remote sensing

MDPI

Technical Note

Improved Object Detection with Content and Position Separation in Transformer

Yao Wang [1] and Jong-Eun Ha [2,*]

[1] Graduate School of Automotive Engineering, Seoul National University of Science and Technology, Seoul 01811, Republic of Korea; yaowang@seoultech.ac.kr

[2] Department of Mechanical and Automotive Engineering, Seoul National University of Science and Technology, Seoul 01811, Republic of Korea

* Correspondence: jeha@seoultech.ac.kr

Abstract: In object detection, Transformer-based models such as DETR have exhibited state-of-the-art performance, capitalizing on the attention mechanism to handle spatial relations and feature dependencies. One inherent challenge these models face is the intertwined handling of content and positional data within their attention spans, potentially blurring the specificity of the information retrieval process. We consider object detection as a comprehensive task, and simultaneously merging content and positional information like before can exacerbate task complexity. This paper presents the Multi-Task Fusion Detector (MTFD), a novel architecture that innovatively dissects the detection process into distinct tasks, addressing content and position through separate decoders. By utilizing assumed fake queries, the MTFD framework enables each decoder to operate under a presumption of known ancillary information, ensuring more specific and enriched interactions with the feature map. Experimental results affirm that this methodical separation followed by a deliberate fusion not only simplifies the task difficulty of the detection process but also augments accuracy and clarifies the details of each component, providing a fresh perspective on object detection in Transformer-based architectures.

Keywords: object detection; Transformer; DETR; decoder

check for updates

Citation: Wang, Y.; Ha, J.-E. Improved Object Detection with Content and Position Separation in Transformer. *Remote Sens.* **2024**, *16*, 353. https://doi.org/10.3390/rs16020353

Academic Editor: Pedro Melo-Pinto

Received: 9 November 2023
Revised: 5 January 2024
Accepted: 12 January 2024
Published: 16 January 2024

Copyright: © 2024 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (https://creativecommons.org/licenses/by/4.0/).

1. Introduction

Object detection, as one of the critical tasks in computer vision, has received extensive attention. It is increasingly important in many real-world applications, such as autonomous driving, monitoring systems, human–computer interaction, medical diagnosis, smart agriculture, and retail analysis. Its broad applicability underscores its importance, driving research efforts to refine and advance detection technology.

The object detection method in the early era mainly used hand-made components to extract features until the emergence and widespread use of convolutional neural networks (CNN) [1–4]. The method of object detection ushered in a paradigm shift. CNN-based frameworks, such as Faster R-CNN [2], YOLO [1,5], and Mask-RCNN [6], not only surpass traditional techniques in accuracy but also exhibit unprecedented efficiency, enabling real-time detection. However, although these models are groundbreaking and achieve good performance, they have also received extensive attention and substantial improvement. Still, they often suffer from complex object relationships and long-range dependencies.

Transformer [7] has achieved great success in NLP (Natural Language Processing) and has also been extended to computer vision [8,9]. With the promise of handling complex spatial relationships and feature dependencies, models such as the DEtection Transformer (DETR) [8] and its derivatives [10–14] set new benchmarks by leveraging the Transformer's inherent self-attention mechanism [7], which enables object detection architecture to open up the DETR paradigm. However, we argue that even in these state-of-the-art architectures, there is an implicit challenge—content and location information are constantly intermixed in attentional computations.

As mentioned earlier, the currently widely used DETR paradigm method usually regards the object detection [8,10–14] task as a single task. It integrates location and classification when querying the feature map so that the object detection in the middle of the model is always a mixed task of location and classification, as shown in Figure 1a,b.

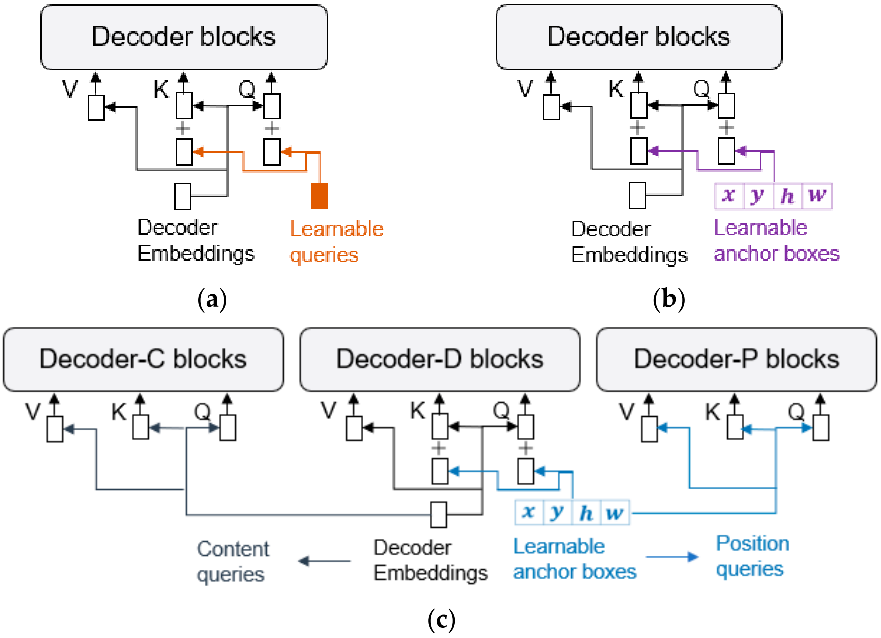


Figure 1. Comparison of DETR-like decoder architecture. (**a**) DETR decoder-query pipeline; (**b**) DAB decoder-query pipeline; (**c**) our decoder-query pipeline.

Compared with simple and single positioning and classification tasks, achieving precise positioning and detailed classification will be more difficult. Furthermore, the richness of content attributes, such as complex textures, patterns, and color gradients, is often obscured. Our work introduces a fine-grained multi-task approach to object detection, recognizing these potential pitfalls and focusing on distinguishing between localization and classification tasks, which can be regarded as learning location information and learning content information, as shown in Figure 1c. This ensures that tasks in different dimensions are optimized, simplifying the object detection task within the model without affecting other dimensions.

Inspired by conditional-DETR [11] and DAB-DETR [8], our approach starts from the joint learning stage to ensure a robust balance between localization and classification. Subsequently, a specialized content learning mechanism queries the content information of objects in the feature map and can identify subtle object attributes, thus providing a more comprehensive representation. At the same time, the positioning learning component adjusts the position in the feature map of interest and allows it to fine-tune the model in stages to ensure accuracy.

