electronics

MDPI

Article

DFS-DETR: Detailed-Feature-Sensitive Detector for Small Object Detection in Aerial Images Using Transformer

Xinyu Cao [1], Hanwei Wang [2], Xiong Wang [1] and Bin Hu [3,*]

[1] School of Information Science and Engineering, Yunnan University, Chenggong District, Kunming 650500, China; xinyucao@stu.ynu.edu.cn (X.C.); wxalbert@stu.ynu.edu.cn (X.W.)
[2] Civil Engineering, The University of Liverpool, Liverpool L3 8HA, UK; sghwan59@liverpool.ac.uk
[3] Department of Computer Science and Technology, Kean University, Union, NJ 07083, USA
* Correspondence: bhu@kean.edu

Abstract: Object detection in aerial images plays a crucial role across diverse domains such as agriculture, environmental monitoring, and security. Aerial images present several challenges, including dense small objects, intricate backgrounds, and occlusions, necessitating robust detection algorithms. This paper addresses the critical need for accurate and efficient object detection in aerial images using a Transformer-based approach enhanced with specialized methodologies, termed DFS-DETR. The core framework leverages RT-DETR-R18, integrating the Cross Stage Partial Reparam Dilation-wise Residual Module (CSP-RDRM) to optimize feature extraction. Additionally, the introduction of the Detail-Sensitive Pyramid Network (DSPN) enhances sensitivity to local features, complemented by the Dynamic Scale Sequence Feature-Fusion Module (DSSFFM) for comprehensive multi-scale information integration. Moreover, Multi-Attention Add (MAA) is utilized to refine feature processing, which enhances the model's capacity for understanding and representation by integrating various attention mechanisms. To improve bounding box regression, the model employs MPDIoU with normalized Wasserstein distance, which accelerates convergence. Evaluation across the VisDrone2019, AI-TOD, and NWPU VHR-10 datasets demonstrates significant improvements in the mean average precision (mAP) values: 24.1%, 24.0%, and 65.0%, respectively, surpassing RT-DETR-R18 by 2.3%, 4.8%, and 7.0%, respectively. Furthermore, the proposed method achieves real-time inference speeds. This approach can be deployed on drones to perform real-time ground detection.

Keywords: aerial images; small object detection; deep learning; transformer; feature fusion

check for updates

Citation: Cao, X.; Wang, H.; Wang, X.; Hu, B. DFS-DETR: Detailed-Feature-Sensitive Detector for Small Object Detection in Aerial Images Using Transformer. *Electronics* **2024**, *13*, 3404. https://doi.org/10.3390/electronics13173404

Academic Editor: Seong G. Kong

Received: 31 July 2024
Revised: 21 August 2024
Accepted: 22 August 2024
Published: 27 August 2024

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

1. Introduction

With the rapid development of drone technology, a large amount of aerial images with subsurface information are produced. By leveraging advanced image-processing techniques, real-time monitoring of ground vehicles and other potential obstacles can be achieved, thereby providing critical support for urban planning, security surveillance, and environmental protection, among other areas. Despite the increasing prevalence of real-time detection applications for drones, achieving accuracy and real-time performance remains a challenge.

Object detection in aerial images is of paramount importance. However, it presents considerable challenges. Aerial images frequently present intricate features, such as diminutive target sizes and occlusions among objects, intensifying the difficulty of object detection. Moreover, practical applications necessitate rapid detection capabilities. In aviation, notably in-flight monitoring and navigation, there exists a demand for object-detection systems characterized by real-time performance. These systems must promptly and precisely discern targets amidst swiftly evolving environments to guarantee navigation accuracy.

Traditional object detection in aerial images primarily relies on manual visual inspection and traditional machine vision methods. However, both approaches have inherent limitations. Manual visual inspection is subjective and lacks real-time capability,

while traditional machine vision methods often depend on handcrafted feature extractors, requiring adjustment and optimization based on domain expertise and experience. In recent years, deep learning has garnered significant attention and has been successfully applied across various domains. Deep learning detection methods have emerged as increasingly popular alternatives, offering advantages such as real-time capability, high accuracy, and robustness.

Detailed information is paramount in object detection within aerial images. Given the abundance of intricate details present in aerial images, this detailed information is indispensable for precise object detection and localization. Sensitivity to details assists models in comprehending target features within the image, consequently bolstering detection accuracy and robustness. Notably, object-detection methodologies in aerial images frequently leverage multi-scale feature-fusion techniques to enhance the model's perceptual acuity toward details.

Initially, SSD [1] endeavored to leverage features from various levels of the backbone for object detection, yielding commendable results. Subsequently, FPN [2] effectively mitigates the deficiency of low-level convolutional features and weak semantic information by establishing a top-down lateral pathway. This optimization enhances the multi-scale detection performance by transmitting high-level semantic features from the top to the bottom layers. However, FPN is not without its limitations: signal transmission is unidirectional and does not convey detailed location information to the upper layers. PANet [3] builds another bottom-up pathway on top of the top-down pathway, transmitting detailed location information to the upper layers. Wang et al. [4] proposed the Adaptive Recursive Feature Pyramid (ARFP), comprising a recursive structure, an Efficient Global Context (EGC) bottleneck module, and a Discriminative Feature-Fusion (DFF) module. Zhang et al. [5] proposed a Laplacian feature pyramid network (LFPN), integrating high-frequency information into the multi-scale pyramid feature representation, thus enhancing the accuracy of object detection. Additionally, some researchers integrate novel feature-fusion modules [6], attention mechanisms [7], or dilated convolutional layers [8] into FPN to acquire more discriminative multi-scale feature representations. While this approach can enhance the detection accuracy of multi-scale targets to some extent, improving the detection capability of small-sized targets under complex background interference remains challenging due to the abundance of small-sized targets and the presence of abundant interfering features in aerial images.

Considering the constraints posed by prior enhancement methodologies and the imperative of refraining from incorporating supplementary modules that could considerably impede inference speed, the primary objective of this study is to augment the detection accuracy of aerial images through a heightened utilization of detail-sensitive multi-scale feature fusion, while the secondary objective is to maintain real-time detection capabilities alongside high precision. In this paper, we initially employ the Cross Stage Partial Reparam Dilation-wise Residual Module (CSP-RDRM) to acquire more comprehensive multi-scale contextual information and facilitate efficient feature extraction. Subsequently, the Detail-Sensitive Pyramid Network (DSPN) integrates Parallel Upsample, Parallel Downsample, and the Cross-Parallel Atrous Fusion Module (CPAFM) to extract multi-scale feature forces, thereby enhancing the perception and comprehension of the image details. Following this, the Dynamic Scale Sequence Feature-Fusion Module (DSSFFM) enables the network to comprehensively exploit multi-scale information. Lastly, we introduce NMPDIoU as the bounding box regression loss to enhance the model's localization ability for small targets.

1. We propose a novel CSP-RDRM, which enhances the model's feature-extraction capability while maintaining a real-time performance through reparameterized dilation-wise residual expansion.
2. For the detection of small objects in aerial images, we propose the DSPN, which exhibits heightened sensitivity to local features and detailed information, thereby mitigating information loss during the process of feature fusion.

3. We propose a novel and practical plug-and-play DSSFFM. By dynamically fusing high-level and low-level features and dynamically weighting them based on the importance of different components, as well as integrating them with the encoder's output, this module enhances the model's sensitivity to details.
4. We optimize the MPDIoU for bounding box regression loss using a normalized Wasserstein distance, thereby accelerating the convergence speed of the model and enhancing the detection accuracy of aerial images.

2. Related Work

The evolution of object detection in aerial images can be broadly categorized into three stages: manual visual inspection, traditional machine vision methods, and deep learning detection methods. Deep learning object detection in aerial images offers superior accuracy, robustness, end-to-end training, and adaptability. Consequently, deep learning object detection is gaining increasing popularity. Deep learning object detection is primarily divided into two or one stages.

Two-stage object detection divides the task into two distinct phases: region proposal generation and object classification. Common two-stage object-detection methods include Faster R-CNN [9], Cascade R-CNN [10], and others. While achieving high accuracy, they also encounter a series of challenges such as complex structures, high computational complexity, and inaccurate region proposals, which restrict their application and dissemination in certain scenarios.

Single-stage object detection offers several advantages: end-to-end design and efficient performance and enabling the direct prediction of object categories and positions from images. Common single-stage object-detection methods include the YOLO series, SSD [1], RetinaNet [11], and DETR series. The YOLO series is primarily utilized for real-time, efficient, and accurate applications, featuring typical algorithms such as YOLO [12], YOLOv3 [13], YOLOv5 [14], YOLOv7 [15], YOLOv8 [16], and YOLOv9 [17]. However, the YOLO series is limited by the design of anchor boxes and the necessity to generate candidate regions prior to object classification and position regression, thus not achieving true end-to-end object detection. DETR [18], as the first Transformer-based detector, streamlines the entire process without the requirement for complex candidate box generation and matching in traditional object-detection algorithms, thereby achieving genuine end-to-end design. However, it is accompanied by challenges such as training difficulty, ambiguous semantics of query vectors, complex model structure, high computational complexity, and inferior performance in detecting small objects. RT-DETR [19] addresses issues such as slow inference time and training difficulty by designing an Efficient Hybrid Encoder and efficiently processing multi-scale features through decoupled intra-scale interactions and cross-scale fusion. Overall, compared to two-stage detectors, single-stage detectors strike a better balance between accuracy and speed, rendering them more practical for real-world applications.

Object detection in aerial images has become a challenging task in the field of computer vision in recent years, with significant hurdles including large-scale variations in object sizes and mutual occlusions among objects. The development of datasets such as AI-TOD [20], VisDrone2019 [21], and NWPU VHR-10 [22] has driven progress in aerial images' object detection. Li et al. [23] proposed the Few-Shot Airborne Object-Detection method (FsCIT) based on confidence-IoU collaborative proposal filtering and small object constraint loss. This method integrates the confidence-IoU collaborative proposal filtering scheme into the Region Proposal Network (RPN) to rescue more foreground proposals from the RPN, aiming to address newly introduced categories. However, the main limitation of this method is the lack of a dedicated few-shot object-detection strategy, which affects the detection accuracy of objects in airborne images. In contrast, Ma et al. [24] proposed the Scale Decoupling Module (SDM) that emphasizes small object features by removing large object features in shallow layers, addressing the problem of object scale disparity. The SDM method has made significant progress in enhancing the small-object-detection

capability, but its handling of large object features could limit its performance in comprehensive scenarios. Chen et al. [25] introduced the Coupled Global–Local (CGL) network, which can be seamlessly integrated into traditional detection models to effectively capture additional information from airborne images. The CGL network enhances detection accuracy through multi-scale feature fusion and adaptive convolution methods. However, these techniques increase the model's complexity and computational overhead, which impacts its practical application in resource-constrained environments. Deng et al. [26] proposed the Hierarchical Adaptive Alignment Network (HAA-Net) with a design that includes the Region Refinement Module (RRM), Feature Alignment Module (FAM), and Potential Label Assignment Module (PLAM), addressing misalignment issues at the region, feature, and label levels, respectively. Although HAA-Net performs excellently in resolving misalignment issues, its model typically contains millions of parameters, posing challenges for deployment on resource-limited embedded devices.

