Real-Time Medical Lesion Screening: Accurate and Rapid Detector

Dangguo Shao
Kunming University of Science and Technology
Jie Jiang
Kunming University of Science and Technology
Lei Ma
Kunming University of Science and Technology
Sanli Yi
20232204087@stu.kust.edu.cn
Kunming University of Science and Technology

Research Article

Keywords: Computer vision, Lesion screening, Attention mechanism, DETR

Posted Date: April 1st, 2024

DOI: https://doi.org/10.21203/rs.3.rs-4168241/v1

Additional Declarations: No competing interests reported.

Version of Record: A version of this preprint was published at Journal of Real-Time Image Processing on July 17th, 2024. See the published version at https://doi.org/10.1007/s11554-024-01512-x.

Real-Time Medical Lesion Screening: Accurate and Rapid Detector

Dangguo Shao[1]· Jie Jiang[1]· Lei Ma[1]· Sanli Yi[1]

Abstract

Deep learning is a rapidly advancing field, and computer vision techniques such as image segmentation and object detection have found extensive applications across various domains. In medical images, computer vision technology proves effective in handling large volumes of medical imaging data, thereby improving screening efficiency. In the realm of practical medical image applications, object detection has not gained the same level of popularity as image segmentation. Even though some object detection algorithms have been employed for lesion detection, the majority are single-stage detection algorithms, primarily based on the YOLO series[1]. However, the emergence of Transformers[2] appears to be altering this landscape. Leveraging the outstanding performance of Transformers, we propose the RPC-DETR model based on DETR[6], further exploring the potential of Transformers in lesion screening detection. We conducted experiments with the RPC-DETR model on the publicly available brain tumor dataset Br35H, minimizing the model's parameter count and reducing its complexity. In our experiments, RPC-DETR achieved high accuracy with only 14 million parameters. In summary, we have achieved greater accuracy in brain tumor detection by employing a more lightweight model.

Keywords Computer vision · Lesion screening · Attention mechanism · DETR

1. Introduction

In recent years, the widely used image segmentation technique based on U-net[9] in the processing of medical images contrasts with the relatively limited application of object detection. In the detection of lesions in brain tumor images, precise pixel segmentation is not necessary; instead, the focus is on identifying the occurrence and location of the lesions. Therefore, detection tasks are more suitable for brain tumor images. Detection algorithms can accurately identify and locate brain tumors, assisting doctors in the early detection of tumors. Through the precise detection of tumor features such as location, size, and shape, doctors can formulate more personalized treatment plans for patients. This contributes to improving treatment effectiveness, reducing unnecessary side effects, and better meeting the needs of patients. Computer vision technologies in the field of medical imaging have consistently faced several key challenges :

(1) Detection technology needs to be highly accurate, capable of pinpointing and identifying potentially early lesions. This is essential to improve the reliability and accuracy of screening to avoid missed or misdiagnosed cases.

(2) Timely detection of early lesions is crucial for treatment and prevention. Therefore, detection techniques need to have the ability for rapid detection, accelerating the screening process, reducing waiting times, and improving the treatment outcomes for patients.

(3) Object detection techniques for early lesions must have the capability to handle large-scale medical image data. This includes efficient data storage, transmission, and processing to cope with the complex and diverse nature of patient data.

While CNN architectures, particularly represented by the YOLO series, have been widely used for detection tasks due to their high accuracy and speed, the introduction of Vision Transformer (VIT)[10] in image processing has brought a new perspective to object detection tasks. The

✉ Sanli Yi
20232204087@stu.kust.edu.cn

1 Faculty of Information Engineering and Automation, Kunming University of Science and Technology, Kunming 650500, Yunnan Pronvince, China