Our main contributions include the following:

(1) Propose a pioneering multi-task object detection framework of simple subtasks. We divide content query tasks and location query tasks and jointly optimize object detection tasks and subtasks, while subtasks do not affect each other.
(2) We design task-specific loss functions and iterative training methods.
(3) Comprehensive evaluations on leading benchmarks confirm our model's excellence in accuracy, object understanding, and scalability while increasing the comprehensibility of the model's internal components.

This paper further digs into the technical complexity, empirical validation, and comprehensive discussion, revealing how our approach defines the contemporary object detection field in a new light.

2. Related Works

2.1. Traditional Object Detection Methods

Traditional object detection methods mainly revolve around CNN-based architectures [1–4]. Mainstream methods such as Faster R-CNN [2] and YOLO [1,5] treat object detection as a combination of localization and classification. Specifically, Faster R-CNN [2] introduces a Region Proposal Network (RPN) to extract and classify regional proposals. At the same time, YOLO [1] tries to predict bounding boxes and class probabilities directly from feature maps at multiple scales. Although CNN-based methods show good performances on many benchmarks [15], they require further improvements in using semantics and spatial information in a global fashion, like Transformer-based methods [8].

2.2. Multi-Stage Object Detection

The multi-stage object detection method is an idea that has received a lot of attention. For example, Cascade R-CNN [16] uses a sequence of detectors trained with increasing IoU thresholds to refine detections progressively. However, the main focus of most multi-stage methods is still limited to the joint optimization of localization and classification. The nature of decomposing tasks for deeper refinement has not been fully explored, especially when addressing content attributes individually.

2.3. Transformer in Object Detection

Facing the problems of CNN architecture, Transformer architecture [7] has attracted attention due to its impressive performance in capturing long-range dependencies and establishing global spatial relationships. The core idea of Transformer is the self-attention mechanism, which allows the model to weigh input features differently based on context. Facebook AI's DEtection Transformer (DETR) [8] brings a paradigm shift in object detection. Rather than relying on region proposals or anchor boxes typical of CNN-based detectors [2,6], DETR treats object detection as a direct set prediction problem. It uses a fixed number of object queries and decodes them parallel to predict bounding boxes and class labels. The self-attention mechanism enables DETR to handle complex spatial relationships in images, making it suitable for scenes with many objects or occluded objects.

The success of DETR has spawned various spin-off works aimed at improving the model and adapting it to different scenarios. For example, DETR with Dynamic Convolution (DETR-DC) [17] incorporates convolution operations into the Transformer architecture, bridging the gap between CNNs and Transformers. Another notable one is Sparse RCNN [18], which utilizes a learnable proposal scheme that eliminates the need for handcrafted anchor boxes or fixed object queries.

Models like DETR [8,10–14,17] incorporate the Transformer's self-attention mechanism to focus on different image parts, effectively encapsulating spatial relationships and feature dependencies. This paradigm shift marks a departure from traditional region-based detection and opens the way for more nuanced approaches to the detection task. Although the DETR model has achieved remarkable results in object detection, we believe that the continuous interweaving of content and location information within the self-attention computation in the model is still an inherent challenge.

2.4. Content Learning and Localization in Vision

Content understanding is crucial in various fields of computer vision. Sun et al. [18] proposed the residual network (ResNet), which is a structure designed for deep networks. It solves the vanishing gradient problem in deep networks through residual connections, thereby better understanding and highlighting the image content. Carion et al. [8] utilize the Transformer structure to handle object detection, thoroughly combining the content and location information of the image. Zhou et al. [19] provide rich annotations, enabling the model to understand various objects and background content in the image; it also shows that understanding the content of images is very important in visual tasks. When we refer to "content," we are talking about the intrinsic properties of the objects in the image—the shape, texture, pattern, color, and other properties that define them, including the core characteristic properties of the objects in the image. We use these distinct attributes to understand the content more fully while often ignoring irrelevant details such as background.

Such content-centric analysis is not common in object detection. Most detection models [2,3,8,10] focus on identifying boundary regions of objects and classifying them. We believe that through a dedicated content learning stage, the model can theoretically gain a deeper understanding of each object's unique properties, thereby improving detection accuracy, especially in challenging scenes with occlusions, shadows, or changing lighting conditions.

On the other hand, localization is a fundamental aspect of object detection [1–6]. It involves identifying the precise location of objects in an image. This is usually achieved by predicting the bounding box of the encapsulated object. Methods such as bounding box regression in architectures such as Faster R-CNN [2] explicitly focus on improving these predictions to improve localization accuracy. References [10–12,20] also attach great importance to spatial position, emphasizing the importance of precise position-learning capabilities for accurate object detection.

In object detection, seamlessly combining content and localization learning is challenging. Object detection tasks do not explicitly separate these aspects. With the emergence of modern Transformer-based architectures [8,10–12,14] and the pursuit of better interpretability, there is a growing need for models that can simultaneously differentiate and optimize content and localization. Our work is also based on this idea, aiming to improve detection results by taking advantage of two aspects. Content understanding ensures detailed object representation, while accurate localization ensures precise object boundaries. Performing these two tasks separately and maintaining a harmonious balance between the two tasks may lead to superior object detection performance.

Also, there are good review papers on object detection [21–23]. Our method provides a separate multi-task training strategy. By decoupling the processes of localization and content learning, each task can receive the attention it deserves during training. The introduction of the content learning stage mainly acts as a bridge, exploiting richer feature interactions and capturing complex object properties. This holistic approach gives our model an advantage, as evidenced by improvements in accuracy, scalability, and component interpretability compared with contemporary methods.

In summary, we separate the object detection method into multiple subtasks in the model and observe the challenge of object detection from the perspective of multiple simple tasks. The method achieves promising results on various benchmarks, triggering deeper exploration, and has the potential to shape future detection strategies.

3. Proposed Method

We propose an approach consistent with DETR-like models that divides the object detection task into multiple subtasks, including content learning related to object information and classification (involving content knowledge of multiple targets) and position learning related to object localization (involving the positioning of multiple objects), and joint learning of positioning and content learning.

Based on the DAB-DETR model, we design and implement our tasks, thus facilitating the concurrent execution of the above functions within the integrated framework. In the following sections, we will first outline the process of our method, explain the combined learning of localization and classification in object detection, explain the structural design of content learning, and detail the scheme of position learning. We will explain the model's loss function and training strategy.

