Don't Let The Information Slip Away

Taozhe Li
University of Oklahoma
Taozhe.Li-1@ou.edu

Guansu Wang
University of Melbourne
guansuw@student.unimelb.edu.au

Bo Yu
University of Utah
bo.yu@utah.edu

Yiming Liu
University of Oklahoma
ymliu@ou.edu

Wei Sun*
University of Oklahoma
wsun@ou.edu

Abstract

Real-time object detection has advanced rapidly in recent years. The YOLO series of detectors is among the most well-known CNN-based object detection models and cannot be overlooked. The latest version, YOLOv26, was recently released, while YOLOv12 achieved state-of-the-art (SOTA) performance with 55.2 mAP on the COCO val2017 dataset. Meanwhile, transformer-based object detection models, also known as DEtection TRansformer (DETR), have demonstrated impressive performance. RT-DETR is an outstanding model that outperformed the YOLO series in both speed and accuracy when it was released. Its successor, RT-DETRv2, achieved 53.4 mAP on the COCO val2017 dataset. However, despite their remarkable performance, all these models let information to slip away. They primarily focus on the features of foreground objects while neglecting the contextual information provided by the background. We believe that background information can significantly aid object detection tasks. For example, cars are more likely to appear on roads rather than in offices, while wild animals are more likely to be found in forests or remote areas rather than on busy streets. To address this gap, we propose an object detection model called Association DETR, which achieves state-of-the-art results compared to other object detection models on the COCO val2017 dataset.

1. Introduction

Artificial intelligence has made rapid progress in recent years and has been applied in various fields, including but not limited to Simultaneous Localization and Mapping (SLAM) [19], Autonomous Driving, Large Language Model [7, 20], Medical Image Analysis, Behavioral Monitoring, Privacy [6, 16, 32] and more. And object detection remains one of the most active and popular research areas among these fields. The mainstream object detection models consist of two branches: one is CNN-based architectures, and the most famous one in this branch is the YOLO series detectors [14, 15, 35]. The other branch comprises end-to-end Transformer-based detectors [2].

The YOLO series detectors primarily focus on trade-off between speed and accuracy, as not all devices—especially those used in autonomous driving possess powerful computational capabilities. This indicates that the performance of YOLO detectors is generally inferior to that of end-to-end Transformer-based detectors. The YOLO series detector recently released its 12th version [35]. However, it just shows limited performance improvement compared to its previous version, YOLOv11 (1.0 mAP in the N, S and M size models, 0.3 mAP in the L size model, and 0.5 mAP in the X size model).

The end to end Transformer based detector (DETRs) have received extensive attention recently. The latest architecture in DETRs named DEIMv2 [12], it has achieved state-of-the-art (SOTA) performance with a 57.80 mAP on the COCO[21] val 2017 dataset, which is remarkable. However, the best-performing version of DEIMv2(DEIMv2-X) is built by almost 50 million parameters, making it extremely difficult to deploy on the onboard devices that typically lack strong computational power, which is very common in the autonomous driving field. This means that the speed is critical for object detectors.

Due to the high performance and streamlined architecture of DETRs, as well as the urgent need for speed, some researchers have concentrated on real-time DETRs. The work RT-DETR [44] is a DETRs that cannot be overlooked. It outperforms the YOLO series detectors in both speed and accuracy at the time it was proposed, achieving state-of-the-art performance.

However, although the aforementioned YOLO series detectors and DETRs have achieved remarkable performance, they still allow valuable information to slip away. Most of these models focus primarily on foreground information

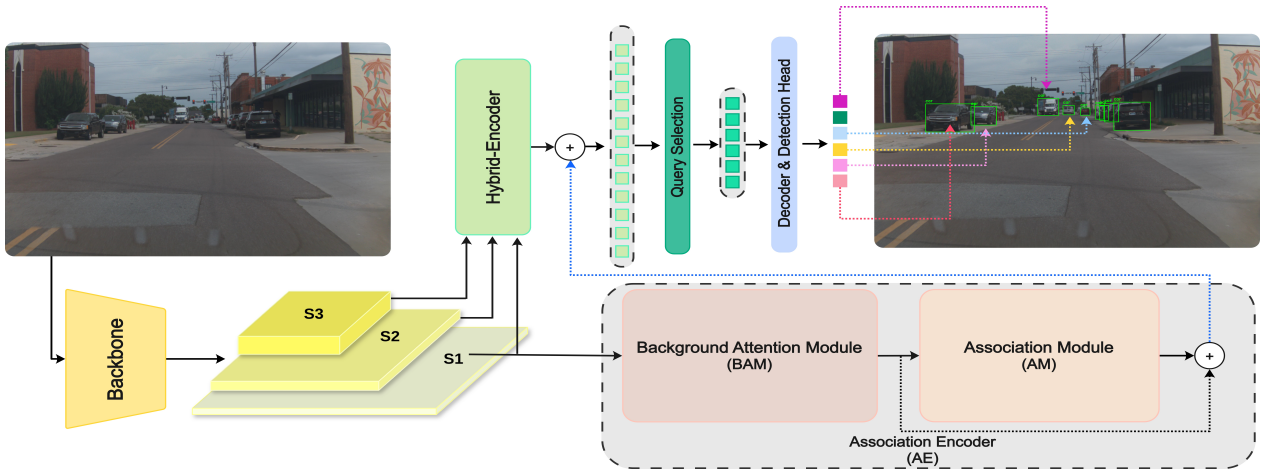

Figure 1. Association DETR Overview. The input image is first fed into the backbone network. The shallowest image feature is denoted as S_1, the second shallowest as S_2, and the deepest as S_3. we feed the shallowest feature S_1 into the Background Attention Module, which is designed to capture background information. Also in Figure 4. We visualize some samples for better understanding the function of the Background Attention Module. And The Features S_1, S_2, and S_3 are fed into the Hybrid Encoder for both intra-feature and inter-feature enhancement. The output F_b of the Background Attention Module is then fed into the Association Module, which performs feature enhancement related to background information. After that, the output of the Association Module, F_a performs an addition operation with F_b. Additionally, Feature F_b is added to F_3 as $F_{\hat{3}}$, the F_3 refers to the output of the Hybrid Encoder corresponding to the input S_3. Finally, the features F_1, F_2, and $F_{\hat{3}}$ undergo query selection and are passed into the Decoder and Detection Head to predict object bounding boxes and classes.