The end-to-end aerial images' object-detection algorithm based on Transformer networks is an emerging approach in the field of object detection, showing potential advantages in aerial images. YOLOS [27] introduces a novel design for object detection by integrating the encoder–decoder architecture of DETR with the encoder backbone of ViT. While this approach offers a fresh perspective, its detection performance is still suboptimal, and there is considerable room for improvement in the pretrained representations. Hu et al. [28] proposed Efficient-Matching-Oriented Object Detection with Transformers (EMO2-DETR), which excels in handling objects of varying scales and orientations, thereby enhancing accuracy in aerial images. Despite these advantages, the model struggles with achieving end-to-end comparability with DETR-like methods and exhibits a convergence speed significantly slower than that of CNN-based approaches. Dai et al. introduced the Arbitrary-Oriented Object DEtection TRansformer (AO2-DETR) framework, designed to explicitly generate orientation queries. This framework improves positional priors for feature aggregation and enhances cross-attention in the Transformer decoder. However, AO2-DETR faces challenges such as convergence difficulties and high computational costs. Li et al. [29] developed TransUNetCD, an end-to-end hybrid Transformer model that combines the strengths of Transformers and UNet. This model addresses issues related to redundant information extraction from low-level features in the UNet framework and aims to optimize feature-difference representations while improving inter-layer relationship modeling. Wang et al. [30] proposed the Vision Transformer Detector (ViTDet), which extracts multi-scale features for aerial image detection and can be integrated into various detector architectures. However, in practical applications, particularly in real-time target-detection scenarios, ViTDet's inference speed can be relatively slow.

3. RT-DETR

RT-DETR is the first real-time end-to-end object detector released by Baidu in 2023, which outperforms the state-of-the-art YOLO detector and DETR series detectors in both speed and accuracy. RT-DETR mainly consists of a backbone network, encoder, and decoder. Its major contribution is the adoption of a novel and Efficient Hybrid Encoder that processes multi-scale features by decoupling intra-scale interactions and cross-scale fusion. Although RT-DETR demonstrates a strong overall performance, its efficacy in aerial image detection is limited, primarily due to the following two shortcomings:

1. Despite using a traditional ResNet as the backbone network, the resolution of feature maps gradually decreases with the increase in network depth, making it challenging to capture the features of small objects.
2. In the structure of the Efficient Hybrid Encoder, although deep features are fused with shallow features from ResNet, multiple convolution and upsampling operations result in the loss of local details and positional information in the feature maps, which is detrimental to the detection of small objects. While shallow features in the backbone network contain rich positional information, the Efficient Hybrid Encoder fails to fully leverage this characteristic.

4. DFS-DETR

4.1. Overall Framework

This paper proposes an aerial image real-time object detector based on a Transformer, and its overall framework is shown in Figure 1. DFS-DETR has four main contributions. Firstly, the CSP-RDRM enhances the feature-extraction capability of the backbone network. Secondly, the DSPN utilizes Parallel Upsample and Parallel Downsample techniques to employ multiple pathways for feature extraction, thereby enriching the diversity of feature representation. Additionally, it integrates a Cross-Parallel Atrous Fusion Module, which improves the network's feature-extraction capabilities while preserving feature resolution. Next, the DSSFFM allows the network to fully leverage multi-scale information by dynamically fusing detailed features from the backbone with higher-order features. The introduction of Multi-Attention Add incorporates distinct attention mechanisms to process the input, merging their outputs to enrich the model's understanding and representation of the input. Finally, the optimization of the loss function with NMPDIoU enables a more precise quantification of the similarity between target and predicted bounding boxes, facilitating the model's acquisition of more reliable object-detection boundaries.

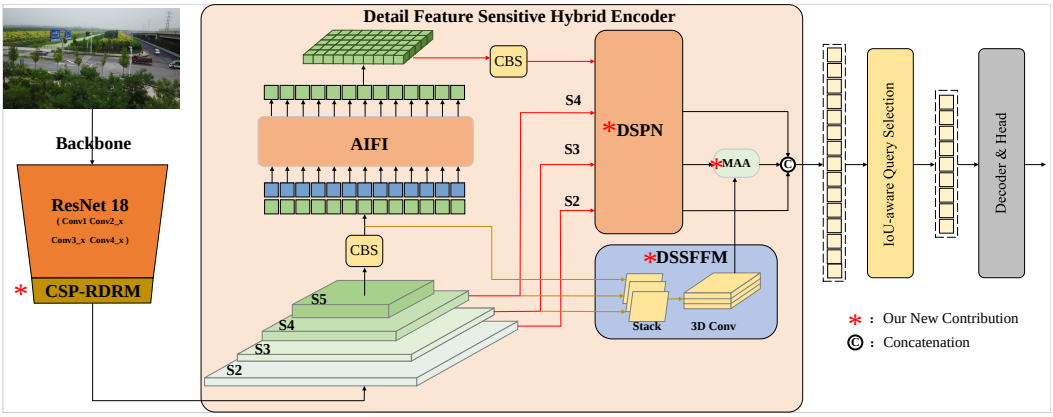


Figure 1. The structure of DFS-DETR.

4.2. CSP-RDRM

The core of the CSP-RDRM is the reparameterized dilated residual module (RDRM), which adopts an efficient method for obtaining multi-scale contextual information. It is primarily designed using a residual approach, as shown in Figure 2.

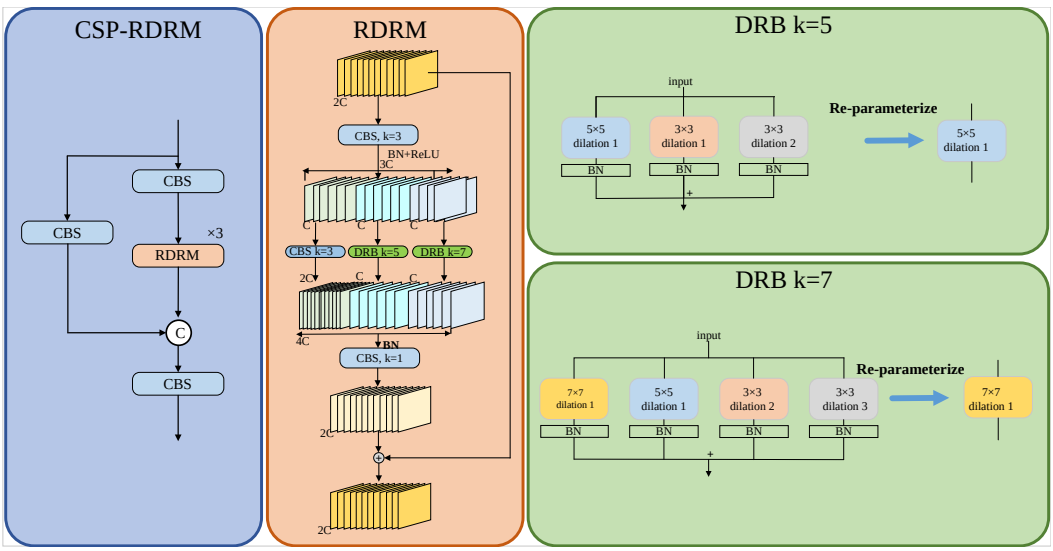


Figure 2. The structures of RDRM, CSP-RDRM, and DRB.

The design concept and structure of the RDRM: within the module, a two-step method is used to efficiently draw multi-scale contextual information, followed by the fusion of feature maps generated from multi-scale receptive fields.

First step: The initial step involves generating relevant residual features from the input features, resulting in a series of concise feature maps with different region sizes. This step primarily involves a 3×3 convolution, followed by batch normalization (BN) and ReLU layers. The 3×3 convolution is used for initial feature extraction, while the ReLU activation function plays a crucial role in activating region features and making them concise.

Second step: The reparameterized dilated block (DRB) [31] is employed to enhance the morphological filtering of features from different region sizes using the reparameterization concept. By learning the required concise regional feature maps and reverse matching the receptive field based on its size, the DRB effectively achieves this operation. To accomplish this, the regional feature maps are divided into several groups, and then different groups undergo reparameterized dilated blocks. The schematic diagram of the reparameterized dilated block is shown in Figure 2, where an expanded small kernel transformation layer is used to enhance the unexpanded large kernel layer. Through the use of structural reparameterization, after training, the BN layer is merged into the convolution layer so that the small kernel convolution can be effectively merged into the large kernel convolution during inference.

Through the aforementioned two steps, the reparameterized dilated block can efficiently obtain multi-scale contextual information. After drawing multi-scale contextual information, aggregation is performed on multiple outputs. Specifically, all feature maps are concatenated together, followed by batch normalization and pointwise convolution for feature merging, ultimately forming the RDRM.

The CSP-RDRM is an improvement of the CSP structure, where we replace the bottleneck in the CSP structure with the reparameterized dilated residual module (RDRM), as depicted in Figure 2. By replacing Conv 5_x of ResNet 18 with the CSP-RDRM, more multi-scale contextual information and efficient feature extraction can be obtained.

4.3. Detail-Sensitive Pyramid Network (DSPN)

4.3.1. The Intuition and Motivation Behind the Design of the DSPN

In RT-DETR, the Efficient Hybrid Encoder utilizes the cross-scale feature-fusion module for multi-scale feature fusion, which significantly impacts the final detection task. Improving the Efficient Hybrid Encoder is an effective approach to enhance the detection performance of RT-DETR and serves as the primary focus of this study.

The key to improving RT-DETR's detection capability for small objects lies in enhancing its sensitivity to detail. As the network depth increases, the positional information of detail gradually diminishes. Furthermore, the cross-scale feature-fusion module in the Efficient Hybrid Encoder is located at a deeper position in the model. Therefore, enhancing the Efficient Hybrid Encoder's detection accuracy for small objects will be crucial for boosting the overall detection performance of RT-DETR.

