IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Date of publication xxxx 00, 0000, date of current version xxxx 00, 0000.

Digital Object Identifier 10.1109/ACCESS.2023.1120000

An Enhanced End-to-End Object Detector for Drone Aerial Imagery

QUAN YU[1,2], QIANG TONG[1,2], LIN MIAO[1,2], LIN QI[3], and XIULEI LIU[1,2]
[1]Beijing Advanced Innovation Center for Materials Genome Engineering, Beijing Information Science and Technology University, Beijing 100101, China
[2]Laboratory of Data Science and Information Studies, Beijing Information Science and Technology University, Beijing 100101, China
[3]School of Economics and Management, Beijing Information Science and Technology University, Beijing 100101, China

Corresponding author: Xiulei Liu (e-mail: liuxiulei@bistu.edu.cn)

This work was supported in part by the National Key R&D Program of China (2021YFB2600600), R&D Program of Beijing Municipal Education Commission (KM202111232003), and the Research Fund of Beijing Information Science and Technology University.

ABSTRACT DETR-like detectors have gained increasing popularity in current practical applications. However, we observe that their pipeline still suffer from several challenges, including unbalanced distribution of positive and negative samples, low-quality initial prediction boxes, and unreasonable gradient structure in the decoding stage. These challenges hinder both the convergence speed and detection performance of the model. To address these issues, we propose an enhanced DETR-like model called EM-DETR. It combines three innovative methods, including Dynamic Groups Assignment, Mixed Query Re-Selection, and Look Forward Stage. Dynamic Groups Assignment employs adaptive parameters to balance the number of positive and negative samples, providing more effective supervision signals for ground-truth boxes. Mixed Query Re-Selection utilizes high-quality bounding boxes regressed by subnet to initialize decoder queries, offering superior prior information for the decoder. Look Forward Stage introduces a more rational gradient structure which eliminates inter-layer information bias between decoders. We conduct extensive experiments to evaluate the effectiveness of our proposed method. On VisDrone2021-DET, EM-DETR with ResNet50 achieved 23.9% AP after 12 epochs of training. Compared to the baseline, this represents an improvement of 4.7% AP. Moreover, the excellent performance of EM-DETR on AI-TOD and Crowdhuman proves the generalization capability of the proposed method.

INDEX TERMS drone aerial imagery, object detection, end-to-end object detector, Detection Transformer

I. INTRODUCTION

THE advancement of technology has made the application of drones in daily life increasingly common. This has led to drone aerial images object detection becoming a research focus in the field of computer vision [1]. The mainstream detectors in the current drone aerial images object detection are conventional detectors [2]–[6] and its variants [7]–[9]. However, the hand-designed components (such as NMS and anchor boxes) that contribute to the exceptional performance of these detectors introduce a high level of complexity to the model pipeline.

In recent years, researchers have increasingly focused on DETR-like (Detection Transformer-like) models to simplify the detection process, making significant progress in various fields such as steel surface defect detection [10], early fire warning [11], remote sensing object detection [12], smart city intelligent transportation [13] and vehicle detection [14]. Unlike classical detectors, DETR-like models do not rely on Non-Maximum Suppression (NMS) for post-processing, which allows them to avoid the instability in speed and accuracy that NMS introduces. This makes DETR-like models more advantageous in practical applications. As a result, DETR-like models have undoubtedly become a key direction for the development of fast and accurate object detectors. The recent introduction of RT-DETR [15] further highlights this point. It not only surpasses YOLOv7 [16] and YOLOv8 [17] in detection speed, but also outperforms YOLOv9 [18] and YOLOv10 [19] in accuracy when processing large-scale images [20]. This further confirms the immense potential of DETR-like models in efficient and precise object detection. Based on these advantages, we advocate for the adoption of DETR-like object detectors for drone aerial imagery object detection.

DETR [21] undertakes a fresh perspective on the object detection from the viewpoint of point sets. It establishes associations between bounding boxes and ground truth by bipartite matching without NMS. This innovative model enables object detectors to alleviate the limitations imposed by hand-designed components. As the first truly end-to-end model, DETR has demonstrated great potential in object detection. Subsequently, researchers have proposed various methods to improve DETR, such as introducing attention mechanisms with lower computational complexity [22], [23], introducing auxiliary branches for one-to-many label assignment [24], [25], and enhancing the prior information used for initializing the decoder [26], [27]. Among these methods, the fusion of low-complexity attention and auxiliary branches for one-to-many label assignment has emerged as the popular approach [28]–[30]. This approach has demonstrated significant efficacy across numerous datasets.

However, we observe that DETR models with auxiliary branch still face several challenges in their pipelines. First, their label assignment strategy fails to achieve a balance between positive and negative samples. Although introducing auxiliary branch increases the number of positive samples, it also brings in an equivalent number of negative samples. As a result, the ratio of positive to negative samples remains unchanged and the gap between their quantities even widens. This imbalance can potentially affect the stability of model training. Second, the quality of initial prediction boxes is low. Existing models predominantly use the TopK method to select bounding boxes. However, with the substantial increase in the number of decoder queries, many low-score bounding boxes are also included in the initialization queue. These low-quality bounding boxes not only fail to accelerate the convergence of the decoder but also make it difficult for the model to learn the relationship between prediction boxes and ground truth. Finally, there is information bias among the decoder layers. In the decoding stage, different layers of the decoder serve distinct purposes. However, current methods apply the same gradient across all decoder layers during optimization, which introduces inter-layer information bias. This bias forces the model to require additional iterations to correct it, thereby slowing down the training convergence speed.

To address these issues, we propose an enhanced DETR-like model, named EM-DETR. This model incorporates three innovative methods to improve the convergence speed and detection performance. First, we introduce an adaptive label assignment strategy called Dynamic Groups Assignment. This strategy allocates appropriate supervision signals to each ground truth based on the positive-to-negative sample ratio. It effectively increases the number of positive samples during the decoding stage while dynamically adjusting the positive-to-negative sample ratio, enhancing the model's robustness and training stability. Second, we propose Mixed Query Re-Selection, a decoding query initialization method. This approach selects high-quality bounding boxes regressed by a subnetwork to initialize decoder queries. This enables the main branch to learn decoding features more effectively based on precise positional information, while the auxiliary branch provides more accurate gradient optimization directions. Third, we introduce a novel iterative box refinement method called Look Forward Stage. This method divides the decoding stage into a coarse-grained localization phase and a fine-grained detection phase. By separating the gradient structures of these two phases, it effectively eliminates inter-layer information bias in the decoder, thereby accelerating the model's convergence. Additionally, we incorporate Distinct Query Selection [31], a method designed for filtering decoding queries between layers. The purpose is to remove redundant prediction boxes between decoder layers and clarify the optimization objectives for bipartite matching.

We summarize the contributions of this paper as follows:

- We propose an adaptive label assignment strategy named Dynamic Groups Assignment. It automatically calculates the supervision quantity for each ground truth box based on the ratio of positive to negative samples, enabling more effective supervision of them.
- We propose a decoding query initialization method named Mixed Query Re-Selection. It selectively initializes decoding queries using high-quality bounding boxes regressed from subnet, thereby providing a more comprehensive prior information for the decoding phase.
- We propose a new gradient structure called Look Forward Stage. This structure adjusts the gradient propagation style within the decoder. It ensures close connectivity between layers while eliminating information bias.
- We integrated the above three methods to propose an enhanced DETR-like model named EM-DETR. Through extensive ablation studies, we validated the effectiveness of these different approaches. As a result, EM-DETR achieved state-of-the-art performance on VisDrone2021-DET [32], surpassing all DETR-like detectors, as well as some conventional object detectors like Cascade RCNN [33] and RetinaNet [3]. Notably, EM-DETR demonstrated varying degrees of improvement over the baseline on AI-TOD and CrowdHuman datasets, indicating the strong generalization capacity of our proposed methods.

II. RELATED WORK

A. OBJECT DETECTION FOR DRONE AERIAL IMAGERY

Presently, drone aerial imagery object detection primarily relies on conventional object detectors and their variants. Based on the processing pipeline of object detection, classical object detectors can be divided into two types: one-stage and two-stage detectors. One-stage object detectors such as YOLO (You Only Look Once) [5], [16], [34]–[37] and SSD (Single Shot MultiBox Detector) [38] directly predict object positions and categories in a single network. They typically use methods such as sliding a fixed-size window over the image or utilizing dense anchor points for predictions. These methods are fast but might perform less effectively in cases of object overlap or significant size variations. On the other hand, two-stage object detectors like Faster R-CNN [2] and Mask R-CNN [39] divide the object detection task into two stages. In the first stage, they use RPN(region proposal network) to generate candidate object regions that are likely to contain targets. In the second stage, these candidate regions are further processed for object classification and localization. While two-stage methods tend to be more accurate, they are slower to process than one-stage methods.

Furthermore, researchers are inclined towards refining the YOLO series models to enhance detection accuracy or inference speed. TPH-YOLOv5 [40] addresses unique challenges in drone aerial object detection, such as motion blur induced by high-altitude flights and dense objects within images, by incorporating self-attention modules into the detection head of YOLOv5. Building upon TPH-YOLOv5, TPH-YOLOv5++ [41] introduces an additional detection head and sparse local attention modules to detect tiny objects, significantly reducing model computational costs. Recently, with the release of YOLOv7 [16], several improvements based on YOLOv7 have gradually gained attention. Efficient YOLOv7-Drone [8] significantly enhances the efficiency and accuracy of drone aerial imagery object detection by improving the hierarchical detection head levels and employing target-guided mask strategy. MS-YOLOv7 [7] introduces a novel detection head network with CBAM convolutional attention modules to extract features at different scales, thereby enhancing detection accuracy across various scales.

B. DETR AND ITS VARIANTS

DETR [21] is an end-to-end object detection model based on the Transformer structure, proposed by Facebook AI Research. The detector comprises three principal components: a backbone network for feature extraction, multiple layers of Transformer encoders, and multiple layers of Transformer decoders. Initially, the detector extracts feature maps from the input image using the backbone. Subsequently, these feature maps are transformed into fixed-length embeddings and fed into the multi-layer encoders. Lastly, the multi-layer decoders utilize the encoding features and decoding queries to localize objects in the image.

While DETR has demonstrated commendable performance, it still faces challenges such as longer convergence periods and poor predictive performance of small objects. Numerous researchers have endeavored to address these issues by refining model components, including attention mechanisms and encoding queries. As a significant innovation with the DETR-like framework, Deformable-DETR [22] introduces deformable attention. Computational complexity is greatly reduced by computing the attention of surrounding keypoints. Compared with the classical global self-attention, this method significantly improves the performance. Conditional-DETR [42] decouples the decoding query into positional and content queries, enabling a higher-quality optimization during the decoding stage. This method accelerates the convergence of the DETR model. Based on deformable attention, Sparse-DETR [43] introduces a scoring network to selectively learn to encoding queries and remove redundancy for more efficient self-attention computation.

