remote sensing

Article

Small Object Detection Based on Deep Learning for Remote Sensing: A Comprehensive Review

Xuan Wang [1], Aoran Wang [1], Jinglei Yi [1], Yongchao Song [1] and Abdellah Chehri [2,*]

[1] School of Computer and Control Engineering, Yantai University, Yantai 264005, China; xuanwang91@ytu.edu.cn (X.W.); 20195858214@s.ytu.edu.cn (A.W.); jingleiyi@s.ytu.edu.cn (J.Y.); ycsong@ytu.edu.cn (Y.S.)

[2] Department of Mathematics and Computer Science, Royal Military College of Canada, Kingston, ON K7K 7B4, Canada

* Correspondence: chehri@rmc.ca

Abstract: With the accelerated development of artificial intelligence, remote-sensing image technologies have gained widespread attention in smart cities. In recent years, remote sensing object detection research has focused on detecting and counting small dense objects in large remote sensing scenes. Small object detection, as a branch of object detection, remains a significant challenge in research due to the image resolution, size, number, and orientation of objects, among other factors. This paper examines object detection based on deep learning and its applications for small object detection in remote sensing. This paper aims to provide readers with a thorough comprehension of the research objectives. Specifically, we aggregate the principal datasets and evaluation methods extensively employed in recent remote sensing object detection techniques. We also discuss the irregularity problem of remote sensing image object detection and overview the small object detection methods in remote sensing images. In addition, we select small target detection methods with excellent performance in recent years for experiments and analysis. Finally, the challenges and future work related to small object detection in remote sensing are highlighted.

Keywords: artificial intelligence; deep learning; object detection; remote sensing

check for updates

Citation: Wang, X.; Wang, A.; Yi, J.; Song, Y.; Chehri, A. Small Object Detection Based on Deep Learning for Remote Sensing: A Comprehensive Review. *Remote Sens.* **2023**, *15*, 3265. https://doi.org/10.3390/rs15133265

Academic Editors: Dusan Gleich and Dong Liu

Received: 12 May 2023
Revised: 19 June 2023
Accepted: 22 June 2023
Published: 25 June 2023

1. Introduction

The rapid advancement of science and technology has made remote-sensing image technology indispensable for various applications. Some examples of these applications include monitoring for diseases, transportation planning, environmental monitoring, crop harvest analysis, geological surveys, and identifying objects used in military operations [1–6].

The primary objective of object detection and recognition, which is one of the primary challenges in remote sensing, is to locate the items that can be noticed through digital images. However, images obtained using remote sensing include a vast range of scales because they cover such a large area. This results in a variety of object sizes. Because of this, implementing algorithms for object detection in images obtained from remote sensing will always encounter substantial obstacles.

To meet this challenge, this paper introduces a series of efficient small object detection methods to help the reader understand the current development status and choose the appropriate solution for specific problems. So far, the traditional object detection methods are the Histograms of Oriented Gradients (HOG) feature extraction algorithm [7–9], and the Deformable Part Model (DPM) algorithm [10–13].

The HOG algorithm starts by creating a grid out of the input image, then uses the created feature table to create a histogram for each cell of the grid, extracts the region of interest to generate features, and then feeds it into the support vector machines (SVM) classifier so that it can be detected.

The DPM algorithm is an upgrade and extension of the HOG algorithm. It has a more effective technique for finding a solution to the problem of the object's multiple perspectives. On the other hand, because these algorithms are focused primarily on detecting pedestrians, the detection effect on remote sensing images is not very good.

In recent years, Convolutional Neural Networks (CNNs), feed-forward neural networks with a convolutional structure, have been widely used. The aforementioned architecture is proficient in diminishing the amount of memory of deep neural networks. The reduction of parameters in a network and the alleviation of model overfitting can be effectively achieved through the implementation of three fundamental operations: local perceptual fields, weight sharing, and pooling layers.

In general, CNNs have several convolutional and pooling layers. They use alternating convolutional and pooling layers, i.e., one convolutional layer is connected to one pooling layer, and so on. Each neuron of the output feature map in the convolutional layer is locally connected to its input local. The corresponding connection weights are weighted and summed with the local inputs, and bias values are added to obtain the neuron input values.