3.1. Overall Structure of the Proposed Model

As shown in Figure 2, the architecture of our proposed method is decoder-centric, where object detection is treated as multi-task learning. The decoder consists of three sub-modules related to content learning, position learning, and joint learning of content and position. Each sub-module is displayed with the same color in Figure 2.

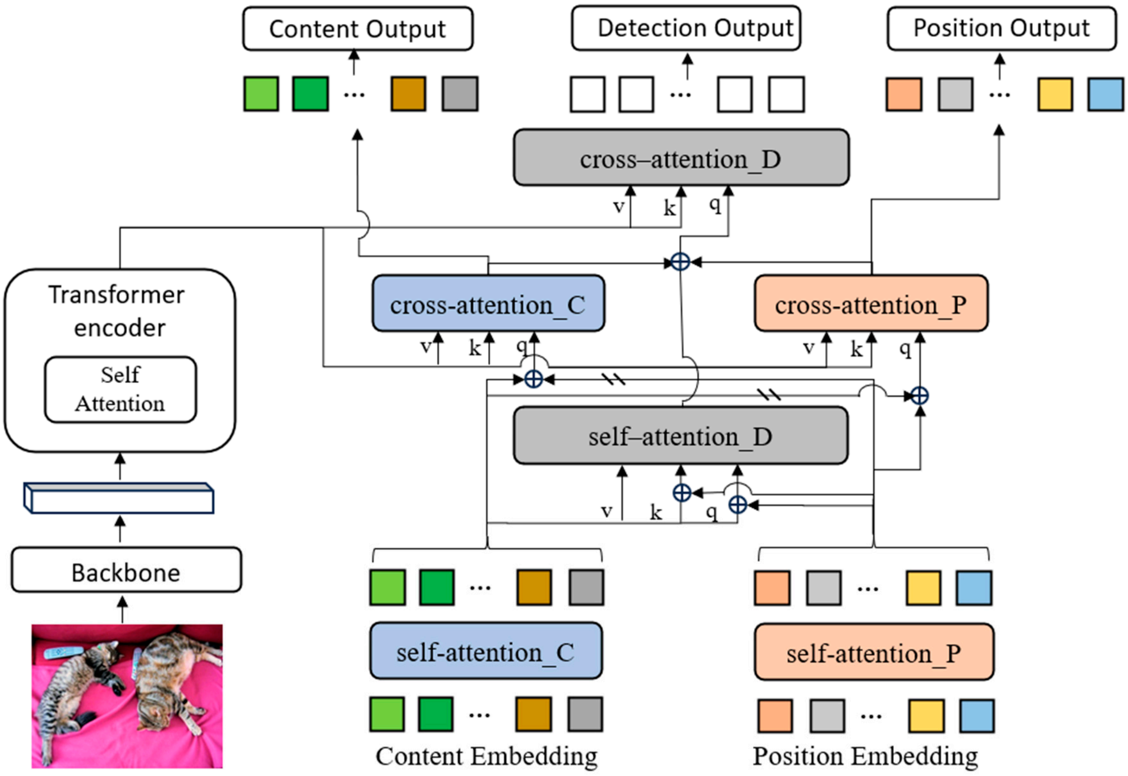


Figure 2. The pipeline of our proposed decoder method. We divide the decoder into three mutually optimized sub-task decoders (each one has different colors), corresponding to object detection, content learning, and position learning, represented as attention_D block, attention_C block, and attention_P block, respectively.

Content learning: For a more in-depth study, we separate the classification task in object detection as a content learning subtask, as self-attention-C and cross-attention-C blocks are shown in Figure 2. Our content learning mechanism is specifically tailored to emphasize the complex textures, patterns, and color gradients of objects. Unlike traditional models, our design emphasizes content attributes for object detection, enhancing the model's classification accuracy.

Position learning: Similarly, we separate the positioning task in target detection as a position learning sub-task, as self-attention-P and cross-attention-P blocks are shown in Figure 2. Position learning focuses on the area with objects in the feature map and ignores its specific category. Incorporating into the base layer fine-tunes the model's ability to predict the spatial location of objects in an image accurately.

Joint learning of content and position: The core of our method is the object detection decoder based on the DETR-like model. By querying, object queries are used to query the feature map and learn to obtain the information required for the object detection task, as self-attention_D and cross-attention_D blocks are shown in Figure 2.

In DETR paradigm architecture, the decoder's information query functions of positioning and classification are always integrated. In traditional object detection paradigms, merging these tasks often results in a precision or recall trade-off. The object detection task will also be more complicated than a single positioning and classification task. Therefore, by using DETR's decoder as the base layer of our model, we leverage the localization decoder and classification decoder as auxiliary layers, allowing us to treat localization and classification as separate but intertwined tasks. This approach ensures that the model understands the fundamental properties of object detection from the beginning and makes the model better interpretable.

The synergy of these three learning processes forms a cohesive end-to-end training workflow that follows a progression from simple to complex, ensuring that every aspect of object detection is addressed with precision and depth. The following subsections delve into the complexities and design philosophies underpinning each stage.

3.2. Content Learning Stage

In the development of object detection tasks, a richer understanding of image content becomes crucial. Our proposed content learning phase explicitly addresses this requirement. Although existence detection methods [8,10–12,14] focus on the spatial representation of objects, including no clear distinction between classification and localization within DETR-like models, the complex details of objects (including their unique features, textures, and relationships) are often not fully explored. Our approach aims to fill this gap by deploying a content query process.

Attention is a key mechanism in Transformer [7]. It operates on queries (Q), keys (K), and values (V), and it is defined as follows.

$$\text{Attention}(\text{Q}, \text{K}, \text{V}) = softmax\left(\frac{QK^T}{\sqrt{d_k}}\right)V \tag{1}$$

where d_k is the dimension of queries and keys.

As shown in Figure 2, the content learning stage is similar to the primary object detection stage. Also, it utilizes both the self-attention layer and cross-attention layer, which we define as $self\_attention\_c$ and $cross\_attention\_c$. This stage is to separate the classification in object detection into a single content learning task. Its core goal is not only to identify objects in the image but to conduct more in-depth research and provide a detailed understanding of its attributes (but does not include its specific location).