while neglecting the background. The study [39] demonstrates that background information can be extremely useful for computer vision tasks. In that work, even when the foreground object is removed from the image, the classification model can still achieve around 50% accuracy, compared to only 11.11% accuracy from random guessing. From our perspective, the way background information aids computer vision tasks, such as object detection or classification, is similar to the associative ability humans possess. More specifically, when we see a photo taken indoors, if we need to guess what objects are present, we are likely to guess a person, a sofa, or a clock on the wall, and it is highly unlikely we would guess a car or a traffic light. To address the issue of losing background information, we propose an Association DETR equipped with an Association Encoder. The contributions of our work are summarized as follows:

1. We propose **Association DETR**, a detector that jointly exploits foreground and background information. By enhancing the interaction between these complementary cues, the proposed method achieves state-of-the-art performance on the COCO 2017 dataset, reaching **54.6 mAP**.
2. We introduce the **Association Encoder**, a lightweight plug-in module with only **3 million parameters**. The module can be seamlessly integrated into existing DETR-based detectors and consistently improves their performance with minimal computational overhead.

2. Related Work

2.1. Real-time Object Detectors

YOLOv1 [29] is the first CNN-based, one-stage object detector to achieve true real-time object detection. After years of continuous development, the YOLO series of detectors has significantly advanced the field.

Outperforming other one-stage object detectors [9, 24], YOLO has become synonymous with real-time object detection and has been applied in various fields. Moreover, the YOLO series of detectors can be classified into two categories: Anchor-based [1, 13, 28, 29] and anchor-free [14, 15, 35, 36], methods.But only anchor-free YOLO series detectors achieve a reasonable trade-off between speed and accuracy, as demonstrated in [44].

Moreover, the YOLO series of detectors has evolved rapidly, with the latest version announced as YOLO26 [31]. Additionally, the X version of YOLOv10 [36], YOLOv11 [14], and YOLOv12 [35] achieve impressive performance on the COCO 2017 dataset [21], with mean Average Precision (mAP) scores of 54.4, 54.7, and 55.2, respectively.

However, all these models focus exclusively on foreground information, neglecting background information that could potentially further enhance model performance.

2.2. End-to-End Object Detectors

End-to-end object detectors based on Transformers, also known as Detection Transformers, were first proposed by

Table 1. **Comparison with SOTA**. All models are trained on the COCO Dataset. And the FPS is reported on T4 GPU with TensorRT. For Evaluation, all input size are fixed on 640×640, Our Association DETR-R34 achieves 54.60 in AP^{val}, 71.6 in AP^{val}_{50} and 153 FPS.Meanwhile, our Association DETR-R50 achieves 55.7 in AP^{val}, 74.0 in AP^{val}_{50} and 104 FPS. Both of them outperform other YOLO detectors and DETR models with similar scale.

Model	Backbone	#Epochs	#Params (M)	$\#FPS_{bs=1}$	AP^{val}	AP^{val}_{50}
YOLOv10-M	-	300	15.4	210	51.1	68.1
YOLOv10-L	-	300	24.4	137	53.2	70.1
YOLOv10-X	-	300	29.5	93	54.4	71.6
YOLOv11-M	-	300	20.1	212	51.5	67.9
YOLOv11-L	-	300	25.3	161	53.4	70.1
YOLOv12-M	-	300	20.2	206	52.5	69.5
YOLOv12-L	-	300	26.4	148	53.7	70.6
RT-DETR	R34	72	31	161	48.9	66.8
RT-DETRv2-M	R34	72	31	161	49.9	67.5
RT-DETRv3-R34	R34	72	31	161	49.9	67.7
Association-DETR-R34 (Ours)	R34	72	34.1	153	**54.6**	**71.6**
DETR-DC5	R50	500	41	-	43.3	63.1
DETR-DC5	R101	500	60	-	44.9	64.7
Deformable-DETR	R50	108	44	-	45.1	65.4
RT-DETR	R50	72	42	108	53.1	71.3
RT-DETR	R101	72	76	74	54.3	72.7
YOLOv11-X	-	300	56.9	88	54.7	71.6
YOLOv12-X	-	300	59.1	85	55.2	72.2
RT-DETRv2-L	R50	72	42	145	53.4	71.6
RT-DETRv2-X	R101	72	76	74	54.3	72.8
RT-DETRv3-R50	R50	72	42	108	53.4	71.7
DEIM-RT-DETRv2-X	R101	72	76	–	55.4	73.7
Association-DETR-R50 (Ours)	R50	72	45.1	104	**55.7**	**74.0**

Table 2. **Effectiveness**. We integrate our Association Encoder(AE) into various models. All these models outperform their respective baselines after integrating our Association Encoder (AE). Notably, the Association-DETR-R34 and Association-DETR-R50 models achieve state-of-the-art performance compared to other models.