The CNN is named because the process is equivalent to the convolution process, and the schematic diagram of CNN target detection is shown in Figure. With the development of deep learning, a large number of deep learning-based target detection algorithms have been proposed and have achieved remarkable results on remote sensing image datasets. The principle diagram of CNN object detection is shown in Figure 1. The algorithm for detecting objects in remote sensing images has emerged as a significant area of research, with a multitude of experiments and studies conducted on the subject.

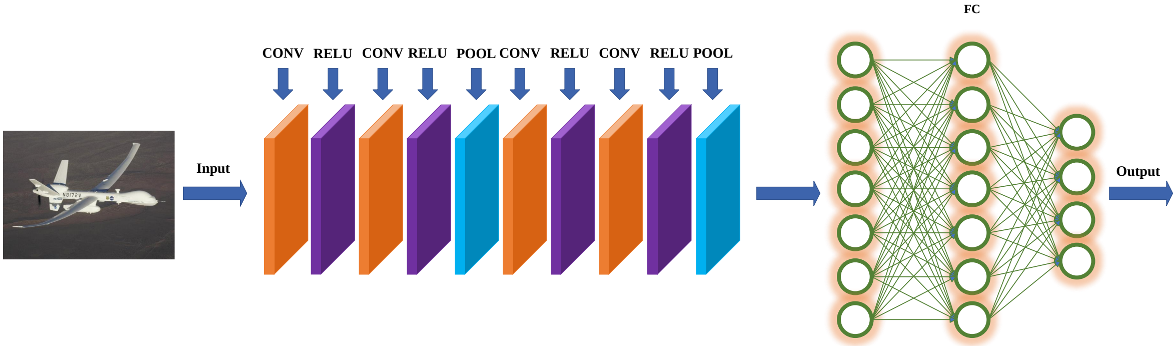


Figure 1. The principle diagram of CNN object detection.

As the pioneering work of object detection algorithms based on deep learning, Region-based Convolutional Neural Networks (RCNNs) [14] successfully link convolutional neural networks with object detection. However, because RCNNs consist of four parts—generating candidate windows, feature extraction, SVM classification, and window regression—the detection efficiency of the algorithm is relatively low. Based on this problem, subsequent SPPNet [15], Fast RCNN [16], Faster RCNN [17], FPN [18], Mask RCNN [19], etc., improved the shortcomings of the previous algorithm to enhance the detector performance. With the introduction of detectors such as the YOLO series [20–23] and SSD [24], the performance of object detection algorithms has been improved, and the technology has been continuously developed.

Several researchers have achieved some results in summarizing the overview of object detection algorithms [25–29]. They mainly review the problems faced by high-resolution object detection and the proposed methodological approaches, remote sensing image datasets, and the performance of the leading detection methods at this time.

This paper provides an in-depth analysis of the remote sensing images and evaluation metrics that are commonly used for object detection, which differs from the existing literature. The article focuses on various categories of object detection techniques, the constraints associated with remote sensing images, and the challenges caused by object irregularities, along with the strategies for addressing them. Additionally, it explores methods for detecting small objects in remote sensing imagery.

The applications of small object detection on remote sensing images, especially rotating small objects, are summarized. We classify the existing processes into six categories based on different technical bases, including more recent techniques within the last two years. In addition, we re-measure the mean Average Precision (mAP), Floating Point Operations (FLOPs), number of parameters (Params), and Frames Per Second (FPS) transmitted for six of the best methods. These algorithms are evaluated on this basis.

In this article, we present a comprehensive review of object identification methods and how those methods have been applied to remote sensing in recent years. In addition, we give a great deal of focus to developing algorithms and applications for detecting small objects. The overview of this paper is shown as Figure 2.