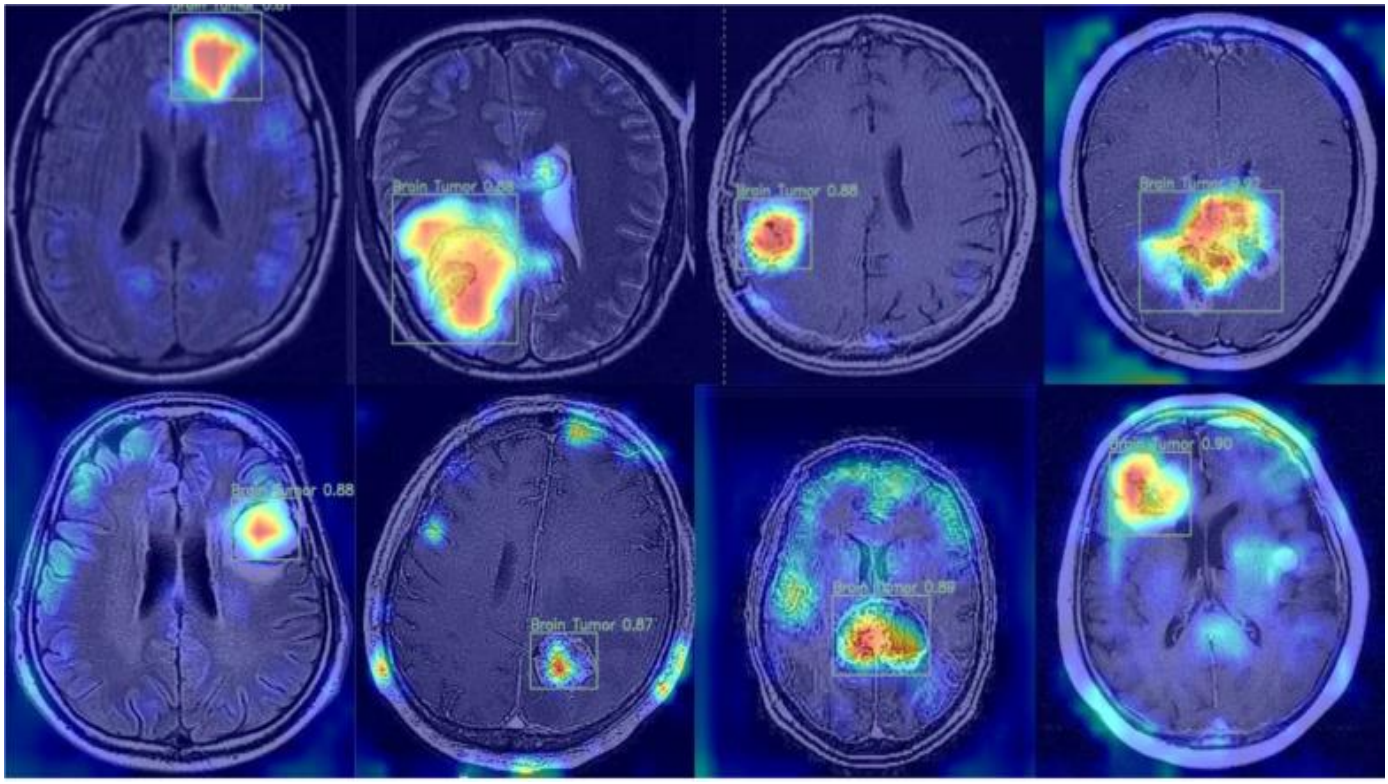

Figure 2：Our detector based on the Transformer, which generates the heat map based on the objects attended to by the Attention mechanism.

proposal of VIT has sparked a trend in applying Transformer principles to computer vision applications. Among them, Detection With Transformer (DETR) introduced a different post-processing approach for object detection, marking the birth of the DETR series. Even models like DETR, which circumvent NMS operations, face challenges due to the inherent complexity of the model compared to typical CNN networks, making it difficult to achieve high accuracy while maintaining lightweight characteristics. Therefore, building upon the DETR framework, we propose RPC-DETR.

Shortly thereafter, the Baidu team introduced Real-Time Detection With Transformer (RT-DETR)[11], which is a remarkable work that, with astonishing results, outperformed all compared YOLO models. This achievement underscores the enormous potential of DETR. Building upon RT-DETR, we conducted improvement experiments and applied it to medical image processing, resulting in our RPC-DETR, achieving the results as shown in Figure 1 with the mAP50 metric.

This work primarily aims to maintain high accuracy while striving to lightweight the model and reduce unnecessary computational redundancy. Inspired by the reparameterization in RepVGG[12] and the partial convolution concept in Fasternet[13], we conducted analysis and experiments within the framework of RT-DETR, leading to the proposal of RPC-DETR, which we believe is suitable for medical image detection.

The main contributions of this article are summarized as follows:

- The DETR detector architecture has been applied to lesion detection screening in medical images, showing improvements in various metrics compared to traditional real-time detectors such as the YOLO series.
- We propose the RPC-block, integrated into the backbone of RPC-DETR, to replace the convolutional part of the multiscale feature extraction in the Resnet network[14]. This replacement not only enhances accuracy but also significantly reduces the parameter count, achieving the goal of a lightweight model.
- We have re-examined the loss function in object detection, adopting the more reasonable Shape-IoU[15] as the loss function for RPC-DETR. This adjustment effectively further enhances the accuracy of detection.

2. Related work

2.1. Real-time object detectors

Current real-time object detectors primarily rely on deep learning techniques, especially convolutional neural networks. Advanced architectures such as Fast R-CNN[16], SSD[17], YOLO, RetinaNet[18], and FCOS[19]have achieved significant success in real-time object detection. These technologies not only improve accuracy but also strike a balance between speed and efficiency. They commonly share several characteristics: (1)These detectors feature streamlined and straightforward network architectures, contributing to their efficiency. (2)Stable Loss Functions: They employ stable loss functions, ensuring robust training and reliable optimization. (3)Effective Feature Extraction and Multi-Scale Feature Fusion: These methods employ strategies for efficient feature extraction and the fusion of multi-scale features, enhancing their overall performance.

Anchor-Based Detectors: YOLOv5, as one of the most popular models for object detection in industrial applications, adopts the Anchor-based prediction paradigm. The core idea is to pre-define a set of anchor boxes on the image and then adjust these predicted anchor boxes through the model's loss training for precise localization and image classification. Models of this type typically involve two stages: anchor box generation and target prediction. However, this model has notable drawbacks. It requires generating a large number of prediction anchor boxes, which may lead to wastage of computational and storage resources. Additionally, in the post-processing of model anchor boxes, Non-Maximum Suppression (NMS) is employed to remove redundant detection results, introducing extra hyperparameters and computational complexity that may need manual tuning.

Anchor-Free Detectors: YOLOv1[20] stands out as a classic example of anchor-free detectors. In YOLOv1, anchor boxes are not employed for prediction. Instead, bounding boxes are predicted through points near the center of objects. To generate high-quality detection results, only points close to the object's center are utilized. This design choice results in YOLOv1 having a lower recall compared to later versions like YOLOv2[21] that utilize an anchor-based approach. Subsequent models, such as FCOS, as well as updated versions like YOLOv6 and YOLOv8, have addressed this limitation, aiming to improve performance in the context of anchor-free detection.

Despite significant progress in the implementation of object detectors, there are still challenges to address, including accuracy in complex scenes, detection of small objects, and balancing real-time performance with accuracy. Future research directions may involve exploring new network architectures, more effective strategies for real-time performance optimization, and the integration of cross-domain applications.

2.2. Detection with Transformer

With the success of Transformers in the field of natural language processing, as seen in models like BERT and the GPT series, researchers have started exploring the application of Transformers in object detection tasks. The Transformer architecture has demonstrated outstanding performance in sequence modeling, which potentially gives it an advantage in handling object detection tasks with irregular arrangements.

An important milestone in this line of work is the introduction of ViT, which marked the first instance of applying the Transformer architecture to the field of computer vision. ViT divides an image into a series of patches and then maps these patches into the Transformer using an embedding layer, allowing the model to globally model the entire image. The success of ViT has inspired subsequent research, including the development of the DETR model. Building on the success of ViT, researchers have proposed various frameworks for object detection using Transformers.

