
Dynamic Object Queries for Transformer-based Incremental Object Detection

Jichuan Zhang^{1*}, Wei Li^{1*}, Shuang Cheng², Ya-Li Li^{1†}, Shengjin Wang¹

¹Department of Electronic Engineering, Tsinghua University

²Institute of Computing Technology, Chinese Academy of Sciences, Beijing, China

University of Chinese Academy of Science

{zhangjc22, 1w22}@mails.tsinghua.edu.cn

chengshuang22@mails.ucas.ac.cn

{liyali13, wgsgj}@tsinghua.edu.cn

Abstract

Incremental object detection (IOD) aims to sequentially learn new classes, while maintaining the capability to locate and identify old ones. As the training data arrives with annotations only with new classes, IOD suffers from catastrophic forgetting. Prior methodologies mainly tackle the forgetting issue through knowledge distillation and exemplar replay, ignoring the conflict between limited model capacity and increasing knowledge. In this paper, we explore *dynamic object queries* for incremental object detection built on Transformer architecture. We propose the **D**ynamic object **Q**uery-based **D**Etection **T**ransformer (DyQ-DETR), which incrementally expands the model representation ability to achieve stability-plasticity tradeoff. First, a new set of learnable object queries are fed into the decoder to represent new classes. These new object queries are aggregated with those from previous phases to adapt both old and new knowledge well. Second, we propose the isolated bipartite matching for object queries in different phases, based on disentangled self-attention. The interaction among the object queries at different phases is eliminated to reduce inter-class confusion. Thanks to the separate supervision and computation over object queries, we further present the risk-balanced partial calibration for effective exemplar replay. Extensive experiments demonstrate that DyQ-DETR significantly surpasses the state-of-the-art methods, with limited parameter overhead. Code will be made publicly available.

1 Introduction

Humans inherently possess the ability to incrementally learn novel concepts without forgetting previous ones, capable of acquiring and accumulating knowledge from past experiences. Traditional object detection models [10, 64, 24, 46, 65] rely on supervised learning with fixed data, where all classes are predefined and known beforehand. However, real-world data continuously evolve over time, leading to non-stationary distributions. Due to the *stability-elasticity dilemma*, fine-tuning models directly on new class data leads to *catastrophic forgetting* [35, 36], whereas joint training is expensive in both computation and storage. Therefore, incremental object detection (IOD) has attracted increasing attention in both research and practical applications.

Recent advances for IOD adopt knowledge distillation and exemplar replay to address forgetting. Knowledge distillation-based methods [48, 21, 40, 9, 18, 39] typically involve distillation on the non-background predictions of the old model to circumvent the imbalance issue caused by an excessive

*These authors contributed equally to this work.

†Corresponding authors.

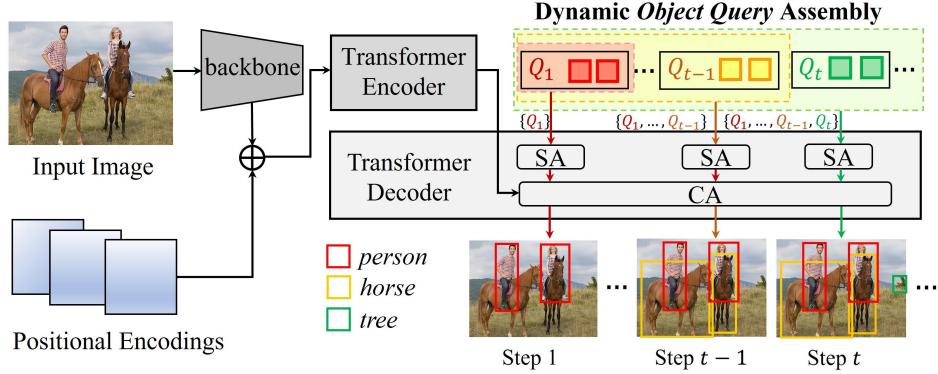


Figure 1: **Illustration of DyQ-DETR.** Built on Detection Transformer, a new set of queries is assigned for the newly-arriving classes at each step and different groups of queries are responsible for detecting specific classes annotated in corresponding steps. Q_t denotes the query group in step t . SA and CA refer to the self-attention and cross-attention modules, respectively.

number of background predictions. Exemplar replay-based methods [45, 16, 31, 32], on the other hand, operate by preserving a small subset of samples from past training data (exemplars), replaying them in subsequent phases to mitigate the forgetting of old data. Despite the progress, there still exist limitations. First, due to the fixed model capacity, severe conflicts between preserving knowledge of old classes and incrementally learning new ones exist. This not only impedes the accommodation of new class knowledge but also leads to the overwriting and forgetting of old class knowledge. Second, exemplars with incomplete labels are inadequately equipped for replay. Even if pseudo labeling is an intuitive way, the low-quality supervision inherited from old knowledge will hamper the adaption.

To address the above concerns, we propose **Dynamic object Query Assembly based DEtection TTransformer**, abbreviated as **DyQ-DETR**, for incremental object detection. Inspired by dynamic networks for incremental classification [14, 25, 51, 54, 56, 63, 8], we motivate to investigate the dynamic architecture for expanding model capacity in IOD. Specifically, the *object queries* serve as the class-specific representative essence for memorization. The object queries from old classes can be assembled with learnable new queries for dynamically expandable representations with limited memory overhead and acceptable time overhead. As demonstrated in Fig.1, we first obtain the visual features of the whole image with a CNN attached Transformer encoder. Then the set of object queries from previous steps are assembled with learnable ones corresponding to new classes in each incremental step. Moreover, we disentangle the self-attention and isolate the bipartite matching to remove the interaction between object queries from different steps. Besides, we propose the risk-balanced partial calibration to tackle incomplete annotations for effective exemplar replay.

Generally, we circumvent the *catastrophic forgetting* in IOD from the perspective of dynamic networks, with the inspiration of incrementally adding class tokens has been proven effective in incremental learning [63, 8, 47, 53]. Notably, for object detection, multiple objects of a single class may co-exist in one image (sometimes even many). We adopt a many-to-many rather than one-to-one matching of dynamic queries, for which sparse object queries implicitly associate with the content and reference positional information of one or multiple seen old classes. We propose to memorize the set of object queries for sequentially arrived classes of data. An isolated group of queries is responsible for detecting the objects from a set of classes arriving at the same time step. By incrementally aggregating the category-wise object queries from previous steps and learning new class embedding, we decouple the representations for old and new class knowledge with lightweight query embeddings, simultaneously maintaining the stability and plasticity. As for exemplar replay, we propose to reserve images with moderate matching losses as exemplars, further perform partial calibration on the outputs of corresponding queries with only the annotated classes. Such risk-balanced partial calibration avoids over focusing on classes from any particular stage and eliminates the reliance on low-quality pseudo labels. Through dynamic object queries and risk-balanced partial calibration, our proposed DyQ-DETR significantly alleviates the forgetting in IOD. The main contributions are three-fold:

- We propose a novel approach to integrate dynamic object queries into Detection Transformer for incremental learning. By dynamically incorporating object queries into DETR, our approach

provides a simple yet effective way to expand the model capability compatible with new knowledge adaption and old knowledge preserving.

- We propose the disentangled self-attention for dynamic object queries. For dynamic knowledge expansion, the isolated bipartite matching over the object queries from different phases further decouples the representation learning of old classes and new ones.
- A risk-balanced selection mechanism is proposed to explore informative and reliable exemplars. The partial calibration is further associated to tackle the incomplete annotations for incremental detection with exemplar replay.

Extensive experiments on public benchmarks demonstrate the superiority of our proposed DyQ-DETR, outperforming the state-of-the-art methods by a large margin. It achieves the average **4.3% AP** improvement in non-exemplar scenarios and **2.9% AP** improvement with exemplar replay.

2 Related Work

Incremental Learning (IL). Prevailing IL methods can be broadly divided as regularization-based, distillation-based and structure-based ones. Regularization-based methods[1, 19, 38, 59] estimate parameter importance and penalize updating of crucial parameters to maintain previous knowledge. Distillation-based methods build the mapping between the old and new model by matching logits[26, 45], feature maps[7], or other information[15, 42, 49, 50, 51], which leverage the knowledge transfer to prevent forgetting. Structure-based methods[14, 25, 51, 54, 56, 63] dynamically expand the representative network, *e.g.*, backbone, prompt, to fit the evolving data stream.

Incremental Object Detection (IOD). As the typical extension of incremental learning, IOD involves multiple objects belonging to the old and new classes appearing simultaneously. This co-occurrence makes knowledge distillation an inherently effective strategy for IOD, since it allows for the utilization of old class objects from new training samples to minimize the discrepancies in responses between the previous and current updating model. As a pioneering work, ILOD[48] distills the responses for old classes to counteract catastrophic forgetting on Fast R-CNN[10]. The idea of knowledge distillation is then extended to other detection frameworks, such as CenterNet[64] (SID[40]), RetinaNet[28] (RIOD[21]), GFLV1[24] (ERD[9]), Faster R-CNN[46] (CIFRCN[12]), Faster ILOD[39], DMC[61], BNC[6], IOD-ML[17] and Deformable DETR[65] (CL-DETR[32]). Built on Deformable DETR instead of conventional detectors such as Faster R-CNN, DyQ-DETR can efficiently expand queries rather than inefficiently enlarging the backbone or specific convolutional layers. Note that DyQ-DETR also uses knowledge distillation techniques. As for exemplar replay,[16] proposes maintaining an exemplar set and fine-tuning the model on the exemplars after each incremental step.[31] proposes an adaptive sampling strategy to achieve more efficient exemplar selection and [32] proposes distribution-preserving calibration, which selects exemplars to match the training distribution. They usually finetune directly using the exemplar set with incomplete annotations and overlook the amount of information and reliability of the annotated objects.