In addition, there are some algorithms for improving the structure of DETR. TSP-RCNN [44] proposes that the primary factor contributing to the slow convergence speed of DETR lies in the computational complexity of the Hungarian matching and the cross-attention. To address this concern, they only retained the encoder as a post-processing component of the model, leading to a substantial acceleration in the convergence speed. With this strategy adjustment, they achieve satisfactory results after only 36 training epochs. On the other hand, D^2-DETR [45] takes the perspective that replacing the decoder in the model is challenging. Consequently, they put forth the idea of using a lightweight Transformer backbone instead of the encoder, with the aim of reducing computational complexity. These methods provide valuable insights for addressing challenges related to the convergence speed and computational efficiency of DETR-like models.

Although existing research has made significant progress in improving convergence speed and small object detection performance, some challenges remain. For instance, the low quality of initial prediction boxes makes small objects difficult to learn, and a portion of training iterations is spent addressing the information bias introduced during the decoding stage, which results in slower convergence speeds. These issues seem solvable through optimization of the model design. To address these challenges, we propose three novel methods: Dynamic Groups Assignment, Mixed Query Re-Selection, and the Look Forward Stage. These methods aim to balance the number of positive and negative samples, improve the quality of initial prediction boxes, and eliminate inter-layer information bias in the decoder, respectively. Detailed descriptions of these methods can be found in Section III.

C. TRAINING STRATEGY FOR DETR-LIKE DETECTORS

Numerous training strategies for DETR-like models have been proposed. The key of these strategies is stronger supervision on the ground truth boxes [28]. DN-DETR [46] introduces a denoising strategy. It generates a set of distinctive decoding queries by applying random noise to the ground truth . These queries stabilizes the initial training phase by explicitly defining optimization objectives for training period. Building upon this, DINO [29] proposes an enhanced denoising strategy known as Contrastive DeNoising Training. It not only introduces random noise to the ground truth serving as positive samples but also adds it to the background, which serves as negative samples. This strategy further reduces the instability of the binary matching process by simultaneously learning image foreground and background. This work marks the first instance where DETR-like models achieved optimal performance on the COCO dataset. In addition, DDQ [31] introduces a query filtering strategy to eliminate similar queries. This strategy provides a stable foundation for subsequent bipartite graph matching.

In addition to the noise strategy, adding auxiliary branches is also the current mainstream training strategy. $\mathcal{H}$-Deformable DETR [28] introduces a hybrid query matching scheme, which implements one-to-many label assignment through auxiliary queries. This strategy can strengthen the supervision of the ground truth and improve the convergence speed of training. Similarly, Group-DETR [24] also uses auxiliary queries to enhance the supervision of the ground truth, thereby accelerating the training of the model. $\mathcal{C}$o-DETR [30] uses the output of the detection head of the conventional object detector to initialize the auxiliary branch. This strategy strengthens the supervision of the encoding phase and improves the prior information of the decoding phase. DETR-SQR [47] solves the problem that some intermediate prediction results are better than the final prediction results by modifying the model structure from linear structure to tree structure. This strategy ensures the high utilization of correct prediction results by optimizing the cascade structure.

III. METHODS

A. MODEL OVERVIEW

We proposed the EM-DETR model comprising three primary components: a backbone network for feature extraction, a multi-layer encoder for enhancing feature representations, and a multi-layer decoder incorporating various attention mechanisms, similar to [21], [22], [28], [46], [48]. The framework of EM-DETR is illustrated in Figure 1.

In EM-DETR, we use a backbone network to extract image features and utilize a multi-layer encoder to enrich the extracted feature maps. Subsequently, we introduce an innovative method, Mixed Query Re-Selection, for initializing decoding queries. Mixed Query Re-Selection initializes decoding queries using anchor boxes regressed from the encoding features. It is essential to note that this method does not initialize the content queries of the main branch but sets them as learnable parameters. Further details on Mixed Query Re-Selection are provided in Section III-C. During the decoding phase, we use Distinct Query Selection and Look Forward Stage in each decoder layer. Distinct Query Selection helps eliminate redundant queries within each decoder layer. Look Forward Stage adjusts the gradient structure between adjacent decoder layers, enabling the model to rapidly identify suitable gradient optimization directions during the decoding phase. Distinct Query Selection and Look Forward Stage will be presented in Section III-E and Section III-D. Simultaneously, to balance the number of positive and negative samples during model training, we propose a novel training strategy called Dynamic Groups Assignment. This method stabilizes the training process and enhances model's generalization. Dynamic Groups Assignment will be presents in Section III-B.

B. DYNAMIC GROUPS ASSIGNMENT

The one-to-many label assignment strategy in conventional object detectors has been proven to be an effective training strategy [33] [3]. The hybrid matching scheme [28] is mainly designed to simulate this label assignment strategy. The core idea is to increase the number of positive samples, which enhances the model's supervision of each ground truth box. This scheme divides the decoding queries into a main branch and an auxiliary branch, as shown in Figure 2(b). During the loss computation phase, the main branch uses the Hungarian algorithm for binary matching between predictions and ground truth, while the auxiliary branch performs one-to-many label assignment, similar to traditional object detectors.

However, while the auxiliary branch increases the number of positive samples, it also introduces an equivalent number of negative samples. Therefore, although the number of positive samples increases, the ratio between positive and negative samples remains unchanged, and the gap between them may even widen. This imbalance can hurt the model's performance and generalization ability [49]. Additionally, using a hyperparameter to fix the number of supervision signals for each ground truth does not match the one-to-many assignment strategy used in conventional object detectors. To address this issue, we propose an adaptive sample assignment strategy, called Dynamic Groups Assignment, which is closer to the one-to-many label assignment used in conventional object detectors. This strategy dynamically allocates positive and negative samples for each iteration based on the dataset distribution and the decoding query state. An illustration of this is shown in Figure 2(c). The specific implementation is outlined as follows:

We establish two sets representing the main and auxiliary branches within decoding queries: $Q = \{Q_1, Q_2, \ldots, Q_t\}$ and $Q' = \{Q'_1, Q'_2, ..., Q'_t\}$. Similar to $\mathcal{H}$-Deformable-DETR [28], we employ L layers of decoders during the decoding phase to process the main branch queries from the 0th layer, generating L sets of queries. The queries from each layer serve as the predicted results for that layer. By performing bipartite matching, we associate predicted results with ground truth values and calculate the loss. We use $\mathcal{L}_{main}$ to represent the loss of the main branch of the decoding queries. The computation process is expressed as Equation 1:

$$\mathcal{L}_{main} = \sum_{i=0}^{L} \mathcal{L}_{Hun}(P_i, G) \tag{1}$$

where P_i represents the i-th layer predict results and G represents the ground truth boxes. We choose the same loss functions as DETR [21], denoted as $\mathcal{L}_{Hun}(.)$, including focal loss [3] for classification, $\mathcal{L}_1$ loss and GIoU loss for regression.

Then, the auxiliary branch go through the same L decoders and similarly obtain L auxiliary branch prediction results. Based on the number of ground truth boxes, we calculate an adaptive coefficient λ using the formula as shown in Equation 2.

$$\lambda = \lfloor \frac{Q_n}{(r+1)G_n} - 1 \rfloor \tag{2}$$

where G_n represents the number of ground truth boxes. Q_n represents the number of decoding queries. And r represents the positive-negative sample ratio. Inferring from the number of targets in each image, it can be deduced that the suitable range for the value of r lies between 1 and 9.

Finally, the predictions from the auxiliary branch are used along with λ times the ground truth boxes to compute the auxiliary loss. We use $\mathcal{L}_{aux}$ to represent the loss of the auxiliary

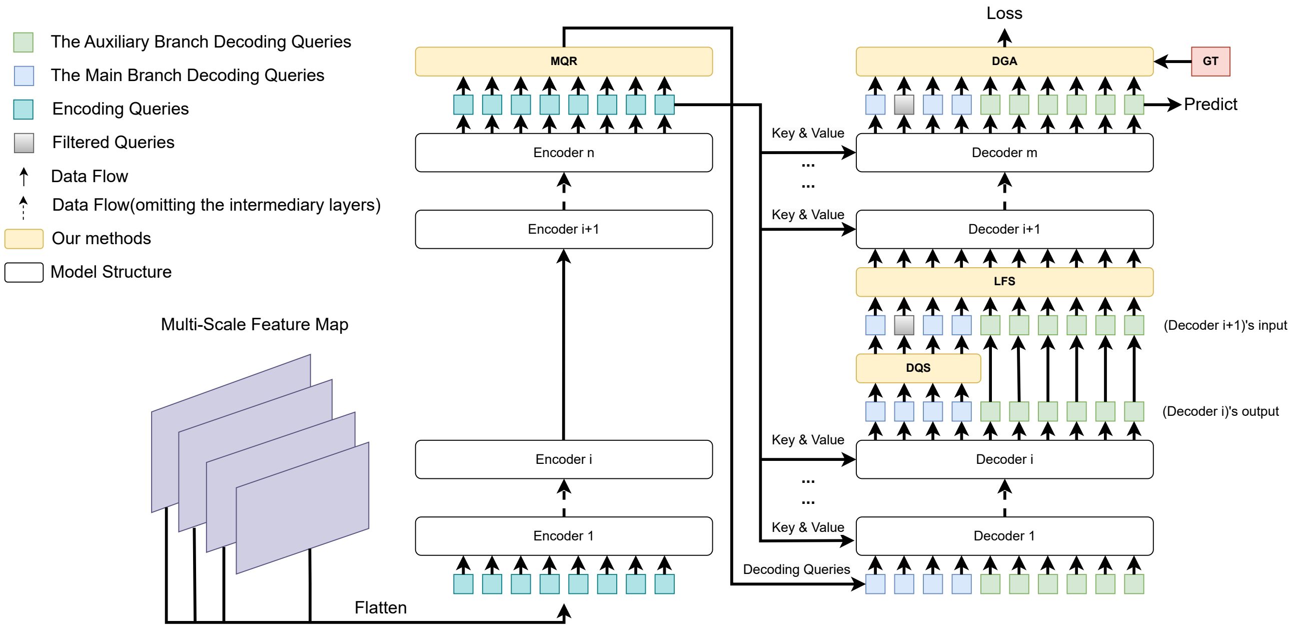
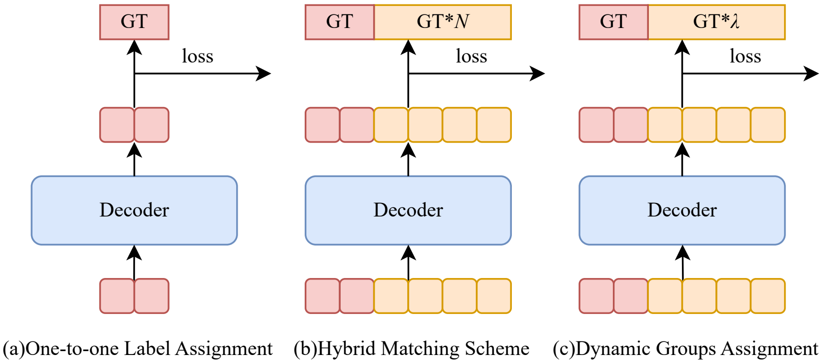


FIGURE 1. The framework of EM-DETR. Our improvements primarily focus on the Transformer encoder and decoder. We use the terms "DGA", "MQR", "DQS", and "LFS" to denote "Dynamic Groups Assignment", "Mixed Query Re-Selection", "Distinct Query Selection" and "Look Forward Stage", respectively.