In previous works, enhancing the detail-sensitive aspect was mainly achieved through attention mechanisms. These mechanisms assist the network in focusing its attention on crucial details, thus improving the model's perceptual capabilities regarding these details. By introducing attention mechanisms, the network can more effectively learn and utilize details, enhancing its ability to recognize small-scale objects or subtle structures, thereby improving the overall performance. Poorly designed attention mechanisms may result in an excessive focus on local information while neglecting global context, leading to a decrease in the model generalization ability or an insufficient perception of overall features.

How to enhance sensitivity to details within the cross-scale feature-fusion module is a question that needs to be addressed. To improve the detection of small objects across multiple scales, there is a need to develop a novel Pyramid Attention Network that can increase sensitivity to details.

4.3.2. The Architecture of the DSPN

The DSPN adopts a Parallel Upsample and Parallel Downsample, providing multiple feature-extraction pathways for the network. Additionally, the CPAFM efficiently extracts features at different scales by employing parallel dilated convolutions with different dilation rates. In this section, we introduce the architecture of the developed DSPN and discuss its characteristics. The proposed DSPN structure is illustrated in Figure 3.

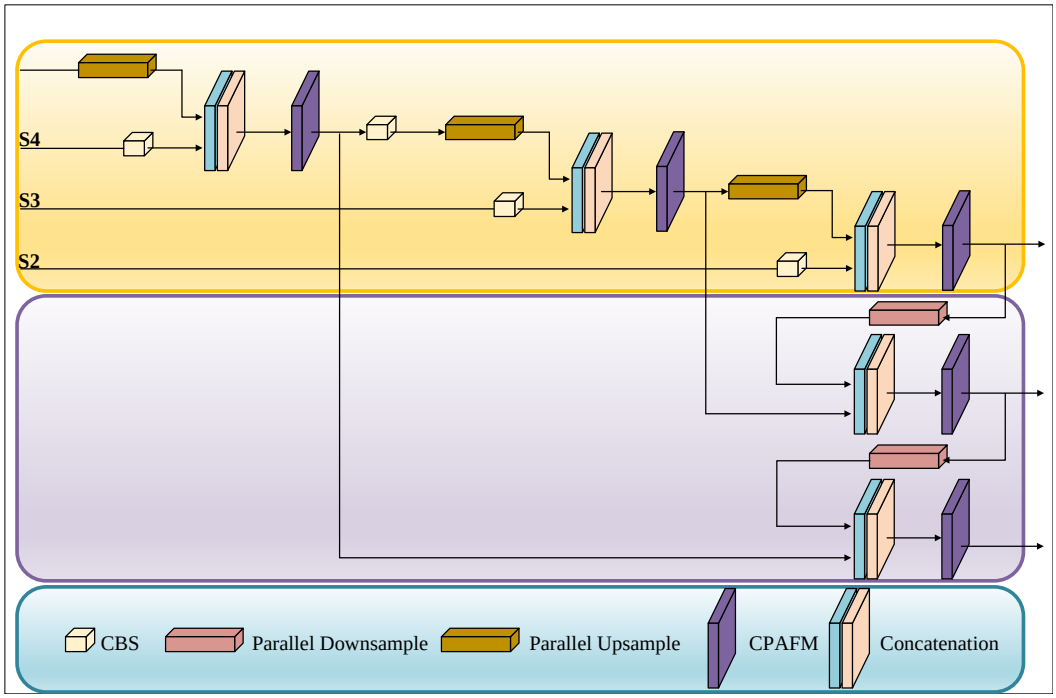


Figure 3. The structure of DSPN.

4.3.3. Parallel Upsample and Parallel Downsample

RT-DETR employs a traditional Upsample and Downsample, primarily used to adjust the resolution of images. This traditional approach facilitates the network in acquiring feature information across multiple scales and extracting features at various levels. However, it performs poorly in prioritizing important features and optimizing feature representation, often resulting in a significant loss of detailed information. Therefore, we propose Parallel Upsample and Parallel Downsample, whose structural diagram is illustrated in Figure 4.

We illustrate the Parallel Upsample process. In Path one, the input feature map undergoes upsampling using Nearest Neighbor interpolation, effectively doubling its height and width. Subsequently, the feature map's channels are halved through CBS. Path two entails reducing the size and dimensions of the feature layer via Transpose. Path three involves passing the input feature map through an AdaptiveAvgPool layer, resulting in a feature map with dimensions of 1×1. Following this, it traverses a CBS layer and is finally activated using the *hardsigmoid* activation function, thereby normalizing the feature map. It is important to note that different applications may necessitate various activation functions, and selecting the appropriate one is pivotal in neural network training and testing. The *hardsigmoid* function is mathematically defined as follows: the path three result is considered as the learned weights for dynamically adjusting the concatenation results of paths one and two. This approach enhances sensitivity to subtle features.

$$hardsigmoid(x) = \begin{cases} 0 & x < -2.5 \\ 0.2x + 0.5 & -2.5 \leq x \leq 2.5 \\ 1 & x > 2.5 \end{cases} \quad (1)$$

In summary, Parallel Upsample and Parallel Downsample provide multiple pathways for feature extraction, enriching the diversity of feature representation. Path three is then utilized to perform feature selection on the sampled features, enhancing more meaningful features while suppressing redundant or irrelevant ones, thereby improving the effectiveness of feature representation.

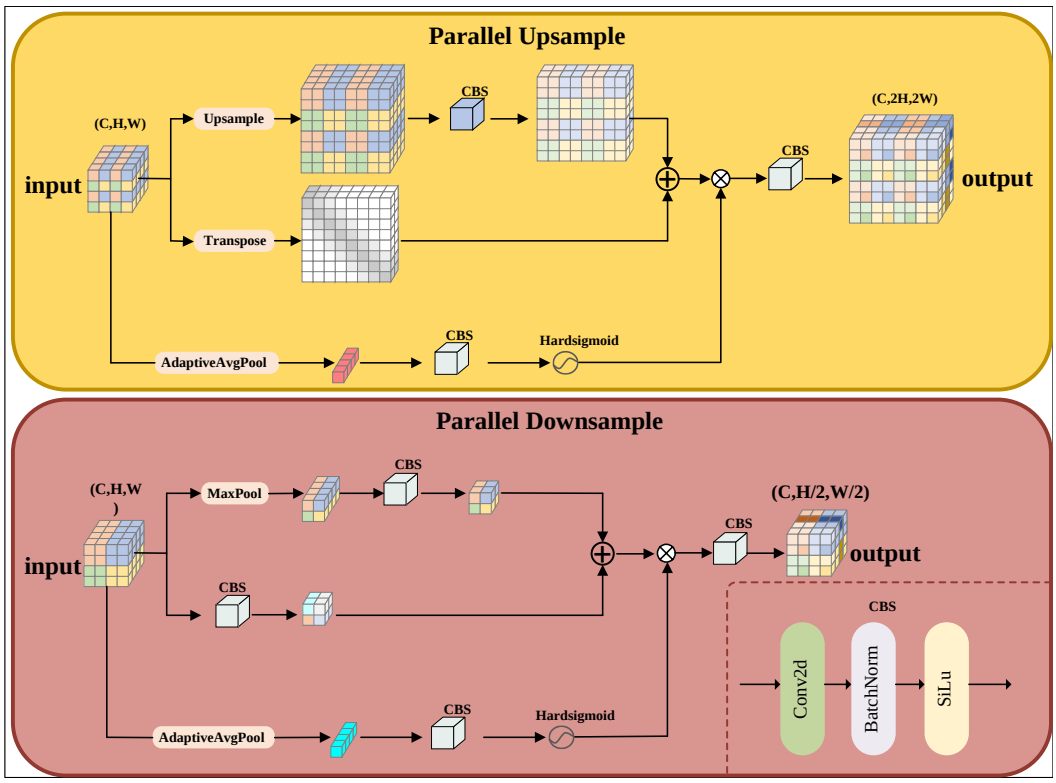


Figure 4. The structure of Parallel Upsample and Parallel Downsample.

4.4. Cross-Parallel Atrous Fusion Module (CPAFM)

Atrous convolution is a commonly used technique in CNNs, which enlarges the receptive field by introducing holes between convolutional kernel elements. This enables the network to effectively capture information over a wider range. A larger receptive field plays a crucial role in detecting small objects, making atrous convolution more effective compared to regular convolution.

In atrous convolution, the dilation rate determines the sampling interval of the convolutional kernel in the atrous convolution. Different dilation rates can achieve different scales of receptive fields. Our improved Parallel Atrous Convolution (PAC) utilizes atrous convolutions with dilation rates of 1, 2, and 3 simultaneously to obtain feature representations at multiple scales, as shown in Figure 5. This parallel operation enables the neural network to better capture both local and global information in the data, thereby enhancing the network's ability to represent complex patterns.

Through Parallel Atrous Convolutions, neural networks can acquire feature representations simultaneously across multiple scales. These representations effectively capture intricate details and the overarching structures of target objects across various scales. The integration of multi-scale information significantly enhances the network's capacity for object recognition and precise localization, thereby improving its generalization performance and robustness. In contrast to traditional single-scale convolutions, Parallel Atrous Convolutions mitigate the loss of feature resolution, thereby further augmenting the network's performance.

In summary, PAC employs Parallel Atrous Convolutions with diverse dilation rates to proficiently extract features across multiple scales. This facilitates the network in capturing

both local details and contextual information, thereby augmenting its capability to represent intricate patterns.

We integrated the refined Parallel Atrous Convolution into the CSP architecture, yielding the Cross-Parallel Atrous Fusion Module (CPAFM), and the structure is illustrated in Figure 5. In comparison to RT-DETR's RepC3, the CPAFM facilitates superior feature information acquisition across diverse scales. This enhancement bolsters the network's feature-extraction prowess while circumventing any compromise in feature resolution, thereby elevating its generalization performance and robustness.

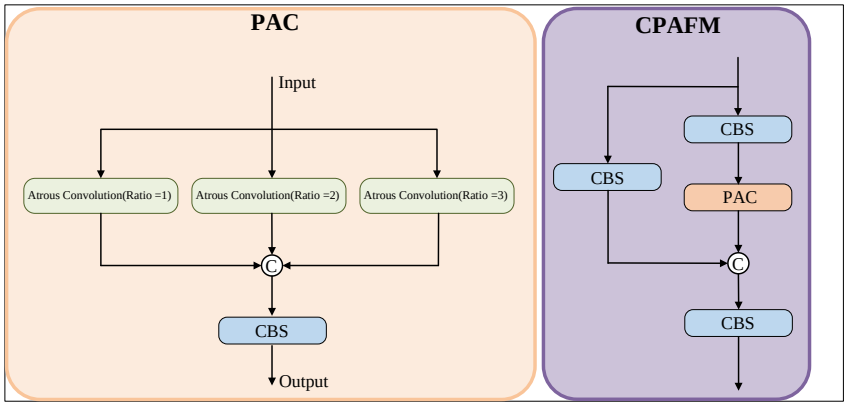


Figure 5. The structure of PAC and CPAFM.