Model	Backbone	#Epochs	#Params (M)	$\#FPS_{bs=1}$	AP^{val}	AP^{val}_{50}
RT-DETR-R34 (base)	R34	72	31	161	48.9	66.8
AE + DETR-R34 (Ours)	R34	72	34.1	153	**54.6**(↑**5.7**)	**71.6**(↑**4.8**)
RT-DETR-R50 (base)	R50	72	42	108	53.1	71.3
AE + DETR-R50 (Ours)	R50	72	45.1	104	**55.7**(↑**2.6**)	**74.0**(↑**2.7**)
RT-DETRv2-M (base)	R34	72	31	161	49.9	67.5
AE + RT-DETRv2-M	R34	72	34.1	153	**52.4**(↑**2.5**)	**70.2**(↑**2.7**)
RT-DETRv2-L (base)	R50	72	42	145	53.4	71.6
AE + RT-DETRv2-L	R50	72	45.1	137	**53.7**(↑**0.3**)	**72.1**(↑**0.6**)
DETR-R50 (base)	R50	500	41	-	43.3	63.1
DETR-R101 (base)	R101	500	60	-	44.9	64.7
AE + DETR	R50	500	63.1	-	**46.0**(↑**2.7**)	**66.7**(↑**3.6**)
Deformable-DETR(base)	R50	108	44	-	45.1	65.4
AE + Deformable DETR	R50	108	47.1	-	**47.7**(↑**2.6**)	**68.6**(↑**3.2**)

Carion et al. [2]. These models have attracted extensive attention since they have been posted. Compared to the YOLO series detectors, they possess distinctive features, a streamlined structure, and significant potential. However, they also face several challenges, including slow training convergence, high computational cost, and difficult-to-optimize queries.

Many DETR variants have been developed to address these issues.

For solving the problem of slow convergence time, the Deformable-DETR [45] introduce utilizing multi-scale features and enhancing the efficiency of the attention mechanism. And the SAM-DETR [41] improves convergence speed through semantic information matching. Moreover, DAB-DETR [23] and DN-DETR [17] further enhance performance by introducing iterative refinement schemes and denoising training.

To reduce computational cost, Group-DETR [3, 4] employs group-wise one-to-many assignment. RT-DETR [44] proposes a hybrid-efficient encoder and an improved query selection method to further reduce computation. The upgraded version, RT-DETRv2 [26], introduces selective multi-scale sampling to better handle objects of varying sizes. Additionally, Efficient DETR [40] and Sparse DETR [30] reduce computational cost by decreasing the number of encoder and decoder layers or the number of updated queries. And the Lite DETR [18] improves encoder efficiency by lowering the update frequency of low-level features.

To decreasing the optimization difficulty of queries. The features are processed in an interleaved manner, which is utilized in Conditional DETR [27] and Anchor DETR [37]. Also, work [45] proposes a query selection method that is specialized for two-stage DETR.

However, all the aforementioned DETRs and YOLO series detectors focus exclusively on foreground information and executing inter and intra-enhancement on it, but none of them explicitly leverage background information. Our Association-DETR bridges this gap, it achieves SOTA performance on the COCO 2017 dataset with a 54.6 mAP as presented in Table 1. Also, according to the experimental results in Table 2, the proposed light Association Encoder can be integrated into any existing DETR model to effectively boost its performance with minimal impact on inference time.

2.3. Spatial Attention & Channel Attention

Spatial attention refers to the mechanism by which cognitive and computational systems selectively focus on specific spatial regions to enhance the processing of relevant visual information. It has been widely used in computer vision tasks to improve model performance. For example, the Convolutional Block Attention Module (CBAM) [38] is a notable advancement in this field. It enhances the representational capacity of convolutional neural networks (CNNs) by incorporating both channel and spatial attention mechanisms. By strengthening the network's representational power, CBAM improves classification performance on the ImageNet-1K dataset [5]. Additionally, the Coordinate Attention (CA) [11] incorporates positional information into channel attention, further boosting model performance in classification and object detection tasks. Moreover, compared to CBAM and CA, the proposed Receptive-Field Attention (RFA) in [43] not only focuses on spatial features but also addresses the issue of parameter sharing in convolutional kernels. RFA achieves better performance than CBAM and CA on both classification and object detection tasks. Furthermore, the combination of RFA and CBAM, termed RFCBAMConv, attains state-of-the-art performance on the ImageNet-1K classification task. Therefore,We will utilize RFCBAMConv in our subsequent Background Attention Module (BAM) because it demonstrates better performance in image classification than RFCAConv, as reported in [43], and exhibits comparable performance in semantic segmentation tasks. The experimental results are presented in Table 3 in Section 4.4.

3. Association DETR

3.1. Overview

The overview of Association DETR is presented in Figure 1. Our model uses the powerful RT-DETR [44] as the baseline, upon which we build a Association Encoder which consists of two modules: the Background Attention Module and the Association Module. Each component of these modules is carefully designed to achieve an optimal balance between speed and efficiency. The input image is first fed into the backbone network. For fair comparison with other models and also to ensure stable performance, we select ResNet-34 and ResNet-50 as our backbones. We extract multi-level features from the backbone for further processing. The shallowest image feature is denoted as S_1, the second shallowest as S_2, and the deepest as S_3. Utilizing multi-level features is a common technique in computer vision field, as it has been shown to improve model performance [22] in different computer vision tasks. Typically, shallow image features captured by neural network is mainly consists of basic information such as edges, corners, and textures, while deeper neural network captures high-level features, including semantic information, object parts, or even specific objects. For these reasons, we feed the shallowest feature S_1 into the Background Attention Module, which is designed to capture background information. Also in Figure 4. We visualize some sample for better understanding the function of the Background Attention Module. The visualization is plotted by tool pytorch-grad-cam [8] .And The Features S_1, S_2, and S_3 are fed into the Hybrid Encoder for both intra-feature and inter-feature enhancement. The output F_b of the Background Attention Module is then fed into the Association Module, which performs feature enhancement related to background information. After that, the output of the Association Module, F_a performs an addition operation with F_b. This design helps preventing the vanishing gradient problem and enriches the extracted background information. Additionally, Feature