DETR takes a step further by treating the object detection task as a sequence generation problem, providing a more intuitive way to capture relationships between objects. DETR adopts the encoder-decoder structure of the Transformer, where the encoder processes features from the input image, and the decoder generates a sequence of target class and position information. This sequence generation approach, distinct from traditional regression or classification methods, allows DETR to simultaneously output position and class information for multiple objects. Unlike methods that introduce candidate box generation with subsequent post-processing using NMS, DETR eliminates duplicate candidate boxes through decoder self-attention and one-to-one supervision, ensuring that each true object is associated

with a single candidate box.

In subsequent work, such as RT-DETR, it has been demonstrated that for real-time detectors requiring NMS post-processing, anchor-free detectors outperform anchor-based detectors with equivalent accuracy. The reason behind this lies in the fact that anchor-based detectors generate more prediction boxes than anchor-free detectors.

2.3. Bounding box loss

In recent years, there has been rapid development in object detectors, accompanied by further exploration of bounding box loss functions. Initially, IoU (Intersection over Union) was commonly used to assess the degree of bounding box regression. Subsequently, a series of loss functions such as GIoU (Generalized IoU)[22], DIoU (Distance IoU)[23], CIoU (Complete IoU), and others emerged[24]. These loss functions iteratively build upon the foundation of IoU to achieve improved detection performance.

IoU: The most commonly used bounding box loss function in object detection tasks, IoU has been employed in numerous classical detection works. Its definition is as follows:

$$IoU = \frac{|B \cap B^{gt}|}{|B \cup B^{gt}|} \tag{1}$$

B denotes the predicted bounding box, while B^{gt} represents the ground truth bounding box.

GIoU : In IoU, situations may arise in bounding box regression where the Ground Truth (GT) box and the Anchor box do not overlap, leading to gradient vanishing issues and hindering normal convergence. To address this scenario, GIoU is proposed as an extension to IoU:

$$GIoU = IoU - \frac{|C - B \cap B^{gt}|}{|C|} \tag{2}$$

C represents the minimum enclosing bounding box between the Ground Truth (GT) and the Anchor.

DIoU: In comparison to its predecessors, DIoU introduces distance constraints between bounding boxes. Leveraging the centroid, it normalizes the distance loss component, further enhancing the precision of regression results. Its definition is as follows:

$$DIoU = IoU - \frac{\rho^2(b, b^{gt})}{c^2} \tag{3}$$

where b and b^{gt} represent the centroids of the Anchor and GT boxes, respectively. ρ denotes the Euclidean distance between the centroids, and c is the diagonal distance of the minimum enclosing bounding box between b and b^{gt}.

CIoU: Incorporating considerations for the similarity of shapes between Ground Truth (GT) and Anchor, CIoU builds upon DIoU by introducing new shape loss terms to reduce differences in aspect ratios between Anchor and GT. Its definition is as follows:

$$CIoU = IoU - \frac{\rho^2(b, b^{gt})}{c^2} - \alpha v \tag{4}$$

$$\alpha = \frac{v}{(1 - IoU) + v} \tag{5}$$

$$v = \frac{4}{\pi}\left(arc\,tan\frac{w^{gt}}{h^{gt}} - arc\,tan\frac{w}{h}\right)^2 \tag{6}$$

where w^{gt} and h^{gt} represent the width and height of the Ground Truth (GT), and w and h represent the width and height of the Anchor, respectively.

In the DETR series of works, matching costs and Hungarian losses are employed to score bounding boxes. Unlike many detectors that utilize predicted boxes, this approach simplifies implementation and introduces issues related to the relative scaling of losses. To mitigate this problem, the authors adopted a linear combination of IoU loss and L1 loss, where the scale of this combination remains invariant.

3. Method

3.1. Model Overview

The model RPC-DETR, designed for lesion detection in medical images, primarily consists of three main components: backbone, encoder, and decoder. The specific architecture of the model is illustrated in Figure 3. Building upon the DETR model's principles, RPC-DETR integrates the Transformer framework into a conventional CNN-based detector. Leveraging the Transformer's self-attention mechanism, it captures global information from images. As the Transformer operates concurrently on the entire global context, it is not constrained by a fixed-size receptive field.

Traditional CNN detectors typically rely on methods such as predicting anchor boxes (predominantly in Anchor-based detectors) and post-processing techniques like Non-Maximum Suppression (NMS) to determine the final prediction results. In contrast, the RT-DETR (Real-Time DETR) work provides a notable comparison. Traditional CNN detectors, especially in Anchor-based models, generate a significantly higher number of predicted boxes than

