miss objects that can still be detected by QSE-FCOS, such as the people in the fourth column of images and the skateboard and the backpack in the fifth column. These detection examples further illustrate the effectiveness of our method in challenging situations.

5 Conclusions

In this paper, we propose a novel query-selection encoder specially designed for end-to-end object detection, which incorporates a hierarchical feature attention mechanism to mitigate the slow training convergence and inferior performance problem in current end-to-end approaches. The QSE can effec-

Unedited

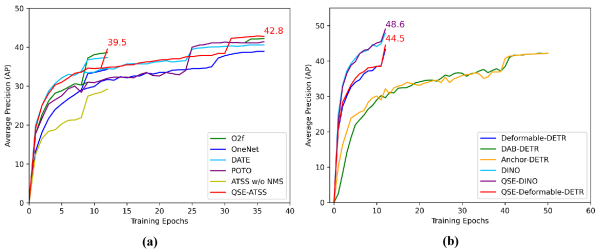

Fig. 5 Convergence curves of different detectors on the MS COCO dataset. (a) The results of transformer-based detectors: QSE-DINO and QSE-Deformable-DETR converged faster and achieved better performance after 12 epochs than other DETR-based methods and the baseline, DINO and Deformable-DETR. (b) The results of CNN-based detectors: the performance of QSE-ATSS was always ahead of the baseline (ATSS without NMS). Compared with other end-to-end detectors, our method ensured a faster convergence curve and achieved better performance

tively enhance the interaction between different levels of features, suppressing similar feature representations and highlighting discriminative ones. With the help of the QSE, end-to-end detectors can improve training efficiency by accelerating the feature selection process and obtain better performance. We conducted extensive experiments on three popular benchmarks. Results showed that our QSE could be applied to both CNN-based and transformer-based detectors outperforming previous end-to-end CNN-based methods and boosting the performance of transformer-based detectors. We hope this work can motivate researchers to design more effective algorithms to further improve the performance of end-to-end object detection.

Unedited

Contributors

Zuyi WANG designed the research and drafted the paper. Zhimeng ZHENG helped organize the paper. Jun MENG and Li XU revised and finalized the paper.

Conflict of interest

All the authors declare that they have no conflict of interest.

References

Carion N, Massa F, Synnaeve G, et al., 2020. End-to-end object detection with transformers. Proc 16th European Conf on Computer Vision, p.213-229. https://doi.org/10.1007/978-3-030-58452-8_13

Chen K, Wang JQ, Pang JM, et al., 2019. MMDetection: Open mmlab detection toolbox and benchmark. https://arxiv.org/abs/1906.07155

Chen Q, Chen XK, Wang J, et al., 2023. Group detr: Fast detr training with group-wise one-to-many assignment. Proc IEEE/CVF Int Conf on Computer Vision, p.6633-6642. https://doi.org/10.1109/ICCV51070.2023.00610

Chen YQ, Chen Q, Hu QH, et al., 2022. Date: Dual assignment for end-to-end fully convolutional object detection. https://arxiv.org/abs/2211.13859v1

Dai XY, Chen YP, Yang JW, et al., 2021. Dynamic detr: End-to-end object detection with dynamic attention. Proc IEEE/CVF Int Conf on Computer Vision, p.2988-2997. https://doi.org/10.1109/ICCV48922.2021.00298

Deng J, Dong W, Socher R, et al., 2009. Imagenet: A large-scale hierarchical image database. Proc IEEE Conf on Computer Vision and Pattern Recognition, p.248-255. https://doi.org/10.1109/CVPR.2009.5206848

Everingham M, Van Gool L, Williams CK, et al., 2010. The pascal visual object classes (voc) challenge. *Int J Comput Vis*, 88(2):303-338. https://doi.org/10.1007/s11263-009-0275-4

Girshick R, 2015. Fast r-cnn. Proc IEEE Int Conf on Computer Vision, p.1440-1448. https://doi.org/10.1109/ICCV.2015.169

He KM, Zhang XY, Ren SQ, et al., 2016. Deep residual learning for image recognition. Proc IEEE Conf on Computer Vision and Pattern Recognition, p.770-778. https://doi.org/10.1109/CVPR.2016.90