The encoder processes the image to achieve this, producing high-dimensional image features, denoted as T. At the same time, we deploy a set of learnable queries. Unlike traditional queries, these queries are mapped exclusively to the content space and are therefore called content queries Q_c. Their main functionality revolves around extracting the intricate details of an object independent of its spatial positioning.

However, to ensure that content queries can effectively focus on image content during the cross-attention stage, we introduce the concept of "fake location ground truth" Q_{pf}. It acts as a guide, providing auxiliary location information for content queries. But crucially, it remains irrelevant to training optimization, effectively blocking gradients. At this stage, we treat the latter as actual bounding box parameters by combining content queries with fake location ground truth elements. This strategy allows for a scenario where the model extracts underlying content details, especially object categories, assuming the location is known.

This process can be represented as follows.

$$Q_{c1} = s\_\ attn\ \_c(Q_c) \tag{2}$$

$$Q_{pf} = detach(Q_{p1}) \tag{3}$$

$$Q_{c2} = c\_attn\_c\left(Q_{c1} + Q_{pf},\ T,\ T\right) \tag{4}$$

where s_attn_c and c_attn_c mean the self-attention and cross-attention layers of the content decoder. Q_{pf} denotes the fake position queries extracted from position queries Q_p.

Therefore, the output Q_{c2} of this stage is considered to be the object category in the known region.

3.3. Position Learning Stage

While content recognition is crucial in fully understanding objects, accurate object localization remains the cornerstone of any effective object detection system, and learning how and obtaining more accurate location information is also a highlight of the improvement plans of other DETR-like models [10,11]. The position learning stage aims to improve the accuracy of spatial recognition and localization of objects in images, focusing on understanding the location of objects in the image. As shown in Figure 2, similar to the content learning stage, our location learning stage follows a structure that includes self-attention and cross-attention mechanisms. However, its core difference from the content learning stage is that it focuses exclusively on the spatial properties of objects without any interference from content details.

The inputs to this stage are high-dimensional image features T from the encoder and a set of learnable positional queries Q_p, which are aimed at finding and refining the spatial coordinates of objects within the image. To further guide these location queries, we introduce the concept of "fake content ground truth" Q_{cf} as auxiliary category information. However, similar to its counterpart in the content learning phase, Q_{cf} remains irrelevant to training optimization and only serves as a guide.

Essentially, by mixing Q_p with Q_{cf}, the latter is treated as the actual category label. Therefore, this stage effectively localizes multiple known objects by querying the location of that object in the image features under the assumption that a specific object exists in the known image.

Similarly, for these location queries to perform optimally, they need to undergo a self-attention mechanism that allows each query to adaptively adjust its focus based on insights gleaned from other queries, which we defined as $self\_attention\_p$. This adaptive refinement helps reduce overlap and redundancy.

The following steps involve the cross-attention mechanism, defined as *cross_attention_p*. Here, refined location queries Q_p are fused with fake content labels Q_{cf} and are then interacted with image features T, allowing each query to focus on a specific spatial region of the image. This interaction brings sharper spatial focus, ensuring higher accuracy in predicting bounding boxes.

After the cross-attention stage, similar to the object detection stage, the location query is transformed to produce predictions of object space boundaries. These predictions are represented in the form of bounding box coordinates.

The procedural steps can be summarized by the following equation:

$$Q_{p1} = s\_\ attn\ \_p(Q_p) \tag{5}$$

$$Q_{cf} = detach(Q_{c1}) \tag{6}$$

$$Q_{p2} = c\_attn\_p\left(Q_{p1} + Q_{cf},\ T,\ T\right) \tag{7}$$

Therefore, the output Q_{p2} of this stage is used to locate the area where objects exist in the image features.

3.4. Joint Cotent and Position Learning Stage

We describe object detection as a joint learning process of localization and classification. This stage builds on the structural framework of other decoders similar to the DETR model [10,11,14], employing a collaborative learning approach to localize and classify objects in images, that we defined as decoder_D, which includes self-attention_D and cross-attention_D, as shown in Figure 2.

The encoder first processes the input image and generates a set of high-dimensional image features, denoted as T. Meanwhile, the decoder uses several predefined object queries Q as input, each Q potentially corresponding to an object in the image. These queries are abstract representations, and they are converted into localization as a bounding box and classification of object category predictions.

Self-attention mechanism: Object queries undergo a self-attention mechanism before interacting with image features. This step allows queries to be correlated with each other, ensuring that predictions do not overlap or become connected in the feature space. We can also argue that allowing queries to talk or communicate with each other creates an environment for them to interpret the scene together.

Cross-attention mechanism: After self-attention, cross-attention is calculated between the output queries as Q, and the image feature T. This process allows each query to focus on a specific area of the image, focusing on particular details, textures, patterns, and also spatial features. The result of this mechanism is a refined set of features for each query, tuned and predictively prepared for their respective focus areas.

Prediction phase: As features are enriched, each query undergoes a series of transformations to predict the location and classification of its corresponding potential object. Locations are bounding boxes with coordinates, and classifications are probability distributions over predefined object categories. During the inference phase, our network structurally uses detection outputs to predict outcomes. However, the detection output utilizes a fusion mechanism to integrate information from the content output and position output. Specifically, the content output provides class information about detected objects, while the location output provides spatial information about the location of these objects. The detection output then combines these two pieces of information to produce a final confidence score and bounding box for object detection. This fusion approach ensures that the model can utilize the complete information learned through separated learning, content, and location information during inference to improve detection accuracy.

This collaborative learning mechanism ensures that with each iteration, the model can better recognize the exact location and category of objects in the image. Over time, through backpropagation and optimization routines, learnable object queries fine-tune themselves to match real-world objects, ensuring accurate object detection.

This stage is the decoder process of the DETR-like model. During the training process, the queries used for positioning and classification are fused and participate in the calculation to achieve basic object detection tasks. The use mechanism of query and the decoding mechanism of queries in this process have laid the foundation for introducing more detailed subtask implementation.

3.5. Training Strategy and Loss Computation

We adopt an end-to-end training strategy, meaning that all components (including object detection, content learning, and location learning) are trained simultaneously, enabling queries to detect objects in images while enhancing object location and category awareness. In the content learning stage, the model focuses on the image content to ensure that the content query can capture the content information of the object. While in the position learning stage, the model focuses on the possible object regions in the image, ensuring that the position query can accurately capture the position information of the foreground region.