Transformer-based Object Detection. The infusive work DETR (DEtection TRansfomer)[3] formulates object detection as a set prediction problem, with an elegant transformer-based architecture. It captures global context and reasons object relations with attention mechanism. With a small set of learnable object queries and Hungarian bipartite matching[20], it eliminates the need for the complex non-maximum suppression and many other hand-designed components in object detection while demonstrating good performance. Deformable DETR[65] introduces sparse attention on multi-level feature maps, thereby accelerating the convergence of DETR and improving the performance, particularly for small objects. There also exists many other DETR variants[4, 37, 22, 30] designed to accelerate convergence speed and enhance detection performance. Without loss of generality, we build our method on the commonly used Deformable DETR.

3 Methodology

Preliminaries In the paradigm of IOD, object detection is performed in multiple steps from sequentially arrived training data to recognize and localize objects of all seen classes in test images. Let D be a dataset with samples (x, y) , where x is the image, y is the set of object class labels and the associated bounding boxes. Suppose there are T steps. At time step t , the incoming dataset is denoted as $D_t = (x_t, y_t)$ and the objects belong to seen classes in C_t . Specially, the images in D_t are only

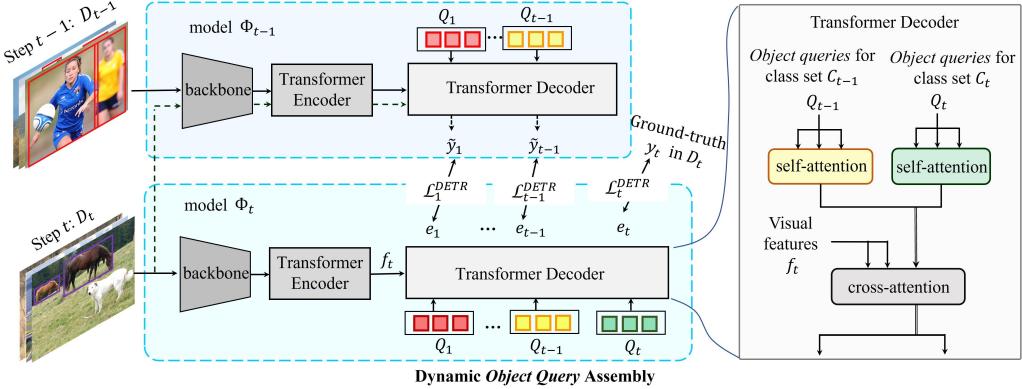


Figure 2: The overview of our proposed DyQ-DETR. The dynamic object queries serve as the input for Transformer decoder in incremental learning. At each incremental time step t , for an image $x \in D_t$, the training loss is independently computed. The total loss is the weighted sum of the knowledge distillation loss \mathcal{L}_i^{DETR} ($1 \leq i < t$) caused by pseudo labels and the standard DETR loss \mathcal{L}_t^{DETR} caused by ground-truth labels.

annotated for objects in C_t . Moreover, the class sets at different time steps are mutually exclusive from each other, *i.e.*, $C_t \cap C_\tau = \emptyset$, $t \neq \tau$. Since D_t only contains the annotations with classes in C_t , each class can be learned only once in a specific step. Due to the missing annotations of old classes $\{C_1, C_2, \dots, C_{t-1}\}$ in current data D_t , detection models are prone to forget the knowledge of old classes and bias towards new ones. To overcome the catastrophic forgetting, we introduce the dynamic object queries into Detection Transformer and propose DyQ-DETR for IOD.

We construct the DyQ-DETR based on Detection Transformer for incremental object detection. In particular, we elaborate the details of dynamic object queries with the structure design and training strategy in Sec. 3.2. The risk-balanced exemplar selection and partial calibration are in Sec. 3.3.

3.1 Transformer-based Incremental Object Detection

We build up the DyQ-DETR based on the DETR architecture [3, 65]. Other than the backbone, a DETR consists of an encoder, a Transformer decoder, and the predictor to generate object classes and locations. The encoder takes the images as inputs and outputs visual features. Then those visual features and learnable object queries are fed into Transformer decoder for prediction. Notably, we propose to aggregate the object queries from up-to-far learning steps to resist the forgetting of old class knowledge. Due to the absence of annotations from previous seen classes in incremental step t , knowledge distillation is applied for preserving the class-specific old knowledge. As in Fig.2, we select the non-background predictions of old classes by thresholding. Pseudo labels \tilde{y}_τ ($1 \leq \tau < t$) are generated by the last previous model. Instead of mixing the pseudo labels \tilde{y}_τ ($1 \leq \tau < t$) with the ground-truth labels y_t , we use \tilde{y}_τ and y_t separately to supervise the model training. The pseudo labels for old classes and the real ground-truths are used to guide the learning of object queries being Q_τ and Q_t , respectively. Besides, object queries from different groups share the weight parameters of the Transformer decoder. To allow for computational complexity to grow linearly rather than quadratically, the self-attention over different groups of object queries are eliminated.

As for IOD with exemplar replay, where a small number of exemplar images ϵ_t from dataset D_t in different time steps are stored, we introduce a risk-balanced selection mechanism. At time step t , we use the trained model Φ_t to score the annotated objects from images in D_t . The computed loss from the partial bipartite matching is considered as the risk score for exemplar selection. We choose the sample images with moderate risk score for the tradeoff between the annotation importance and quality. Specifically, to build the exemplar dataset $\epsilon_{1:t}$, we select the samples falling into the middle part after sorting, because they are informative and reliably annotated. Considering the image in $\epsilon_{1:t}$ is incompletely annotated for specific classes, we adapt the partial calibration. We leverage the incomplete annotations to calibrate the output for corresponding object queries in each group. Since $\epsilon_{1:t}$ is balanced, the partial calibration will prohibit the model being biased towards certain classes.

3.2 Dynamic Object Query Assembly

Existing DETR models employ a fixed set of object queries (*i.e.*, learnable embedding) as the inputs of Transformer decoder. These object queries are progressively optimized to map into object instances in images. Despite various designs, the learnable object queries are highly relevant with the specific classes. For IOD, the object queries are expected to be associated with objects belonging to sequentially arrived classes. Since new classes are considered as backgrounds in previous time steps, the preservation of old knowledge naturally contradicts the knowledge updating from new data learning, especially from the perspective of object queries. Moreover, the conflict between a fixed network and the continually emerged class-specific information severely undermines the performance of incremental learning, especially in non-exemplar scenarios.

To cope with incremental classes without extra modules in network architecture, we focus on tackling the forgetting issue with dynamic object queries. At time step t , for a set of new classes C_t , a new set of learnable object queries Q_t is added. The newly added object queries Q_t are aggregated with previous sets of queries $Q_\tau, 1 \leq \tau < t$. The assembly of object queries $\{Q_1, Q_2, \dots, Q_t\}$ serves as the input of Transformer encoder in step t . The visual embeddings e_t corresponding to the newly expanded queries Q_t are used to predict the objects of new classes C_t , and the old classes $C_\tau (1 \leq \tau < t)$ is detected with the embeddings e_τ of old queries Q_τ . By dynamically expanding object queries, the new and old classes are segregated by class embeddings, significantly alleviating the conflict between the old knowledge and continuously evolving new knowledge.

Based on the dynamic assembly of object queries, we further investigate the decoder design to restrict the computational burden. In standard DETR, object queries interact visual features with cross-attention for refinement. Besides, those object queries interact with each other by self-attention. Through self-attention, duplicated detections can be removed, but the computational complexity increases quadratically with the number of object queries. Considering that the object instances from different class sets rarely overlap, we disentangle the self-attention in Transformer decoder. The self-attention is computed among separate set of object queries, as:

$$a_{i,j} = \begin{cases} \text{Softmax}(Q_{m,i} \cdot Q_{n,j}^T / \sqrt{d}) & m = n \\ 0 & m \neq n \end{cases}, \quad (1)$$

where $a_{i,j}$ denotes the attention weight between two queries $Q_{m,i}$ and $Q_{n,j}$ from the query set Q_m and Q_n , respectively. By eliminating the attention interaction between queries of different groups as shown in Eq.(1), we achieve a linear growth in computational complexity almost for free. Upon the addition of new queries, the capability of old queries to detect old classes is fully preserved. We perform the decoder forward passes as many as the time steps, obtaining the embedding vectors from different sets of queries as:

$$e_\tau = \text{decoder}(f_t, Q_\tau), 1 \leq \tau \leq t, \quad (2)$$

where f_t denotes the visual feature extracted from image x by the CNN and Transformer encoder. Each decoder forward pass is executed with a different set of queries Q_τ , resulting in different task-specific embeddings e_τ to obtain detection predictions of the corresponding classes C_τ .