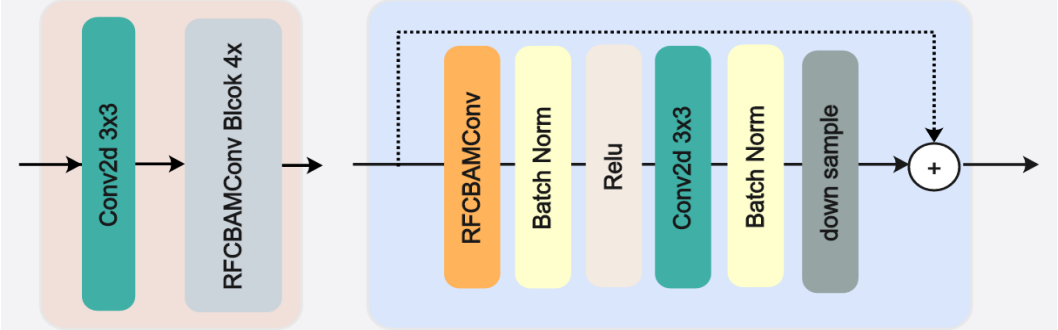

Figure 2. Background Attention Module & Single RFCBAMConv Block. On the left side is the structure of the Background Attention Module, and on the right side are the details of a single RFCBAMConv Block, which is located within the Background Association Module.

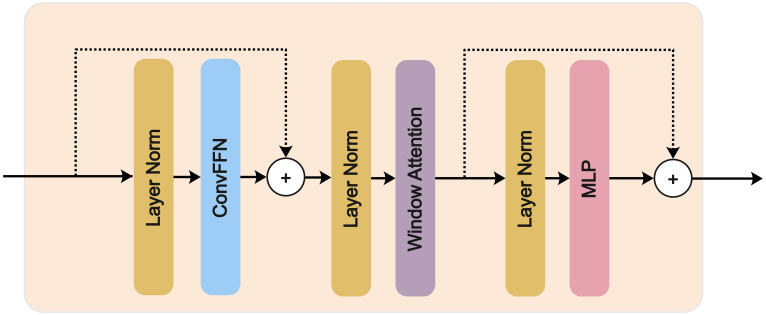

Figure 3. Association Module. We incorporate ConvFFN and Window Attention for trade off between performance and speed.

F_b is added to F_3 as $F_{\hat{3}}$, the F_3 refers to the output of the Hybrid Encoder corresponding to the input S_3. This operation further enriches the original image feature and serves as another feature enhancement technique. Finally, the features F_1, F_2, and $F_{\hat{3}}$ undergo query selection and are passed into the Decoder and Detection Head to predict object bounding boxes and classes.

3.2. Association Encoder

The Association Encoder (AE) is a light encoder we carefully elaborate with only 3.1 million parameters. It consists of two components: the Background Attention Module (BAM) and the Association Module (AM). These modules are designed for background feature extraction and intra-feature enhancement respectively. Furthermore, we find that the AE is an efficient plug-in module that can be integrated into any DETR detector to improve the model's performance, as demonstrated by the experimental results in Table 2. The details of each module are described below.

3.2.1. Background Attention Module

The background attention module is designed to extract background information effectively. The detailed structure of our proposed background attention module is illustrated in Figure 2. We carefully designed each component to optimize both performance and speed. The detailed architecture of the RFCBAMConv block is shown on the right side of Figure 2. The RFCBAMConv is derived from the RFA (Receptive-Field Attention) method [43]. It combines RFA and CBAM, achieving state-of-the-art (SOTA) performance in classification tasks. This also demonstrates that RFCBAMConv has a strong capability for extracting image features. We apply this RFCBAMConv in the BM Module for better extract spatial image features, specifically background information. Additionally, some readers may notice that the structure of this BAM resembles ResNet blocks. This similarity is intentional because we pre-train the BAM before applying it to object detection tasks. The goal of this module is to extract background information efficiently while using fewer parameters. The simplest approach is to build a whole model similar to ResNet, maintaining the same number of layers and parameters but replacing all convolutional layers with RFCBAMConv. However, this results in a large number of parameters; even the smallest model contains 11 million parameters. To address this issue, we share the first two blocks with our backbone (ResNet has a total of four blocks) and train only the two blocks which located in BAM for background information extraction. This approach is effective and requires fewer parameters compared to the previously mentioned method (it is one-quarter of the number parameters of previous method). It is pre-trained on the Stanford Background Dataset [10] as a classification task. It includes a total of 9 background categories [10], such as grass, road,

tree, and sky. Then, the global average pooling layer and prediction head are removed before integrating it into any object detection model. This method significantly reduces the number of parameters required by the model and increases inference speed. If we were to use a complete ResNet-like model for the background attention module, it would increase the number of parameters by approximately five times compared to our method, resulting in much slower inference. The visualization results of background attention module that tested on some random images are presented in Figure 4. As shown, the module effectively captures background information across various scenarios. In the first image featuring bears, it successfully identifies the grass behind them. In the second image, it accurately detects grass and even fencing, despite fencing is not included in our training categories. Furthermore, in the third and fourth images, the module correctly identifies the road, sky, and grass.