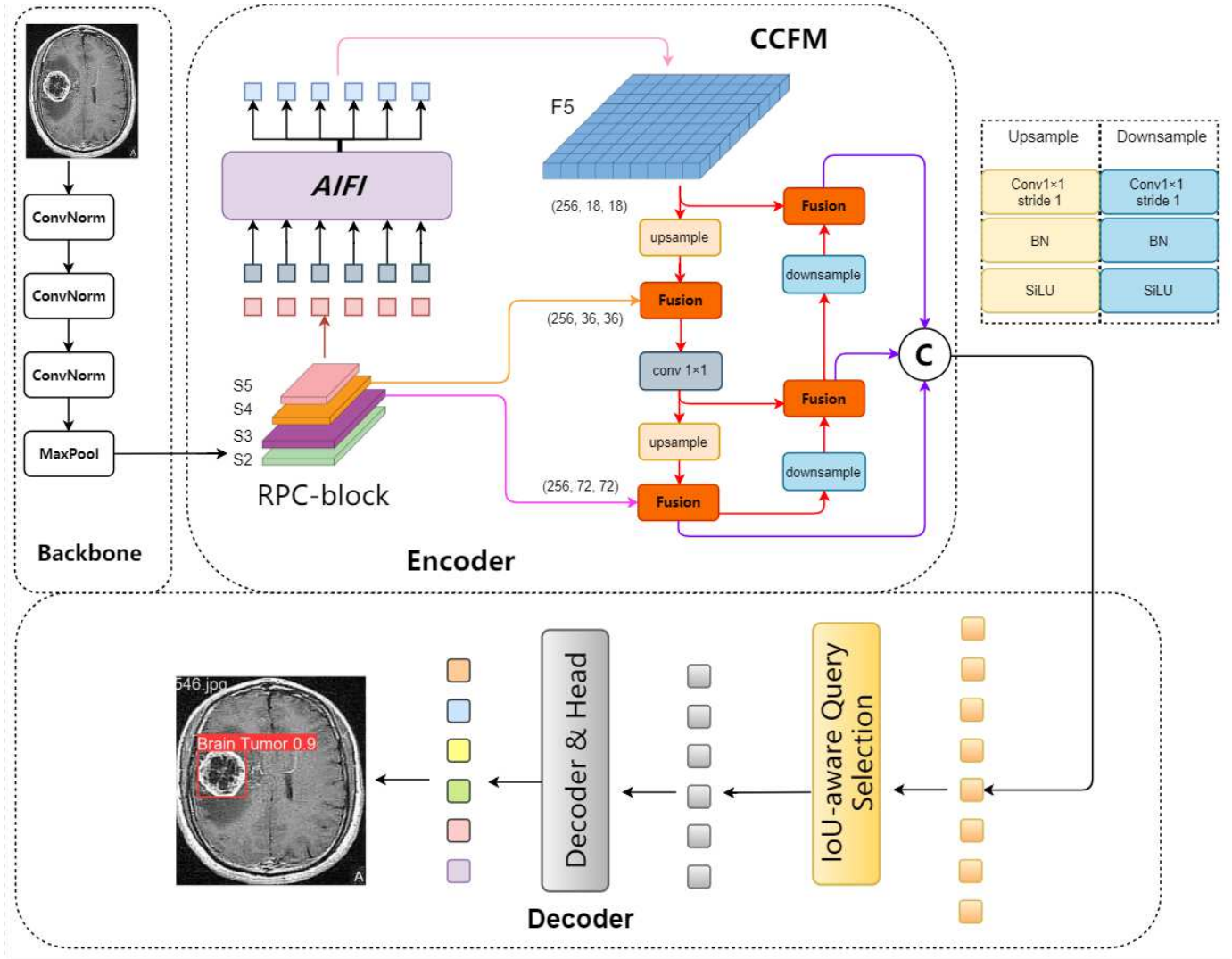

Figure 3: Overview of RPC-DETR. The input image, sized at 640×640, undergoes feature extrac-tion through conventional convolutional layers and normalization, followed by max-pooling. Subsequently, the RPC-block is applied to {S2, S3, S4, S5}, with S5 serving as the feature input to AIFI for scale-aware feature interaction within the same scale. The CCFM module is then em-ployed to perform cross-scale feature fusion on S3, S4, and F5. Ultimately, before proceeding to Head detection and classification, an IoU-based query selection module is employed to optimize object queries.

Anchor-free models, leading to a substantial increase in the time spent on NMS during post-processing. DETR, on the other hand, directly outputs the positional information of objects of different classes using attention mechanisms, thereby avoiding some of the complexities in traditional methods.

Moreover, DETR uses a fixed number of positional encodings instead of predicting boxes, making the model more flexible and suitable for varying numbers and sizes of targets. The DETR decoder identifies candidates through cross-attention, interacting with image features, and performs one-to-one supervision by using self-attention to filter out redundant candidates. The latter part is similar to NMS post-processing, while the former resembles most detectors in the initial stages.

In the backbone section, RPC-DETR primarily adopts the ResNet-18 framework. What sets it apart is the modification made at the forefront of the network. To enhance model performance, reduce model size, and address gradient flow issues during retraining, three 3×3 convolutional layers are employed instead of the original 7×7 convolutional layer in the ResNet architecture. Using three 3×3 convolutional layers instead of a single 7×7 layer reduces