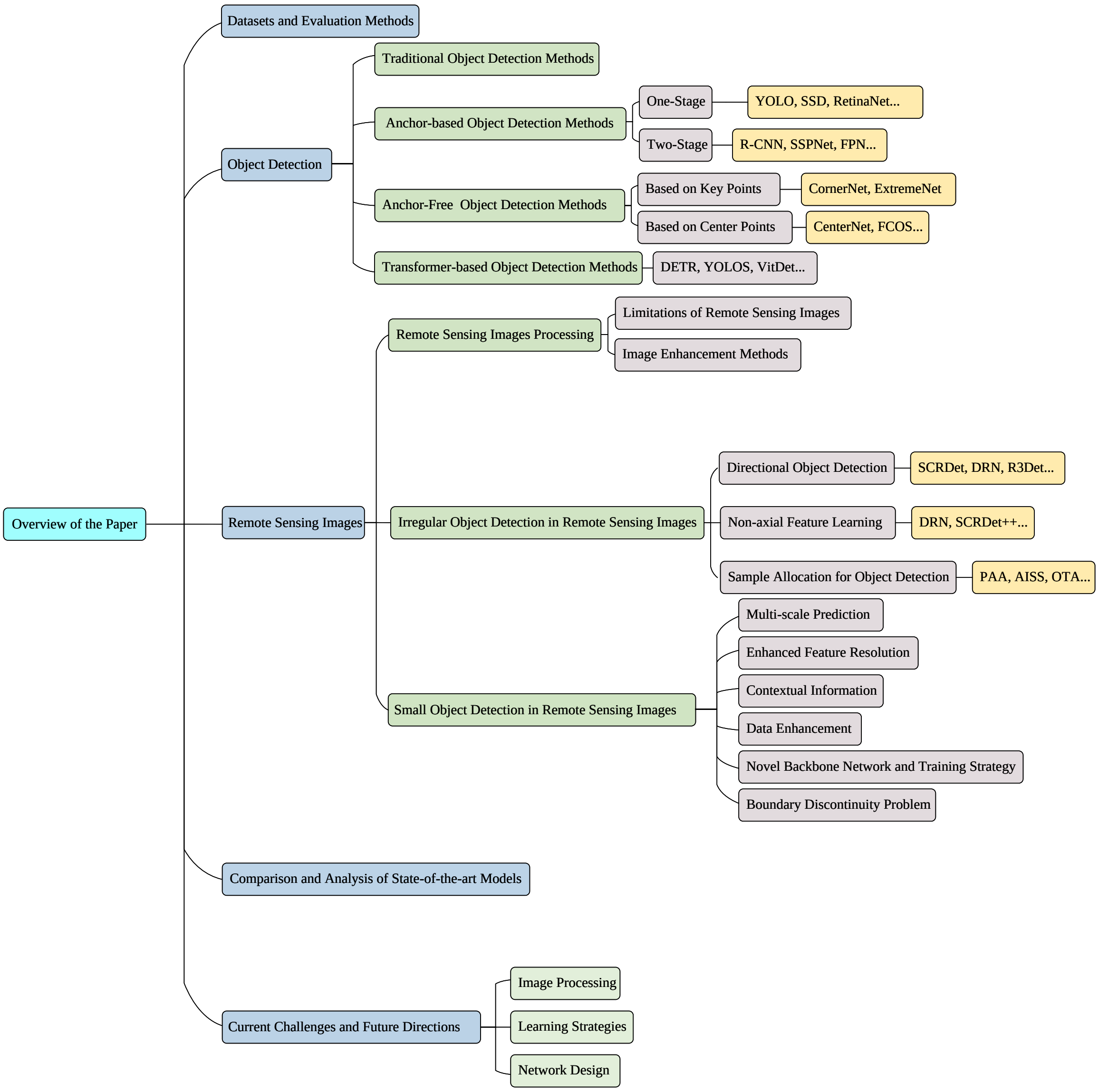

Overview of the Paper
Datasets and Evaluation Methods
Object Detection
Traditional Object Detection Methods
Anchor-based Object Detection Methods
One-Stage
YOLO, SSD, RetinaNet...
Two-Stage
R-CNN, SSPNet, FPN...
Anchor-Free Object Detection Methods
Based on Key Points
CornerNet, ExtremeNet
Based on Center Points
CenterNet, FCOS...
Transformer-based Object Detection Methods
DETR, YOLOS, VitDet...
Remote Sensing Images
Remote Sensing Images Processing
Limitations of Remote Sensing Images
Image Enhancement Methods
Irregular Object Detection in Remote Sensing Images
Directional Object Detection
SCRDet, DRN, R3Det...
Non-axial Feature Learning
DRN, SCRDet++...
Sample Allocation for Object Detection
PAA, AISS, OTA...
Small Object Detection in Remote Sensing Images
Multi-scale Prediction
Enhanced Feature Resolution
Contextual Information
Data Enhancement
Novel Backbone Network and Training Strategy
Boundary Discontinuity Problem
Comparison and Analysis of State-of-the-art Models
Current Challenges and Future Directions
Image Processing
Learning Strategies
Network Design

Figure 2. Hierarchical structure of this paper.