4.5. Dynamic Scale Sequence Feature-Fusion Module (DSSFFM)

4.5.1. The Structure of DSSFFM

We employ the Dynamic Scale Sequence Feature-Fusion Module (DSSFFM) to augment the network's capability to detect small objects by integrating high-level and low-level features. During the Image-Downsample process, the size of shallow feature maps varies, while the scale-invariant features remain consistent. S2 represents the high-resolution feature level, encompassing a significant portion of crucial information for object detection. The DSSFFM is devised based on the S2 level. Initially, a two-dimensional Gaussian filter is applied to smoothen the input image, yielding an output image with identical resolution but varying scales, yielding

$$F_0(w,h) = G_0(w,h) \times f(w,h) \tag{2}$$

$$G_0(w,h) = \frac{1}{2\pi\sigma^2} e^{-\frac{w^2+h^2}{2\sigma^2}} \tag{3}$$

where $f(w,h)$ represents the two-dimensional input image with width w and height h. $F_0(w,h)$ is generated by smoothing through a series of convolutions using the two-dimensional Gaussian filter $G_0(w,h)$. σ is the scaling parameter for the standard deviation of the two-dimensional Gaussian filter used for convolution. Below outlines the design rationale of the DSSFFM:

1. Utilize pointwise convolution to uniformly adjust the channel dimensions of three feature maps from the shallow layer to a predetermined number of channels.
2. Employ dysample for dynamic upsampling of the two larger-sized feature maps from the preceding step, ensuring uniformity in size across all three feature maps.
3. Implement the unsqueeze operation to append an extra dimension to the original three-dimensional tensor, transitioning it from a three-dimensional tensor (height, width, and channel) to a four-dimensional tensor (depth, height, width, and channel).
4. Concatenate the 4D feature maps along the depth dimension to generate a unified 3D feature map.
5. Conclude by employing 3D convolution, batch normalization, and the SiLU activation function to execute sequence feature extraction effectively.

4.5.2. Multi-Attention Add (MAA)

To integrate the detailed features of the DSSFFM into the results of the encoder, we use Multi-Attention Add (MAA) to add the output of the DSSFFM to the encoder results with the same channels through attention mechanisms. It undergoes both channel attention and local attention sequentially. The structure is illustrated in Figure 6.

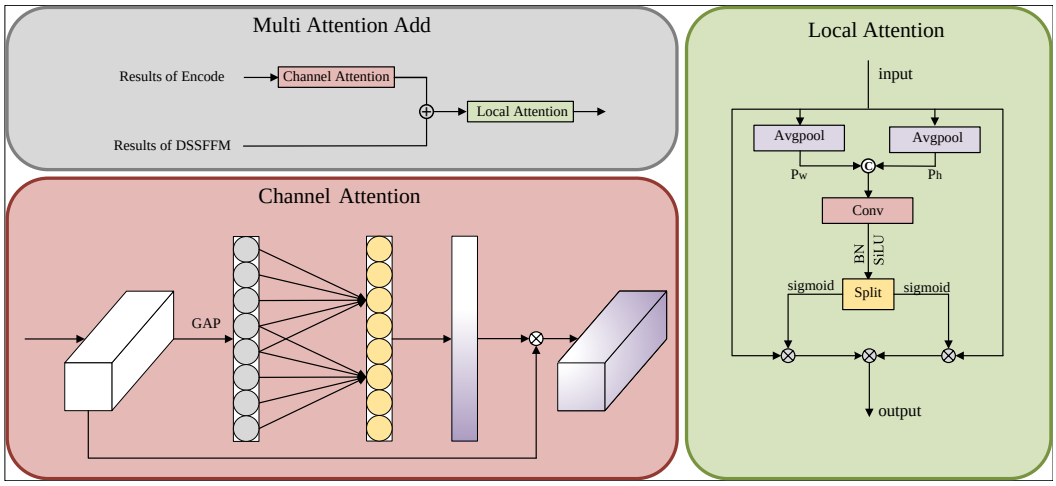


Figure 6. The structure of Multi-Attention Add.

Channel attention, known as Efficient Channel Attention [32], is designed to model the channel dimension effectively, thereby enhancing the model's efficiency in utilizing channel features. ECA captures channel correlations, amplifies crucial features, and reduces computational costs. It implements channel attention weighting in a straightforward and efficient manner, consequently enhancing the model's efficiency and inference speed.

Local attention combines the output features of Efficient Channel Attention with those of the DSSFFM to seamlessly integrate detailed multi-scale feature information. Merging the output of channel attention with the DSSFFM features not only provides complementary insights but also facilitates the extraction of crucial positional information from each unit through local attention. In comparison to channel attention, local attention initially partitions the input feature map based on its width and height, subsequently processing them independently along the horizontal axis (pw) and vertical axis (ph) for feature encoding. Eventually, these encoded features undergo pooling along the axes to retain the spatial structural information of the feature map and are ultimately merged to yield the output.

In summary, channel attention and local attention within Multi-Attention Add utilize distinct attention mechanisms to process the input, with their outputs subsequently merged to enrich the model's comprehension and representation of the input. Unlike simple addition operations (e.g., x = input1 + input2), Multi-Attention Add offers the advantage of discerning the significance of various input components and accordingly adjusting their weighting. Through this adaptive approach, the model gains greater flexibility in handling inputs and effectively captures crucial information, thereby enhancing its performance and generalization capabilities.

4.6. Improvement of Loss Function

The loss function of RT-DETR includes bounding box regression loss and classification loss. The initial model uses GIoU loss as the bounding box regression loss function. The definition of GIoU loss is as follows: suppose we have the coordinates of the predicted box and the ground truth box, denoted as $B^p = (x_1^p, y_1^p, x_2^p, y_2^p)$ and $B^g = (x_1^g, y_1^g, x_2^g, y_2^g)$:

$$L_{\text{GIoU}} = 1 - \text{GIoU} \tag{4}$$

$$\mathrm{GIoU} = \mathrm{IoU} - \frac{A^C - U}{A^C} \tag{5}$$

$$\mathrm{IoU} = \frac{I}{U} = \frac{I}{A^g + A^p - I} \tag{6}$$

where A^g represents the area of B^g and A^p represents the area of B^p, calculated as $A^g = (x_2^g - x_1^g)(y_2^g - y_1^g)$ and $B^g = (x_2^p - x_1^p)(y_2^p - y_1^p)$, respectively. A^c represents the area of the smallest bounding box that contains both A^g and B^g. GIoU loss cannot be optimized when the predicted box has the same aspect ratio as the ground truth box but with completely different width and height values. However, this situation is common in object detection, so we introduce the more effective MPDIoU to overcome this limitation. MPDIoU is a loss based on the minimum point distance, which can achieve a faster convergence speed and more accurate regression results. The corresponding formulas are shown in (7) and (8):

$$L_{\mathrm{MPDIoU}} = 1 - \mathrm{MPDIoU} \tag{7}$$

$$\mathrm{MPDIoU} = \mathrm{IoU} - \frac{d_1^2}{h^2 + w^2} - \frac{d_2^2}{h^2 + w^2} \tag{8}$$

where $\mathrm{IoU} = \frac{I}{U}$, $d_1^2 = (x_1^p - x_1^g)^2 + (y_2^p - y_1^g)^2$, and $d_2^2 = (x_2^p - x_2^g)^2 + (y_2^p - y_2^g)^2$. w and h represent the width and height of the input image, respectively.

MPDIoU [33] accelerates the convergence speed and improves the accuracy compared to GIoU. However, for a large number of small objects in aerial images, MPDIoU shows some limitations. Therefore, a normalized Wasserstein-distance-based NWD position regression loss function [34] is introduced to optimize MPDIoU, resulting in the NWD-MPDIoU (NMPDIoU) loss function. NWD is insensitive to scale transformations, making it more suitable for calculating the similarity between predicted boxes and ground truth boxes of small objects in aerial images. NWD uses a two-dimensional Gaussian distribution to calculate the similarity between predicted boxes and ground truth boxes and calculates their normalized Wasserstein distance according to Formula (10). Finally, the calculated Wasserstein distance is combined with MPDIoU to obtain the final loss, as shown in Formula (11). The parameter 'a' can be flexibly adjusted based on the number of small objects in aerial images to achieve the best performance:

$$NWD(N_a, N_b) = \exp\left(-\frac{\sqrt{W_2^2(N_a, N_b)}}{C}\right) \tag{9}$$

$$W_2^2(N_a, N_b) = \left\| \left(\left[cx_a, cy_a, \frac{w_a}{2}, \frac{h_a}{2} \right]^T, \left[cx_b, cy_b, \frac{w_b}{2}, \frac{h_b}{2} \right]^T \right) \right\|_2^2 \tag{10}$$

$$Loss = \alpha \cdot L_{MPDIoU} + (1 - \alpha) \cdot NWD \tag{11}$$

5. Experimental Results and Analysis

5.1. Dataset

VisDrone2019 Dataset: Collected by the AISKYYE team from Tianjin University's Machine Learning and Data Mining Laboratory. It was captured using drones from multiple angles and scenes in various cities across China. The VisDrone2019 dataset has several characteristics: large scale, rich diversity, high resolution, and multitasking. It provides strong support for drone-application research and is widely used in academia and industry. The VisDrone2019 dataset consists of images and their corresponding annotation files, including 6471 training set images, 548 validation set images, and 1610 test set images. The image sizes range from 2000 × 1500 to 480 × 360, and each image contains a large number of detected objects, with 2.6 million labeled boxes. There are instances of occlusion and overlap between objects. The dataset contains 10 categories: pedestrian (C1), people (C2), bicycle (C3), car (C4), van (C5), truck (C6), tricycle (C7), awning tricycle (C8), bus (C9),

and motorbike (C10). The images were captured from drone flights conducted at altitudes ranging from several hundred meters to several thousand meters, utilizing optical systems primarily from DJI's Mavic and Phantom series (3, 3a, 3SE, 3P, 4, 4a, and 4P).

AI-TOD Dataset: This dataset is designed for small object detection in aerial images and includes 28,036 images with a total of 700,621 instance objects across eight categories. Each original image is sized at 800 × 800 pixels, with an average target size of 12.8 pixels. The dataset uses 11,214 images for training, 2804 images for validation, and 14,018 images for testing. The dataset includes eight categories: airplane (Al), bridge (BR), storage tank (ST), ship (SH), swimming pool (SP), vehicle (VE), person (PE), and windmill (WM).