Considering that our model has to handle multiple tasks, we use a multi-task learning loss function to train the model. Following the approach of DETR-like models, we first find the best bipartite match between the predictions produced by a detection query and the ground truth and design a loss function accordingly. According to our task design, the loss function mainly consists of object detection loss, content loss, and positional loss. Among them, detection loss also includes positioning loss and content loss. The three outputs of the model correspond to object query, content query, and location query. We will optimize each output according to these three losses and adjust the proportion of each loss to obtain the best performance. Specifically, the localization loss combines $L1$ loss and $GIOU$ loss to quantify the difference between predicted and ground truth bounding boxes. In addition, the classification loss uses focal loss. Therefore, the loss function is formed as follows:

$$L_{total} = \alpha L_{detection} + \beta L_{content} + \gamma L_{position} \tag{8}$$

where α, β, and γ are the weights of each loss, which is used to adjust the importance of each loss in the overall loss. The choice of these weights is usually tuned based on performance on the validation dataset. After experimental verification, we set α, β, and γ to 2, 2, and 1, respectively. For the specific hyperparameter settings in detection, we follow the settings in DAB-DETR [10].

4. Experimental Results

4.1. Datasets

We evaluate with the COCO 2017 object detection dataset, split into train2017 and val2017. MS-COCO is composed of 160K images with 80 categories. These images are divided into train2017 with 118K images, val2017 with 5K images, and test2017 with 41K images. Following the common practice [15], we report the standard mean average precision (AP) results on the COCO validation dataset under different IoU thresholds and object scales.

4.2. Implementation Details

Following the approach of DAB-DETR [10], we utilize various ResNet [24] models, which have been pre-trained on ImageNet, as our backbone. Regarding hyperparameters, we use the same values in DAB-DETR's configuration by employing a 6-layer Transformer encoder and a 6-layer Transformer decoder with a hidden dimension of 256.

Our decoder includes three self-attention layers and three cross-attention layers as a block, corresponding to object detection, content learning, and position learning, as shown in Figure 2.

4.3. Object Detection Performance Comparison

Our proposed approach is of plug-and-play type. Therefore, we use our proposed method to insert several popular DETR-like models and compare them to evaluate the effectiveness of our method in the object detection task.

Our goal is to observe and validate the effectiveness of our model in improving the performance of object detection tasks and the role of its separated components in explaining the object detection task during model decoding.

Table 1 shows the comparison results between various DETR-like models after using our method. For a fair comparison, we use the same parameter settings as each method, and the difference only exists in whether to insert our proposed subtask module.

Table 1. Results for our method on baselines and comparison between baselines. Each one has the same settings as the baseline.

Model	Epochs	AP	AP50	AP75	APS	APM	APL	Params
DETR [8]	500	42.0	62.4	44.2	20.5	45.8	61.1	41 M
DAB-DETR [10]	50	42.2	63.1	44.7	21.5	45.7	60.3	44 M
+Ours	50	42.8 (0.6 ↑)	63.3 (0.2 ↑)	45.4 (0.7 ↑)	22.4 (0.9 ↑)	46.1 (0.4 ↑)	61.1 (0.8 ↑)	45 M
	60	42.9	63.5	45.6	23.2	46.3	60.9	45 M
	70	43.1	63.5	45.9	23.7	46.3	61.1	45 M
	80	43.2	63.8	46.3	23.5	46.4	61.4	45 M
	100	43.4	63.9	46.4	24.1	46.3	62.1	45 M
	110	43.2	63.8	46.1	24.1	46.1	61.9	45 M
DAB-Deformable-DETR [10]	50	46.9	66.0	50.8	30.1	50.4	62.5	44 M
+Ours	50	47.8 (0.9 ↑)	67.1 (1.1 ↑)	51.3 (0.5 ↑)	30.7 (0.6 ↑)	51.0 (0.6 ↑)	62.6 (0.1 ↑)	47 M
DINO [12]	36	50.9	69.0	55.3	34.6	54.1	64.6	47 M
+Ours	36	51.2(0.3 ↑)	69.0(0.0 ↑)	55.8(0.5 ↑)	34.4(0.2 ↓)	54.5(0.4 ↑)	64.7(0.1 ↑)	51 M

Our approach gives superior performance compared with each model, like in DAB-DETR, with a 0.6% improvement in Average Precision (AP) compared with the baseline. Specifically, for small objects, our method attains a 0.9% enhancement in performance, indicating that our approach offers improved detection efficacy on small objects.

This kind of comparison validates our approach's effectiveness and points out areas of potential improvement that simplify the model's task internally. Through this rigorous analysis, we can identify the strengths of our model and areas where further refinements could be beneficial.

4.4. Ablation Experiments

We conduct a series of component ablation experiments to gain insight into the contribution of individual components in our proposed method to the performance.

Baseline model: We first need to determine the baseline model, which is trained without any of our specific added components, that is, the original model. For example, if we conduct a series of ablation implementations based on DAB-DETR, the baseline model is the DAB-DETR model. This provides a reference point that allows us to evaluate the performance improvement of each component.

Single component addition:

Baseline plus content learning: this configuration only added a content learning phase to the baseline model.

Baseline plus position learning: we add a position learning phase to the baseline model in this setting.

Baseline plus content learning and position learning: this allows us to observe performance changes when the two main components are present simultaneously and thus evaluate their synergy.

We trained and evaluated the model in each configuration, recording key performance metrics, as shown in Table 2.

Table 2. Ablation experiments of components.