(a)One-to-one Label Assignment (b)Hybrid Matching Scheme (c)Dynamic Groups Assignment

FIGURE 2. The illustration of different label assignment methods.

branch of the decoding queries. The method for computing the loss function in the auxiliary branch is the same as that in the main branch, as shown in Equation 3.

$$\mathcal{L}_{aux} = \sum_{i=0}^{L} \mathcal{L}_{Hun}(P'_i, \lambda G) \tag{3}$$

where P'_i represents the predicted results of the auxiliary branch at the i-th layer.

In summary, all the loss functions of the Dynamic Groups Assignment are presented in Equation 4.

$$\begin{aligned}\mathcal{L}_{total} &= \mathcal{L}_{main} + \mathcal{L}_{aux} + \mathcal{L}_{encoder} \\ &= \sum_{i=0}^{L}[\mathcal{L}_{Hun}(P_i, G) + \mathcal{L}_{Hun}(P'_i, \lambda G)] + \mathcal{L}_{Hun}(P_e, G)\end{aligned} \tag{4}$$

where P_e represents the prediction results regressed from encoding queries.

To accelerate the training speed, we adopted self-attention mask similar to that utilized in $\mathcal{H}$-Deformable-DETR [28] to concurrently process two branches, as shown in Figure 3. This approach not only prevents interaction between the two branches but also avoids substantial additional training costs. Detailed comparative results between our approach and

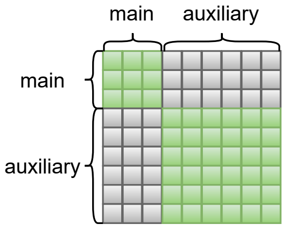
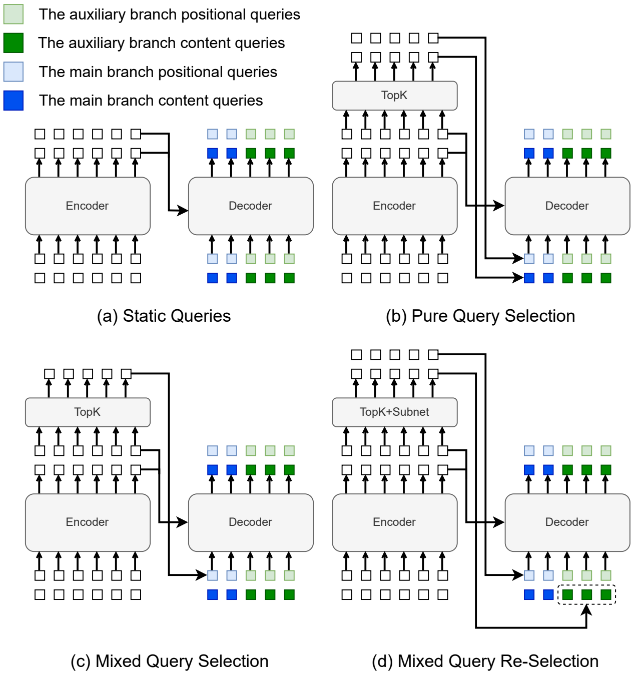


FIGURE 3. The illustration of self-attention mask in Dynamic Groups Assignment.


The auxiliary branch positional queries
The auxiliary branch content queries
The main branch positional queries
The main branch content queries
Encoder
Decoder
TopK
TopK+Subnet

(a) Static Queries

(b) Pure Query Selection

(c) Mixed Query Selection

(d) Mixed Query Re-Selection

FIGURE 4. Comparison of four different query initialization methods.

hybrid branch scheme are presented in the experimental section. As a summary, Dynamic Groups Assignment enhances the model's ability to learn of data distribution by stabilizing the ratio of positive and negative samples. Compared to the hybrid matching scheme, our approach aligns more closely with the one-to-many label assignment process inherent in conventional object detectors.

C. MIXED QUERY RE-SELECTION

In Static Queries [21], decoding queries are constructed as static embeddings, as illustrated in Figure 4(a). These queries directly learn object information from encoding features without relying on any prior knowledge. In contrast, Pure Query Selection [22] employs the TopK method to filter anchor boxes from encoding features as prior knowledge to initialize decoding queries, as shown in Figure 4(b). Building upon Pure Query Selection, Mixed Query Selection [29] utilizes prior knowledge only to initialize positional queries, as depicted in Figure 4(c). This approach aids decoding queries in extracting more comprehensive content information from refined features.

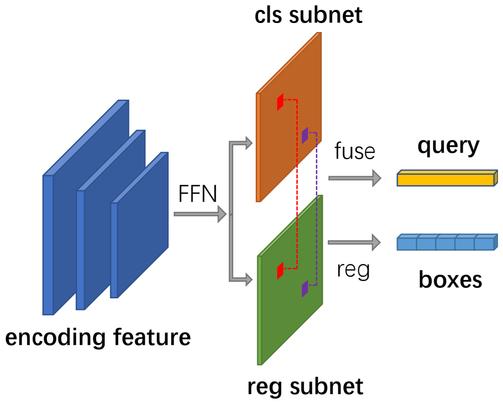


FIGURE 5. The illustration of subnet structure of Mixed Query Re-Selection.

However, we've found two limitations in the Mixed Query Selection after introducing the auxiliary branch. First, the quality of the initialized anchor boxes within the auxiliary branch is low. As the number of decoding queries increases significantly, this results in many bounding boxes with low scores being selected into the initialization queue. These low-quality boxes not only fail to help the decoder converge faster but also introduce additional noise, forcing the model to spend significant training cost on recognizing such noisy data.

Secondly, the content queries in the auxiliary branch are static queries and are not initialized using prior knowledge. The approach of initializing only the content queries is mainly intended to avoid interference from complex prior information during the one-to-one label assignment process. However, in the decoding process, the auxiliary branch simulates the one-to-many label assignment found in conventional object detectors. Multiple supervision signals optimize the same target during decoding. Therefore, this issue does not occur in DETR-like models with an auxiliary branch. On the other hand, setting the content queries in the auxiliary branch as static queries slows down the process of finding the optimal gradient optimization direction.

To overcome these limitations, we propose a decoding query initialization method termed Mixed Query Re-Selection, depicted in Figure 4(d). Addressing the first issue mentioned, we devised a dedicated subnet for initializing the auxiliary branch, as illustrated in Figure 5. Similar to the role of the RPN (Region Proposal Network) [2] in two-stage object detectors, the subsidiary network, composed of convolutional and feedforward neural networks, employs a sliding window mechanism to directly regress higher-quality anchor boxes from encoding features. Its main role is to enhance the initialization quality of anchor boxes within the auxiliary branch. This helps the decoder in rapidly localizing object positions within images and prevents misleading influences from prior knowledge obtained from the encoder.

To address the second issue, Mixed Query Re-Selection initializes both the positional queries and content queries in the auxiliary branch using high-quality bounding boxes. As discussed above, the variance in optimization targets across adjacent iterations is one of the main causes of training instability. However, since the auxiliary branch in EM-DETR uses a one-to-many label assignment strategy, it does not encounter the issue of inconsistent optimization targets. Instead, initializing the content queries in the auxiliary branch becomes essential for quickly determining the direction of gradient descent. Compared to Mixed Query Selection, Mixed Query Re-Selection demonstrates faster convergence. In Section IV, we conduct comprehensive ablation experiments on Mixed Query Re-Selection. The results show that, compared to other methods, Mixed Query Re-Selection achieves superior detection performance, as detailed in Section IV-D.

In summary, the decoding query is initialized as shown in Equation 5.

$$\begin{aligned} Q^d &\leftarrow [Pos^d_{main} \cdot Pos^d_{aux}, Con^d_{main} \cdot Con^d_{aux}] \\ Pos^d_{main} &\leftarrow TopK(G_{box}(Q^e)) \\ Pos^d_{aux} &\leftarrow FFN(3 \times Conv_{3\times3}(Q^e)) \\ Con^d_{main} &\leftarrow S_e \\ Con^d_{aux} &\leftarrow G_{mem}(FFN(3 \times Conv_{3\times3}(Q^e))) \end{aligned} \tag{5}$$

where Q^d and Q^e represent the decoding query and encoding query. Pos^d_{main}, Pos^d_{aux}, Con^d_{main}, and Con^d_{aux} correspond to the main branch positional queries, auxiliary branch positional queries, main branch content queries, and auxiliary branch content queries of the decoding query, respectively. S_e denotes the static queries. The functions $TopK(\cdot)$, $FFN(\cdot)$, and $Conv_{3\times3}(\cdot)$ represent the TopK method, the feedforward neural network layer, and the 3×3 Conv-GN-ReLU network layer, respectively. Additionally, G_{box} and G_{mem} refer to the generation of bounding boxes based on encoding features and the generation of corresponding features based on the bounding boxes.

D. LOOK FORWARD STAGE

We introduce a novel decoder gradient structure in this section. Deformable-DETR [22] draws inspiration from iterative refinement in optical flow estimation and proposes a method called Look Forward Once to ensure the stability of training. This method maintains the relative independence between layers during the training process by obstructing the gradient backpropagation from the i-th layer to the (i-1)-th layer. This means that the optimization of the i-th layer will only affect the parameters of the i-th layer, and will not affect the parameter updates of other layers. Further, DINO [29] proposed another iterative box refinement method called Look Forward Twice. This strategy opens up the gradients across the entire decoding phase, enabling predictive results from later stages to participate in the optimization of parameters in the preceding stages of the model. The key of this method is to provide the direction of gradient optimization for the early stage through the prediction results in the later stage. The structure of these two methods are depicted in Figure 6.