Hou XQ, Liu MQ, Zhang SL, et al., 2024. Relation detr: Exploring explicit position relation prior for object detection. Proc 18th European Conf on Computer Vision, p.89-105. https://doi.org/10.1007/978-3-031-72973-7_6

Jia D, Yuan YH, He HD, et al., 2023. Detrs with hybrid matching. Proc IEEE/CVF Conf on Computer Vision and Pattern Recognition, p.19702-19712. https://doi.org/10.1109/CVPR52729.2023.01887

Jocher G, Chaurasia A, Qiu J, 2023. Ultralytics yolov8, . https://github.com/ultralytics/ultralytics

Kim K, Lee HS, 2020. Probabilistic anchor assignment with iou prediction for object detection. Proc 16th European Conf on Computer Vision, p.355-371. https://doi.org/10.1007/978-3-030-58595-2_22

Law H, Deng J, 2018. Cornernet: Detecting objects as paired keypoints. Proc 15th European Conf on Computer Vision, p.765-781. https://doi.org/10.1007/978-3-030-01264-9_45

Li F, Zhang H, Liu SL, et al., 2022. Dn-detr: accelerate detr training by introducing query denoising. Proc IEEE/CVF Conf on Computer Vision and Pattern Recognition, p.13609-13617. https://doi.org/10.1109/CVPR52688.2022.01325

Li F, Zeng AL, Liu SL, et al., 2023a. Lite detr: An interleaved multi-scale encoder for efficient detr. Proc IEEE/CVF Conf on Computer Vision and Pattern Recognition, p.18558-18567. https://doi.org/10.1109/CVPR52729.2023.01780

Li S, Li MH, Li RH, et al., 2023b. One-to-few label assignment for end-to-end dense detection. Proc IEEE/CVF Conf on Computer Vision and Pattern Recognition, p.7350-7359. https://doi.org/10.1109/CVPR52729.2023.00710

Fig. 6 Comparison of qualitative examples between FCOS, One-to-few and our method. Yellow arrows denote the redundant bounding boxes or the missed objects. One-to-few output redundant bounding boxes when detecting some larger objects, such as the person, traffic lights and train in the first three columns of images, respectively. In scenes with dense and small objects, FCOS and One-to-few may miss targets, such as the people, skateboard and backpack in the fourth and the fifth columns of images. In comparison, QSE-FCOS performed better in these situations

Unedited

Lin TY, Maire M, Belongie S, et al., 2014. Microsoft coco: common objects in context. Proc 13th European Conf on Computer Vision, p.740-755. https://doi.org/10.1007/978-3-319-10602-1_48

Lin TY, Goyal P, Girshick R, et al., 2017. Focal loss for dense object detection. Proc IEEE Int Conf on Computer Vision, p.2999-3007. https://doi.org//10.1109/ICCV.2017.324

Liu SL, Li F, Zhang H, et al., 2022. Dab-detr: Dynamic anchor boxes are better queries for detr. Proc 10th Int Conf on Learning Representations.

Liu W, Anguelov D, Erhan D, et al., 2016. Ssd: Single shot multibox detector. Proc 14th European Conf on Computer Vision, p.21-37. https://doi.org/10.1007/978-3-319-46448-0_2

Pu SL, Zhao W, Chen WJ, et al., 2021. Unsupervised object detection with scene-adaptive concept learning. *Front Inf Technol Electron Eng*, 22(5):638-651. https://doi.org/10.1631/FITEE.2000567

Qin XF, Hu WK, Xiao C, et al., 2023. Attention-based efficient robot grasp detection network. *Front Inf Technol Electron Eng*, 24(10):1430-1444. https://doi.org/10.1631/FITEE.2200502

Redmon J, Divvala S, Girshick R, et al., 2016. You only look once: Unified, real-time object detection. Proc IEEE Conf on Computer Vision and Pattern Recognition, p.779-788. https://doi.org/10.1109/CVPR.2016.91

Ren SQ, He KM, Girshick R, et al., 2015. Faster r-cnn: Towards real-time object detection with region proposal networks. Proc 29th Int Conf on Neural Information Processing Systems, p.91-99.