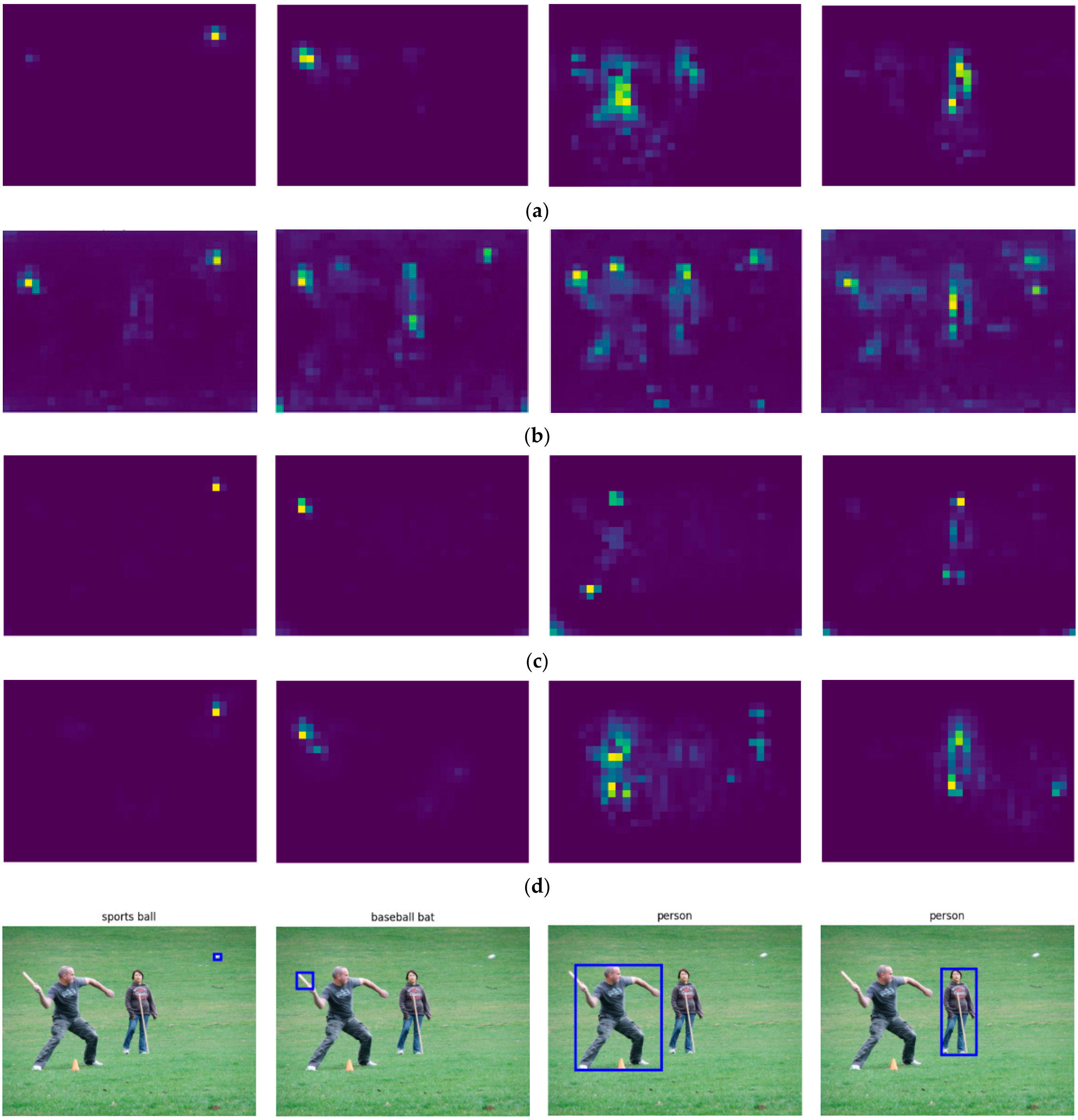
	B	**B + C**	**B + P**	**B + C + P**	**AP**	**AP50**	**AP75**	**APS**	**APM**	**APL**
a	√				42.2	63.1	44.7	21.5	45.7	60.3
b		√			42.1	63.5	44.3	22.1	45.8	60.4
c			√		42.5	63.6	44.9	22.5	46.0	60.6
d				√	42.8	63.3	45.4	22.4	46.1	61.1

The results show that the content and position learning stages are necessary for performance improvement, where position learning contributes most significantly. Through these ablation experiments, we confirm the effectiveness and necessity of the component design of our method.

4.5. Attention Visualization of Decoders

To more intuitively understand the decoding mechanism of feature maps by different decoders in our model, we followed DETR attention visualization methods [10,25]. Also, we conducted a series of attention map visualization experiments. This method uses a 'hook' to hook the attention weights of the decoder layer and then visualizes each detected object. Through its visualization results, we can analyze which part of the image is being looked at by the decoder's object queries. The part where a query is looking at can acquire more attention weight, and then the model uses this to predict specific bounding boxes and classes. We use this visualization method to observe which parts of the image the queries indicated by our different decoders are looking at simultaneously.

Cross-attention visualization of DAB-DETR: We first show cross-attention maps in the original DAB-DETR model, as shown in Figure 3a. This provides a baseline for comparison, revealing the focus of the decoder's attention without any specific improvement component.


(**a**)
(**b**)
(**c**)
(**d**)

(**e**)

Figure 3. Visualizing cross-attention map of decoders for predicted objects and corresponding object queries (**a**) DAB-DETR decoder attention; (**b**) content decoder attention; (**c**) position decoder attention; (**d**) detection decoder attention; (**e**) target object with a blue rectangle.

Cross-attention visualization of content learning: We first analyzed the cross-attention weight in the content learning stage. As shown in Figure 3b, for content query, we can see that the model mainly focuses on the core part of the object in the image to capture the object content and ignores the background.

Cross-attention visualization of position learning: In the visualization of the position learning stage, as shown in Figure 3, second row, the model's attention is more focused on the boundary area of the object, and relatively less attention is paid to the object's interior.

This proves that the model successfully learns the location information of the object at this stage.

Cross-attention map visualization of joint content and position learning: Figure 3d is the decoder attention map that combines the content learning and position learning stages, revealing the synergistic effect when these two components work at the same time and their impact on the focus of attention and target detection performance of the original model's detection queries.