We further adapt the knowledge distillation for incremental detector training. The foreground predictions with the pseudo labels generated from old model are kept for distillation and used for supervision. As in [32], the foreground predictions with high confidence from the old model are selected. A probability threshold θ_p (typically 0.4) is set over the prediction scores. An additional IoU threshold θ_{IoU} (typically 0.7) is used to restrict the predictions not too close to the ground-truth bounding boxes of new classes. It helps filter out incorrect predictions about new class objects that are misclassified as old classes. The highly-confident predictions after filtering are used as pseudo labels $\tilde{y}_\tau (1 \leq \tau < t)$, which contain two parts of annotations (*i.e.*, the predicted object labels and bounding boxes). Notably, instead of merging the pseudo labels with real grounding truths, we compute the separate bipartite matching losses with different set of object queries in an independent way. The loss for retaining old class knowledge $\mathcal{L}_\tau^{DETR} (1 \leq \tau < t)$, as well as the loss for learning new class knowledge \mathcal{L}_t^{DETR} can be formulated as in Eq.(3):

$$\mathcal{L}_\tau^{DETR} = \mathcal{L}^{DETR}(e_\tau, \tilde{y}_\tau), 1 \leq \tau < t; \mathcal{L}_t^{DETR} = \mathcal{L}^{DETR}(e_t, y_t), \quad (3)$$

Note that the specific embeddings $e_\tau, 1 \leq \tau \leq t$ are only supervised with the corresponding pseudo labels or real annotations, leading to decoupled representations. The total loss is the weighted sum as:

$$\mathcal{L}_{total} = \sum_{\tau=1}^t w_\tau \mathcal{L}_\tau^{DETR}. \quad (4)$$

To tackle the varying number of classes in each step and prevent the model from being biased towards class sets with fewer classes, we set the weight w_τ of the class set C_τ to $|C_\tau|/|C_{1:t}|$.

3.3 Risk-balanced Partial Calibration

For exemplar replay based IOD paradigm, an exemplar memory is formed to store a small number of samples for subsequent incremental learning. At time step t , the exemplar set ϵ_t is a subset of the entire dataset D_t . Let $\epsilon_{1:t} = \epsilon_1 \cup \dots \cup \epsilon_t$. The exemplars in ϵ_t are used to represent the samples with objects in C_t . We intend to choose the class annotated images which are *substantial* for detector training. An intuitive way is to directly compute the loss between the model output and the real label y_t for sample selection. However, since the image $x \in D_t$ only contains annotations for the classes C_t of interest, this will result in the loss being dominated by the absence of old classes, posing challenges to foreground-background balancing.

Benefiting from the internal decoupling of dynamic object queries, we are able to exclusively detect new classes C_t using the corresponding queries Q_t . Considering that the image is only annotated for the specific classes, the partial loss is more reliable. For computing the partial loss, the old classes are considered as backgrounds by Q_t , which is compatible with the incomplete real labels y_t . After each incremental step of training, the partial loss from Eq.(5) is considered as the risk score to guide the subsequent selection of exemplars.

$$\Upsilon \leftarrow \mathcal{L}_{partial} = \mathcal{L}_t^{DETR}(e_t, y_t). \quad (5)$$

The risk score can measure the quality of incomplete class labels and bounding boxes. Additionally, it also takes the number of annotated objects like in [32] into account. As in Fig. 3, images with low risk, which constitute a high proportion, provide limited information for optimization, while those with high risk are likely to be outliers with incorrect annotations. Based on the risk estimation, we construct the exemplar set ϵ_t by sorting and selecting the middle part of risk-balanced samples in D_t . The exemplar set ϵ_t is merged with ϵ_{t-1} to form a set $\epsilon_{1:t}$ for partial calibration of the model.

In incremental step t , the exemplar sets $\epsilon_{1:t}$ are used to finetune the model after training with D_t . Previous IOD methods typically finetune the model directly with dataset $\epsilon_{1:t}$, without addressing the issue of missing labels. That is, an image x in the balanced exemplar set $\epsilon_{1:t}$ is only annotated for a specific class subset from $\{C_1, \dots, C_{t-1}, C_t\}$, with the absence of annotations for other classes. The confusing, even contradictive supervision hinders the process of prediction calibration. An intuitive way is to use pseudo labels, yet the quality of pseudo labels is hard to guarantee. Thanks to the dynamic object queries and associative disentangled computation, we propose to perform partial calibration that relies only on the incomplete real labels. Specifically, we calculate the partial loss between the outputs of the corresponding queries and the ground-truth annotations in $\epsilon_{1:t}$. This type of partial calibration further mitigates the forgetting.

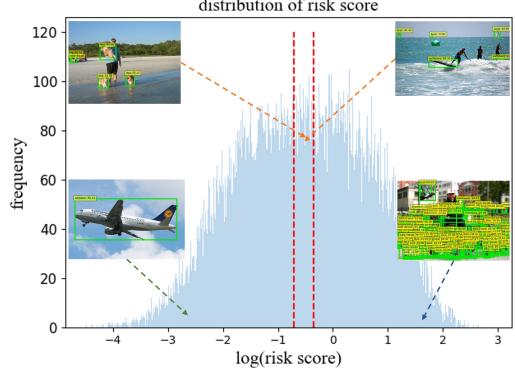


Figure 3: Illustration of risk-balanced exemplar selection. We choose the middle part with moderate risk score to serve as the exemplars.

<i>1)</i>	Method	AP	AP ₅₀	AP ₇₅	AP _S	AP _M	AP _L	<i>2)</i>	Method	AP	AP ₅₀	AP ₇₅	AP _S	AP _M	AP _L
40+40	LwF[26]†	17.2	25.4	18.6	7.9	18.4	24.3	40+40	Upper Bound	41.0	59.8	44.2	25.0	43.5	54.2
	CL-DETR[32]†	39.2	56.1	42.6	21.0	42.8	52.6		LwF[26]†	23.9	41.5	25.0	12.0	26.4	33.0
	DyQ-DETR (ours) †	41.4	59.7	44.9	24.1	45.2	54.3		CL-DETR[32]†	36.2	52.6	39.5	18.7	39.5	49.4
	RILOD[21]	29.9	45.0	32.0	15.8	33.0	40.5		DyQ-DETR (ours) †	39.1	57.1	42.5	21.3	42.7	51.8
	SID[40]	34.0	51.4	36.3	18.4	38.4	44.9		iCaRL[45]	33.4	52.0	36.0	18.0	36.4	45.5
	ERD[9]	36.9	54.5	39.6	21.3	40.4	47.5		ERD[9]	36.0	55.2	38.7	19.5	38.7	49.0
70+10	CL-DETR[32]	42.0	60.1	45.9	24.0	45.3	55.6	70+10	CL-DETR[32]	37.5	55.1	40.3	20.9	40.8	50.7
	DyQ-DETR (ours)	42.4	60.5	45.9	23.9	46.3	56.7		DyQ-DETR (ours)	39.7	57.5	43.0	21.6	42.9	53.8
	LwF[26]†	7.1	12.4	7.0	4.8	9.5	10.0		Upper Bound	43.3	61.8	47.0	25.3	46.1	57.9
	CL-DETR[32]†	35.8	53.5	39.5	19.4	41.5	46.1		LwF[26]†	24.5	36.6	26.7	12.4	28.2	35.2
	DyQ-DETR (ours) †	39.5	56.4	43.1	22.5	43.1	53.0		CL-DETR[32] †	34.0	48.0	37.2	15.5	37.7	49.7
	RILOD[21]	24.5	37.9	25.7	14.2	27.4	33.5		DyQ-DETR (ours) †	39.6	57.6	43.5	23.4	43.3	51.8
70+10	SID[40]	32.8	49.0	35.0	17.1	36.9	44.5	70+10	iCaRL[45]	35.9	52.5	39.2	19.1	39.4	48.6
	ERD[9]	34.9	51.9	37.4	18.7	38.8	45.5		ERD[9]	36.9	55.7	40.1	21.4	39.6	48.7
	CL-DETR[32]	40.4	58.0	43.9	23.8	43.6	53.5		CL-DETR[32]	40.1	57.8	43.7	23.2	43.2	52.1
	DyQ-DETR (ours)	42.4	60.4	46.3	24.5	45.7	57.5		DyQ-DETR (ours)	41.9	60.1	45.8	24.1	45.3	55.8

Table 1: IOD results (%) on COCO 2017 under the 40+40 and 70+10 scenarios. † indicates that the results are obtained without exemplar replay. “Upper bound” refers to the result of joint training with all previous data available at each step.

4 Experiment

4.1 Experimental Setup

Dataset and evaluation metrics. Following[32], we conduct experiments on the widely-used COCO 2017 dataset[29], which consists of images from 80 object categories in natural scenes. The standard COCO metrics as AP, AP₅₀, AP₇₅, AP_S, AP_M, AP_L are used for performance evaluation.