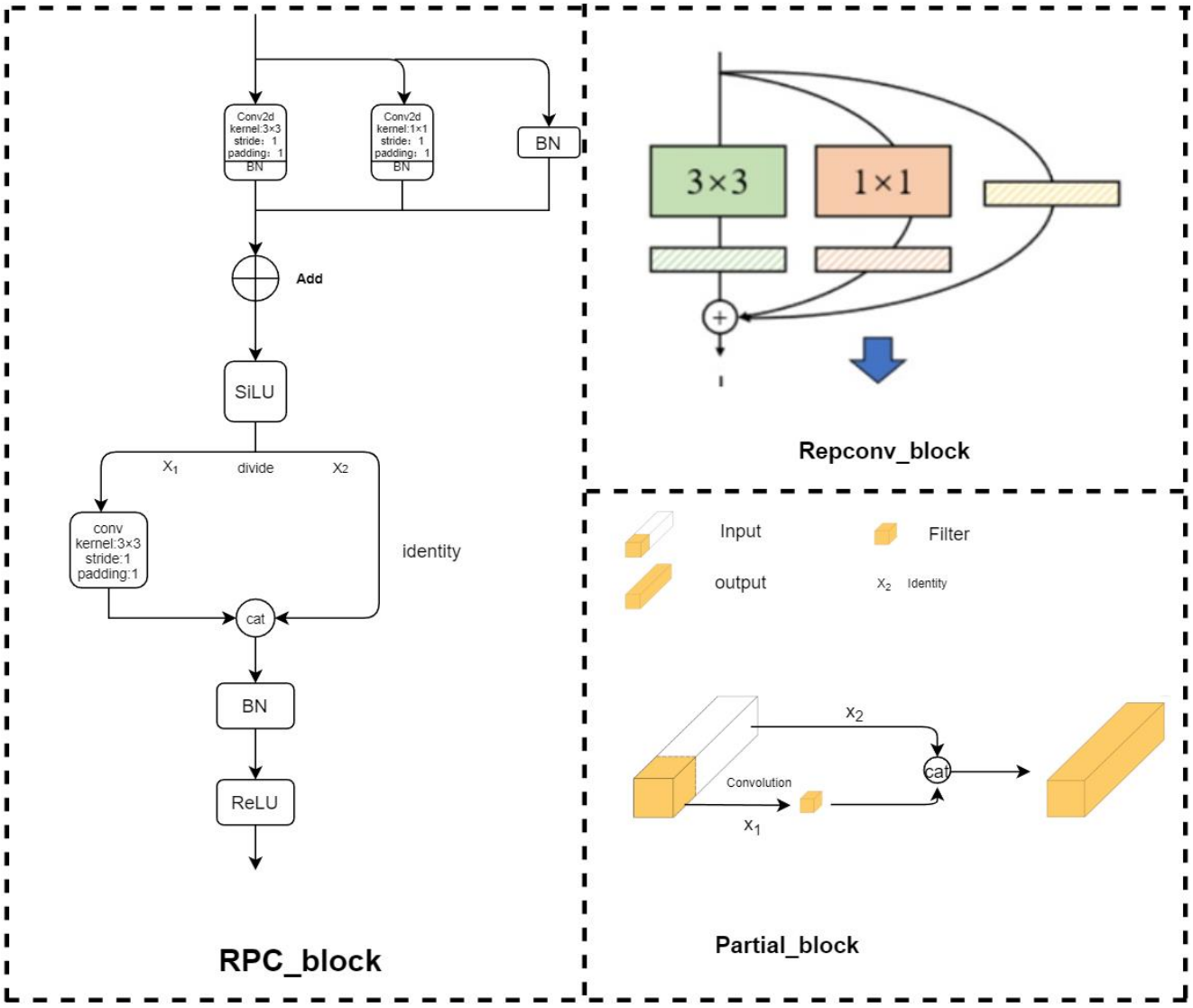

Figure 4: The RPC-block consists of two components, Repconv and Partialconv. Initially, the in-put is fed into Repconv. During the training process, three branches are formed, and Batch Nor-malization operations are applied to the results of each branch. The outputs of the branches are then combined through an Add operation, followed by the SiLU activation function, yielding the output of the first stage. In Partialconv, the output of the first stage serves as the input for the second stage. The input X is divided into two parts, X1 and X2, through a divide operation. After applying a regular 3×3 convolution operation to X1 and concatenating the result with X2, the combined output undergoes BN and ReLU operations, resulting in the final output of the RPC-block.The three layers of 3×3 convolution have a total of 27 parameters, whereas a single 7×7 convolutional layer would require 49 parameters. To align with our lightweight improvement goals, we opted for smaller convolutional kernels. This not only reduces the parameter count but also diminishes model complexity, thereby lowering the risk of overfitting.

parameter count, decreases model complexity, and mitigates overfitting risks. Employing consecutive 3×3 convolutional layers enables a larger receptive field and deeper non-linearity, enhancing the neural network's capability to learn complex patterns and features, thereby improving representational power.

Through consecutive use of the RPC-block {S2, S3, S4, S5} for cross-scale feature extraction, the output of S5 is eventually combined with positional encodings and fed into the AIFI module for scale-aware feature interaction. Since only scale-aware interaction is applied to S5, AIFI further reduces computational overhead. This self-attention

operation captures relationships between conceptual entities in the image, facilitating subsequent modules for object detection. Subsequently, the CCFM module fuses features extracted from F5 and the previous scales S3 and S4 across different scales. The fused output is then input into the IoU-aware Query Selection, providing encoder features with more precise classification and accurate location information for object queries, thereby enhancing the accuracy of the detector.

Finally, the output of the IoU-aware Query Selection serves as the input to the DETR decoder. In this process, DETR introduces class embedding information into the earlier encoder section, seamlessly integrating category information into the object detection procedure. This integration aids in enhancing the discriminative capability across different classes.

3.2. RPC-block

In DETR, the {S2, S3, S4, S5} layers were originally extracted using ordinary convolutional layers of different scales from ResNet-18. In this work, inspired by RepVGG and FasterNet, we introduce the RPC-block module to reduce redundancy in the convolutional process, aiming to lightweight the model while enhancing its effective feature extraction capabilities. The specific representation of the RPC-block module is illustrated in Figure 4.

After extensive validation through various experiments, conventional convolutional layers indeed possess numerous advantages in neural networks. However, they also expose a crucial drawback, namely, subpar feature extraction performance. In RPC, we drew inspiration from the design of RepVGG, incorporating different-sized convolutional kernels in distinct branches. The first branch employs a 3×3 kernel, the second branch uses a 1×1 kernel, and the third branch plays a role similar to the shortcut in ResNet. This design enables the model to learn to capture features at different scales and levels, enhancing the model's representational capacity for complex inputs. Following reparameterization, the different branches share the same convolutional kernel weights, promoting model generalization and reducing the requirements for storage and computational resources. This achieves the goal of lightweight and efficient model design. Subsequently, Batch Normalization is applied to all three branches, and their outputs are combined through an Add operation to generate a new output. The SiLU activation function is then applied to the output of this stage.