After exhaustive experiments, we observe that Look Forward Once performs well in coarse-grained object detection, while Look Forward Twice excels in fine-grained cases. In Look Forward Once, each layer of the decoder is treated independently, but in fact there is inherent interconnection between layers. This limitation arises from the reliance on each layer's limited information for computation, which may lead to accurate intermediate results but incorrect final outcomes. In contrast, Look Forward Twice features excessively tight interconnections between layers, which can cause information deviation across distant layers. This high degree of interconnection may influence the optimization direction, especially in the earlier stages, potentially leading to suboptimal results. For example, the sixth layer's results could impact the optimization of the first layer's parameters, which primarily focus on coarse localization.

To mitigate these limitations, we propose a novel iterative box refinement method, Look Forward Stage, which combines the strengths of both approaches. Specifically, we divide the decoding phase into two stages: the coarse-grained detection stage and the fine-grained detection stage. In the coarse-grained stage, the model relies on decoding features and prior information to roughly localize objects, with gradient detach used to prevent unnecessary information propagation between layers. In the fine-grained stage, we allow gradient flow to enable subsequent layers to guide earlier ones for more precise object localization.

More specifically, we assume the model possesses L layers of decoders. We partition them into early stage and late stage based on the functionalities of each decoder layer. The early stage comprises n decoder layers, while the corresponding late stage comprises $(L-n)$ decoder layers. For Decoder i in the early stage $(0 < i \leq n)$, we utilize Equation 6 to calculate the prediction boxes b^p_i. And for Decoder j in the late stage $(n < j \leq L)$, we employ Equation 7 to compute the prediction boxes b^p_j.

$$b^p_i = Detach(b'_{i-1}) + FFN(Decoder_i(q_{i-1})) \tag{6}$$

$$b^p_j = b'_{j-1} + FFN(Decoder_j(q_{j-1})) \tag{7}$$

where q_{m-1} represents the input queries of Decoder m. b'_{m-1} represents the bounding box obtained by Decoder $(m-1)$. $Detach(.)$ denotes gradient detach for two tensors. $FFN(.)$ represents a feedforward neural network. $Decoder_m$ represents the operation of Decoder m. Through this approach, we not only prevent information loss between layers but also alleviate information deviation, thereby enhancing the overall performance of the model.

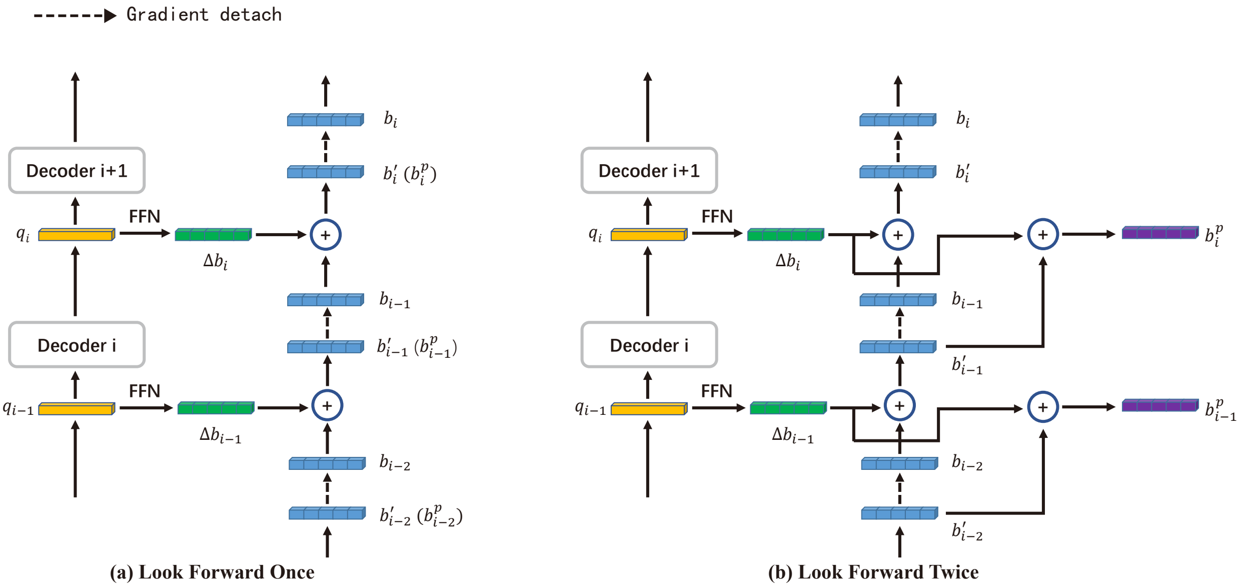


FIGURE 6. Comparison of different decoder gradient structure.

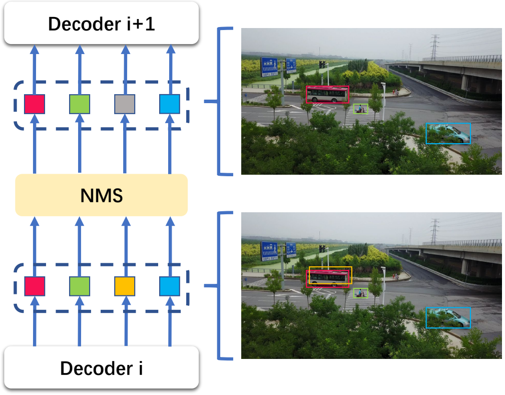


FIGURE 7. The structure of Distinct Query Selecti.

E. DISTINCT QUERY SELECTION

Between the decoder layers of our model, we use an innovative decoding query selection method called Distinct Query Selection [31] to further enhance performance. Distinct Query Selection primarily utilizes the NMS algorithm to filter queries between decoder layers, as illustrated in Figure 7. The purpose of this filtering method is to eliminate redundant bounding boxes, making it clearer to determine the optimization targets during binary matching. By introducing NMS into the query processing within the decoder, we effectively reduce the number of duplicate bounding boxes, thereby improving the accuracy and interpretability of the detection results. It's worth noting that we apply this filtering method between layers during both training and inference, rather than as a post-processing step on the prediction results. As a result, the model can still be considered an end-to-end object detector.

IV. EXPERIMENTS

A. DATASET

VisDrone dataset. We conducted an evaluation on the VisDrone2021-DET dataset [32] for object detection. The dataset is composed of a training set consisting of 6471 images, a validation set containing 548 images, and a test set comprising 1610 images. The range of image resolutions spans from 960 × 540 to 2000 × 1500. The dataset includes 10 distinct object classes, namely pedestrian, people, bicycle, car, van, truck, tricycle, awning-tricycle, bus and motor. The label number of each object category is shown in Figure 8(a). Notably, each image in this dataset has a relatively elevated average count of objects. Moreover, the dataset presents challenges such as a large number of small objects, significant object occlusions and the presence of visually similar objects. The number of object labels of different sizes is shown in Figure 8(b). Hence, this dataset falls into the category of relatively challenging detection datasets.

Other datasets. In addition to the VisDrone2021-DET dataset, we conducted evaluations using two other datasets, AI-TOD [50] and CrowdHuman [51], to validate the generalizability of our proposed method. AI-TOD is a dataset primarily focused on detecting tiny objects. It comprises 28,036 aerial images, encompassing 8 distinct categories and a total of 700,621 object instances. The average size of objects within AI-TOD stands at 12.8 pixels, with over 80% of instances in the dataset having sizes smaller than 16 pixels. On the other hand, the CrowdHuman dataset is a substantial-scale dataset specifically designed for dense pedestrian detection. It consists of 15,000 images for training, 4,370 for validation, and 5,000 for testing. In total, the dataset

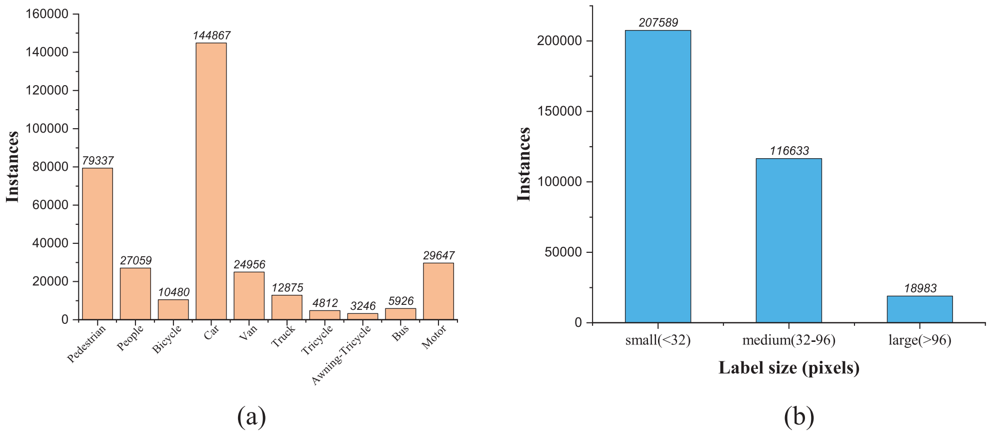


FIGURE 8. (a) The label number of each object category in the VisDrone2021-DET dataset. (b) The number of object labels of different sizes in the VisDrone2021-DET dataset.

encompasses 470,000 human instances. These datasets serve as crucial benchmarks for evaluating the stability of our proposed method across various scenarios, including scenes predominantly featuring tiny objects (such as AI-TOD) and densely object environments (such as CrowdHuman).

Evaluation measures. In this paper, we use various average precisions(AP) based on different IoU thresholds as the main evaluation metrics to judge the accuracy of the model. These include AP with an IoU threshold of 0.5 (represented by AP_{50}), AP with an IoU threshold of 0.75 (represented by AP_{75}), and AP with an IoU threshold of 0.5 to 0.95 separated by 0.05 (represented by $AP_{50:95}$). We also introduce AP of different object scales as an auxiliary evaluation metrics(represented by AP_S, AP_M and AP_L). AP_S, AP_M, and AP_L denote the AP values across various object size categories: [2, 32], [32, 96], and [96, $+\infty$], with $+\infty$ representing positive infinity. In addition, the indicator of giga floating-point operations per second(GFLOPs) is used to evaluate the computational amount of the model. The parameter indicator is used to evaluate the number of parameters of the model.

Implementation details. The operating system used for all experiments in this paper is Ubuntu 20.04. The experimental environments were Python 3.7, Pytorch 1.13.0, and CUDA 12.0. All models were trained on a single NVIDIA GeForce RTX3090 GPU. The experiment was implemented using mmdetection object detection framework. In terms of optimizer, AdamW optimizer [52] was used in this paper. In terms of learning rate, EM-DETR was first warmed up through 2000 iterations and then used the same learning rate setting as DETR [21]. The training batch was set to 2 for all models. In order to make a fair comparison, all the models in the experiment were trained from scratch without any additional data.