Protocols. We evaluate DyQ-DETR with two protocols: *1)* traditional protocol [48] (Tab.1-left) and *2)* revised protocol proposed by[32] (Tab.1-right). Protocol 2) avoids observing the same images at different stages. Therefore, we adopt protocol 2) for all subsequent experiments, and the details of protocol *1)* can be found in the appendix. For protocol 2), we adopt both two-phase and multiple-phase settings, which can be formulated in the form of $c_1 + c_2 + \dots + c_T$, where c_t represents the number of new classes in incremental step t ($c_t = |C_t|$) and the sum of c_t is denoted as c ($c = |C_{1:T}|$). At time step t , we observe a fraction $\frac{c_t}{c}$ of the training samples with c_t new categories annotated. We test settings $c_1 + c_2 + \dots + c_T = 40 + 40, 70 + 10, 40 + 20 \times 2$, and $40 + 10 \times 4$. Following [32], we also set the total memory budget for the exemplars to be 10% of the total dataset size.

Implementation details. Following [32], we build DyQ-DETR on top of Deformable DETR [65] without iterative bounding box refinement and the two-stage variant. The backbone is ResNet-50[13] pretrained on ImageNet[5] and the training configurations for the initial stage are consistent with[32] to maintain uniform performance in the initial phase. We denote the initial number of queries for the detector as N (typically 300). For both two-phase and multi-phase settings, we dynamically expand by N queries at each incremental step t and the initial parameters of Q_t are inherited from Q_{t-1} . At step t , old queries $Q_{1:t-1}$ are frozen during the incremental training and unfrozen in subsequent exemplar replay. We train the model for 50 epochs, and for an additional 20 epochs during fine-tuning. All the experiments are performed on 8 NVIDIA GeForce RTX 3090, with a batch size of 8.

4.2 Quantitative Results

Two-phase setting. We compare our DyQ-DETR with LwF[26], iCaRL[45], RILOD[21], SID[40], ERD[9], and the previous SOTA method CL-DETR[32]. For each setting, we provide the performance of different methods with/without Exemplar Replay (ER). The metrics by joint training are also presented as the upper bound for reference. Tab.1 shows that, in the two-phase settings, our proposed DyQ-DETR consistently outperforms the aforementioned methods under different protocols with significant margins. For protocol 2), with exemplar replay, the DyQ-DETR achieves the AP of 39.7% and 41.9% under 40+40 and 70+10 settings. It surpasses CL-DETR by 2.2% AP and 1.8% AP under the 40+40 and 70+10 settings, respectively. Compared with the upper bound, the DyQ-DETR obtains an average performance gap of 1.4%, which is much smaller than the 3.4% gap of CL-DETR.

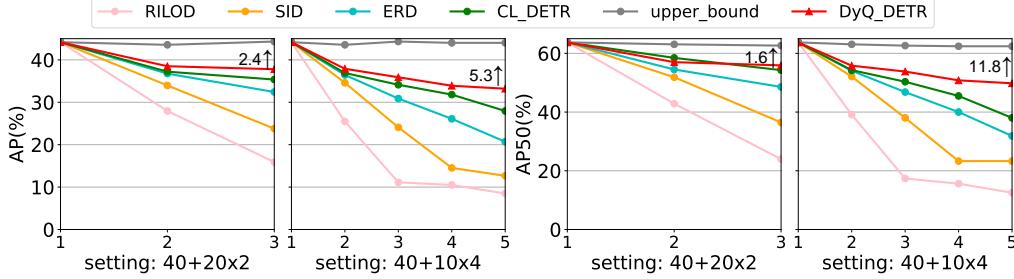


Figure 4: IOD results (AP/AP_{50} , %) in the multi-phase $40+20\times 2$ and $40+10\times 4$ settings. The results of all other works are from [32].

To evaluate the model’s ability to preserve old knowledge and learn new knowledge, we conduct a comparison with CL-DETR in the $40+40$ setting, where both capabilities are equally important. CL-DETR achieves an old class AP of 39.7% and a new class AP of 36.3%, while our method achieves much better results with 41.3% AP for old classes and 38.6% AP for new classes, respectively. This validates that our proposed method achieves better stability and plasticity, thus addressing the catastrophic forgetting effectively.

Additionally, Tab.1 include a performance comparison without exemplar replay (*w/o* ER) to show the effectiveness of dynamic object queries. In non-exemplar scenarios, DyQ-DETR demonstrates a more significant advantage, outperforming CL-DETR by 2.9% AP and 5.6% AP in the $40+40$ and $70+10$ settings, respectively. It is noteworthy that the performance of our method without ER is comparable or even exceeds that of the existing method with ER. For example, the performance of our DyQ-DETR *w/o* ER is 1.6% AP higher than that of CL-DETR *w/* ER in the $40+40$ setting.

Multiple-phase setting. We conduct experiments in the more challenging $40+20\times 2$ and $40+10\times 4$ settings. The changing AP and AP_{50} along with time steps are presented in Fig.4. Our DyQ-DETR consistently outperforms other IOD methods. Moreover, in both settings, the AP improvements of DyQ-DETR become more pronounced with the increase of incremental steps.

Scalability. As shown in Fig.5-left, by expanding queries rather than the network structure, the additional parameter overhead of our method is almost negligible. As illustrated in Fig.5-right, the computational overhead of our method grows linearly because of the removed inter-group query interaction. Since the computational load of the decoder (excluding the portion shared by different query groups) constitutes a small portion ($\sim 6\%$) of the entire model’s computation, our computational complexity increases at a slow linear rate. Specifically, with 20 stages and an increment of 100 queries per stage (note that the standard Deformable DETR has 300 queries), DyQ-DETR only increases the parameters and GFLOPs by 2% and 39% respectively compared to the standard Deformable DETR, confirming its scalability.

4.3 Ablation Study

Effect of dynamic object queries. Tab.3 illustrates the ablation study over dynamic object queries in the $70+10$ setting. Compared to the baseline, the naive way of expanding queries (+Nat Query) increases the AP to 34.5% and AP_{50} to 52.7%, by 1.1% and 4.2%, respectively. This can be attributed to model capacity. By equipping the dynamic object queries (+Dy Query) with isolated matching on independent query set, we further obtain 3.6% AP and 3.5% AP_{50} improvements. It validates that dynamic object queries is effective to retain old knowledge during the learning of new knowledge. Furthermore, freezing the task-specific old queries (+DyFro Query) during the incremental training leads to slightly more improvement of 0.4% AP .

	AP	AP_{50}	AP_{75}	AP_S	AP_M	AP_L	DS	RS	DC	PC	AP	AP_{50}	AP_{75}	AP_S	AP_M	AP_L
Baseline	34.5	48.5	37.7	16.7	37.4	49.8	✓	✗	✓	✗	40.5	58.7	44.5	24.7	44.0	53.1
+Nat Query	35.6	52.7	38.5	20.3	37.8	46.4	✗	✓	✓	✗	41.0	59.1	44.7	23.0	44.3	55.7
+Dy Query	39.2	56.2	43.1	21.6	42.8	52.1	✓	✗	✗	✓	41.4	59.6	45.0	23.8	44.4	55.5
+DyFro Query	39.6	57.6	43.5	23.4	43.3	51.8	✗	✓	✗	✓	41.9	60.1	45.8	24.1	45.3	55.8

Table 2: Ablation on dynamic object queries. Table 3: Effects of components in exemplar replay.

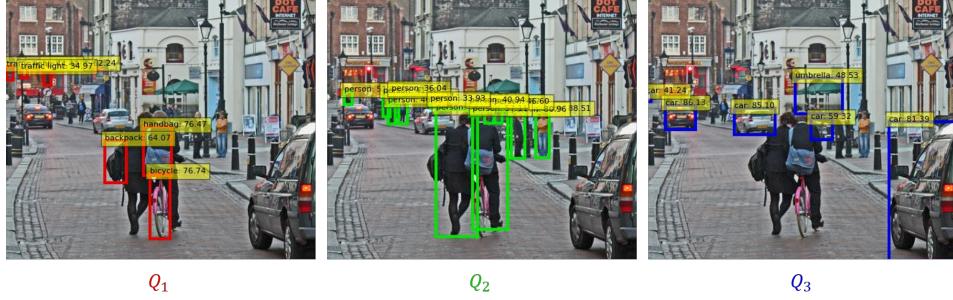


Figure 6: Visualization of the behavior of decoupled queries after training in the 40+20x2 setting. Different query groups are responsible for detecting different sets of classes. More visualization results can be found in the appendix.

Components for exemplar replay. We introduce the risk-balanced exemplar selection (RS) and partial calibration (PC) for exemplar replay. The ablative results of the two components in the 70+10 setting are provided in Tab.3. We compare the RS and PC with the distribution-preserving selection (DS) and direct calibration (DC) proposed in CL-DETR [32], respectively. By separately applying RS and PC, we increase the AP by 0.5% and 0.9%, respectively. Applying the whole risk-balanced exemplar selection and partial calibration obtains 41.9% AP , exceeding the baseline by 1.5%.

Setting	DSA	AP	AP_S	AP_M	AP_L
40+40	w/o	39.8	23.5	42.8	53.4
	w/	39.7	21.6	42.9	53.8
70+10	w/o	41.7	24.3	45.2	55.7
	w/	41.9	24.1	45.3	55.8

Table 4: Ablation results (%) for disentangled self-attention (DSA) in the 40+40 and 70+10 settings.