Rezatofighi H, Tsoi N, Gwak J, et al., 2019. Generalized intersection over union: A metric and a loss for bounding box regression. Proc IEEE/CVF Conf on Computer Vision and Pattern Recognition, p.658-666. https://doi.org/10.1109/CVPR.2019.00075

Shao S, Zhao ZJ, Li BX, et al., 2018. Crowdhuman: A benchmark for detecting human in a crowd. https://arxiv.org/abs/1805.00123

Sun PZ, Jiang Y, Xie EZ, et al., 2021a. What makes for end-to-end object detection? Proc 38th Int Conf on Machine Learning, p.9934-9944.

Sun PZ, Zhang RF, Jiang Y, et al., 2021b. Sparse r-cnn: End-to-end object detection with learnable proposals. Proc IEEE/CVF Conf on Computer Vision and Pattern Recognition, p.14449-14458. https://doi.org/10.1109/CVPR46437.2021.01422

Tian Z, Shen CH, Chen H, et al., 2019. Fcos: Fully convolutional one-stage object detection. Proc IEEE/CVF Int Conf on Computer Vision, p.9626-9635. https://doi.org/10.1109/ICCV.2019.00972

Wang A, Chen H, Liu LH, et al., 2024. Yolov10: Real-time end-to-end object detection. Proc 38th Annual Conf on Neural Information Processing Systems.

Wang CY, Bochkovskiy A, Liao HYM, 2023. Yolov7: Trainable bag-of-freebies sets new state-of-the-art for real-time object detectors. Proc IEEE/CVF Conf on Computer Vision and Pattern Recognition, p.7464-7475. https://doi.org/10.1109/CVPR52729.2023.00721

Wang JF, Song L, Li ZM, et al., 2021. End-to-end object detection with fully convolutional network. Proc IEEE/CVF Conf on Computer Vision and Pattern Recognition, p.15844-15853. https://doi.org/10.1109/CVPR46437.2021.01559

Wang YN, Zhang XY, Yang T, et al., 2022. Anchor detr: Query design for transformer-based detector. Proc 36th AAAI Conf on Artificial Intelligence, 36(3):2567-2575. https://doi.org/10.1609/aaai.v36i3.20158

Yao ZY, Ai JB, Li BX, et al., 2021. Efficient detr: improving end-to-end object detector with dense prior. https://arxiv.org/abs/2104.01318

Ye MQ, Ke L, Li SY, et al., 2023. Cascade-detr: delving into high-quality universal object detection. Proc IEEE/CVF Int Conf on Computer Vision, p.6681-6691. https://doi.org/10.1109/ICCV51070.2023.00617

Zhang H, Li F, Liu SL, et al., 2023. Dino: Detr with improved denoising anchor boxes for end-to-end object detection. Proc 11th Int Conf on Learning Representations.

Zhang SF, Chi C, Yao YQ, et al., 2020. Bridging the gap between anchor-based and anchor-free detection via adaptive training sample selection. Proc IEEE/CVF Conf on Computer Vision and Pattern Recognition, p.9756-9765. https://doi.org/10.1109/CVPR42600.2020.00978

Zhang SL, Wang XJ, Wang JQ, et al., 2023. Dense distinct query for end-to-end object detection. Proc IEEE/CVF Conf on Computer Vision and Pattern Recognition, p.7329-7338. https://doi.org/10.1109/CVPR52729.2023.00708

Zhou XY, Wang DQ, Krähenbühl P, 2019. Objects as points. https://arxiv.org/abs/1904.07850

Zhu XZ, Su WJ, Lu LW, et al., 2021. Deformable detr: Deformable transformers for end-to-end object detection. Proc 9th Int Conf on Learning Representations.

Zong ZF, Song GL, Liu Y, 2023. Detrs with collaborative hybrid assignments training. Proc IEEE/CVF Int Conf Computer Vision, p.6725-6735. https://doi.org/10.1109/ICCV51070.2023.00621

Unedited