B. MAIN RESULTS

Comparison with different models. Through improvements in gradient structure and training strategies, we significantly accelerate the convergence speed and enhance predictive performance. Table 1 presents a comparison between our proposed model and several strong baselines, including two-stage object detectors (Cascade RCNN [33]), single-stage object detectors (RetinaNet [3]), and several SOTA (state-of-the-art) DETR-like detectors. We evaluate the performance of different models using ResNet50 [53] as the backbone across two training scenarios: 12 epochs and 36 or 50 epochs. As shown in Table 1, EM-DETR achieves a notable 23.8% Average Precision (AP) on the VisDrone2021-DET dataset after 12 epochs, without the inclusion of additional training data. Under similar conditions, this result outperforms conventional object detectors by 5.3% AP (17.5% vs 23.8%) and surpasses the best existing DETR-like object detector by 1.1% AP (22.7% vs 23.8%). Furthermore, when all models are fully trained (36 or 50 epochs), EM-DETR achieves SOTA performance, further demonstrating its ability to accelerate convergence and improve accuracy.

Comparison with baseline. We also conduct a comprehensive evaluation of the proposed EM-DETR and $\mathcal{H}$-Deformable-DETR [28] (baseline). This evaluation considers various backbone networks (ResNet50 [53], Swin-Transformer [55]) and different numbers of training epochs (12, 24, 36). The results are presented in Table 2. It is clear from the table that EM-DETR significantly improves model performance across three different backbone networks: ResNet50 [53], Swin-Tiny [55], and Swin-Large [55]. For instance, with 12 training epochs, EM-DETR demonstrates performance improvements of 4.6%, 4.0%, and 3.7%, respectively, compared to the baseline model. Table 2 presents results for 12, 24, and 36 training epochs, while more detailed comparative data can be found in Figure 9. Notably, when ResNet50 [53] is utilized as the backbone network, EM-DETR's performance even surpasses that of the baseline model using Swin-Large [55] as the backbone network. In terms of complexity, our method introduces only a marginal increase in computational load and parameter count (10GFLOPs and 2M). These results substantiate that our proposed method consistently yields substantial improvements

TABLE 1. The detection results of EM-DETR and other detectors using ResNet50 [53] as backbone on the Visdrone 2021-DET dataset. In particular, denotes a two-stage model. The bolded numerical values represent the best detection results. The values marked with a horizontal line at the bottom indicate the second best detection results.**

Methods	Epochs	AP	AP_{50}	AP_{75}	AP_S	AP_M	AP_L
Cascade-RCNN [33]	12	17.5	29.2	18.9	9.0	27.5	36.1
RetinaNet [3]	12	14.5	24.8	15.3	6.1	24.3	32.2
Deformable-DETR** [22]	12	18.1	33.8	17.4	10.4	26.7	32.9
Deformable-DETR-Glimpse [54]	12	17.5	33.3	16.6	9.5	26.2	33.9
DAB-Deformable-DETR [48]	12	16.2	29.1	16.3	8.2	24.3	33.1
DN-Deformable-DETR [46]	12	15.9	29.0	15.6	7.6	24.4	33.7
DINO [29]	12	22.7	40.7	22.4	13.2	33.5	**46.5**
$\mathcal{C}$o-DETR [30]	12	21.9	38.4	22.3	12.1	33.3	39.4
EM-DETR(ours)	12	**23.8**	**43.1**	**23.4**	**14.3**	**34.6**	45.5
Cascade-RCNN [33]	36	19.9	32.5	21.3	10.5	31.1	38.5
RetinaNet [3]	36	16.8	28.6	17.6	7.3	27.1	36.9
Deformable-DETR** [22]	50	20.2	37.0	19.8	11.6	30.3	41.4
Deformable-DETR-Glimpse [54]	50	21.6	39.0	21.4	12.1	32.4	45.3
DAB-Deformable-DETR [48]	50	22.6	38.0	22.8	**15.7**	31.2	38.3
DN-Deformable-DETR [46]	50	22.7	38.1	22.8	15.0	32.1	41.1
DINO [29]	36	**25.1**	**45.0**	**25.0**	15.3	**36.6**	46.2
$\mathcal{C}$o-DETR [30]	36	23.0	40.0	23.4	13.4	33.9	42.2
EM-DETR(ours)	36	25.0	44.4	**25.0**	15.4	36.1	**47.0**

TABLE 2. Comparison of EM-DETR and $\mathcal{H}$-Deformable-DETR under various backbone networks and epochs. The bolded numerical values represent the detection results of EM-DETR, while the regular numerical values represent the detection results of $\mathcal{H}$-Deformable-DETR-Deformable-DETR.

Methods	Backbone	Epochs	AP	AP_{50}	AP_{75}	GFLOPs	Params
$\mathcal{H}$-Deformable-DETR [28] **EM-DETR**	ResNet50 [53]	12	19.2 **23.9**(+4.7)	35.2 **43.3**	18.7 **23.5**	281 **291**	47M **48M**
		24	21.3 **24.5**(+3.2)	38.3 **44.2**	21.2 **24.2**		
		36	22.1 **25.0**(+2.9)	39.2 **44.4**	21.9 **25.0**		
	Swin-Tiny [55]	12	20.7 **24.7**(+4.0)	37.8 **45.5**	20.3 **24.1**	287 **297**	48M **49M**
		24	22.2 **26.1**(+3.9)	40.2 **47.0**	21.9 **25.7**		
		36	23.1 **26.8**(+3.7)	41.3 **48.0**	22.9 **26.4**		
	Swin-Large [55]	12	23.1 **26.8**(+3.7)	41.3 **48.6**	22.6 **26.2**	925 **935**	217M **219M**
		24	24.2 **27.7**(+3.5)	42.9 **49.0**	23.9 **27.5**		
		36	24.5 **27.4**(+3.1)	43.0 **48.6**	24.4 **27.1**		

in object detection performance across various experimental settings.

Other datasets results. In addition to validating the proposed method on the VisDrone2021-DET dataset, we conducted further validation on the sparsely distributed AI-TOD dataset and the densely distributed CrowdHuman dataset. The experimental results are recorded in Table 3. The results indicate that our methods achieve performance gains of 1.9% AP on AI-TOD and 3.7% AP on CrowdHuman after 12 training epochs. With an extended training of 36 epochs, the performance improvements become more substantial, reaching 2.5% AP on AI-TOD and 2.9% AP on CrowdHuman. These results further validate that our method demonstrates exceptional adaptability across various scenarios, showcasing remarkable generalizability.

C. ABLATION STUDY

We present the results of ablation experiments conducted on the VisDrone2021-DET dataset, as shown in Table 4. We employed $\mathcal{H}$-Deformable-DETR (denoted as $\mathcal{H}$) [28] as the baseline. To demonstrate the effectiveness of our proposed methods, we developed an optimized version of $\mathcal{H}$-Deformable-DETR (denoted as $\mathcal{H}$*). The optimized $\mathcal{H}$-Deformable-DETR incorporates several existing techniques, such as setting dropout to 0 and increasing the hidden layer dimension to 2048. It exhibits a relative improve-

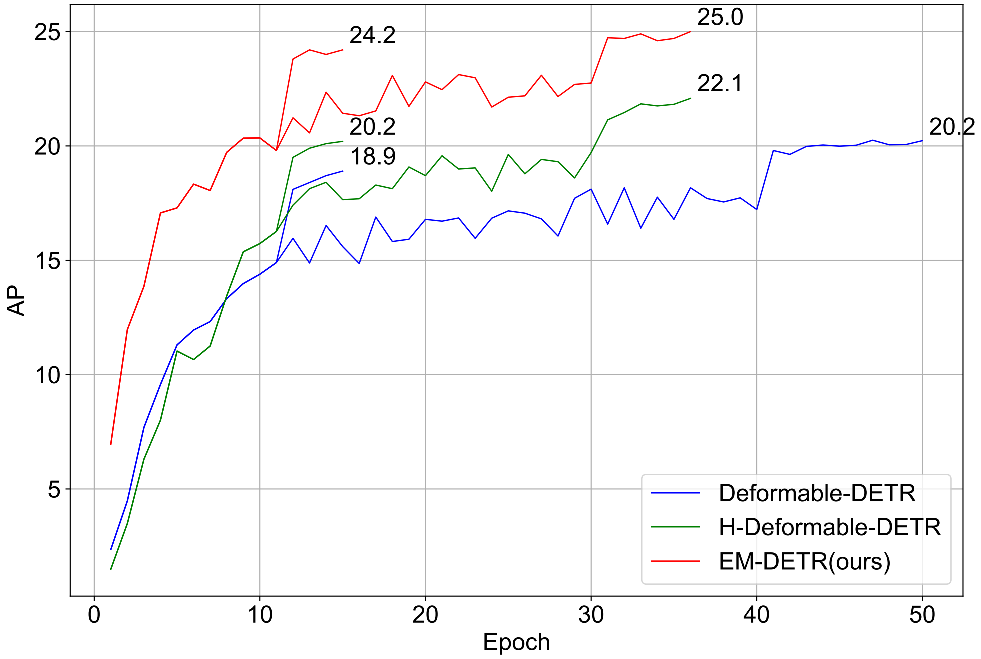


FIGURE 9. Training convergence curves of EM-DETR and two previous SOTA models using multi-scale features evaluated on the VisDrone2021-DET.

TABLE 3. The results of EM-DETR and $\mathcal{H}$-Deformable-DETR using ResNet50 as backbone on AI-TOD and CrowdHuman. $\mathcal{H}$ in table stands for $\mathcal{H}$-Deformable-DETR.

Dataset	Method	Epochs	AP	AP_{50}	AP_{75}	AP_S	AP_M	AP_L
AI-TOD	$\mathcal{H}$	12	16.5	43.3	8.9	16.0	33.2	-
	EM-DETR	12	$18.4^{+1.9}$	52.0	11.1	20.0	34.2	-
	$\mathcal{H}$	36	20.0	50.9	11.0	19.5	35.2	-
	EM-DETR	36	$22.5^{+2.5}$	60.0	15.3	24.1	40.2	-
CrowdHuman	$\mathcal{H}$	12	45.0	79.1	44.8	26.5	42.6	50.2
	EM-DETR	12	$48.7^{+3.7}$	84.9	51.8	33.9	49.2	53.4
	$\mathcal{H}$	36	47.2	81.3	47.7	30.9	45.9	54.3
	EM-DETR	36	$50.1^{+2.9}$	86.0	53.5	37.2	51.7	54.7

ment of 2.1%AP compared to the baseline. Subsequently, we performed ablation experiments on the four methods proposed in this paper. The experimental results indicate that our proposed methods lead to respective enhancements of 1.0/0.8/0.4/0.4%AP. These findings highlight a significant enhancement in the model's detective performance, further validating the effectiveness of the methods proposed in this study.