3.2.2. Association Module

The AM is designed to convert the extracted background information from the background-attention module into association information relevant to object detection. Essentially, it functions as a feature enhancement module from our perspective. Its structure is illustrated in Figure 3. To balance performance and speed, we incorporate ConvFFN [33] and Window Attention [42] within this module. ConvFFN serves as a feature extraction component that is significantly more efficient than self-attention. We made minor modifications to adapt it for the AM in the object detection task, as it was originally developed for super-resolution tasks. Additionally, Window Attention is also a optimization between performance and speed. It achieves comparable performance to multi-head attention but with a time complexity of only $O(n \times w)$, whereas the latter has a quadratic time complexity of $O(n^2)$. Here, n represents the number of image patches, and w denotes the fixed number of windows. For evaluating the importance of this module, we conducted ablation experiments presented in Section 4.4. And the experimental result is presented in Table 3. The module contributes an improvement of 1.3 mAP while adding only 0.7 million parameters. Furthermore, in the ablation study, we replaced this module with an EL (basic encoder layer) and found that although the EL has 0.4 million more parameters than the AM, it performs worse both individually and when combined with BAM.

4. Experimental Results

4.1. Experimental Settings

All experiments were conducted on a workstation equipped with one NVIDIA A100 GPU with 80 GB of VRAM and an Intel Xeon CPU with 128 GB of RAM. We train the Association-DETR-R34 and the Association-DETR-R50 in the same configurations. More specifically, we use AdamW[25] as optimizer, the base learning rate is $1e-4$, the training epoch is 72, the batch-size for training and evaluation is 16, the number of Hybrid Encoder layer is 1, the number of selection query is 300, the number of Background Attention Module and Association Module are 1, the embedding dims is 256. Most of our training parameters are the same as our baseline RT-DETR[44]. The hyper-parameter details are presented in table 5. Moreover, the hyper-parameters for pre-training the Background Attention Module on the Stanford Background Dataset is same parameters as [34].

4.2. Dataset

4.2.1. Stanford Background Dataset

The Stanford Background Dataset was introduced by Gould et al. [10] to evaluate methods for geometric and semantic scene understanding. It contains total 715 images selected from public datasets and the selection criteria required images to depict outdoor scenes, contain at least one foreground object, and include a visible horizon within the image. Semantic and geometric labels were obtained using Amazon Mechanical Turk. Moreover, there are total 9 kind of background class. They are sky, tree, road, grass, water, building, mountain, foreground and unknown respectively.

4.2.2. COCO

We conduct evaluation on the COCO 2017 val set, a widely used benchmark for object detection and related vision tasks. The validation split contains 5,000 images spanning 80 object categories, with annotations including bounding boxes, instance masks, category labels, and human keypoints where available. Its large visual diversity, rich annotations, and standardized evaluation protocol make it a reliable benchmark for fair comparison with prior methods.

4.3. SOTA comparison

Table 1 compares our Association DETR with current real-time detectors (YOLO models) and end-to-end detectors (DETR models). Specifically, we compare the M, L, and X variants of YOLOv10, YOLOv11, and YOLOv12, as all achieve outstanding performance on the COCO dataset. Our Association DETR and the YOLO detectors use a common input size of 640 × 640 pixels, while other DETR models use an input size of 800 × 1333 pixels. The FPS measurements were obtained on an NVIDIA T4 GPU using TensorRT FP16 precision. All YOLO detectors employed official pre-trained models available on their respective websites. Our Association DETR-R34 achieves 54.60 in AP^{val}, 71.6 in AP^{val}_{50} and 153 FPS, representing the best performance among models with fewer than 40 million parameters. Meanwhile, our Association DETR-R50 achieves 55.7 in AP^{val}, 74.0 in AP^{val}_{50} and 104 FPS. Both Association DETR-R34 and Association DETR-R50 outperform other YOLO detectors and DETR

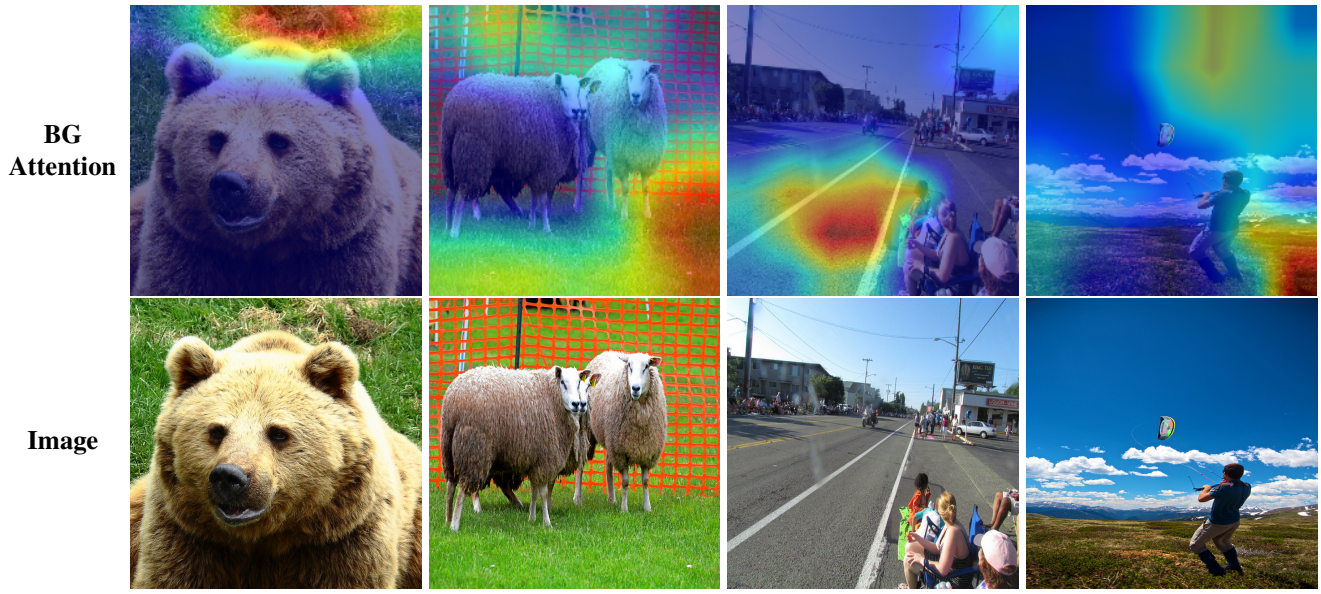

Figure 4. Visualization of Background Attention. BG refers to background. The figure is plotted by pytorch-grad-cam. The BAM (Background Attention Module) effectively captures background information across various scenarios. In the first image featuring bears, it successfully identifies the grass behind them. In the second image, it accurately detects grass and even fencing, despite fencing is not included in our training categories. Furthermore, in the third and fourth images, the module correctly identifies the road, sky, and grass.