Query No.	AP	AP_S	AP_M	AP_L
50	41.5	23.6	45.0	55.7
100	42.0	24.0	45.3	56.6
200	42.2	24.3	45.4	56.6
300	41.9	24.1	45.3	55.8

Table 5: Ablation results (%) for the number of expanded queries in the 70+10 setting.

Effect of disentangled self-attention. The comparison in Tab.4 under both the 40+40 and 70+10 settings shows that removing the self-attention interaction between different query groups has almost no impact on performance. This is reasonable because the class sets detected by different query groups do not overlap, thus eliminating the need for self-attention interaction to remove duplicate predictions. Combined with Fig.5-right, by disentangling the self-attention computation, the computational complexity can be reduced from quadratic to linear growth without performance drop.

Effect on the number of expanded object queries. Tab.5 presents the APs with changing number of object queries. It indicates that the expanded queries number has a trivial impact. By expanding fewer queries, we can decrease complexity with negligible harm to performance.

Visualization of decoupled queries. In Fig.6, we visualize the decoupling behavior of dynamic queries on the test set, based on the 40+20x2 setting. The classes "bicycle", "person", and "car" appear in phases 1, 2, and 3, respectively. They are detected by queries in Q_1 , Q_2 and Q_3 accordingly, in a decoupling manner. It can be observed that Q_1 detects the "bicycle" class accurately, while Q_2 and Q_3 consider it as background. Once a query group Q_t learns to detect the class set C_t at step t , its class-specific knowledge remains unchanged thereafter, requiring only the maintenance of old knowledge, which significantly improves the performance of IOD.

5 Conclusion

In this paper, we propose DyQ-DETR for incremental object detection. Distinct from the mainstream methods focus on the distillation mechanism, we address the catastrophic forgetting by taking inspiration from dynamic networks for model capability expansion. In particular, we propose the dynamic object queries with incremental assembly of new object queries, disentangled self-attention computation and isolated bipartite matching over object queries from different time. The DyQ-DETR can alleviate the conflict between outdated background knowledge and continually emerged classes, thereby achieving stability-plasticity tradeoff. Benefiting from the isolated supervision of dynamic object queries, we further propose risk-balanced partial calibration for effective exemplar replay, with the idea to select exemplars based on risk and partially finetunes the model without relying on low-quality pseudo labels. Extensive experiments demonstrate that our proposed DyQ-DETR surpasses existing IOD methods by a large margin, with quite limited memory overhead. Besides of dynamic query, we hope that more various ways of model expansion can be explored in IOD.

References

- [1] R. Aljundi, F. Babiloni, M. Elhoseiny, M. Rohrbach, and T. Tuytelaars, *Memory aware synapses: Learning what (not) to forget*, 2018. arXiv: 1711.09601 [cs.CV].
- [2] K. Buettner and A. Kovashka, *Investigating the role of attribute context in vision-language models for object recognition and detection*, 2023. arXiv: 2303.10093 [cs.CV].
- [3] N. Carion, F. Massa, G. Synnaeve, N. Usunier, A. Kirillov, and S. Zagoruyko, “End-to-end object detection with transformers,” in *European conference on computer vision*, Springer, 2020, pp. 213–229.
- [4] Z. Dai, B. Cai, Y. Lin, and J. Chen, “Up-detr: Unsupervised pre-training for object detection with transformers,” in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2021, pp. 1601–1610.
- [5] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, “Imagenet: A large-scale hierarchical image database,” in *2009 IEEE conference on computer vision and pattern recognition*, Ieee, 2009, pp. 248–255.
- [6] N. Dong, Y. Zhang, M. Ding, and G. H. Lee, “Bridging non co-occurrence with unlabeled in-the-wild data for incremental object detection,” *Advances in Neural Information Processing Systems*, vol. 34, pp. 30492–30503, 2021.
- [7] A. Douillard, M. Cord, C. Ollion, T. Robert, and E. Valle, “Podnet: Pooled outputs distillation for small-tasks incremental learning,” in *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XX 16*, Springer, 2020, pp. 86–102.
- [8] A. Douillard, A. Ramé, G. Couairon, and M. Cord, “Dytox: Transformers for continual learning with dynamic token expansion,” in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022, pp. 9285–9295.
- [9] T. Feng, M. Wang, and H. Yuan, “Overcoming catastrophic forgetting in incremental object detection via elastic response distillation,” in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022, pp. 9427–9436.
- [10] R. Girshick, “Fast r-cnn,” in *Proceedings of the IEEE international conference on computer vision*, 2015, pp. 1440–1448.
- [11] X. Gu, T.-Y. Lin, W. Kuo, and Y. Cui, “Open-vocabulary object detection via vision and language knowledge distillation,” *arXiv preprint arXiv:2104.13921*, 2021.
- [12] Y. Hao, Y. Fu, Y.-G. Jiang, and Q. Tian, “An end-to-end architecture for class-incremental object detection with knowledge distillation,” in *2019 IEEE International Conference on Multimedia and Expo (ICME)*, IEEE, 2019, pp. 1–6.
- [13] K. He, X. Zhang, S. Ren, and J. Sun, “Deep residual learning for image recognition,” in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 770–778.
- [14] C.-Y. Hung, C.-H. Tu, C.-E. Wu, C.-H. Chen, Y.-M. Chan, and C.-S. Chen, “Compacting, picking and growing for unforgetting continual learning,” *Advances in Neural Information Processing Systems*, vol. 32, 2019.

- [15] K. Joseph, S. Khan, F. S. Khan, R. M. Anwer, and V. N. Balasubramanian, “Energy-based latent aligner for incremental learning,” in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022, pp. 7452–7461.
- [16] K. Joseph, S. Khan, F. S. Khan, and V. N. Balasubramanian, “Towards open world object detection,” in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2021, pp. 5830–5840.
- [17] K. Joseph, J. Rajasegaran, S. Khan, F. S. Khan, and V. N. Balasubramanian, “Incremental object detection via meta-learning,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 44, no. 12, pp. 9209–9216, 2021.
- [18] M. Kang *et al.*, “Alleviating catastrophic forgetting of incremental object detection via within-class and between-class knowledge distillation,” in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2023, pp. 18 894–18 904.
- [19] J. Kirkpatrick *et al.*, “Overcoming catastrophic forgetting in neural networks,” *Proceedings of the national academy of sciences*, vol. 114, no. 13, pp. 3521–3526, 2017.
- [20] H. W. Kuhn, “The hungarian method for the assignment problem,” *Naval research logistics quarterly*, vol. 2, no. 1-2, pp. 83–97, 1955.
- [21] D. Li, S. Tasci, S. Ghosh, J. Zhu, J. Zhang, and L. Heck, “Rilod: Near real-time incremental learning for object detection at the edge,” in *Proceedings of the 4th ACM/IEEE Symposium on Edge Computing*, 2019, pp. 113–126.
- [22] F. Li, H. Zhang, S. Liu, J. Guo, L. M. Ni, and L. Zhang, “Dn-detr: Accelerate detr training by introducing query denoising,” in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022, pp. 13 619–13 627.
- [23] L. H. Li *et al.*, “Grounded language-image pre-training,” in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022, pp. 10 965–10 975.
- [24] X. Li *et al.*, “Generalized focal loss: Learning qualified and distributed bounding boxes for dense object detection,” *Advances in Neural Information Processing Systems*, vol. 33, pp. 21 002–21 012, 2020.
- [25] X. Li, Y. Zhou, T. Wu, R. Socher, and C. Xiong, “Learn to grow: A continual structure learning framework for overcoming catastrophic forgetting,” in *International Conference on Machine Learning*, PMLR, 2019, pp. 3925–3934.
- [26] Z. Li and D. Hoiem, “Learning without forgetting,” *IEEE transactions on pattern analysis and machine intelligence*, vol. 40, no. 12, pp. 2935–2947, 2017.
- [27] C. Lin *et al.*, “Learning object-language alignments for open-vocabulary object detection,” *arXiv preprint arXiv:2211.14843*, 2022.
- [28] T.-Y. Lin, P. Goyal, R. Girshick, K. He, and P. Dollár, “Focal loss for dense object detection,” in *Proceedings of the IEEE international conference on computer vision*, 2017, pp. 2980–2988.
- [29] T.-Y. Lin *et al.*, “Microsoft coco: Common objects in context,” in *Computer Vision–ECCV 2014: 13th European Conference, Zurich, Switzerland, September 6–12, 2014, Proceedings, Part V 13*, Springer, 2014, pp. 740–755.
- [30] S. Liu *et al.*, “Dab-detr: Dynamic anchor boxes are better queries for detr,” *arXiv preprint arXiv:2201.12329*, 2022.
- [31] X. Liu, H. Yang, A. Ravichandran, R. Bhotika, and S. Soatto, “Multi-task incremental learning for object detection,” *arXiv preprint arXiv:2002.05347*, 2020.
- [32] Y. Liu, B. Schiele, A. Vedaldi, and C. Rupprecht, “Continual detection transformer for incremental object detection,” in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2023, pp. 23 799–23 808.
- [33] C. Ma, Y. Jiang, X. Wen, Z. Yuan, and X. Qi, *Codet: Co-occurrence guided region-word alignment for open-vocabulary object detection*, 2023. arXiv: 2310.16667 [cs.CV].
- [34] Z. Ma *et al.*, *Open-vocabulary one-stage detection with hierarchical visual-language knowledge distillation*, 2022. arXiv: 2203.10593 [cs.CV].
- [35] M. McCloskey and N. J. Cohen, “Catastrophic interference in connectionist networks: The sequential learning problem,” in *Psychology of learning and motivation*, vol. 24, Elsevier, 1989, pp. 109–165.
- [36] K. McRae and P. A. Hetherington, “Catastrophic interference is eliminated in pretrained networks,” in *Proceedings of the 15th Annual Conference of the Cognitive Science Society*, vol. 1, 1993, p. 2.