The primary reason for using the SiLU activation function here is its smooth nature and continuous derivative, facilitating stable gradient propagation and making the network easier to train. SiLU, as a smooth activation function, exhibits adaptability because its shape depends on the input. When the input is close to zero, the output of SiLU is approximately linear, while elsewhere it demonstrates nonlinear behavior. This adaptability helps the network better adapt to different input distributions. The specific expression is as follows:

$$F(x) = x * \sigma(x) \tag{7}$$

$$\sigma = \frac{1}{1 + e^{-x}} \tag{8}$$

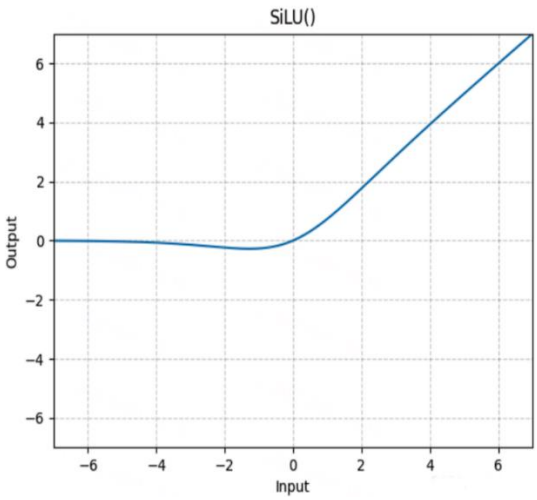

Figure 5: SiLU (Sigmoid Gated Linear Unit) Activation Function

Taking the output after SiLU as the input for the second stage Partialconv, the Partialconv layer divides the input X into two parts, X1 and X2. The X1 portion undergoes a 3×3 convolution operation for further feature extraction, while X2 remains unchanged and serves as a shortcut path. After concatenating the convolutional X1 with the original X2, the combined output goes through Batch Normalization to alleviate the issue of gradient vanishing and ReLU activation to introduce non-linearity. This process results in the output of our RPC-block. In this sequence, the Repconv stage is employed to learn features at different scales and levels, capturing a broader range of features.

Simultaneously, the Partialconv stage reduces redundancy in the convolutional process, overcoming frequent memory access issues, and forms a simple yet fast and effective block.

3.3. Shape-IoU in DETR

The loss functions in earlier detectors have undergone a series of improvements based on the original IoU loss, such as GIoU, DIoU, CIoU, and others. Undoubtedly, these loss functions represent refinements to previous methods, progressively incorporating considerations for the geometric relationship between ground truth (GT) and anchor boxes. However, in Shape-IoU, it is argued that these methods overlook the intrinsic properties of the bounding box itself, such as shape and scale, as illustrated in Figure 6. Consequently, a new generation of loss function, Shape-IoU, has been introduced. This loss function focuses on calculating loss by considering
the inherent attributes of the bounding box, including its shape and scale. This approach aims to enhance the accuracy of regression in detecting objects.

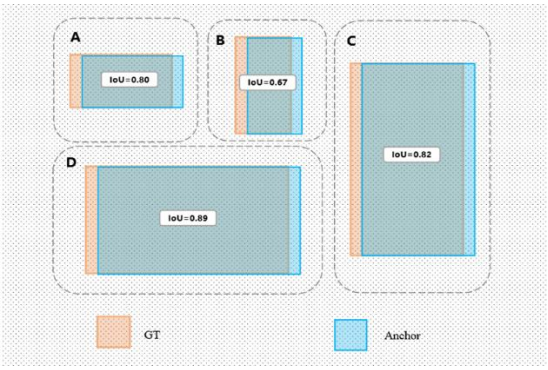

Figure 6 illustrates two sets of examples, denoted as A and B, and C and D, respectively. In both sets, intra-group biases are consistent. However, due to variations in the GT orientation, there is an inconsistency in the directions of the long and short sides. This disparity in orientation results in a significant difference in the final IoU values.

In medical images, the regions of lesions often exhibit a multi-angular nature. We believe that Shape-IoU aligns well with the loss computation requirements of DETR. Therefore, we replaced the original IoU loss function in DETR with Shape-IoU. This substitution allows the loss function to focus on the shape and scale of the bounding box itself, aiming for more precise loss regression. The derivation of the Shape-IoU formula is as follows:

$$ww = \frac{2 \times (w^{gt})^{scale}}{(w^{st})^{scale} + (h^{gt})^{scale}} \tag{9}$$

$$hh = \frac{2 \times (h^{gt})^{scale}}{(w^{st})^{scale} + (h^{gt})^{scale}} \tag{10}$$

$$distance^{shape} = hh \times \frac{\left(x_c - x_c^{gt}\right)^2}{c^2} + ww \times \frac{\left(y_c - y_c^{gt}\right)^2}{c^2} \tag{11}$$

$$\Omega^{shape} = \sum_{t=w,h} (1 - e^{-w_t})^{\theta}, \theta = 4 \tag{12}$$

$$\begin{cases} w_w = hh \times \frac{|w - w^{gt}|}{\max(w, w^{gt})} \\ w_h = ww \times \frac{|h - h^{gt}|}{\max(h, h^{gt})} \end{cases} \tag{13}$$

$$L_{Shape-IoU} = 1 - \text{IoU} + \text{distance}^{shape} + 0.5 \times \Omega^{shape} \tag{14}$$

The "scale" refers to a scaling factor, typically within the range of 0 to 1.5, and is related to the scale of the dataset. A smaller target scale corresponds to a larger scale factor. The parameters "ww" and "hh" denote the weighting coefficients for the horizontal and vertical directions, respectively, and are associated with the shape of the ground truth (GT) bounding box.

4. Experiments

4.1. Setups

Dataset. In our experiments, we employed the publicly available dataset Br35H as the experimental
dataset for our RPC-DETR model. This dataset focuses on brain tumor imaging. For this study, we utilized a total of 500 images as the training set and 201 images as the validation set.

Implementation Details. We adopted ResNet-18 as the backbone network for our RPC-DETR, wherein we replaced the original 7×7 convolutions in ResNet with three consecutive 3×3 ordinary convolutions. Typically, AIFI(Attention-based Intrascale Feature Interaction) module consists of a single Transformer layer, and CCFM(Cross-Scale Context Feature Fusion) is comprised of three Repblocks.