Table 3. **Ablation**. All models are trained on the COCO 2017 dataset. For training, the number of epochs is fixed at 72 to ensure a fair comparison. For evaluation, all input sizes are fixed at 640×640. BAM, AM, and EL represent the Background Attention Module, Association Module, and Basic ViT Encoder Layer, respectively.

Model	BAM	AM	EL	#Params (M)	AP^{val}	AP^{val}_{50}
RT-DETR-R34				31	48.9	66.8
RT-DETR-R34	✓			33.4	52.1(↑**3.2**)	70.4(↑**3.6**)
RT-DETR-R34		✓		31.7	50.2 (↑**1.3**)	68.7(↑**1.9**)
RT-DETR-R34			✓	32.1	49.9(↑1.0)	68.6(↑1.8)
RT-DETR-R34	✓		✓	34.5	53.5(↑4.6)	70.6 (↑3.8)
RT-DETR-R34	✓	✓		34.1	**54.6**(↑**5.7**)	**71.6**(↑**4.8**)
RT-DETR-R50				42	53.1	71.3
RT-DETR-R50	✓			44.4	54.4(↑**1.3**)	73.1(↑**1.8**)
RT-DETR-R50		✓		42.7	53.9(↑**0.8**)	72.3(↑**1.0**)
RT-DETR-R50			✓	43.1	53.5(↑0.4)	72.0(↑0.7)
RT-DETR-R50	✓		✓	45.5	54.7(↑1.6)	72.6(↑1.3)
RT-DETR-R50	✓	✓		45.1	**55.7**(↑**2.6**)	**74.0**(↑**2.7**)

Table 4. **Ablation on gains from Stanford Background Dataset.** These experimental results indicate that there is an association between detected objects and background information, and our proposed AE is also capable of learning it.

Model	Pretrained	AP^{val}	AP^{val}_{50}
(1)AE + RT-DETR-R34	✓	**54.6**	**71.6**
(2)AE + RT-DETR-R34		49.3	68.0
(3)AM only + RT-DETR-R34		50.2	68.7

models with similar scale.

Moreover, to evaluate the effectiveness of our proposed Association Module, we integrate our Association Encoder (AE) into various models, including RT-DETR-v2-R50, RT-DETRv2-R34 [26], DETR [2], and Deformable DETR [45]. The experimental results are presented in Table 2. As shown, our Association Module incurs a slight reduction in speed but achieves higher performance compared to each baseline model. Specifically, for the RT-DETR-R34 model, it increases AP^{val} by 5.7 and AP^{val}_{50} by 4.8, while reducing FPS by less than 5.7%. Similarly, it improves AP^{val} by 2.6 and

AP_{50}^{val} by 2.7 for RT-DETR-R50. Moreover, 2.5 in AP^{val} and 2.7 in AP_{50}^{val} for RT-DETRv2-M; 0.3 in AP^{val} and 0.6 in AP_{50}^{val} for RT-DETRv2-L with R50 as backbone; DETR-R50, it increases 2.7 in AP^{val} and 3.6 in AP_{50}^{val}. We need to noticed that it even outperforms than DETR-R101(base). Moreover, 2.6 in AP^{val} and 3.2 in AP_{50}^{val} for Deformable DETR. All these models outperform their respective baselines after integrating our Association Encoder (AE). Notably, the Association-DETR-R34 and Association-DETR-R50 models achieve state-of-the-art performance compared to other models. These experimental results indicate that the AE is a lightweight plug-in module that enhances model performance by incorporating background information.

4.4. Ablation

For evaluating the importance and effectiveness of each module and comparing them with the basic components (basic vistion transforemr encoder layer), we conduct ablation experiments. The experimental results are presented in Table 3. We streamline each module for of Association Encoder in the Association DETR, you can tell both of these modules only consists of 3.1 million parameters according to the Table 3. More specifically, the BAM (Background Attention Module) is implemented with only 2.4 million parameters, while the AM (Association Module) uses only 0.7 million parameters. The EL (basic vision transformer encoder layer) with multi-head attention (8 heads) is implemented with 1.1 million parameters. Furthermore, we observe that both BAM and AM improve the baseline model. However, BAM contributes an increase of 3.2 in AP^{val} and 3.6 in AP_{50}^{val} for the RT-DETR-R34 model, and an increase of 1.3 in AP^{val} and 1.8 in AP_{50}^{val} in RT-DETR-R50. Similarly, The AM contributes 1.3 in AP^{val} and 1.9 in AP_{50}^{val} for the RT-DETR-R34 model, and 0.8 in AP^{val} and 1.0 in AP_{50}^{val} in another setting. Given that AM is implemented with only 0.7 million parameters, it is reasonable that it yields a smaller performance boost than BAM. Moreover, although EL has 0.4 million more parameters than AM, it performs worse both when compared to AM alone and when combined with BAM. These results indicate that our proposed BAM and AM modules are effective and offer a good trade-off between performance and speed. Moreover, we perform ablation experiments to evaluate performance gains from the Stanford Background Dataset. The experimental result shows in Table 4. Comparing (1) and (2) shows learning from Stanford images drives performance. Further analysis with (3), we find untrained BAM negatively impacts learning performance as (3) outperformed (2). This further indicates an association between background information and detected objects.