- [37] D. Meng *et al.*, “Conditional detr for fast training convergence,” in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2021, pp. 3651–3660.
- [38] I. Paik, S. Oh, T. Kwak, and I. Kim, “Overcoming catastrophic forgetting by neuron-level plasticity control,” in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 34, 2020, pp. 5339–5346.
- [39] C. Peng, K. Zhao, and B. C. Lovell, “Faster ilod: Incremental learning for object detectors based on faster rcnn,” *Pattern recognition letters*, vol. 140, pp. 109–115, 2020.
- [40] C. Peng, K. Zhao, S. Maksoud, M. Li, and B. C. Lovell, “Sid: Incremental learning for anchor-free object detection via selective and inter-related distillation,” *Computer vision and image understanding*, vol. 210, p. 103 229, 2021.
- [41] C. Pham, T. Vu, and K. Nguyen, *Lp-ovod: Open-vocabulary object detection by linear probing*, 2023. arXiv: 2310.17109 [cs.CV].
- [42] M. PourKeshavarzi, G. Zhao, and M. Sabokrou, “Looking back on learned experiences for class/task incremental learning,” in *International Conference on Learning Representations*, 2021.
- [43] A. Radford *et al.*, “Learning transferable visual models from natural language supervision,” in *International conference on machine learning*, PMLR, 2021, pp. 8748–8763.
- [44] H. Rasheed, M. Maaz, M. U. Khattak, S. Khan, and F. S. Khan, *Bridging the gap between object and image-level representations for open-vocabulary detection*, 2022. arXiv: 2207.03482 [cs.CV].
- [45] S.-A. Rebuffi, A. Kolesnikov, G. Sperl, and C. H. Lampert, “Icarl: Incremental classifier and representation learning,” in *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*, 2017, pp. 2001–2010.
- [46] S. Ren, K. He, R. Girshick, and J. Sun, “Faster r-cnn: Towards real-time object detection with region proposal networks,” *Advances in neural information processing systems*, vol. 28, 2015.
- [47] C. Shang, H. Li, F. Meng, Q. Wu, H. Qiu, and L. Wang, “Incrementer: Transformer for class-incremental semantic segmentation with knowledge distillation focusing on old class,” in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2023, pp. 7214–7224.
- [48] K. Shmelykov, C. Schmid, and K. Alahari, “Incremental learning of object detectors without catastrophic forgetting,” in *Proceedings of the IEEE international conference on computer vision*, 2017, pp. 3400–3409.
- [49] C. Simon, P. Koniusz, and M. Harandi, “On learning the geodesic path for incremental learning,” in *Proceedings of the IEEE/CVF conference on Computer Vision and Pattern Recognition*, 2021, pp. 1591–1600.
- [50] X. Tao, X. Chang, X. Hong, X. Wei, and Y. Gong, “Topology-preserving class-incremental learning,” in *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XIX 16*, Springer, 2020, pp. 254–270.
- [51] F.-Y. Wang, D.-W. Zhou, H.-J. Ye, and D.-C. Zhan, “Foster: Feature boosting and compression for class-incremental learning,” in *European conference on computer vision*, Springer, 2022, pp. 398–414.
- [52] J. Wu *et al.*, “Towards open vocabulary learning: A survey,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2024.
- [53] W. Wu, Y. Zhao, Z. Li, L. Shan, H. Zhou, and M. Z. Shou, “Continual learning for image segmentation with dynamic query,” *IEEE Transactions on Circuits and Systems for Video Technology*, 2023.
- [54] S. Yan, J. Xie, and X. He, “Der: Dynamically expandable representation for class incremental learning,” in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2021, pp. 3014–3023.
- [55] L. Yao *et al.*, *Detclipv2: Scalable open-vocabulary object detection pre-training via word-region alignment*, 2023. arXiv: 2304.04514 [cs.CV].
- [56] J. Yoon, E. Yang, J. Lee, and S. J. Hwang, “Lifelong learning with dynamically expandable networks,” *arXiv preprint arXiv:1708.01547*, 2017.

- [57] Y. Zang, W. Li, K. Zhou, C. Huang, and C. C. Loy, “Open-vocabulary detr with conditional matching,” in *Computer Vision – ECCV 2022*. Springer Nature Switzerland, 2022, pp. 106–122, ISBN: 9783031200779. DOI: 10.1007/978-3-031-20077-9_7. [Online]. Available: http://dx.doi.org/10.1007/978-3-031-20077-9_7.
- [58] A. Zareian, K. D. Rosa, D. H. Hu, and S.-F. Chang, “Open-vocabulary object detection using captions,” in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2021, pp. 14 393–14 402.
- [59] F. Zenke, B. Poole, and S. Ganguli, “Continual learning through synaptic intelligence,” in *International conference on machine learning*, PMLR, 2017, pp. 3987–3995.
- [60] H. Zhang *et al.*, *Glipv2: Unifying localization and vision-language understanding*, 2022. arXiv: 2206.05836 [cs.CV].
- [61] J. Zhang *et al.*, “Class-incremental learning via deep model consolidation,” in *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, 2020, pp. 1131–1140.
- [62] Y. Zhong *et al.*, “Regionclip: Region-based language-image pretraining,” in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022, pp. 16 793–16 803.
- [63] D.-W. Zhou, Q.-W. Wang, H.-J. Ye, and D.-C. Zhan, “A model or 603 exemplars: Towards memory-efficient class-incremental learning,” *arXiv preprint arXiv:2205.13218*, 2022.
- [64] X. Zhou, D. Wang, and P. Krähenbühl, “Objects as points,” *arXiv preprint arXiv:1904.07850*, 2019.
- [65] X. Zhu, W. Su, L. Lu, B. Li, X. Wang, and J. Dai, “Deformable detr: Deformable transformers for end-to-end object detection,” *arXiv preprint arXiv:2010.04159*, 2020.

A Appendix

In Sec.B, we present the algorithmic pipeline of DyQ-DETR. In Sec.C, we introduce a simple but strong baseline with only exemplar replay to emphasize the importance and rationale of the non-exemplar setting. In Sec.D, we provide more detailed experimental results. This includes details of the traditional protocol (Sec.D.1), more fine-grained results about old classes and new classes (Sect.D.2), and more ablation results (Sec. D.3). In Sec.E, we show more visualization results, including images with different levels of risk, behavior of decoupled queries, and performance comparison with CL-DETR. Please note that all experiments, except for those in Sec.D.1, are based on the revised protocol, using Deformable DETR on COCO 2017, and we will not specify this further.

B Algorithmic Pipeline of DyQ-DETR

To get a more comprehensive understanding of the overall framework, we describe the training pipeline of DyQ-DETR in Algorithm 1. In the τ -th phase, we dynamically add a new set of queries, which are responsible for detecting and recognizing new classes. We disentangle self-attention modules in the decoder, ensuring that queries between different groups remain relatively independent. In the incremental training, we use a score threshold to filter out background predictions, resulting in the pseudo-labels y^{pseudo} . Instead of simply merging pseudo-labels y^{pseudo} and incomplete real labels y , we divide the completed annotations of an image according to the incremental phase/class set. For an image, annotations corresponding to different class sets act on the outputs of respective queries, thus resulting in the decoupled loss $\mathcal{L}_{total}^{DETR}$ for training.

After the incremental training, we utilize the trained model to calculate the partial loss as the risk score for images in D_τ . We sort D_τ based on the risk score and simply select the middle 10% of these samples to serve as exemplars. Then, for an image in the exemplar set $\epsilon_{1:\tau}$, we identify the specific stage corresponding to its annotation, and we use the respective output and real labels to compute the partial loss $\mathcal{L}_{partial}$ for calibration.