The main contributions of this paper are as follows:

- We present a detailed overview of the process of object detection using deep learning, covering topics such as problem definition, the history of development, the current status of research, datasets, and assessment methodologies.

- We take a comprehensive approach to organize, classify, and compare the various methods for object detection based on the various differentiation principles. The irregularity problem in object detection for remote sensing is addressed using a variety of different approaches and methodologies. The most up-to-date methods, as well as the method of remote sensing photos and the detection of small objects, are addressed here.
- For small object detection methods in remote sensing, we have conducted a detailed literature classification and analysis. We classify small object detection algorithms into six categories, including multi-scale prediction, enhanced feature resolution, contextual information, data enhancement, novel backbone network and training strategy, and boundary discontinuity problem.
- In this paper, we provide an in-depth analysis of the issues and difficulties associated with the detection of small objects in remote sensing images from various viewpoints, and we clarify the future development trends and directions.

The rest of this review is organized as follows. In Section 2, we organize and analyze the commonly used datasets for object detection and their evaluation methods. In Section 3, we focus on different classes of object detection methods. In Section 4, we focus on object detection methods and their applications for remotely sensing images. In Section 5, we compare the classical object detection algorithms and the visualization of the results of small object detection algorithms. In Section 6, We analyze the problems and challenges faced by remote sensing image small object detection from multiple perspectives and clarify the future development trends and directions. Finally, we conclude the research work in Section 7.

2. Datasets and Evaluation Methods

2.1. Datasets

(1) *DIOR dataset* [26]: This is a public dataset for large-scale benchmarking of optical remote sensing image object detection. The dataset contains 20 types of objects, such as airplanes, stadiums, bridges, dams, ports, etc. Its total number is 23,463 images.

(2) *RSOD dataset* [30,31]: This dataset contains much smaller types and images than the DIOR dataset. The dataset includes only four types of objects: aircraft, oil drums, overpasses, and sports fields—a total of 976 images.

(3) *NWPU VHR-10 dataset* [32–36]: The NWPU-RESISC45 dataset was proposed by researchers at Northwestern Polytechnical University, with a total of 45 categories and a total of 31,500 images. The experimental results were not entirely satisfactory due to the low resolution of the images. As a solution, the NWPU VHR-10 dataset was created. The images of this dataset are VHR images with a total of 10 categories, which are widely used in object detection tasks. The disadvantage of this dataset is that there are no small-sized objects marked, so the recognition effect in small object detection algorithms could be better.

(4) *DOTA dataset* [37]: This dataset has a total of 2806 images, and it contains 15 types of objects with various scales, orientations, and shapes, so the detection algorithm using this data is more stringent.

(5) *VEDAI dataset* [38]: This dataset contains a large number of vehicles and is mainly used for remote sensing image vehicle identification. Compared with tubing, sports fields, etc., vehicles are small objects so this dataset can be used for remote sensing image small object detection.

(6) *ITCVD dataset* [39]: This dataset contains images used for remote sensing image vehicle detection, with a total of 23,678 images. It contains 23,543 test images with many vehicle objects, and each vehicle is manually labeled.

(7) *COCO dataset* [40]: This dataset is one of the most commonly used datasets for object detection, especially small object detection. The dataset contains a large number of small objects, a total of 91 types of objects, and the number of images is as high as 328,000.

(8) *UCAS-AOD dataset* [41]: This dataset contains 2819 car images and 3210 aircraft images.
(9) *RSC11 dataset* [42]: This dataset contains 11 similar scene classes, so the classification of scenes becomes difficult.

The Horizontal Bounding Box (HBB) is commonly used to represent objects oriented horizontally in the labeling of datasets. Objects that do not rotate are typically depicted using the Oriented Bounding Box (OBB) method.

The HBB (Hierarchical Bounding Box) requires the box to be oriented perpendicular to the coordinate axis. This orientation restricts the box from fully encompassing partially distorted large objects. The orientation and scale of the box are determined by OBB, which considers the object's shape. The box is not necessarily perpendicular to the coordinate axis. The generated inclusive box is comparatively more compact than the oriented bounding box.