5. Conclusion

In this paper, for bridging the gap that no object detection model explicitly utilizes background information and just let it slip away, we propose Association DETR, which leverages background information and achieves state-of-the-art performance on the COCO dataset with 55.7 mAP. Moreover, the proposed Association Encoder (AE) is a lightweight plug-in module that can be integrated into any existing DETR model, enhancing its performance while sacrificing minimal speed.

Table 5. hyperparameters of Association DETR. Most of hyperparameters are as same as our baseline RT-DETR

Item	Value
optimizer	AdamW
base learning rate	1e-4
learning rate of backbone	1e-5
freezing BN	True
linear warm-up start factor	0.001
linear warm-up steps	2000
weight decay	0.0001
clip gradient norm	0.1
ema decay	0.9999
number of encoder layers	1
number of Max Pooling layers	1
number of input channels C_i	256
number of reduced channels C_j	32
feedforward dim	1024
number of selected queries	300
number of Assocaition Encoder Layer	1
number of Background Attention Module	1
number of Association Module	1
number of decoder layers	1
max object detections	100
epoch	72
batch-size	8

References

[1] Alexey Bochkovskiy, Chien-Yao Wang, and Hong-Yuan Mark Liao. Yolov4: Optimal speed and accuracy of object detection. *arXiv preprint arXiv:2004.10934*, 2020. 2

[2] Nicolas Carion, Francisco Massa, Gabriel Synnaeve, Nicolas Usunier, Alexander Kirillov, and Sergey Zagoruyko. End-to-end object detection with transformers, 2020. 1, 3, 7

[3] Qiang Chen, Jian Wang, Chuchu Han, Shan Zhang, Zexian Li, Xiaokang Chen, Jiahui Chen, Xiaodi Wang, Shuming Han, Gang Zhang, et al. Group detr v2: Strong object detector with encoder-decoder pretraining. *arXiv preprint arXiv:2211.03594*, 2022. 4

[4] Qiang Chen, Xiaokang Chen, Jian Wang, Shan Zhang, Kun Yao, Haocheng Feng, Junyu Han, Errui Ding, Gang Zeng, and Jingdong Wang. Group detr: Fast detr training with group-wise one-to-many assignment. In *Proceedings of*

the IEEE/CVF international conference on computer vision, pages 6633–6642, 2023. 4

[5] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *2009 IEEE Conference on Computer Vision and Pattern Recognition*, pages 248–255, 2009. 4

[6] Chen Feng and Ioannis Patras. MaskCon: Masked Contrastive Learning for Coarse-Labelled Dataset. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2023. 1

[7] Chen Feng, Minghe Shen, Ananth Balashankar, Carsten Gerner-Beuerle, and Miguel R. D. Rodrigues. Noisy but valid: Robust statistical evaluation of LLMs with imperfect judges. In *The Fourteenth International Conference on Learning Representations (ICLR)*, 2026. 1

[8] Jacob Gildenblat and contributors. Pytorch library for cam methods. https://github.com/jacobgil/pytorch-grad-cam, 2021. 4

[9] Ross Girshick. Fast r-cnn. In *Proceedings of the IEEE international conference on computer vision*, pages 1440–1448, 2015. 2

[10] Stephen Gould, Richard Fulton, and Daphne Koller. Decomposing a scene into geometric and semantically consistent regions. In *2009 IEEE 12th international conference on computer vision*, pages 1–8. IEEE, 2009. 5, 6

[11] Qibin Hou, Daquan Zhou, and Jiashi Feng. Coordinate attention for efficient mobile network design. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 13713–13722, 2021. 4

[12] Shihua Huang, Yongjie Hou, Longfei Liu, Xuanlong Yu, and Xi Shen. Real-time object detection meets dinov3. *arXiv*, 2025. 1

[13] Glenn Jocher. Ultralytics yolov5, 2020. 2

[14] Glenn Jocher and Jing Qiu. Ultralytics yolo11, 2024. 1, 2

[15] Glenn Jocher, Ayush Chaurasia, and Jing Qiu. Ultralytics yolov8, 2023. 1, 2

[16] Zong Ke, Shicheng Zhou, Yining Zhou, Chia Hong Chang, and Rong Zhang. Detection of ai deepfake and fraud in online payments using gan-based models. In *2025 8th International Conference on Advanced Algorithms and Control Engineering (ICAACE)*, pages 1786–1790. IEEE, 2025. 1

[17] Feng Li, Hao Zhang, Shilong Liu, Jian Guo, Lionel M Ni, and Lei Zhang. Dn-detr: Accelerate detr training by introducing query denoising. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 13619–13627, 2022. 4

[18] Feng Li, Ailing Zeng, Shilong Liu, Hao Zhang, Hongyang Li, Lei Zhang, and Lionel M Ni. Lite detr: An interleaved multi-scale encoder for efficient detr. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 18558–18567, 2023. 4

[19] Taozhe Li and Wei Sun. MLP-SLAM: Multilayer perceptron-based simultaneous localization and mapping with a dynamic and static object discriminator. *arXiv e-prints*, pages arXiv–2410, 2024. 1

[20] Zichao Li and Zong Ke. Domain meets typology: Predicting verb-final order from universal dependencies for financial and blockchain nlp. In *Proceedings of the 7th Workshop on Research in Computational Linguistic Typology and Multilingual NLP*, pages 156–164, 2025. 1