C Baseline with only Exemplar Replay

We introduce a simple baseline, which only combats forgetting by saving exemplars of the same size. As shown in Tab.6, it completely forgets past knowledge after the incremental training, but its performance significantly improves after the simple exemplar replay. The total memory budget

Algorithm 1: DyQ-DETR (the τ -th phase)

Input: new class data D_τ ; old class exemplars $\epsilon_{1:\tau-1}$; old model Φ^{old} .
Output: new model Φ ; new exemplar set $\epsilon_{1:\tau}$.

```

1 Let  $\Phi \leftarrow \Phi^{old}$ ;
2 Dynamically expand new queries, i.e.,  $\{Q^{old}, Q^{new}\} \leftarrow \{Q^{old}\}$ ;
3 Disentangle self-attention, i.e.,  $attn(Q_{m,i}, Q_{n,j}) = 0, m \neq n$ ;
4 for epochs do // dynamic query for incremental training
5   for mini-batches  $(x, y) \in D_\tau$  do
6     Let  $\hat{y}^{old} \leftarrow \Phi^{old}(x)$  and get  $y^{pseudo}$  from  $\hat{y}^{old}$  ;
7     Let  $\hat{y} \leftarrow \Phi(x)$ ;
8     Compute the decoupled loss  $\mathcal{L}_{total}^{DETR}(\hat{y}, y^{pseudo} \cup y)$ ;
9     Update  $\Phi$  via optimizer;
10    for mini-batches  $(x, y) \in D_\tau$  do // score by risk
11      Let  $\hat{y} \leftarrow \Phi(x)$  and select  $\tau$ -th part  $\hat{y}_\tau$  from  $\hat{y}$ ;
12      Compute risk score using  $\mathcal{L}_{partial}(\hat{y}_\tau, y)$  ;
13    Sort  $D_\tau$  by risk score, and select the middle 10% as  $\epsilon_\tau$  ;
14    Let  $\epsilon_{1:\tau} \leftarrow \epsilon_{1:\tau-1} \cup \epsilon_\tau$ ;
15    for epochs do // partial calibration
16      for mini-batches  $(x, y) \in \epsilon_{1:\tau}$  do
17        Let  $\hat{y} \leftarrow \Phi(x)$  and select the corresponding part  $\hat{y}_k$  from  $\hat{y}$ ;
18        Compute partial calibration loss  $\mathcal{L}_{partial}(\hat{y}_k, y)$  ;
19        Update  $\Phi$  via optimizer;

```

Setting	ER	AP	AP ₅₀	AP ₇₅	AP _S	AP _M	AP _L	AP _{old}	AP _{new}
40+40	w/o	20.1	29.6	21.8	11.5	22.9	26.4	0	40.5
	w/	31.4	46.4	34.0	16.5	34.1	42.2	26.1	37.7

Table 6: Results (%) of the simple baseline in the 40+40 setting.

for exemplars is typically set as 10% of the total dataset size. This relatively large proportion may obscure issues in the incremental training and potentially diminish the advantages of superior methods. Therefore, we believe that comparisons in non-exemplar settings are more suitable for evaluating the performance of different methods.

D More Detailed Experimental Results

D.1 Traditional protocol and results

The main difference between the traditional protocol[48, 21, 40, 9] and the revised protocol[32] lies in data partitioning. In the traditional protocol, the model can observe all images containing at least one object of the currently interested classes, which may result in an image appearing across multiple phases. Formally, let $D = \{(x, y)\}$ denotes a dataset with images x and corresponding object annotations y . First, we divide the total class set into non-overlapping parts $\{C_1, C_2, \dots, C_T\}$, one for each incremental phase. For each phase τ , all samples in D retain annotations of C_τ and drop others. The incremental training dataset in phase τ consists of images that contain at least one

Methods	All classes				Old classes				New classes			
	AP	AP _S	AP _M	AP _L	AP	AP _S	AP _M	AP _L	AP	AP _S	AP _M	AP _L
CL-DETR [32]	37.7	21.1	40.7	50.9	39.7	22.9	40.4	53.7	36.3	19.7	41.4	49.1
DyQ-DETR	39.6	22.9	43.0	53.2	41.3	25.3	42.7	55.3	38.6	21.0	43.9	52.4

Table 7: Fine-grained results (%) for old and new classes in the 40 + 40 setting.

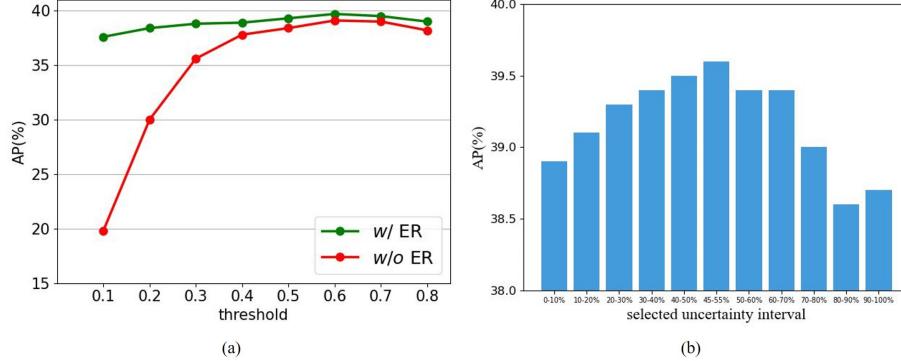


Figure 7: (a) Performance with different score thresholds. (b) Performance with different selected risk intervals. Both experiments are conducted in the 40+40 setting.

annotation for C_τ . Additionally, the revised protocol shuffles the class order, while the traditional one does not, leading to different class partitions.

Tab.1 shows that our proposed DyQ-DETR consistently outperforms the state-of-the-art (SOTA) method in the traditional IOD protocol, especially in the non-exemplar settings. Specifically, without exemplars, our DyQ-DETR surpasses the SOTA by 2.2% AP and 3.7% AP in the 40+40 and 70+10 settings, respectively. With exemplars, the advantage decreases to 0.4% AP and 2.0% AP, respectively. The substantial advantage of our DyQ-DETR in non-exemplar settings demonstrates the effectiveness of dynamic query, which is the core of our method.

D.2 Results about old classes and new classes

Tab.7 presents fine-grained results in the 40+40 setting about old classes and new classes, which respectively represent the model’s stability and plasticity. The results show that, compared to the SOTA, our DyQ-DETR achieves improvements of 1.6% AP for old classes and 2.3% AP for new classes, indicating enhanced stability and plasticity in our method.

D.3 More ablation results

Effect of the score threshold. In the 40+40 setting, we set the score threshold to 0.6, while for other settings, it is set to 0.4. This is because, in other settings, there is a higher proportion of old classes where stability is preferred, and more old class pseudo-labels help combat forgetting. The ablation study on the score threshold is shown in Fig.7(a), where the optimal threshold for the 40+40 setting is 0.6. We also find that when the score threshold is low, the pseudo-label noise is significant, leading to poor performance in non-exemplar settings. However, after exemplar replay, the performance greatly improves, further illustrating that exemplar replay may mask issues in the incremental training.

Effect of the selected risk interval. After sorting the images by risk in ascending order, the results of changing the selection interval are illustrated in Fig.7(b). It can be observed that selecting images with moderate risk is optimal. This selection strategy is straightforward, and our contribution lies more in demonstrating the feasibility of using the model to optimize the selection.

E More visualization results

Images with different levels of risk. As illustrated in Fig.8, images with low risk are considered as simple samples containing only a few annotated objects, while images with high risk are characterized by a dense distribution of objects, accompanied by severe occlusion or inaccurate annotations. Samples in the middle part are considered informative and reliable, and such an active selection strategy promotes more effective exemplar replay.

Behavior of decoupled queries. As depicted in Fig.9, our designed dynamic query set achieves the goal of only detecting the corresponding class set. The old query set is responsible for memorizing

knowledge of old classes, while the new query set focuses on learning new knowledge. This decoupling significantly improves the performance in non-exemplar settings.

Performance comparison with CL-DETR. Fig.10 shows the visualized detection results of CL-DETR and our DyQ-DETR in non-exemplar and exemplar-based settings. In both settings, DyQ-DETR presents more robust performance for remaining old knowledge and learning new knowledge, thus greatly mitigating catastrophic forgetting.

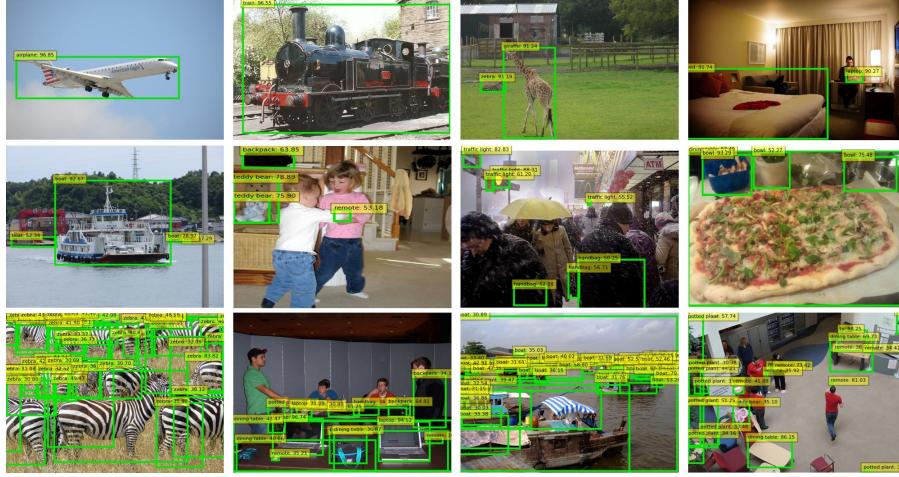


Figure 8: Supplementary to Fig.3 (main paper). Visualization of images with different levels of risk. Rows 1, 2, and 3 respectively represent images with low risk, moderate risk (selected by our DyQ-DETR), and high risk.