Regarding the creation methods of OBB, the Principal Component Analysis (PCA) method [43] is the dominant method. OBB first uses PCA to obtain the three principal directions of the point cloud to obtain the center of mass and calculate the covariance. The covariance matrix is then obtained, and the eigenvalues and eigenvectors of the covariance matrix are found. Among them, the eigenvectors are the principal directions. In the second step, OBB converts the input point cloud to the origin using the principal directions and the center of mass.

The principal direction coincides with the coordinate system direction to build the enclosing box of the point cloud transformed into the origin. Finally, OBB sets the principal direction and enclosing box to the input point cloud and achieves the final effect by the inverse transformation of the input point cloud to the origin point cloud transformation.

As shown in Table 1, the composition of a benchmark dataset, including the number of objects, classes, instances, and annotation style, significantly affects the training and testing of a model. The effective training of the model can be facilitated by using rich instances, diverse classes, and a suitable annotation style. In Table 1, the classes of DIOR and DOTA are 20 and 15, respectively, and their number of instances is much higher than other datasets. In addition, the annotation style of OBB helps to improve the detection of rotating objects. DIOR uses a combination of both HBB and OBB annotation styles, whereas RSOD and NWPU VHR-10 solely employ the HBB annotation style. Furthermore, DOTA incorporates all OBB. The distinctive characteristics of DIOR and DOTA differentiate them from other remote sensing datasets.

Table 1. Comparison of classical datasets.

Dataset	Amount	Classes	Instance	Annotation Style	Description
DIOR [26]	23,463	20	192,472	HBB + OBB	Aircraft, stadiums, bridges, dams, ports, etc.
RSOD [30]	976	4	6950	HBB	Aircraft, oil drums, overpasses, sports fields
NWPU VHR-10 [32]	800	10	3775	HBB	Aircraft, ships, stadiums, ports, bridges, etc.
DOTA [37]	2806	15	188,282	OBB	Aircraft, vehicles, stadiums, etc.
VEDA [38]	1210	9	3640	OBB	Vehicles
ITCVD [39]	173	1	29,088	OBB	Vehicles
UCAS-AOD [44]	910	2	6029	HBB + OBB	Airplane, car
RSC11 [42]	1213	11	-	Scene Class	Dense forests, grasslands, buildings, ports, etc.

2.2. Evaluation Methods

In this section, We focus on five evaluation metrics commonly used to evaluate object detection performance, including Intersection over Union (IoU), Precision, Recall, Average Precision (AP), and mean Average Precision (mAP).

(1) *IoU*: a detection frame is generated when detecting an object. IoU is the ratio of overlap and union of a priori frame and real frame area. Generally, the threshold is set to 0.5, which is also the threshold for the cross-union ratio. When the value is more

significant than 0.5, the detected object is considered to be detected. The crossover ratio is defined as follows:

$$\mathrm{IoU} = \frac{A \cap B}{A \cup B}. \tag{1}$$

(2) *Precision*: Precision represents the ratio of the model finding the correct sample to the total sample in the prediction result. When the intersection-union ratio is greater than the threshold, the result is classified as True Positive (TP), and vice versa as False Positive (FP). If the detector does not detect an object in the detection frame labeled with the sample, the object is classified as False Negative (FN). Accuracy is defined as follows:

$$\text{Precision} = \frac{\text{True Positive}}{\text{True Positive} + \text{False Positive}} = \frac{\text{True Positive}}{\text{All Observations}}. \tag{2}$$

(3) *Recall*: Recall rate indicates the number of positive samples recovered by the model in the total positive samples, which is an important indicator to measure whether the model is "found all". Recall is defined as:

$$\text{Recall} = \frac{\text{True Positive}}{\text{True Positive} + \text{False Negative}} = \frac{\text{True Positive}}{\text{All Ground Truth}}. \tag{3}$$

(4) *AP* [45]: Average Precision is the precision averaging on a $[0, 1]$ recall. The higher the AP value, the better the detector's detection performance for a certain type of object in the dataset. Average Precision is defined as follows:

$$AP_u = \frac{1}{\Omega_u} \sum_{i \subset \Omega_u} \frac{\sum_{j \subset \Omega_u} h(p_{uj} < p_{ui}) + 1}{p_{ui}}, \tag{4}$$

where Ω_u denotes the Ground Truth result, p_{uj} denotes the location of object j, and $p_{uj} < p_{ui}$ denotes that object j is ranked before item i in the recommendation list.