NWPU VHR-10 Dataset: This dataset includes 715 color images obtained from Google Earth, with spatial resolutions ranging from 0.5 m to 2 m. Additionally, it contains 85 full-resolution color infrared (CIR) images from the Vaihingen dataset, with a spatial resolution of 0.08 m. The Vaihingen data are provided by the German Society for Photogrammetry, Remote Sensing, and Geoinformation (DGPF). The dataset encapsulates a spectrum of target categories, encompassing airplanes (c1), ships (c2), oil tanks (c3), baseball diamonds (c4), tennis courts (c5), basketball courts (c6), ground track fields (c7), harbors (c8), bridges (c9), and vehicles (c10), thereby encompassing a total of 10 distinct classes. We partitioned the NWPU VHR-10 dataset into 640 images for training and 160 images for validation, respectively.

5.2. Experimental Environment and Parameter Configuration

In the experiment, the flight altitude of the UAV ranges from several hundred meters to several thousand meters, and the primary platform used is the DJI Mavic, Phantom series (3, 3a, 3SE, 3P, 4, 4a, and 4P). We used Win11 as the operating system and PyTorch 2.0.1, CUDA 11.8, and cuDNN 11.8 as the desktop computing software environment. The experiment utilized an NVIDIA GeForce RTX 4080 graphics card as the hardware. Consistent hyperparameters were maintained during the training, testing, and experimental validation processes. The optimizer used was AdamW. The RT-DETR dependency library, ultralytics, version 8.0.157 was employed, and the main training parameters are listed in Table 1.

Table 1. Training parameter settings.

Parameters	Setup
lr0	0.0001
lrf	1.0
Momentum	0.9
Weight decay	0.0001
Batch size	4

To comprehensively evaluate the model's accuracy, this paper adopts *recall* (R), *precision* (P), average precision (AP), and mean average precision (mAP) as the main evaluation metrics. Precision represents the proportion of true positive samples among all samples predicted as positive by the model, while recall indicates the proportion of true positive samples detected by the model among all actual positive samples. The formulas are as follows:

$$precision = \frac{TP}{TP + FP} \times 100\% \tag{12}$$

$$recall = \frac{TP}{TP + FN} \times 100\% \tag{13}$$

where TP denotes positive samples correctly predicted as positive, FP denotes negative samples incorrectly predicted as positive, TN denotes negative samples correctly predicted as negative, and FN denotes positive samples incorrectly predicted as negative. AP represents the precision–recall curve of the model across various confidence thresholds,

calculated by integrating the area under the PR curve. mAP is a composite metric obtained by averaging the AP values, expressed by the following formula:

$$mAP = \frac{AP}{N} = \frac{\int_0^1 p(r)dr}{N} \tag{14}$$

where p represents precision, r represents recall, and N is the number of classes. mAP50 denotes the average precision calculated at an Intersection over Union (IOU) threshold of 0.5. mAP represents the mean average precision calculated as the average precision over the range of IOU thresholds from 0.5 to 0.95. In addition, we classify objects into three categories based on their area: objects with an area ranging from 0 to 32 square pixels are designated as "small", objects with an area between 32 and 96 square pixels are categorized as "medium", and objects with an area exceeding 96 square pixels are classified as "large".

To evaluate the complexity of the model, parameter size and FPS are used as evaluation metrics.

5.3. Selection of Baseline Detection Framework

Under the same experimental hardware and environment, we evaluated the RT-DETR models of different sizes based on their accuracy (P, R, mAP50, and mAP), frames per second (FPS), and Params. From Table 2, it can be observed that the detection performance of RT-DETR-R101 is the best, but it has 56.5M more Params than RT-DETR-R18 (76.6 vs. 20.1). The mAP50 and mAP of RT-DETR-R18 are lower compared to other model sizes, but its Params and FPS are better than those of other models. As real-time detection is required for aerial images, we chose the RT-DETR-R18 model due to its more prominent Params and FPS.

Table 2. The performance evaluation of different baseline models on the VisDrone2019-test. The bold numbers in the table indicate the best results.

Method	P	R	mAP50	mAP	Params/M	FPS
RT-DETR-R18	55.6	38.8	37.6	21.8	**20.1**	**110**
RT-DETR-R34	56.1	39.8	38.6	22.4	30.2	94
RT-DETR-R50	57.3	41.2	39.9	23.3	42.9	73
RT-DETR-R101	**57.9**	**41.5**	**40.3**	**23.8**	76.6	45
RT-DETR-L	57.3	40.2	39.0	22.9	32.9	93

5.4. Comparison Experiment of Loss Functions

To validate the superiority of NMPDIoU, we conducted comparative experiments using NMPDIoU and several mainstream loss functions on RT-DETR-R18 under the same experimental hardware and environment, as shown in Table 3. From the table, it can be observed that when the improved NMPDIoU is used as the bounding box regression loss, the model exhibits a better performance, surpassing MPDIoU and other IoU metrics. The use of the enhanced NMPDIoU results in a 0.6% higher mAP compared to the original GIoU model, as well as a 0.8% higher mAP50, indicating the effectiveness of the improvement.

5.5. Ablation Experiment

We conducted meticulous ablation experiments utilizing RT-DETR-R18 as the foundational detection framework, as depicted in Table 4. The substitution of our developed CSP-RDRM for the baseline backbone notably enhanced the network's detection precision. Furthermore, replacing the original CCFM with MAPN resulted in a noteworthy 1.7% augmentation in mAP50 and Params also being appropriately reduced. This enhancement stemmed from MAPN's heightened sensitivity to local features and detailed information, thereby mitigating information loss during the process of feature fusion. Subsequent integration of the DSSFFM into the model yielded a further improvement of 0.8% in mAP50, showcasing an amplified capacity for the dynamic fusion of both high-level and low-level

features, while concurrently bolstering sensitivity toward nuanced details. Lastly, the incorporation of NMPDIoU into the model culminated in a refined performance enhancement.

Table 3. The comparison of detection results with different loss functions introduced by RT-DETR-R18. The bold numbers in the table indicate the best results.

Loss Function	P	R	mAP50	mAP
GIoU [35]	55.6	38.8	37.6	21.8
DIoU [36]	56.0	38.8	37.8	22.0
CIoU [36]	55.9	**39.0**	37.7	21.9
SIoU [37]	55.4	39.3	38.0	22.0
Inner GIoU [38]	55.3	38.6	37.4	21.6
Inner DIoU [38]	55.3	38.8	37.8	22.0
Inner CIoU [38]	55.4	38.9	37.5	21.6
Inner EIoU [38]	55.0	38.7	37.8	21.9
MPDIoU [33]	54.6	38.8	37.8	22.0
NMPDIoU	**55.5**	38.9	**38.4**	**22.4**

Table 4. Ablation experiments of DFS-DETR on the VisDrone2019-test.

CSP-RDRM	MAPN	DSSFFM	NMPDIoU	P	R	mAP50	mAP	Params/M
				55.6	38.8	37.6	21.8	20.1
✓				55.3	39.0	37.8	22.1	21.3
✓	✓			55.9	41.2	39.5	22.9	19.5
✓	✓	✓		56.9	41.4	40.3	23.7	19.8
✓	✓	✓	✓	57.2	41.7	40.7	24.1	19.8

5.6. Comparison between DFS-DETR and RT-DETR-R18

To delve deeper into understanding the efficacy of DFS-DETR and the baseline model RT-DETR-R18 in aerial image object detection, we present the PR curves of both approaches in Figure 7. Here, the x-axis of the PR curve denotes recall, while the y-axis represents precision, and the area under the curve corresponds to the mAP. Notably, the area encompassed by DFS-DETR surpasses that of RT-DETR-R18. On the VisDrone2019 dataset, the overall mAP@0.5 stands at 0.47%, exhibiting a 3.4 percentage points improvement over RT-DETR-R18. It is pivotal to underscore that within the VisDrone2019 dataset, the categories characterized by small objects encompass bicycle, van, tricycle, awning tricycle, and motor. Noteworthy from the PR curve analysis is the conspicuous enlargement of enclosed areas by DFS-DETR in detecting these five categories of defects when compared to RT-DETR-R18.

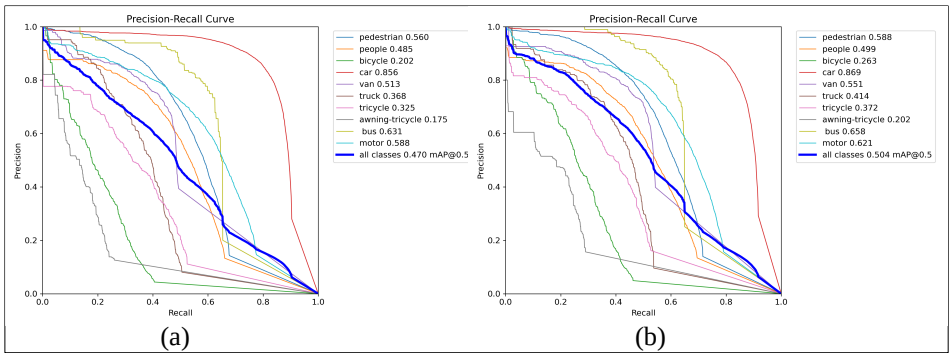


Figure 7. The PR curves for RT-DETR and DFS-DETR on the VisDrone2019 dataset. (**a**) The PR curves of DFS-DETR on the VisDrone2019 dataset. (**b**) The PR curves of DFS-DETR on the VisDrone2019 dataset.

To provide a more vivid comparison of the detection accuracy between the two models, we introduced Figure 8. In the figure, different colors of boxes represent different detection

outcomes: true positive (TP) are depicted by green boxes, false positive (FP) by blue boxes, and false negative (FN) by red boxes. In this visual representation, green boxes indicate objects accurately detected by the model. By observing Figure 8, it is evident that DFS-DETR exhibits a superior detection performance compared to RT-DETR-R18.