F Discussion over Open Vocabulary Object Detection

Open vocabulary object detection (OVD) leverages visually related language data as auxiliary supervision to bridge the gap between base categories and novel label spaces[52, 58]. By using image-caption pairs, it acquires an extensive vocabulary of concepts, allowing the model to learn object detection with annotations only for certain base categories. Consequently, the model can detect both base classes and novel classes that were not present in the training annotations. OVR-CNN[58] is the pioneering work that introduced the concept of open vocabulary object detection and established a comprehensive framework to address this problem. Subsequently, with the introduction of large-scale visual language pretraining models such as CLIP[43], novel approaches have emerged that utilize these pretrained models to enhance the capability of open vocabulary object detection. Some works[11, 41, 44, 34] employ knowledge distillation, using large model text encoders to align visual features with the large model and transfer new class information. Other works address the gap in object detection region localization by converting image-text level pre-training to region-text level[23, 62, 2, 60], while some complete this during the detector training stage after pre-training[27, 57, 55, 33].

The difference between IOD (Incremental Object Detection) and OVD (Open Vocabulary Detection) lies in their focus. IOD emphasizes the continuous learning and updating process of the model, while OVD focuses on the model’s generalization ability. IOD aims to improve the model through multiple learning steps, whereas OVD aims for the model to generalize perfectly to other problems with just training once. Specifically, after training, OVD can detect some objects in a new category space, even if these objects have not been seen in the annotations of base categories, due to the rich semantic concepts obtained during the image-caption alignment process. However, despite the richness of the learned semantic concepts, OVD is difficult to handle newly generated concepts due to the dynamic nature of the environment and requires re-alignment of image-caption pairs to learn these new concepts. In contrast, IOD simplifies this process by allowing incremental learning of new concept based on newly collected and annotated data, thus avoiding the need for retraining the entire model.

IOD and OVD are not mutually exclusive but complementary. They are both important for open-world practical applications. As mentioned above, OVD still shows deficiency in handling the emergence of new concepts without retraining, whereas IOD simplifies this process, saving time and computational resources. However, IOD struggles to resist catastrophic forgetting of old class semantics during continuous model updates. OVD relies on pre-learned rich semantic concepts for novel classes, without consideration of forgetting issue.

G Limitations and Broader Impacts

We propose a dynamic object queries-based detection transformer (DyQ-DETR) to address catastrophic forgetting in incremental object detection. While it scales well to tasks with an incremental step of 20, DyQ-DETR may face challenges in scaling to tasks with longer incremental steps.

High-performance incremental object detection systems have tremendous potential to have a significant impact on various fields and inspire innovative research approaches in robotics, autonomous driving, and beyond. For example, our proposed DyQ-DETR can progressively enhance the recognition and manipulation capabilities of robots, thereby improving efficiency and productivity in industries and healthcare. Furthermore, considering the significant overhead of joint training, DyQ-DETR helps detection models avoid retraining from scratch, providing an effective and efficient approach for incremental updates of large visual models. As for potential widespread impact, our work has not shown any negative social consequences.

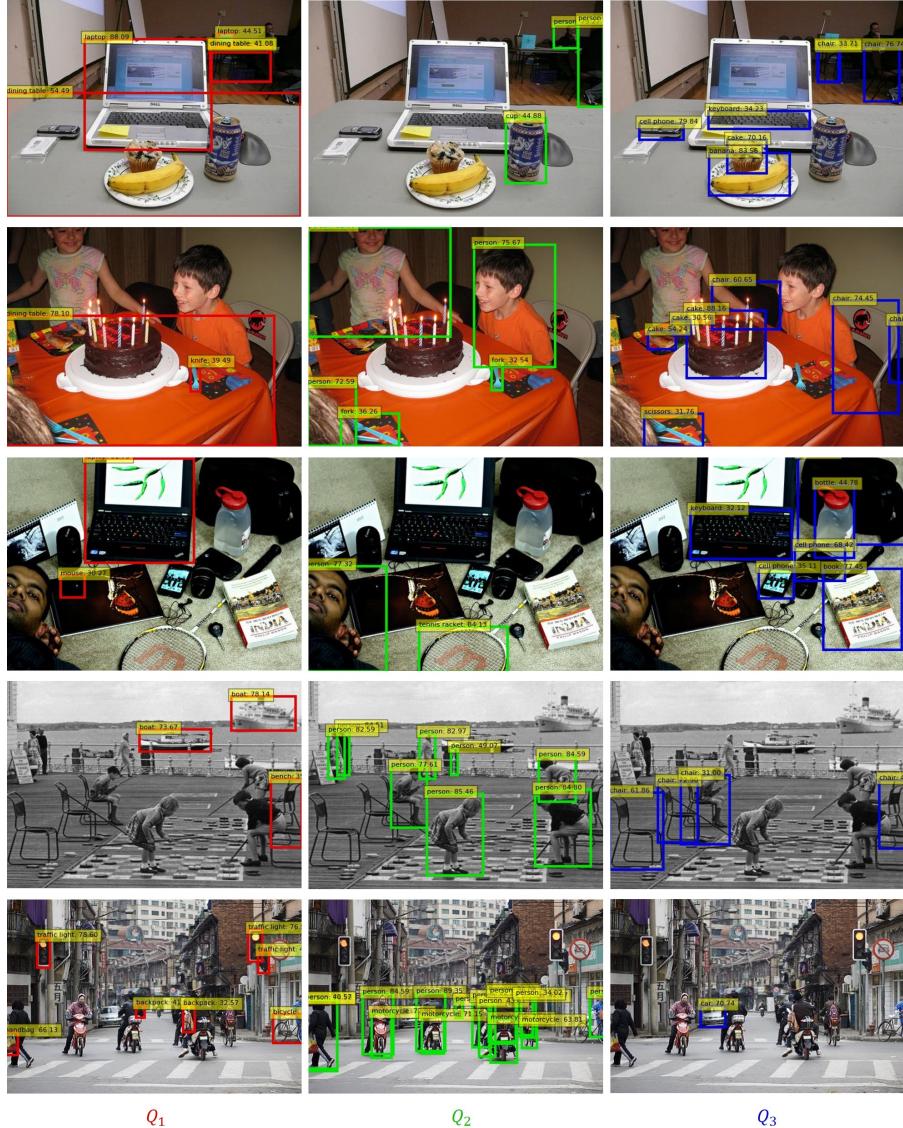


Figure 9: Supplementary to Fig.6 (main paper). Visualization of the behavior of decoupled queries after training in the revised protocol 40+20×2 setting.

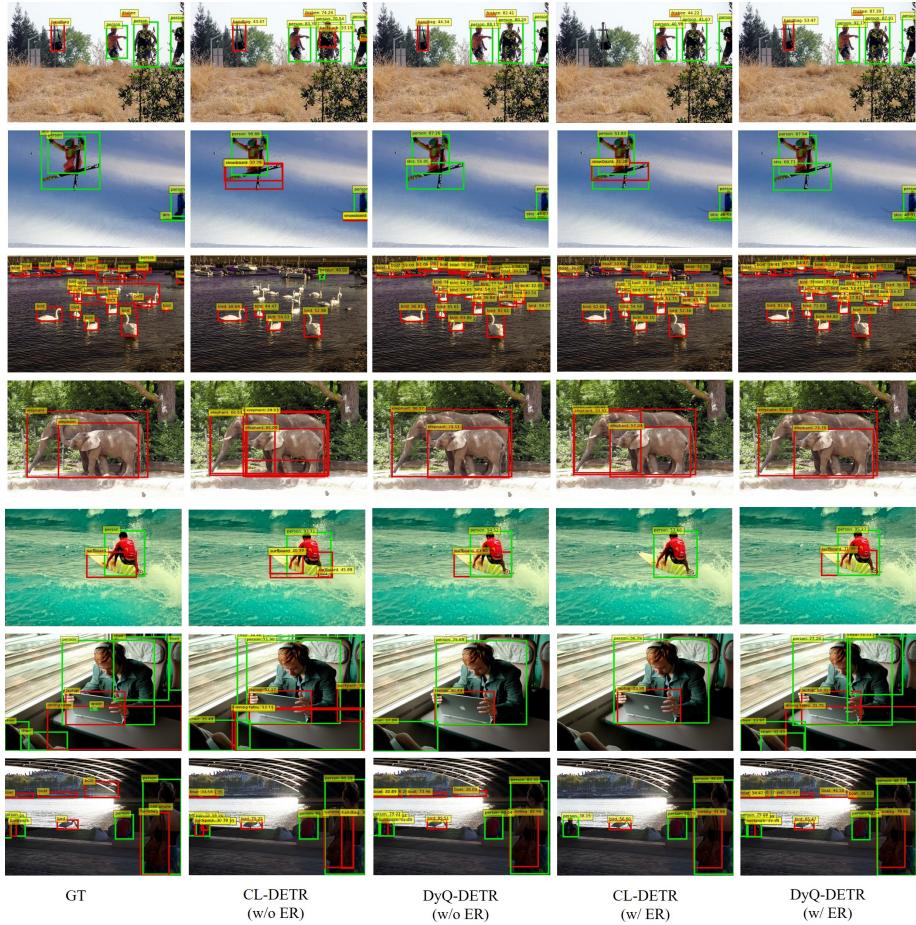


Figure 10: Visualization of performance comparison with CL-DETR, including the non-exemplar setting and exemplar-based setting. Visualization is based on the 40+40 setting. In this figure, red and green represent old classes and new classes, respectively.