The training strategy and hyperparameters
closely align with those of RT-DETR. We chose AdamW as the optimizer with an initial learning rate of 0.0001, weight decay of 0.0001, and a warm-up iteration of 2000 rounds. Data augmentation techniques include random color distortions, cropping, flipping, and resizing. The training process was conducted over 150 epochs, with a batch size of 8 and

Model	Params	Precision	Recall	AP50	AP50：95	GFLOPs	FPS
YOLOv5s	7M	0.866	0.856	0.903	0.646	15.8	84.03
YOLOv5m	20.9M	0.893	0.821	0.909	0.628	47.9	54.05
YOLOv5l	53..2M	0.896	0.86	0.914	0.629	107.6	49.5
YOLOv6s	16.3M	0.906	0.911	0.934	0.713	44	108.7
YOLOv6m	51.9M	0.895	0.865	0.912	0.69	161.1	40.98
YOLOv6l	110.9M	0.892	0.876	0.912	0.675	391.2	26.88
YOLOv8s	11.1M	0.938	0.906	0.948	0.732	28.4	78.12
YOLOv8m	25.8M	0.938	0.897	0.948	0.725	78.7	59.17
YOLOv8l	43.6M	0.942	0.925	0.953	0.725	164.8	40.49
RPC-DETR(our)	14M	**0.945**	**0.96**	**0.96**	**0.736**	42.8	62.3

Table 1: Comparison with YOLO Series Detectors on the Br35H Public Dataset. Training and inference were conducted using an RTX 4060 8G GPU for all models. Each model underwent 150 epochs of training with a consistent input resolution of 640×640. The evaluation focuses on assessing both model accuracy and complexity.

4 workers. The experiments were conducted on hardware consisting of an AMD Ryzen 9 7940H CPU and a GeForce RTX 4060 laptop GPU for both training and testing phases.

4.2. Comparison with other detectors

In the context of medical image applications, our primary focus lies in evaluating the model's accuracy and complexity after training. The goal is to achieve high precision while concurrently lightweighting the model to reduce computational demands. To address this, we conducted a comparative analysis of RPC-DETR with YOLOv5, YOLOv6, and YOLOv8's L-network models based on metrics such as mAP.

RPC-DETR ultimately achieved a mAP50 of 0.96 and mAP50:95 of 0.736, leading in both accuracy metrics compared to the best-performing YOLOv8l in the YOLO series. Regarding model speed, RPC-DETR achieved a frame rate of 62.3 FPS in our testing environment (using an RTX 4060 laptop GPU), ranking just below YOLO's S-level networks. However, from a comprehensive perspective, RPC-DETR surpassed the accuracy of L-level models while utilizing parameters falling between YOLO's S and M-level models, striking a favorable balance between accuracy and computational efficiency.

4.3. Ablation study on IoU losses

We conducted ablation experiments on commonly used loss functions, and the relevant data is presented in Table 2. Our primary focus lies on mAP 50 and mAP 50:95 to evaluate the model's accuracy. Compared to the previously introduced GIoU, DIoU demonstrates a certain improvement in all aspects. While CIoU slightly outperforms GIoU and DIoU on mAP 50, there is a noticeable decline in performance on mAP 50:95. In contrast, Shape-IoU achieves the highest accuracy.

RPC-block	Shape-IoU	Params	AP50	AP50：95	GFLOPs
		19.9M	0.935	0.732	56.9
√		14M	0.942	0.734	42.8
	√	19.9M	0.947	0.736	56.9
√	√	14M	**0.96**	**0.736**	42.8

Table 2 Ablation study on RPC-block and Shape-IoU. The test model is RT-DETR on RTX4060 laptop GPU evaluate.

Model	precision	recall	mAP50	mAP50:95
RPC-DETR+GIoU	0.944	0.927	0.945	0.733
RPC-DETR+DIoU	0.945	0.949	0.95	0.732
RPC-DETR+CIoU	0.944	0.926	0.956	0.713
RPC-DETR+Shape-IoU	0.945	**0.96**	**0.96**	**0.736**

Table 3: Ablation experiments on IoU loss functions, with the best results highlighted in bold. The scale parameter for Shape-IoU is set to 1.

4.4. Ablation study on RPC-DETR structure

To validate the effectiveness of the structural improvements to the RT-DETR and assess the performance gains, we conducted separate experiments, with a particular focus on the primary RPC-block and Shape-IoU loss function. Our key metrics of interest include the changes in model parameters, mAP50, mAP50:95, and GFLOPs before and after the model modifications. The experimental results are presented in Table 3.

4.5. Result analysis

Given that RPC-DETR is designed for lesion detection in medical images, we should emphasize the model's accuracy as a primary consideration, followed by its speed. A lightweight model for detection can operate in real-time or near-real-time, making it suitable for clinical scenarios where swift diagnostic decisions are required. As shown in Table 1, our proposed RPC-DETR exhibits excellent performance compared to other currently popular detectors. It maintains high accuracy while being relatively lightweight and possessing fast inference speeds. Therefore, we believe this aligns well with the requirements of lesion detection in the field of medical imaging.

5. Conclusion

In this study, we focused on the detection of lesions in medical images, attempting to apply an end-to-end detector to the screening of abnormalities. Leveraging the capability of Transformers in DETR to process global information, our approach allows for a comprehensive consideration of the interactions between features from different regions in

medical images. Inspired by RepVGG and Fasternet, we introduced the RPC-block, conducting further experimental exploration on the original end-to-end detector structure of DETR. Initially, we utilized reparameterized convolutions to learn features at different scales and reduce the parameter count in the feature extraction process. Subsequently, we employed partial convolutions to eliminate redundant features during the convolution process.

Simultaneously, we reevaluated the bounding box loss function used in DETR. To address the diverse shapes and scales present in medical images, we introduced a new Shape-IoU loss function in the DETR detector framework to enhance the accuracy of bounding boxes. As a result, we proposed a novel end-to-end detector, RPC-DETR, which not only improves accuracy in medical image detection but also reduces the device requirements for the model. We sincerely hope that our work can provide valuable assistance in real-life scenarios and anticipate that this research will contribute insights for future endeavors in this field.