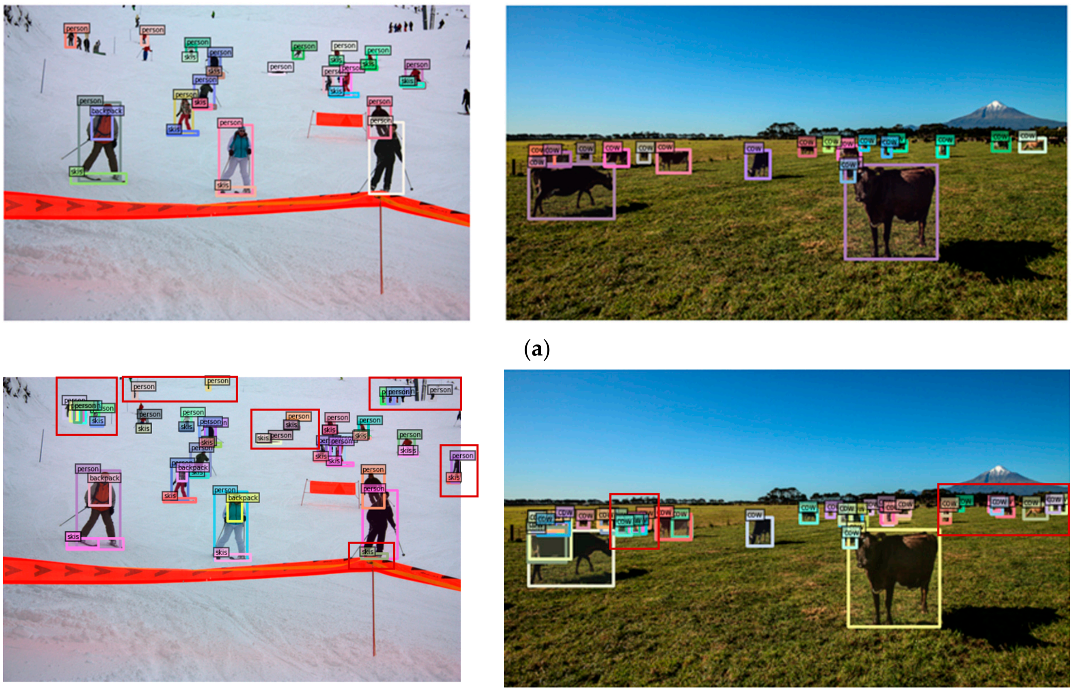
Object detection visualization: We also show the object detection results predicted by the decoder, outlined by a blue rectangle, as shown in Figure 3e. This provides an intuitive display of results. At the same time, the queries that predict the results correspond to the previous attention map. For example, in the query that predicts 'sports ball', the area of focus is shown in the first attention map in the first four rows.

From the above visualizations, we can see that although all decoders use the cross-attention mechanism, they pay attention to different parts of the image at different learning stages. This further proves that the attention distributions of the three decoders we designed are different when dealing with other tasks and are consistent with their design goals.

4.6. Experimental Results on Small Objects

In real-life scenes, detecting small objects in images is challenging because the number of pixels is limited, details are often difficult to capture, and there are higher requirements for image understanding.

The proposed model learns and understands images' content from multiple aspects. Through the performance comparison of Table 1, we can see that one of the achievements of our proposed model is that compared with the baseline, its performance in detecting small objects has been improved and gives better enhancement in the context of the COCO dataset. Figure 4 shows the comparison results of our model and baseline methods on images containing small objects from the COCO dataset.



(**a**)

(**b**)

Figure 4. Comparison of detection results on sample images containing small objects from the COCO dataset. We used the visualization method in [26] to output prediction results and comparisons. (**a**) Baseline method [10]. (**b**) Proposed method: the red box is the small object area that our proposed method detected but was missing in the baseline method.

The detection results in Figure 4 show that our model provides better detection and localization results for small objects on the COCO dataset. At the same time, the baseline method [10] either misses or inaccurately locates some of these objects, such as those in the red box in Figure 4b. Compared with baseline results, our method better detects small objects in the red box area where smaller objects exist without specifically training or fine-tuning the small object dataset. Usually, the detection of small objects requires a better understanding of the details of the image [23], so by analyzing the above small object detection results, our model has a better understanding of the content and details of the image, which is due to our auxiliary independent learning and emphasis on the content information and location information of the image. Also, reflecting the separate stages of content learning and location learning in our model (as described in Sections 3.3 and 3.4) enhances the model's ability to capture complex details of the object.

Our results on the COCO dataset show the promise of our improved method to enhance small object detection, as we believe it has multiple layers and a deeper understanding of image details. While we did not train specifically on a dedicated small object dataset, nor were we designed to generalize across a variety of datasets, our results demonstrate the model's potential in the context in which it was trained. Future work could study its adaptability and efficacy on datasets dedicated to small object detection or investigate its design for generalization capabilities.

5. Conclusions

In this study, we propose an innovative multi-component object detection technique that seamlessly integrates the joint learning stage of localization and classification, the content learning stage, and the location learning stage within the DETR model framework. Our approach divides object detection into three subtle stages. Starting with joint learning of positioning and classification lays a strong premise for in-depth exploration of subsequent content and accurate learning of object locations. We are introducing a decoder structure designed explicitly for understanding object content information and using an independent content learning mechanism to enable the model to capture objects' complex details and characteristics meticulously. Furthermore, our uniquely designed position learning architecture emphasizes capturing precise object locations, ensuring the model can identify object locations in multi-faceted scenarios. Through testing on standard benchmarks, our proposed method consistently exhibits excellent performance in object detection tasks, significantly outperforming established baselines. We believe our approach reveals potential improvements in object detection and demonstrates the effectiveness of refined research directions in this area.

However, we acknowledge that our method, like all research, has limitations. First, although our model performs well in benchmarks, its generalization to multiple datasets and to real-world scenarios needs further exploration. Then, there is also the issue of computational efficiency in processing datasets, especially large-scale datasets, because our multi-stage learning mechanism can be resource-intensive. Moving forward, we aim to address these limitations by optimizing our model's computational architecture, thereby enhancing its efficiency. We will also investigate the application of our method to broader datasets and real-world scenarios to ensure robustness and versatility. Additionally, we plan to delve into integrating unsupervised learning techniques to improve the model's performance in less controlled environments, paving the way for more generalizable object detection systems. In conclusion, our research focuses on more refined and detailed object detection strategies, providing new perspectives for improving accuracy and reliability in this field.

Author Contributions: Conceptualization, J.-E.H. and Y.W.; methodology, J.-E.H. and Y.W.; software, Y.W.; validation, J.-E.H. and Y.W.; formal analysis, J.-E.H. and Y.W.; writing—original draft preparation, Y.W.; writing—review and editing, J.-E.H.; visualization, Y.W.; supervision, J.-E.H.; project administration, J.-E.H.; funding acquisition, J.-E.H. All authors have read and agreed to the published version of the manuscript.

Funding: This study was supported by the Research Program funded by the SeoulTech (Seoul National University of Science and Technology).

Data Availability Statement: Data are contained within the article.

Conflicts of Interest: The authors declare no conflicts of interest.