D. OTHER RESULTS

Different Ratio of Positive and Negative Samples. Through experiments applying Dynamic Groups Assignment with different ratios of positive to negative samples, we observed an instability in prediction performance and a noticeable increase in training duration, as depicted in Figure 10. Based on this analysis, we attribute this phenomenon to an insufficient or excessive number of query groups used for matching with the ground truth in certain images. An insufficient number of query groups can lead to inadequate supervision for ground truth. This may limit the model's learning capacity, as it is unable to fully leverage the information present in the training data. Conversely, an excessively large number of query groups introduces more low-quality queries matched with ground truth, adversely affecting detection performance. Additionally, since we perform Hungarian matching calculations using the CPU, an overly large number of query groups significantly amplifies the training duration.

To address these issues, we introduce a set of hyperparameters to stabilize the loss calculation process. These hyperparameters define a numerical range. We fill the number

TABLE 4. Ablation experiments of the proposed algorithm components. Among them, we use $\mathcal{H}$ to represent $\mathcal{H}$-Deformable-DETR. "*" indicates optimization baseline. The terms "DGA", "MQR", "DQS" and "LFS" stand for "Dynamic Groups Assignment", "Mixed Query Re-Selection", "Distinct Query Selection" and "Look Forward Stage".

Methods	AP	AP_{50}	AP_{75}	AP_S	AP_M	AP_L
$\mathcal{H}$	19.2	35.2	18.7	11.2	28.7	33.8
$\mathcal{H}$*	21.3 (+2.1)	39.8	20.4	12.3	31.2	37.3
$\mathcal{H}$*+DGA	22.3 (+1.0)	40.9	21.8	13.2	32.9	39.9
$\mathcal{H}$*+DGA+MQR	23.1 (+0.8)	42.1	22.7	13.8	33.6	43.3
$\mathcal{H}$*+DGA+MQR+DQS	23.5 (+0.4)	42.4	23.3	14.0	34.4	44.3
$\mathcal{H}$*+DGA+MQR+DQS+LFS	23.9 (+0.4)	43.3	23.5	14.1	35.2	47.5

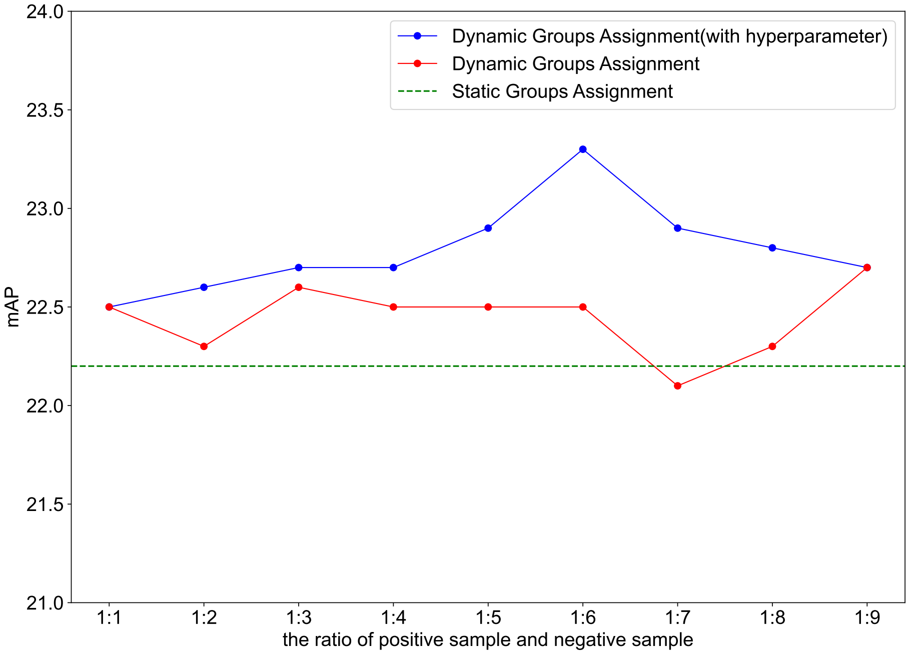


FIGURE 10. Model performance with different ratios of positive and negative samples.

of groups below this range and clip the number of groups above this range. Consequently, all the groups used for loss computation are distributed within this specified range which prevented the adverse impact of insufficient or excessive number groups on the training process. The prediction performance of Dynamic Groups Assignment with proposed hyperparameters, is illustrated by the blue solid line in Figure 10. Its performance is notably superior to the previous version. On the VisDrone2021-DET dataset, the optimal performance is achieved when the positive-to-negative sample ratio is set to 1:6. Hence, we adopt this configuration as the default setting for subsequent experiments.

Different anchor generation strategies and query initialization methods. Table 5 provides a detailed comparison of different anchor box generation strategies and query initialization methods. Under the condition of employing the same query initialization method, the combined TopK and Subnet strategy exhibits a noticeable improvement over the sole use of the TopK strategy, with respective enhancements of 0.8/0.6/0.7%AP. This observation indicates that the quality of newly regressed anchor boxes from Subnet is superior to those with lower scores in the TopK strategy. Additionally, the visual results of the two anchor generation strategies in Figure 13(a) and 13(b) corroborate the same conclusion. Under the premise of employing identical anchor box generation strategies, the query initialization method within Mixed Query Re-Selection demonstrates a slight superiority over Pure Query Selection and Mixed Query Selection. The results in Table 5 imply that Mixed Query Re-Selection significantly enhances the detection accuracy of the model.

Different gradient structure. Table 6 presents the detection performance of different gradient structures. The experiments compare the detection performance of Look Forward Once, Look Forward Twice, and various Look Forward

TABLE 5. Comparing the performance of different anchor box generation strategies and query initialization methods. Among them, "TopK" represents the TopK anchor box generation strategy. "Subnet" means using the TopK method in the main branch and using our proposed subnet to generate the anchor boxes in the auxiliary branch. "Pure" means that all decoding queries are initialized. "Mixed" means that all decoding positional queries is initialized. "Re-Mixed" means that the positional queries of the main branch and all queries of the auxiliary branch are initialized.

TopK	Subnet	Pure	Mixed	Re-Mixed	AP	AP_{50}	AP_{75}
✓		✓			21.3	39.8	20.4
✓			✓		22.1	40.8	21.4
✓				✓	22.3	41.2	21.6
	✓	✓			22.5	40.5	22.4
	✓		✓		22.7	40.8	22.7
	✓			✓	23.0	41.3	22.8

TABLE 6. Compare the detection performance of Look Forward Once, Look Forward Twice and different Look Forward Stage. The "Sample" column records the gradient structure of different methods. "1|2" represents the gradient detach between the first layer decoder and the second layer decoder of the model. "12" represents the intermediate gradient between the first layer decoder and the second layer decoder of the model not detached.

Methods	Sample	AP	AP_{50}	AP_{75}	AP_S	AP_M	AP_L
Look Forward Once	1|2|3|4|5|6	19.2	35.2	18.7	11.2	28.7	33.8
Look Forward Twice	123456	19.5	34.9	19.3	10.7	29.0	37.7
Look Forward Stage	1|2|3|4|56	19.5	35.5	19.2	11.0	28.6	33.4
	1|2|3|456	20.3	35.8	19.5	11.2	29.8	35.5
	1|2|3456	19.9	35.8	19.5	11.2	29.8	36.0
	1|23456	20.6	36.3	20.5	11.7	30.6	37.3
	12|3|4|5|6	19.0	34.9	18.6	10.6	28.3	33.3
	123|4|5|6	19.0	34.6	18.8	10.8	28.5	33.1
	1234|5|6	19.2	34.1	19.2	10.7	28.8	35.5
	12345|6	19.2	34.9	18.8	11.0	28.7	34.9

TABLE 7. Influence of different query selection methods and thresholds on VisDrone2021-DET under 12 training epochs.

NMS	Soft-NMS	AP	AP_{50}	AP_{75}	AP_S	AP_M	AP_L
0.5	-	23.2	42.7	22.4	13.5	34.2	41.8
0.6	-	23.4	42.6	22.9	13.9	34.4	45.2
0.7	-	23.3	42.2	22.9	13.7	34.1	43.7
0.8	-	23.2	41.8	23.1	13.7	34.1	43.6
0.9	-	23.2	41.4	22.9	13.6	33.9	43.8
-	0.5	23.2	41.6	23.0	13.7	34.0	41.2
-	0.6	23.1	41.4	22.8	13.8	33.8	45.9
-	0.7	23.3	41.8	23.1	13.8	34.3	43.7
-	0.8	23.3	41.6	23.0	13.8	34.0	42.3
-	0.9	23.4	41.9	23.3	14.0	34.4	42.8

Stages. The results indicate that Look Forward Stage, with gradient detachment in the early decoding stage and gradient connection in the late decoding stage, achieves the best performance. This approach aligns with the characteristics of DETR-like models, where early decoding typically learns coarse positions, while late decoding focuses on refining positions more accurately. Furthermore, the poor performance observed in the table for the Look Forward Stage with gradient connection in the early decoding stage and gradient detachment in the late decoding stage further supports this conclusion. Based on the experimental results, we select the Look Forward Stage with gradient detachment in the first layer and gradient connection in the subsequent layers as the model's gradient structure.

Different selective methods and thresholds. We also conduct a detailed analysis of the impact of different query selection strategies and thresholds on detection performance. The experimental results indicate that the choice of query selection strategies and thresholds does not significantly affect predictive performance, as shown in Table 7. Across all strategies and threshold settings, the maximum observed difference is only 0.5% AP (ranging from 22.9% to 23.4%). We believe that NMS with any threshold is sufficient to enhance predictive performance. This configuration effectively filters redundant query results. Therefore, even if seemingly stronger methods like Soft-NMS are employed and the threshold is increased, predictive performance will not be significantly improved. Based on these findings, we select NMS as the query selection method with a threshold set at 0.5 under the default settings.