Acknowledgements This work is supported by National Natural Science Foundation of China (Grant NO.62266025).

Declarations

Conflict of interest The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper. The authors declare the following financial interests/personal relationships which may be considered as potential competing interests.

References

[1] Redmon J, Farhadi A. YOLOv3: An Incremental Improvement[M]. arXiv, 2018. http://arxiv.org/abs/1804.02767.

[2] Bochkovskiy A, Wang C Y, Liao H Y M. YOLOv4: Optimal Speed and Accuracy of Object Detection[M]. arXiv, 2020. http://arxiv.org/abs/2004.10934.

[3] Li C, Li L, Jiang H, et al. YOLOv6: A Single-Stage Object Detection Framework for Industrial Applications[M]. arXiv, 2022. http://arxiv.org/abs/2209.02976.

[4] Wang C Y, Bochkovskiy A, Liao H Y M. YOLOv7: Trainable bag-of-freebies sets new state-of-the-art for real-time object detectors[M]. arXiv, 2022. http://arxiv.org/abs/2207.02696.

[5] Vaswani A, Shazeer N, Parmar N, et al. Attention Is All You Need[M]. arXiv, 2023. http://arxiv.org/abs/1706.03762.

[6] Carion N, Massa F, Synnaeve G, et al. End-to-End Object Detection with Transformers[M]. arXiv, 2020. http://arxiv.org/abs/2005.12872.

[7] Chen Q, Wang J, Han C, et al. Group DETR v2: Strong Object Detector with Encoder-Decoder Pretraining[M]. arXiv, 2022. http://arxiv.org/abs/2211.03594.

[8] Zhao C, Sun Y, Wang W, et al. MS-DETR: Efficient DETR Training with Mixed Supervision[M]. arXiv, 2024. http://arxiv.org/abs/2401.03989.

[9] Ronneberger O, Fischer P, Brox T. U-Net: Convolutional Networks for Biomedical Image Segmentation[M]. arXiv, 2015. http://arxiv.org/abs/1505.04597.

[10] Dosovitskiy A, Beyer L, Kolesnikov A, et al. AN IMAGE IS WORTH 16X16 WORDS: TRANSFORMERS FOR IMAGE RECOGNITION AT SCALE[J]. 2021.

[11] Lv W, Zhao Y, Xu S, et al. DETRs Beat YOLOs on Real-time Object Detection[M]. arXiv, 2023. http://arxiv.org/abs/2304.08069.

[12] Ding X, Zhang X, Ma N, et al. RepVGG: Making VGG-style ConvNets Great Again[M]. arXiv, 2021. http://arxiv.org/abs/2101.03697.

[13] Chen J, Kao S hong, He H, et al. Run, Don't Walk: Chasing Higher FLOPS for Faster Neural Networks[M]. arXiv, 2023. http://arxiv.org/abs/2303.03667.

[14] He K, Zhang X, Ren S, et al. Deep Residual Learning for Image Recognition[M]. arXiv, 2015.

http://arxiv.org/abs/1512.03385.

[15] Zhang H, Zhang S. Shape-IoU: More Accurate Metric considering Bounding Box Shape and Scale[M]. arXiv, 2024. http://arxiv.org/abs/2312.17663.

[16] Ren S, He K, Girshick R, et al. Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks[J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2017, 39(6): 1137-1149. http://ieeexplore.ieee.org/document/7485869/. doi: 10.1109/TPAMI.2016.2577031.

[17] Liu W, Anguelov D, Erhan D, et al. SSD: Single Shot MultiBox Detector[M]//Leibe B, Matas J, Sebe N, et al. Computer Vision – ECCV 2016: Vol. 9905. Cham: Springer International Publishing, 2016: 21-37. http://link.springer.com/10.1007/978-3-319-46448-0_2. doi: 10.1007/978-3-319-46448-0_2.

[18] Lin T Y, Goyal P, Girshick R, et al. Focal Loss for Dense Object Detection[M]. arXiv, 2018. http://arxiv.org/abs/1708.02002.

[19] Tian Z, Shen C, Chen H, et al. FCOS: A simple and strong anchor-free object detector[M]. arXiv, 2020. http://arxiv.org/abs/2006.09214.

[20] Redmon J, Divvala S, Girshick R, et al. You Only Look Once: Unified, Real-Time Object Detection[C]//2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR). Las Vegas, NV, USA: IEEE, 2016: 779-788. http://ieeexplore.ieee.org/document/7780460/. doi: 10.1109/CVPR.2016.91.

[21] Redmon J, Farhadi A. YOLO9000: Better, Faster, Stronger[C]//2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR). Honolulu, HI: IEEE, 2017: 6517-6525. http://ieeexplore.ieee.org/document/8100173/. doi: 10.1109/CVPR.2017.690.

[22] Rezatofighi H, Tsoi N, Gwak J, et al. Generalized Intersection Over Union: A Metric and a Loss for Bounding Box Regression[C]//2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). Long Beach, CA, USA: IEEE, 2019: 658-666. https://ieeexplore.ieee.org/document/8953982/. doi: 10.1109/CVPR.2019.00075.

[23] Zheng Z, Wang P, Liu W, et al. Distance-IoU Loss: Faster and Better Learning for Bounding Box Regression[J]. Proceedings of the AAAI Conference on Artificial Intelligence, 2020, 34(07): 12993-13000. https://ojs.aaai.org/index.php/AAAI/article/view/6999. doi: 10.1609/aaai.v34i07.6999.

[24] Gevorgyan Z. SIoU Loss: More Powerful Learning for Bounding Box Regression[M]. arXiv, 2022. http://arxiv.org/abs/2205.12740.

[25] Zhang Y F, Ren W, Zhang Z, et al. Focal and efficient IOU loss for accurate bounding box regression[J]. Neurocomputing, 2022, 506: 146-157. https://linkinghub.elsevier.com/retrieve/pii/S0925231222009018. doi: 10.1016/j.neucom.2022.07.042.