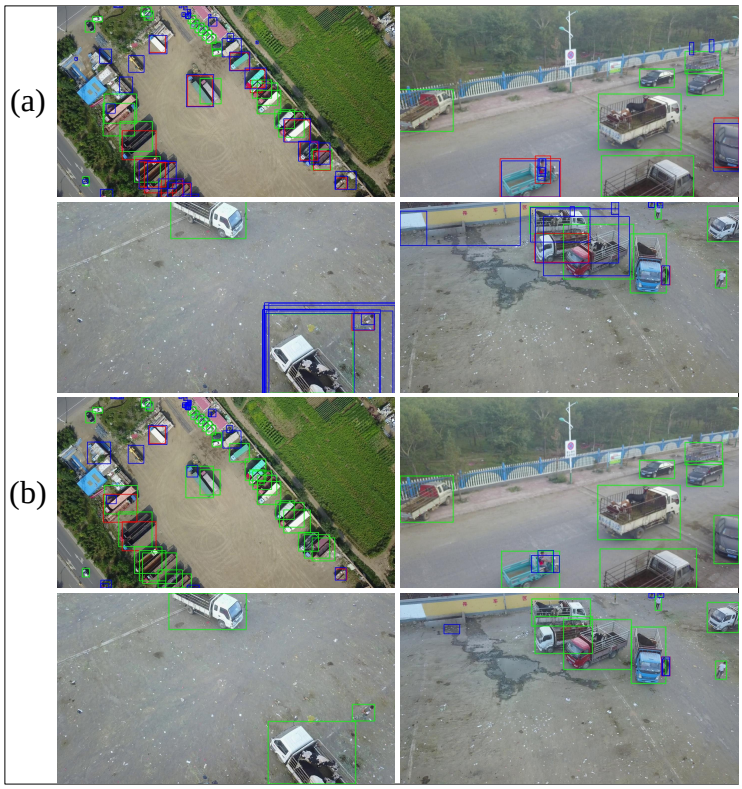


Figure 8. Visualization of detection results using DFS-DETR and RT-DETR-R18. (**a**) represents RT-DETR-R18 and (**b**) represents DFS-DETR.The green, blue, and red boxes are indicative of true positive (TP), false positive (FP), and false negative (FN) predictions, respectively.

In order to visually compare the background-suppression capabilities of DFS-DETR and RT-DETR-R18, we introduced GradCAM, as shown in Figure 9. GradCAM provides an intuitive and easy way to visualize which regions of the feature map the model is focusing on. The darker the gradient color in the feature map, the more attention the model pays to that region, while lighter gradient colors indicate less sensitivity. From the figure, it can be observed that DFS-DETR performs better than RT-DETR-R18, with more focus on the detected objects and higher confidence scores assigned to the corresponding bounding boxes.

5.7. Comparative Experiments

To demonstrate the advantages of DFS-DETR in aerial image detection, we compared it with two-stage object detectors (Cascade R-CNN) and one-stage object detectors (RetinaNet, CornerNet, YOLOv5, YOLOv7, YOLOv8, MS-YOLO, DETR, DAB-DETR, DINO, and RT-DETR) that converged on the VisDrone2019 training set and were evaluated on the VisDrone2019 test set, as shown in Table 5. The categories C1–C10 in the table represent the following specific meanings: pedestrian, people, bicycle, car, van, truck, tricycle, awning tricycle, bus, and motor. We categorize algorithms into two main groups: one primarily for real-time object detection, namely the YOLO series, and the other focusing on accuracy while disregarding the Params and FPS. From the table, it can be observed that DFS-DETR exhibits an excellent performance in terms of detection accuracy, with superior AP values across all categories and a higher overall mAP compared to most models. Particularly noteworthy is its outstanding performance in detecting C2 (people), C4 (car),

and C10 (motor). Regarding the real-time performance, we exclusively compare DFS-DETR with real-time object-detection models based on the Params and FPS. It can be seen from the table that DFS-DETR has fewer Params than the baseline model RT-DETR-R18 and slightly lower FPS, yet it still meets the requirements for real-time detection. To visually illustrate the detection performance of DFS-DETR, we randomly selected four images from the VisDrone2019 test set for visualization, as shown in Figure 10. DFS-DETR outperforms the other three detectors in detecting small objects and objects with occlusions.

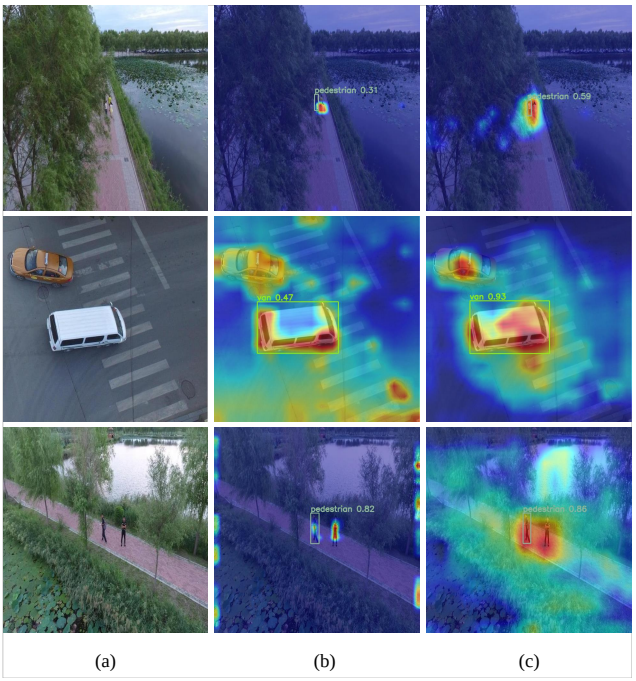


Figure 9. Illustrates the heatmaps of RT-DETR-R18 and DFS-DETR. (**a**) represents the original image, (**b**) represents RT-DETR-R18, and (**c**) represents DFS-DETR, where darker color indicates stronger focus of the model.

Table 5. Results of experiments conducted with various algorithms on the VisDrone2019-test. The best results are marked in bold.

Method	Object Category										mAP	Params	FPS
	C1	C2	C3	C4	C5	C6	C7	C8	C9	C10			
RetinaNet [11]	9.91	2.92	1.32	28.99	17.82	11.35	10.93	8.02	22.21	7.03	12.05	-	-
Cascade R-CNN [10]	16.28	6.16	4.18	37.29	20.38	17.11	14.48	12.37	24.31	14.85	16.74	-	-
FPN [2]	15.69	5.02	4.93	38.47	20.82	18.82	15.03	10.84	26.72	12.83	16.92	-	-
CornerNet [39]	**20.43**	6.55	4.56	40.94	20.23	20.54	14.03	9.25	24.39	12.10	17.3	-	-
YOLOv5m [14]	10.90	5.69	2.98	44.50	21.80	18.30	7.59	7.69	35.10	9.85	16.40	21.2	76
YOLOv5l [14]	12.00	6.41	3.66	46.50	23.40	20.90	8.57	9.82	36.80	11.00	17.90	46.5	70
YOLOv7-tiny [15]	5.61	3.66	1.22	34.60	12.60	8.25	1.73	2.03	22.50	5.74	9.80	**6.0**	**149**
YOLOv8m [16]	12.80	5.99	4.81	47.60	26.70	29.20	12.00	12.00	42.50	13.60	20.70	25.9	90
YOLOv8l [16]	13.10	5.75	5.15	48.60	27.90	31.60	12.70	12.30	44.20	14.20	21.60	43.7	73
MS-YOLO-s [40]	14.80	8.20	5.67	48.90	27.40	26.00	12.10	13.70	41.00	15.00	21.30	9.6	92
MS-YOLO-m [40]	16.80	9.57	8.13	**51.30**	**32.00**	**32.10**	15.50	15.10	45.00	17.70	24.30	27.7	72
RT-DETR-R18 [19]	16.00	10.80	5.66	49.70	25.90	27.70	11.80	10.90	42.40	17.30	21.80	20.1	110
DETR [18]	2.41	1.23	1.01	21.70	10.13	8.92	2.71	1.32	19.29	2.57	7.13	-	-
DAB-DETR [41]	7.88	4.35	5.36	36.40	22.31	20.81	10.01	7.04	34.86	9.26	15.83	-	-
DINO [42]	18.60	10.38	**9.98**	49.01	31.14	30.56	18.25	**16.76**	**45.05**	18.57	**24.83**	-	-
DFS-DETR	18.90	**11.00**	8.80	51.10	28.30	29.70	**15.80**	14.60	43.70	**19.20**	24.10	19.8	89

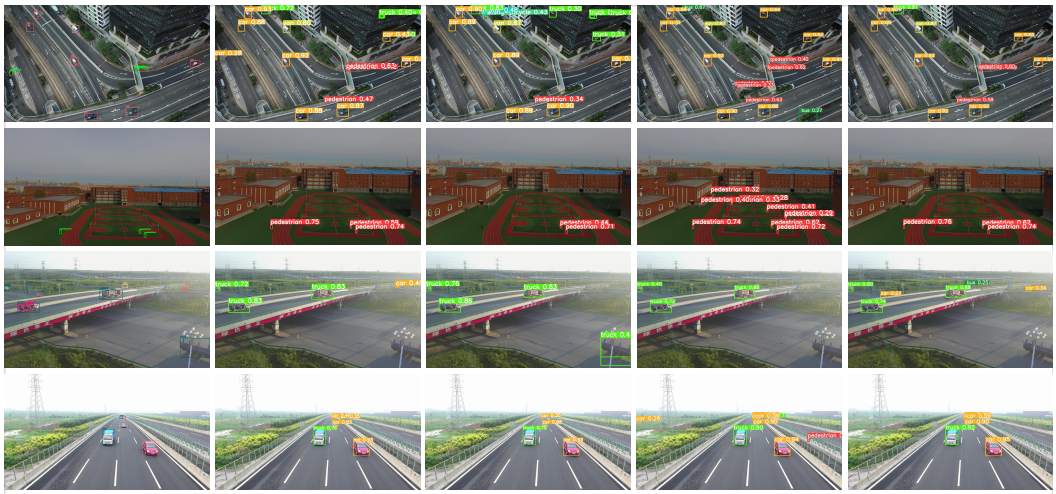


Figure 10. Visual comparison of different object detectors on the VisDrone dataset. (**a**) Ground truth, (**b**) YOLOv5m, (**c**) YOLOv8m, (**d**) RT-DETR-R18, and (**e**) DFS-DETR.

5.8. Generalization to Other Aerial Datasets

5.8.1. Comparison to the AI-TOD Dataset

To demonstrate the generalization ability of DFS-DETR to other aerial datasets, we tested it on the AI-TOD dataset. All algorithms were trained to convergence on the AI-TOD training set and evaluated on the test set, as shown in Table 6. From the table, DFS-DETR performs remarkably well across all categories, with its mAP surpassing that of the baseline model RT-DETR-R18. Additionally, when compared to the latest object-detection algorithm YOLOv8m, DFS-DETR demonstrates a superior performance across various categories. This indicates that DFS-DETR exhibits a high accuracy and generalization ability in object-detection tasks.

The visual detection results of YOLO methods and DFS-DETR under diverse scenes with the same parameter quantity are compared, as shown in Figure 11. It can be observed that DFS-DETR accurately detects objects that were either undetected or misclassified as other categories or uncertain in YOLOv5m, YOLOv8s, and RT-DETR-R18. Objects at small scales in the images are often challenging to detect. DFS-DETR outperforms YOLO methods in this task, demonstrating a superior performance.