[21] Tsung-Yi Lin, Michael Maire, Serge Belongie, Lubomir Bourdev, Ross Girshick, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. Microsoft coco: Common objects in context. *arXiv preprint arXiv:1405.0312*, 2014. 1, 2

[22] Tsung-Yi Lin, Piotr Dollár, Ross Girshick, Kaiming He, Bharath Hariharan, and Serge Belongie. Feature pyramid networks for object detection. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2117–2125, 2017. 4

[23] Shilong Liu, Feng Li, Hao Zhang, Xiao Yang, Xianbiao Qi, Hang Su, Jun Zhu, and Lei Zhang. Dab-detr: Dynamic anchor boxes are better queries for detr. *arXiv preprint arXiv:2201.12329*, 2022. 4

[24] Wei Liu, Dragomir Anguelov, Dumitru Erhan, Christian Szegedy, Scott Reed, Cheng-Yang Fu, and Alexander C Berg. Ssd: Single shot multibox detector. In *European conference on computer vision*, pages 21–37. Springer, 2016. 2

[25] Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. *arXiv preprint arXiv:1711.05101*, 2017. 6

[26] Wenyu Lv, Yian Zhao, Qinyao Chang, Kui Huang, Guanzhong Wang, and Yi Liu. Rt-detrv2: Improved baseline with bag-of-freebies for real-time detection transformer. *arXiv preprint arXiv:2407.17140*, 2024. 4, 7

[27] Depu Meng, Xiaokang Chen, Zejia Fan, Gang Zeng, Houqiang Li, Yuhui Yuan, Lei Sun, and Jingdong Wang. Conditional detr for fast training convergence. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 3651–3660, 2021. 4

[28] Joseph Redmon and Ali Farhadi. Yolov3: An incremental improvement. *arXiv preprint arXiv:1804.02767*, 2018. 2

[29] Joseph Redmon, Santosh Divvala, Ross Girshick, and Ali Farhadi. You only look once: Unified, real-time object detection. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 779–788, 2016. 2

[30] Byungseok Roh, JaeWoong Shin, Wuhyun Shin, and Saehoon Kim. Sparse detr: Efficient end-to-end object detection with learnable sparsity. *arXiv preprint arXiv:2111.14330*, 2021. 4

[31] Ranjan Sapkota and Manoj Karkee. Ultralytics yolo evolution: An overview of yolo26, yolo11, yolov8 and yolov5 object detectors for computer vision and pattern recognition. *arXiv preprint arXiv:2510.09653*, 2025. 2

[32] Zhonglin Sun, Chen Feng, Ioannis Patras, and Georgios Tzimiropoulos. LAFS: Landmark-based Facial Self-supervised Learning for Face Recognition. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2024. 1

[33] Yingxia Tang, Yanxuan Wei, Yupeng Hu, Xiangwei Zheng, and Cun Ji. Convolutional network integrated with frequency adaptive learning for multivariate time series classification. *ACM Transactions on Knowledge Discovery from Data*, 19 (8):1–23, 2025. 6

[34] Sasha Targ, Diogo Almeida, and Kevin Lyman. Resnet in resnet: Generalizing residual architectures. *arXiv preprint arXiv:1603.08029*, 2016. 6

[35] Yunjie Tian, Qixiang Ye, and David Doermann. Yolov12: Attention-centric real-time object detectors, 2025. 1, 2

[36] Ao Wang, Hui Chen, Lihao Liu, Kai Chen, Zijia Lin, and Jungong Han. Yolov10: Real-time end-to-end object detection. *Advances in Neural Information Processing Systems*, 37:107984–108011, 2024. 2

[37] Yingming Wang, Xiangyu Zhang, Tong Yang, and Jian Sun. Anchor detr: Query design for transformer-based detector. In *Proceedings of the AAAI conference on artificial intelligence*, pages 2567–2575, 2022. 4

[38] Sanghyun Woo, Jongchan Park, Joon-Young Lee, and In So Kweon. Cbam: Convolutional block attention module. In *Proceedings of the European conference on computer vision (ECCV)*, pages 3–19, 2018. 4

[39] Kai Yuanqing Xiao, Logan Engstrom, Andrew Ilyas, and Aleksander Madry. Noise or signal: The role of image backgrounds in object recognition. *CoRR*, abs/2006.09994, 2020. 2

[40] Zhuyu Yao, Jiangbo Ai, Boxun Li, and Chi Zhang. Efficient detr: improving end-to-end object detector with dense prior. *arXiv preprint arXiv:2104.01318*, 2021. 4

[41] Gongjie Zhang, Zhipeng Luo, Yingchen Yu, Kaiwen Cui, and Shijian Lu. Accelerating detr convergence via semantic-aligned matching. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 949–958, 2022. 4

[42] Qiming Zhang, Yufei Xu, Jing Zhang, and Dacheng Tao. Vsa: Learning varied-size window attention in vision transformers. In *European conference on computer vision*, pages 466–483. Springer, 2022. 6

[43] Xin Zhang, Chen Liu, Degang Yang, Tingting Song, Yichen Ye, Ke Li, and Yingze Song. Rfaconv: Innovating spatial attention and standard convolutional operation. *arXiv preprint arXiv:2304.03198*, 2023. 4, 5

[44] Yian Zhao, Wenyu Lv, Shangliang Xu, Jinman Wei, Guanzhong Wang, Qingqing Dang, Yi Liu, and Jie Chen. Detrs beat yolos on real-time object detection. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 16965–16974, 2024. 1, 2, 4, 6

[45] Xizhou Zhu, Weijie Su, Lewei Lu, Bin Li, Xiaogang Wang, and Jifeng Dai. Deformable detr: Deformable transformers for end-to-end object detection. *arXiv preprint arXiv:2010.04159*, 2020. 4, 7