E. VISUALIZATION

To clearly demonstrate the superiority of our method, we visualized some feature maps and detection boxes. The visualizations in Figure 11 depict heatmaps of attention distributions from the P3 layer for Deformable-DETR, $\mathcal{H}$-Deformable-DETR, and EM-DETR. It is evident from the images that the attention distribution of EM-DETR is more refined and closely aligns with the original object contours. This observation reports that EM-DETR can better guide the model's focus on the foreground of the image. Figure 12 showcases the prediction results of $\mathcal{H}$-Deformable-DETR

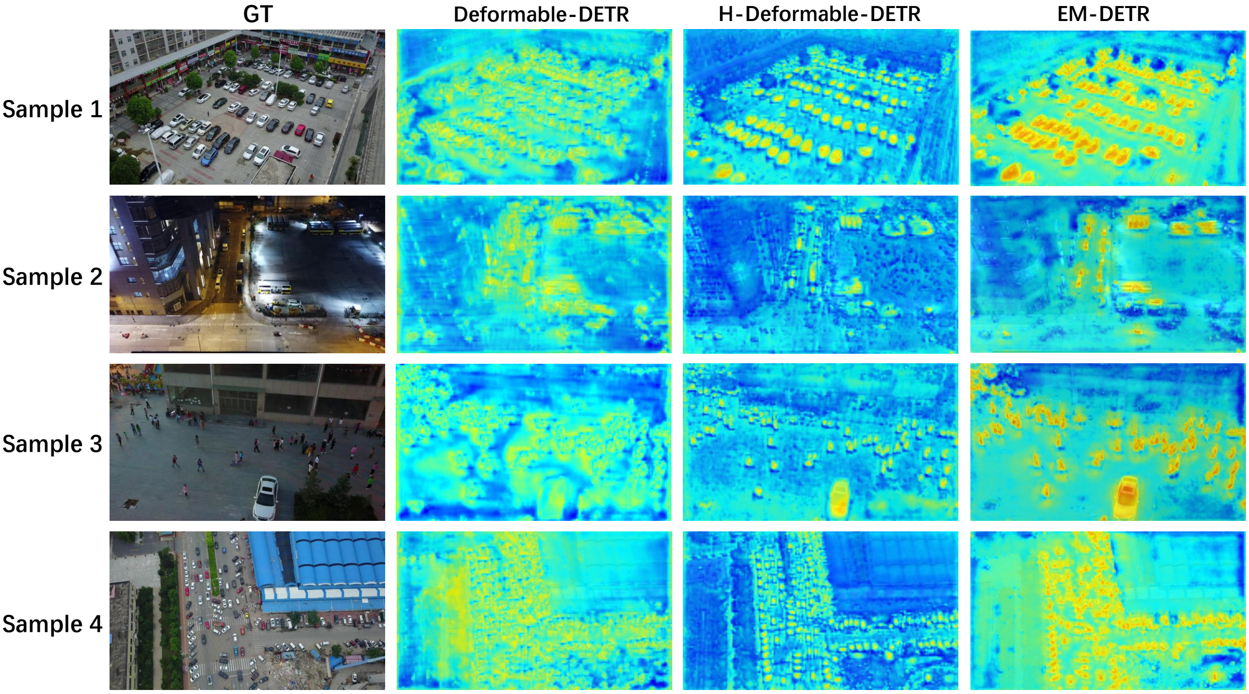
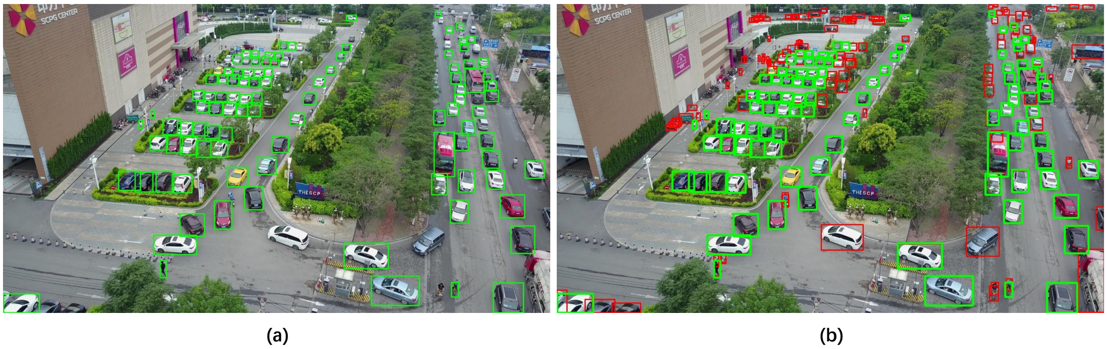
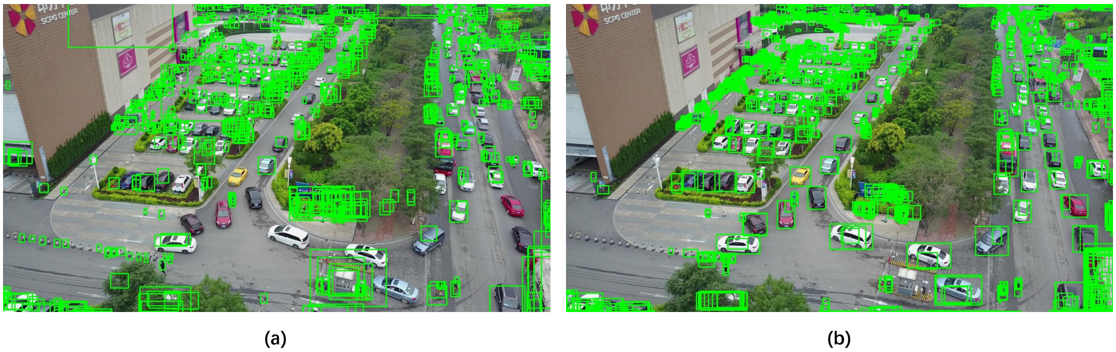


FIGURE 11. Heatmaps of Attention Distribution of Layer P3.


(a)

(b)

FIGURE 12. Detection results of $\mathcal{H}$-Deformable-DETR(a) and EM-DETR(b). The green bounding boxes represents the results detected by both $\mathcal{H}$-Deformable-DETR and EM-DETR. The red bounding boxes represents the results detected by EM-DETR but missed by $\mathcal{H}$-Deformable-DETR.


(a)

(b)

FIGURE 13. Anchor boxes generated by Mixed Query Selection(a) and Mixed Query Re-Selection(b).

and EM-DETR. In this representation, the red bounding boxes denote ground truths correctly detected by EM-DETR but missed by $\mathcal{H}$-Deformable-DETR. It is apparent from the figure that the prediction performance of EM-DETR surpasses that of $\mathcal{H}$-Deformable-DETR. Figure 13 presents the anchor boxes selected by Mixed Query Selection through TopK and those regressed by Mixed Query Re-Selection through the subnet. Similarly, the visual results easily support the conclusion that anchor boxes regressed by the subnet outperform those selected by the TopK method.

V. CONCLUSIONS

In this paper, we propose an enhanced object detector for drone aerial imagery, named EM-DETR. It incorporates three innovative approaches, namely Dynamic Groups Assignment, Mixed Query Re-Selection and Look Forward Stage. Dynamic Groups Assignment stabilizes the training process by rationally assigning positive and negative samples. Mixed Query Re-Selection introduces a subnet to improve the quality of anchor boxes for initializing decoding queries and redefines the initialization method, thereby providing stronger prior knowledge for the decoder. Look Forward Stage introduces a more effective gradient structure tailored for multi-layer decoders, enabling faster identification of optimal gradient directions. Furthermore, we incorporated Distinct Query Selection from DDQ [31] to enhance the queries between the decoder layers.

We conducted comprehensive ablation experiments on the proposed method using the VisDrone2021-DET dataset. The results demonstrate that EM-DETR consistently achieves SOTA performance, whether trained for 12 epochs or more. Meanwhile, we observed that the newly introduced methods only incur a modest increase of 10 GFLOPs and 1M parameters. Hence, our approach achieves superior detection performance with minimal impact on inference speed and model size. Furthermore, these methods exhibit significant performance improvements over baseline models on AI-TOD and Crowdhuman, affirming their strong generalization capabilities. These findings highlight the potential and promising prospects of DETR-like models in object detection.

Despite the substantial improvements made to the model pipeline, EM-DETR still faces some limitations. For example, in real-world scenarios involving drone imagery, factors such as dynamic lighting conditions, varying weather, and moving objects at high speeds can degrade image quality and negatively impact detection performance. Additionally, similar to $\mathcal{H}$-Deformable-DETR [28], the inference speed of EM-DETR currently does not meet the requirements for real-time processing tasks. In the future, we plan to explore more robust preprocessing techniques and lighter model designs to address these challenges, thereby enhancing the model's applicability in real-world scenarios.

Furthermore, EM-DETR is built upon $\mathcal{H}$-Deformable-DETR [28], inheriting its strong adaptability to a wide range of vision tasks. The hybrid matching scheme of $\mathcal{H}$-Deformable-DETR has been successfully applied to tasks such as multi-person pose estimation, multi-object tracking, and multi-view 3D detection [28]. Inspired by this versatility, our future work will focus on extending the improvements made in EM-DETR to other visual tasks, further exploring its potential for broader applications.

REFERENCES

[1] P. Zhu, L. Wen, D. Du, X. Bian, H. Fan, Q. Hu, and H. Ling, "Detection and tracking meet drones challenge," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 44, no. 11, pp. 7380–7399, 2021.

[2] 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.

[3] 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.

[4] Z. Tian, C. Shen, H. Chen, and T. He, "Fcos: Fully convolutional one-stage object detection," in *Proceedings of the IEEE/CVF international conference on computer vision*, 2019, pp. 9627–9636.

[5] A. Bochkovskiy, C.-Y. Wang, and H.-Y. M. Liao, "Yolov4: Optimal speed and accuracy of object detection," *arXiv preprint arXiv:2004.10934*, 2020.

[6] Z. Ge, S. Liu, F. Wang, Z. Li, and J. Sun, "Yolox: Exceeding yolo series in 2021," *arXiv preprint arXiv:2107.08430*, 2021.

[7] L. Zhao and M. Zhu, "Ms-yolov7: Yolov7 based on multi-scale for object detection on uav aerial photography," *Drones*, vol. 7, no. 3, p. 188, 2023.

[8] X. Fu, G. Wei, X. Yuan, Y. Liang, and Y. Bo, "Efficient yolov7-drone: An enhanced object detection approach for drone aerial imagery," *Drones*, vol. 7, no. 10, p. 616, 2023.

[9] M. Pan, W. Xia, H. Yu, X. Hu, W. Cai, and J. Shi, "Vehicle detection in uav images via background suppression pyramid network and multi-scale task adaptive decoupled head," *Remote Sensing*, vol. 15, no. 24, p. 5698, 2023.

[10] H. Mao and Y. Gong, "Steel surface defect detection based on the lightweight improved rt-detr algorithm," *Journal of Real-Time Image Processing*, vol. 22, no. 1, p. 28, 2025.

[11] B. Sun and X. Cheng, "Smoke detection transformer: An improved real-time detection transformer smoke detection model for early fire warning," *Fire*, vol. 7, no. 12, p. 488, 2024.