Table 6. Performance of different algorithms on the AI-TOD test set. The best results are marked in bold.

Method	Object Category								mAP
	AI	BR	ST	SH	SP	VE	PE	WM	
RetinaNet [11]	0.01	6.62	1.84	20.87	0.06	5.67	1.75	0.53	4.70
Faster R-CNN [9]	22.51	5.07	19.48	19.19	8.99	13.10	4.58	0.04	11.6
Cascade R-CNN [10]	25.17	7.09	22.68	24.23	9.72	15.18	5.35	0.04	13.7
CenterNet [43]	18.59	10.58	27.55	22.27	7.53	18.60	9.17	2.03	14.5
YOLOv3 [13]	7.14	2.60	3.66	10.69	0.61	8.50	2.13	0.40	4.50
YOLOv5m [14]	31.20	14.70	34.60	34.80	**17.80**	26.90	11.90	5.55	22.20
YOLOv8s [16]	35.10	12.10	40.20	34.10	8.25	29.80	10.80	2.28	21.60
YOLOv8m [16]	**39.30**	16.50	**42.20**	38.40	12.10	**31.10**	12.50	3.60	**24.50**
YOLOv7-tiny [15]	14.30	0.10	24.30	19.20	0.06	16.40	4.18	0	9.80
RT-DETR-R18 [19]	23.10	14.50	36.90	35.70	1.28	26.90	11.80	3.78	19.20
DFS-DETR	31.60	**20.00**	40.80	**41.50**	7.90	30.70	**13.90**	**5.76**	24.00

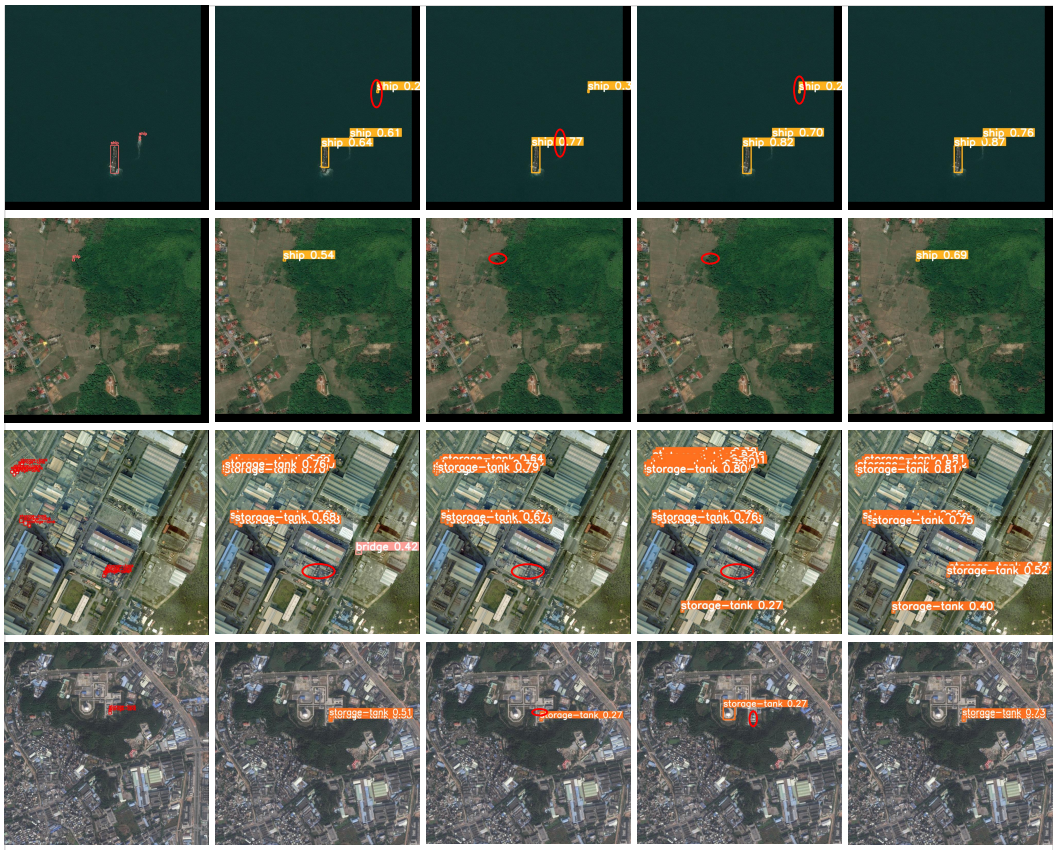


Figure 11. Visual comparison of different object detectors on the AI-TOD dataset. (**a**) Ground truth, (**b**) YOLOv5m, (**c**) YOLOv8m, (**d**) RT-DETR-R18, and (**e**) DFS-DETR. The red circles represent false negatives and false positives.

5.8.2. Comparison to the NWPU VHR-10 Dataset

To verify the performance of DFS-DETR on targets of different scales, we conducted experiments on the NWPU VHR-10 dataset. Table 7 presents a comparison of different algorithms under the same conditions. It is evident from the table that the improved DFS-DETR achieves an outstanding performance in terms of mAP and also obtains satisfactory results in AP across various categories. For visual representations of the difference in accuracy, we conducted visualization operations, as shown in Figure 12. From the figure, it is evident that DFS-DETR has lower false positives and false negatives compared to the baseline model and other object detectors.

Table 7. Performance of different algorithms on the NWPU VHR-10. The best results are marked in bold.

Method	Object Category										mAP	mAP50
	c 1	c2	c3	c4	c5	c6	c7	c8	c9	c10		
YOLOv5-m [14]	64.1	61.1	26.6	75.7	38.6	38.9	81.9	43.5	24.4	53.4	50.8	83.4
YOLOv7 [15]	62.6	63.2	28.5	74.1	49.6	56.7	84.2	44.6	22.1	52.1	53.8	88.7
YOLOv8 [16]	66.4	63.4	32.7	77.4	50.1	61.1	84.3	49.0	28.3	57.7	57.0	89.6
gelan-c [17]	69.2	63.9	**34.5**	78.9	58.3	67.6	86.6	45.8	30.3	**62.8**	59.8	**91.5**
YOLOV9-C [17]	67.8	66.5	34.3	78.1	61.7	68.4	88.9	45.5	27.0	62.5	60.1	90.3
RT-DETR-R18 [19]	64.0	65.4	28.1	77.9	58.2	66.9	87.9	46.9	23.9	61.2	58.0	88.0
DFS-DETR	**77.1**	**70.3**	28.3	**83.2**	**67.3**	**72.0**	**90.3**	**66.3**	**34.0**	60.7	**65.0**	90.7

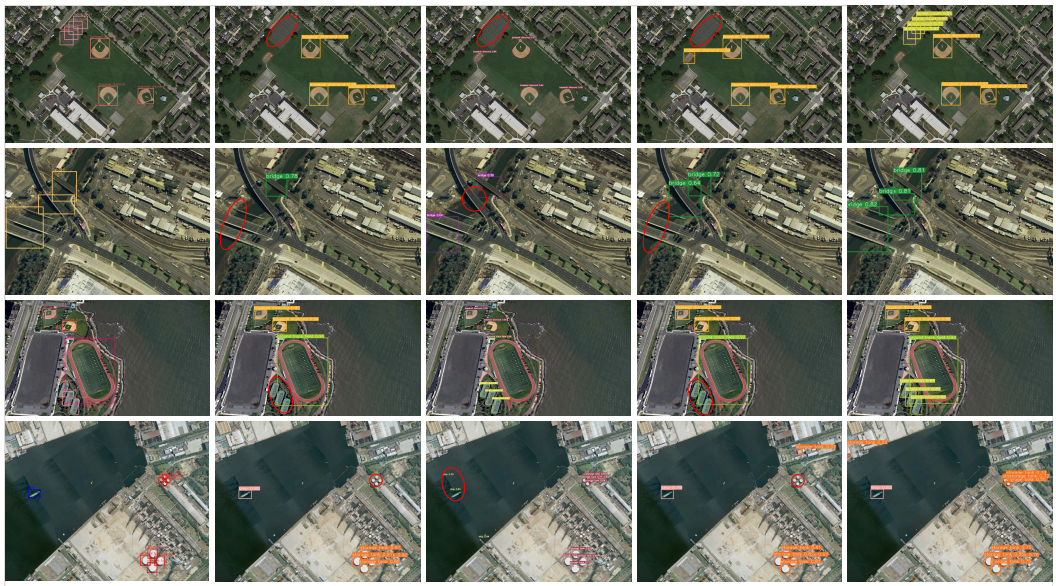


Figure 12. Visual comparison of different object detectors on the NWPU VHR-10 dataset. (**a**) Ground truth, (**b**) YOLOv8m, (**c**) YOLOvv7, (**d**) RT-DETR-R18, and (**e**) DFS-DETR. The red circles represent false negatives and false positives.

6. Conclusions

This paper aims to address prominent challenges in object detection in aerial images. To tackle these challenges, we propose the DFS-DETR network based on RT-DETR-R18. Firstly, we introduce the CSP-RDRM, which adeptly captures complex details in aerial images, thereby enhancing the backbone's feature-extraction capabilities. Secondly, the DSPN module is employed to boost sensitivity to local features, thereby minimizing information loss during the fusion process. Subsequently, the DSSFFM module improves the dynamic fusion of high-level and low-level features, while MAA is utilized to refine feature processing, thereby enhancing the model's ability to understand and represent information more effectively. Finally, the enhanced NMPDIoU accelerates the model's convergence rate and improves the detection accuracy for aerial images. We apply these improvements to RT-DETR-R18, enhancing the performance of object detection in aerial images. The experimental results show that DFS-DETR achieves mAP values of 24.1%, 24.0%, and 65.0% on the VisDrone2019, AI-TOD, and NWPU VHR-10 datasets, respectively, outperforming RT-DETR-R18 by 2.3%, 4.8%, and 7.0%, and other models of similar scale.

Regarding inference speed, our model does not exhibit significant advantages among all the tested models. However, simulation results indicate that the model can meet the requirements of real-time industrial detection. In the future, we will delve into optimizing the model's complexity and researching issues related to training on small datasets. Additionally, we will explore the use of techniques from other domains to enhance the model performance, such as diffusion models and Neural Architecture Search (NAS) technology to reduce network overhead.

Author Contributions: The main manuscript text was written by X.C., with B.H. revising the manuscript and H.W. and X.W. reviewing the manuscript. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Data Availability Statement: The code and data used in this paper can be obtained from the corresponding author through reasonable means.