References

1. Redmon, J.; Farhadi, A. YOLOv3: An Incremental Improvement. *arXiv* **2018**, arXiv:1804.02767.
2. Ren, S.; He, K.; Girshick, R.; Sun, J. Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks. *IEEE Trans. Pattern Anal. Mach. Intell.* **2017**, *39*, 1137–1149. [CrossRef] [PubMed]
3. Tian, Z.; Shen, C.; Chen, H.; He, T. FCOS: Fully Convolutional Onestage Object Detection. In Proceedings of the 2019 IEEE/CVF International Conference on Computer Vision (ICCV), Seoul, Republic of Korea, 27 October–2 November 2019.
4. Lin, T.-Y.; Goyal, P.; Girshick, R.; He, K.; Dollar, P. Focal Loss for Dense Object Detection. *IEEE Trans. Pattern Anal. Mach. Intell.* **2020**, *42*, 318–327. [CrossRef] [PubMed]
5. Ge, Z.; Liu, S.; Wang, F.; Li, Z.; Sun, J. YOLOX: Exceeding YOLO Series in 2021. *arXiv* **2021**, arXiv:2107.08430.
6. He, K.; Gkioxari, G.; Dollar, P.; Girshick, R. Mask R-CNN. In Proceedings of the IEEE International Conference on Computer Vision, Venice, Italy, 22–29 October 2017; pp. 2961–2969.
7. Vaswani, A.; Shazeer, N.; Parmar, N.; Uszkoreit, J.; Jones, L.; Gomez, A.N.; Kaiser, Ł.; Polosukhin, I. Attention is All You Need. In Proceedings of the Advances in Neural Information Processing Systems 30 (NIPS 2017): Annual Conference on Neural Information Processing Systems 2017, Long Beach, CA, USA, 4–9 December 2017.
8. Carion, N.; Massa, F.; Synnaeve, G.; Usunier, N.; Kirillov, A.; Zagoruyko, S. End-to-End Object Detection with Transformers. In Proceedings of the European Conference on Computer Vision, Glasgow, UK, 23–28 August 2020; pp. 213–229.
9. Dosovitskiy, A.; Beyer, L.; Kolesnikov, A.; Weisssenborn, D.; Zhai, X.; Unterthiner, T.; Dehghani, M.; Minderer, M.; Heigold, G.; Gelly, S.; et al. An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale. *arXiv* **2020**, arXiv:2010.11929,.
10. Liu, S.; Li, F.; Zhang, H.; Yang, X.; Qi, X.; Su, H.; Zhu, J.; Zhang, L. DAB-DETR: Dynamic Anchor Boxes are Better Queries for DETR. *arXiv* **2022**, arXiv:2201.12329.
11. Meng, D.; Chen, X.; Fan, Z.; Zeng, G.; Li, H.; Yuan, Y.; Sun, L.; Wang, J. Conditional DETR for Fast Training Convergence. In Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV), Montreal, QC, Canada, 10–17 October 2021; pp. 3651–3660.
12. Zhang, H.; Li, F.; Liu, S.; Zhang, L.; Su, H.; Zhu, J.; Ni, L.M.; Shum, H.-Y. DINO: DETR with Improved Denoising Anchor Boxes for End-to-End Object Detection. *arXiv* **2022**, arXiv:2203.03605.
13. Zhu, X.; Su, W.; Lu, L.; Li, B.; Wang, X.; Dai, J. Deformable DETR: Deformable Transformers for End-to-End Object Detection. In Proceedings of the Ninth International Conference on Learning Representations, Virtual Event, Austria, 3–7 May 2021.
14. Li, F.; Zhang, H.; Liu, S.; Guo, J.; Ni, L.M.; Zhang, L. DN-DETR: Accelerate DETR Training by Introducing Query Denoising. *arXiv* **2022**, arXiv:2203.01305.
15. Papers with Code—Coco Test-Dev Benchmark in Object Detection. Available online: https://paperswithcode.com/sota/object-detection-on-coco (accessed on 30 October 2023).
16. Cai, Z.; Vasconcelos, N. Cascade R-CNN: Delving into High Quality Object Detection. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–23 June 2018; pp. 6154–6162.
17. Dai, X.; Chen, Y.; Yang, J.; Zhang, P.; Yuan, L.; Zhang, L. Dynamic DETR: End-to-End Object Detection with Dynamic Attention. In Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV), Montreal, BC, Canada, 11–17 October 2021; pp. 2988–2997.
18. Sun, P.; Zhang, R.; Jiang, Y.; Kong, T.; Xu, C.; Zhan, W.; Tomizuka, M.; Li, L.; Yuan, Z.; Wang, C.; et al. Sparse R-CNN: End-to-End Object Detection with Learnable Proposals. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Nashville, TN, USA, 20–25 June 2021; pp. 14454–14463.
19. Zhou, B.; Zhao, H.; Puig, X.; Fidler, S.; Barriuso, A.; Torralba, A. Scene Parsing Through ADE20K Dataset. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Honolulu, HI, USA, 21–26 June 2017; pp. 633–641.
20. Zhou, X.; Wang, D.; Krähenbühl, P. Objects as Points. *arXiv* **2019**, arXiv:1904.07850.
21. Sirisha, M.; Sudha, S.V. A Review of Deep Learning-based Object Detection Current and Future Perspectives. In Proceedings of the Third International Conference on Sustainable Expert Systems, Nepal, 9–10 September 2023; Springer: Berlin/Heidelberg, Germany, 2023; pp. 937–951.
22. Li, Y.; Miao, N.; Ma, L.; Shuang, F.; Huang, X. Transformer for Object Detection: Review and Benchmark. *Eng. Appl. Artif. Intell.* **2023**, *126*, 107021. [CrossRef]
23. Rekavandi, A.M.; Rashidi, S.; Boussaid, F.; Hoefs, S.; Akbas, E.; Bennamoun, M. Transformers in Small Object Detection: A Benchmark and Survey of State-of-The-Art. *arXiv* **2023**, arXiv:2309.04902.
24. He, K.; Zhang, X.; Ren, S.; Sun, J. Deep Residual Learning for Image Recognition. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Las Vegas, NV, USA, 27–30 June 2016; pp. 770–778.

25. Available online: https://colab.research.google.com/github/facebookresearch/detr/blob/colab/notebooks/detr_attention.ipynb (accessed on 30 October 2023).
26. Available online: https://github.com/IDEA-Research/DAB-DETR/blob/main/inference_and_visualize.ipynb (accessed on 30 October 2023).

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.