[12] M. Yang, R. Xu, C. Yang, H. Wu, and A. Wang, "Hybrid-detr: A differentiated module-based model for object detection in remote sensing images," *Electronics*, vol. 13, no. 24, p. 5014, 2024.

[13] M. Ahmed, N. El-Sheimy, and H. Leung, "Enhancing object tracking in smart city intelligent transportation systems: A track-by-detection approach utilizing satellite video monitoring," in *2024 IEEE International Conference on Smart Mobility (SM)*. IEEE, 2024, pp. 17–24.

[14] K. SP and P. Mohandas, "Detr-spp: a fine-tuned vehicle detection with transformer," *Multimedia Tools and Applications*, vol. 83, no. 9, pp. 25 573–25 594, 2024.

[15] Y. Zhao, W. Lv, S. Xu, J. Wei, G. Wang, Q. Dang, Y. Liu, and J. Chen, "Detrs beat yolos on real-time object detection," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2024, pp. 16 965–16 974.

[16] C.-Y. Wang, A. Bochkovskiy, and H.-Y. M. Liao, "Yolov7: Trainable bag-of-freebies sets new state-of-the-art for real-time object detectors," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2023, pp. 7464–7475.

[17] G. Jocher, A. Chaurasia, and J. Qiu, "Ultralytics yolov8," 2023. [Online]. Available: https://github.com/ultralytics/ultralytics

[18] C.-Y. Wang, I.-H. Yeh, and H.-Y. Mark Liao, "Yolov9: Learning what you want to learn using programmable gradient information," in *European Conference on Computer Vision*. Springer, 2025, pp. 1–21.

[19] A. Wang, H. Chen, L. Liu, K. Chen, Z. Lin, J. Han, and G. Ding, "Yolov10: Real-time end-to-end object detection," *arXiv preprint arXiv:2405.14458*, 2024.

[20] A. O. Saltık, A. Allmendinger, and A. Stein, "Comparative analysis of yolov9, yolov10 and rt-detr for real-time weed detection," *arXiv preprint arXiv:2412.13490*, 2024.

[21] 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.

[22] X. Zhu, W. Su, L. Lu, B. Li, X. Wang, and J. Dai, "Deformable detr: Deformable transformers for end-to-end object detection," in *International Conference on Learning Representations*, 2021.

[23] P. Gao, M. Zheng, X. Wang, J. Dai, and H. Li, "Fast convergence of detr with spatially modulated co-attention," in *Proceedings of the IEEE/CVF international conference on computer vision*, 2021, pp. 3621–3630.

[24] Q. Chen, J. Wang, C. Han, S. Zhang, Z. Li, X. Chen, J. Chen, X. Wang, S. Han, G. Zhang *et al.*, "Group detr v2: Strong object detector with encoder-decoder pretraining," *arXiv preprint arXiv:2211.03594*, 2022.

[25] Q. Chen, X. Chen, J. Wang, S. Zhang, K. Yao, H. Feng, J. Han, E. Ding, G. Zeng, and J. Wang, "Group detr: Fast detr training with group-wise one-to-many assignment," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2023, pp. 6633–6642.

[26] Z. Yao, J. Ai, B. Li, and C. Zhang, "Efficient detr: improving end-to-end object detector with dense prior," *arXiv preprint arXiv:2104.01318*, 2021.

[27] Y. Wang, X. Zhang, T. Yang, and J. Sun, "Anchor detr: Query design for transformer-based detector," in *Proceedings of the AAAI conference on artificial intelligence*, vol. 36, no. 3, 2022, pp. 2567–2575.

[28] D. Jia, Y. Yuan, H. He, X. Wu, H. Yu, W. Lin, L. Sun, C. Zhang, and H. Hu, "Detrs with hybrid matching," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2023, pp. 19 702–19 712.

[29] H. Zhang, F. Li, S. Liu, L. Zhang, H. Su, J. Zhu, L. Ni, and H.-Y. Shum, "Dino: Detr with improved denoising anchor boxes for end-to-end object detection," in *The Eleventh International Conference on Learning Representations*, 2024.

[30] Z. Zong, G. Song, and Y. Liu, "Detrs with collaborative hybrid assignments training," in *Proceedings of the IEEE/CVF international conference on computer vision*, 2023, pp. 6748–6758.

[31] S. Zhang, X. Wang, J. Wang, J. Pang, C. Lyu, W. Zhang, P. Luo, and K. Chen, "Dense distinct query for end-to-end object detection," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2023, pp. 7329–7338.

[32] Y. Cao, Z. He, L. Wang, W. Wang, Y. Yuan, D. Zhang, J. Zhang, P. Zhu, L. Van Gool, J. Han *et al.*, "Visdrone-det2021: The vision meets drone object detection challenge results," in *Proceedings of the IEEE/CVF International conference on computer vision*, 2021, pp. 2847–2854.

[33] Z. Cai and N. Vasconcelos, "Cascade r-cnn: Delving into high quality object detection," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2018, pp. 6154–6162.

[34] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, "You only look once: Unified, real-time object detection," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 779–788.

[35] J. Redmon and A. Farhadi, "Yolo9000: better, faster, stronger," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2017, pp. 7263–7271.

[36] ——, "Yolov3: An incremental improvement," *arXiv preprint arXiv:1804.02767*, 2018.

[37] C. Li, L. Li, H. Jiang, K. Weng, Y. Geng, L. Li, Z. Ke, Q. Li, M. Cheng, W. Nie *et al.*, "Yolov6: A single-stage object detection framework for industrial applications," *arXiv preprint arXiv:2209.02976*, 2022.

[38] W. Liu, D. Anguelov, D. Erhan, C. Szegedy, S. Reed, C.-Y. Fu, and A. C. Berg, "Ssd: Single shot multibox detector," in *Computer Vision–ECCV 2016: 14th European Conference, Amsterdam, The Netherlands, October 11–14, 2016, Proceedings, Part I 14*. Springer, 2016, pp. 21–37.

[39] K. He, G. Gkioxari, P. Dollár, and R. Girshick, "Mask r-cnn," in *Proceedings of the IEEE international conference on computer vision*, 2017, pp. 2961–2969.

[40] X. Zhu, S. Lyu, X. Wang, and Q. Zhao, "Tph-yolov5: Improved yolov5 based on transformer prediction head for object detection on drone-captured scenarios," in *Proceedings of the IEEE/CVF international conference on computer vision*, 2021, pp. 2778–2788.

[41] Q. Zhao, B. Liu, S. Lyu, C. Wang, and H. Zhang, "Tph-yolov5++: Boosting object detection on drone-captured scenarios with cross-layer asymmetric transformer," *Remote Sensing*, vol. 15, no. 6, p. 1687, 2023.

[42] D. Meng, X. Chen, Z. Fan, G. Zeng, H. Li, Y. Yuan, L. Sun, and J. Wang, "Conditional detr for fast training convergence," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2021, pp. 3651–3660.

[43] P. Sun, R. Zhang, Y. Jiang, T. Kong, C. Xu, W. Zhan, M. Tomizuka, L. Li, Z. Yuan, C. Wang *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*, 2021, pp. 14 454–14 463.

[44] Z. Sun, S. Cao, Y. Yang, and K. M. Kitani, "Rethinking transformer-based set prediction for object detection," in *Proceedings of the IEEE/CVF international conference on computer vision*, 2021, pp. 3611–3620.

[45] J. Lin, X. Mao, Y. Chen, L. Xu, Y. He, and H. Xue, "Dˆ 2etr: Decoder-only detr with computationally efficient cross-scale attention," *arXiv preprint arXiv:2203.00860*, 2022.

[46] 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.

[47] F. Chen, H. Zhang, K. Hu, Y.-K. Huang, C. Zhu, and M. Savvides, "Enhanced training of query-based object detection via selective query recollection," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2023, pp. 23 756–23 765.

[48] S. Liu, F. Li, H. Zhang, X. Yang, X. Qi, H. Su, J. Zhu, and L. Zhang, "Dab-detr: Dynamic anchor boxes are better queries for detr," in *International Conference on Learning Representations*, 2023.

[49] S. Zhang, C. Chi, Y. Yao, Z. Lei, and S. Z. Li, "Bridging the gap between anchor-based and anchor-free detection via adaptive training sample selection," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2020, pp. 9759–9768.

[50] J. Wang, W. Yang, H. Guo, R. Zhang, and G.-S. Xia, "Tiny object detection in aerial images," in *2020 25th international conference on pattern recognition (ICPR)*. IEEE, 2021, pp. 3791–3798.

[51] S. Shao, Z. Zhao, B. Li, T. Xiao, G. Yu, X. Zhang, and J. Sun, "Crowdhuman: A benchmark for detecting human in a crowd," *arXiv preprint arXiv:1805.00123*, 2018.

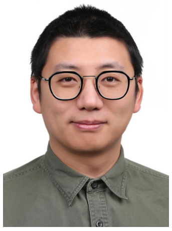
[52] I. Loshchilov and F. Hutter, "Decoupled weight decay regularization," *arXiv preprint arXiv:1711.05101*, 2017.

[53] 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.

[54] Z. Chen, J. Zhang, and D. Tao, "Recurrent glimpse-based decoder for detection with transformer," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022, pp. 5260–5269.

[55] Z. Liu, Y. Lin, Y. Cao, H. Hu, Y. Wei, Z. Zhang, S. Lin, and B. Guo, "Swin transformer: Hierarchical vision transformer using shifted windows," in *Proceedings of the IEEE/CVF international conference on computer vision*, 2021, pp. 10 012–10 022.

QUAN YU received the B.E. degree from School of Information Science and Technology, Beijing University of Chemical Technology, in 2020. He is now a third-year graduate student at Computer School at Beijing Information Science and Technology University, China. His research interests are deep learning, aerial imagery object detection, etc.



QIANG TONG received the PhD degree in department of computer science and technology from Tsinghua University in 2012. He is a lecturer since 2018.08 in Computer School, Beijing Information Science and Technology University, 100101, P.R. China. His research interests are in several areas including Image Recognition, Computer Vision, Machine Learning, and so on.

LIN MIAO received the Ph.D. degree in Ben-Gurion University of the Negev in 2022. She is now a lecturer in Beijing Information Science and Technology University. She has broad interests in artificial intelligence, pattern recognition, knowledge graph, etc.

LIN QI is a professor in School of Economics and Management, Beijing Information Science and Technology University, 100101, P.R. China.

XIULEI LIU received the PhD degree in computer science from Beijing University of Posts and Telecommunications in 2013.03. He is a professor since 2022.01 in Computer School, Beijing Information Science and Technology University, 100101, P.R. China. He was a visiting PhD student in CCSR in University of Surrey from 2008.10 to 2010.10. His research interests are in several areas including semantic sensor, semantic web, knowledge graph, semantic information retrieval, and so on.