Acknowledgments: We sincerely thank the corresponding author for their technical and financial support.

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

References

1. Liu, W.; Anguelov, D.; Erhan, D.; Szegedy, C.; Reed, S.; Fu, C.Y.; Berg, A.C. Ssd: Single shot multibox detector. In Proceedings of the Computer Vision—ECCV 2016: 14th European Conference, Amsterdam, The Netherlands, 11–14 October 2016; pp. 21–37.
2. Lin, T.Y.; Dollár, P.; Girshick, R.; He, K.; Hariharan, B.; Belongie, S. Feature pyramid networks for object detection. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July 2017 ; pp. 2117–2125.
3. Liu, S.; Qi, L.; Qin, H.; Shi, J.; Jia, J. Path aggregation network for instance segmentation. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–23 June 2018; pp. 8759–8768.
4. Wang, J.; Yu, J.; He, Z. ARFP: A novel adaptive recursive feature pyramid for object detection in aerial images. *Appl. Intell.* **2022**, *52*, 12844–12859. [CrossRef]
5. Zhang, W.; Jiao, L.; Li, Y.; Huang, Z.; Wang, H. Laplacian feature pyramid network for object detection in VHR optical remote sensing images. *IEEE Trans. Geosci. Remote Sens.* **2021**, *60*, 5604114. [CrossRef]
6. Cheng, G.; He, M.; Hong, H.; Yao, X.; Qian, X.; Guo, L. Guiding clean features for object detection in remote sensing images. *IEEE Geosci. Remote Sens. Lett.* **2021**, *19*, 8019205. [CrossRef]
7. Shi, L.; Kuang, L.; Xu, X.; Pan, B.; Shi, Z. CANet: Centerness-aware network for object detection in remote sensing images. *IEEE Trans. Geosci. Remote Sens.* **2021**, *60*, 5603613. [CrossRef]
8. Yang, X.; Zhang, X.; Wang, N.; Gao, X. A robust one-stage detector for multiscale ship detection with complex background in massive SAR images. *IEEE Trans. Geosci. Remote Sens.* **2021**, *60*, 5217712. [CrossRef]
9. Girshick, R. Fast r-cnn. In Proceedings of the IEEE International Conference on Computer Vision, Santiago, Chile, 7–13 December 2015; pp. 1440–1448.
10. Cai, Z.; Vasconcelos, N. Cascade r-cnn: Delving into high quality object detection. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–23 June 2018; pp. 6154–6162.
11. Lin, T.Y.; Goyal, P.; Girshick, R.; He, K.; Dollar, P. Focal Loss for Dense Object Detection. In Proceedings of the IEEE International Conference on Computer Vision (ICCV), Venice, Italy, 22–29 October 2017.
12. Redmon, J.; Divvala, S.; Girshick, R.; Farhadi, A. You only look once: Unified, real-time object detection. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 27–30 June 2016; pp. 779–788.
13. Redmon, J.; Farhadi, A. Yolov3: An incremental improvement. *arXiv* **2018**, arXiv:1804.02767.
14. Jocher, G.; Stoken, A.; Borovec, J.; Changyu, L.; Hogan, A.; Diaconu, L.; Poznanski, J.; Yu, L.; Rai, P.; Ferriday, R.; et al. ultralytics/yolov5: v3.0. *Zenodo* **2020**. [CrossRef]
15. 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. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Vancouver, BC, Canada, 17–24 June 2023; pp. 7464–7475.
16. Jocher, G.; Chaurasia, A.; Qiu, J. YOLO by Ultralytics. 2023. Available online: https://github.com/ultralytics/ultralytics (accessed on 7 October 2023).
17. Wang, C.Y.; Yeh, I.H.; Liao, H.Y.M. YOLOv9: Learning What You Want to Learn Using Programmable Gradient Information. *arXiv* **2024**, arXiv:2402.13616.
18. Carion, N.; Massa, F.; Synnaeve, G.; Usunier, N.; Kirillov, A.; Zagoruyko, S. End-to-end object detection with transformers. In Proceedings of the European Conference on Computer Vision, Glasgow, UK, 23–28 August 2020; pp. 213–229.
19. Lv, W.; Xu, S.; Zhao, Y.; Wang, G.; Wei, J.; Cui, C.; Du, Y.; Dang, Q.; Liu, Y. Detrs beat yolos on real-time object detection. *arXiv* **2023**, arXiv:2304.08069.
20. Xu, C.; Wang, J.; Yang, W.; Yu, L. Dot distance for tiny object detection in aerial images. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Nashville, TN, USA, 20–25 June 2021; pp. 1192–1201.
21. Du, D.; Zhu, P.; Wen, L.; Bian, X.; Lin, H.; Hu, Q.; Peng, T.; Zheng, J.; Wang, X.; Zhang, Y.; et al. VisDrone-DET2019: The vision meets drone object detection in image challenge results. In Proceedings of the IEEE/CVF International Conference on Computer Vision Workshops, Seoul, Republic of Korea, 27–28 October 2019; pp. 213–226.
22. Cheng, G.; Han, J.; Zhou, P.; Guo, L. Multi-class geospatial object detection and geographic image classification based on collection of part detectors. *ISPRS J. Photogramm. Remote Sens.* **2014**, *98*, 119–132. [CrossRef]
23. Li, L.; Yao, X.; Wang, X.; Hong, D.; Cheng, G.; Han, J. Robust Few-Shot Aerial Image Object Detection via Unbiased Proposals Filtration. *IEEE Trans. Geosci. Remote Sens.* **2023**, *61*, 5617011. [CrossRef]
24. Ma, Y.; Chai, L.; Jin, L. Scale Decoupled Pyramid for Object Detection in Aerial Images. *IEEE Trans. Geosci. Remote Sens.* **2023**, *61*, 4704314. [CrossRef]
25. Chen, X.; Wang, C.; Li, Z.; Liu, M.; Li, Q.; Qi, H.; Ma, D.; Li, Z.; Wang, Y. Coupled Global–Local object detection for large VHR aerial images. *Knowl.-Based Syst.* **2023**, *260*, 110097. [CrossRef]
26. Deng, C.; Jing, D.; Han, Y.; Chanussot, J. Toward Hierarchical Adaptive Alignment for Aerial Object Detection in Remote Sensing Images. *IEEE Trans. Geosci. Remote Sens.* **2023**, *61*, 5615515. [CrossRef]
27. Fang, Y.; Liao, B.; Wang, X.; Fang, J.; Qi, J.; Wu, R.; Niu, J.; Liu, W. You Only Look at One Sequence: Rethinking Transformer in Vision through Object Detection. In Proceedings of the Advances in Neural Information Processing Systems, New Orleans, LA, USA, 6–14 December 2021; Volume 34, pp. 26183–26197.

28. Hu, Z.; Gao, K.; Zhang, X.; Wang, J.; Wang, H.; Yang, Z.; Li, C.; Li, W. EMO2-DETR: Efficient-Matching Oriented Object Detection with Transformers. *IEEE Trans. Geosci. Remote Sens.* **2023**, *61*, 5616814. [CrossRef]
29. Li, Q.; Zhong, R.; Du, X.; Du, Y. TransUNetCD: A Hybrid Transformer Network for Change Detection in Optical Remote-Sensing Images. *IEEE Trans. Geosci. Remote Sens.* **2022**, *60*, 5622519. [CrossRef]
30. Wang, L.; Tien, A. Aerial Image Object Detection with Vision Transformer Detector (ViTDet). In Proceedings of the IGARSS 2023—2023 IEEE International Geoscience and Remote Sensing Symposium, Pasadena, CA, USA, 16–21 July 2023; pp. 6450–6453. [CrossRef]
31. Ding, X.; Zhang, Y.; Ge, Y.; Zhao, S.; Song, L.; Yue, X.; Shan, Y. Unireplknet: A universal perception large-kernel convnet for audio, video, point cloud, time-series and image recognition. *arXiv* **2023**, arXiv:2311.15599.
32. Wang, Q.; Wu, B.; Zhu, P.; Li, P.; Zuo, W.; Hu, Q. ECA-Net: Efficient channel attention for deep convolutional neural networks. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Seattle, WA, USA, 13–19 June 2020; pp. 11534–11542.
33. Siliang, M.; Yong, X. MPDIoU: A loss for efficient and accurate bounding box regression. *arXiv* **2023**, arXiv:2307.07662.
34. Wang, J.; Xu, C.; Yang, W.; Yu, L. A normalized Gaussian Wasserstein distance for tiny object detection. *arXiv* **2021**, arXiv:2110.13389.
35. Rezatofighi, H.; Tsoi, N.; Gwak, J.; Sadeghian, A.; Reid, I.; Savarese, S. Generalized intersection over union: A metric and a loss for bounding box regression. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Long Beach, CA, USA, 15–20 June 2019; pp. 658–666.
36. Zheng, Z.; Wang, P.; Liu, W.; Li, J.; Ye, R.; Ren, D. Distance-IoU loss: Faster and better learning for bounding box regression. In Proceedings of the AAAI conference on artificial intelligence, New York, NY, USA, 7–8 February 2020; Volume 34, pp. 12993–13000.
37. Gevorgyan, Z. SIoU loss: More powerful learning for bounding box regression. *arXiv* **2022**, arXiv:2205.12740.
38. Zhang, H.; Xu, C.; Zhang, S. Inner-IoU: More Effective Intersection over Union Loss with Auxiliary Bounding Box. *arXiv* **2023**, arXiv:2311.02877.
39. Law, H.; Deng, J. CornerNet: Detecting Objects as Paired Keypoints. In Proceedings of the European Conference on Computer Vision (ECCV), Munich, Germany, 8–14 September 2018.
40. Cao, X.; Duan, M.; Ding, H.; Yang, Z. MS-YOLO: Integration-based multi-subnets neural network for object detection in aerial images. *Earth Sci. Inform.* **2024**, *7*, 2085–2106. [CrossRef]
41. Liu, S.; Li, F.; Zhang, H.; Yang, X.; Qi, X.; Su, H.; Zhu, J.; Zhang, L. Dab-detr: Dynamic anchor boxes are better queries for detr. *arXiv* **2022**, arXiv:2201.12329.
42. Zhang, H.; Li, F.; Liu, S.; Zhang, L.; Su, H.; Zhu, J.; Ni, L.M.; Shum, H.Y. Dino: Detr with improved denoising anchor boxes for end-to-end object detection. *arXiv* **2022**, arXiv:2203.03605.
43. Duan, K.; Bai, S.; Xie, L.; Qi, H.; Huang, Q.; Tian, Q. CenterNet: Keypoint Triplets for Object Detection. In Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV), Seoul, Republic of Korea, 27 October–2 November 2019.

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