(5) *mAP* [45]: mAP averages the average accuracy of each class of objects detected by the detector. Higher mAP values indicate better detector performance for the entire dataset. The mean average accuracy is defined as:

$$mAP = \frac{\sum_{u \in U} AP_u}{|U|}. \tag{5}$$

(6) *FPS*: FPS is used to evaluate the target detection speed, i.e., the number of images that can be processed in each second. The higher the FPS, the faster the detection speed of the model.

(7) *FLOPs*: FLOPs refers to the number of floating point operations, which can also be interpreted as computations coming. The smaller the FLOPs, the smaller the complexity of the model.

(8) *Params*: Params represents the number of parameters required by the model. The smaller the Params, the less parameters the model needs and the lighter it is.

3. Object Detection

Object detection has been researched and refined for over two decades. Object identification, considered one of the key directions and fundamental challenges of computer vision, is currently developing along with the requirements of many applications. Throughout its development history, object detection can be divided into two main periods, i.e., traditional object detection algorithms and deep learning-based object detection algorithms.

Deep learning-based object detection algorithms are further divided into several technical branches. A diagram of the method development history is shown in Figure 3. This section mainly discusses the different branches of traditional and deep learning-based object detection algorithms.

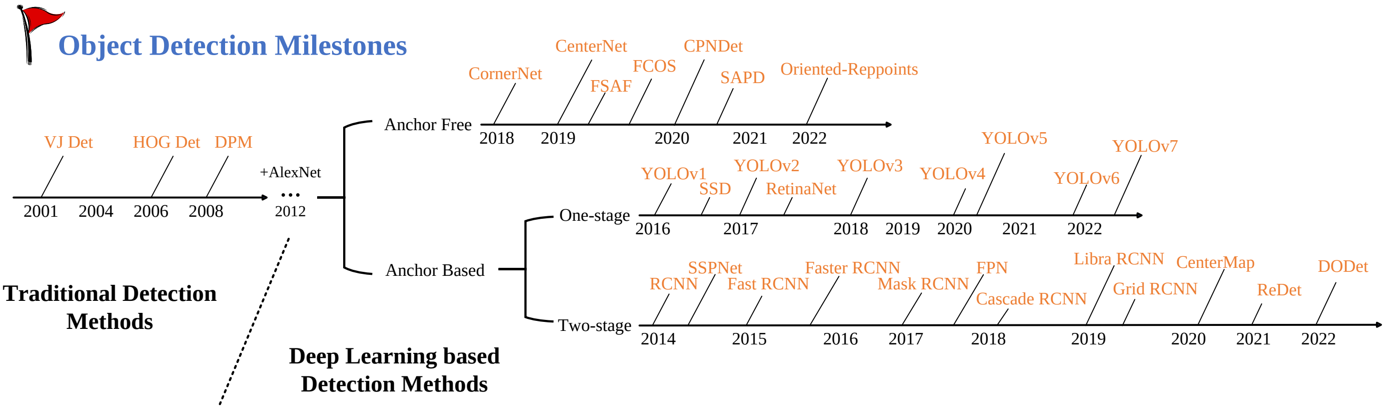


Figure 3. The development history of object detection.

3.1. Traditional Object Detection Methods

In this section, we discuss the Viloa–Jones, HOG, and DPM detectors. The Viloa–Jones detector was proposed in 2001 and is mainly used for face detection. It combines techniques such as integral images, cascade classifiers, and other methods with outstanding performance.

The HOG detector was introduced in 2005. The functionality extends beyond facial recognition. The process is centered on the extraction of features from the object. The DPM detector was conceptualized in 2009 using the detection principle of the HOG detector. The system can detect discrete components of the object and enhance its precision.

The performance of object detectors is inadequate to meet experimental requirements when faced with a large number of images and objects to be detected. The advent of deep learning techniques has led to the emergence of numerous new detectors that outperform traditional object detectors by a significant margin.

3.2. Anchor-Based Object Detection Methods

The Anchor-based object detection methodology aims to produce a multitude of discrete candidate frames for a given pixel, followed by applying filtering, classification, and regression techniques to these frames. This solution offers a partial resolution to the problem of inconsistent object sizes and occlusion.

This technology has the potential to effectively enhance its recall rate when used for the detection of small objects. On the other hand, the approach has the drawback of relying on an extremely high number of manually designed components. In addition, the process of training the test takes a significant amount of time, which results in reduced efficiency [46].

The Anchor-based object detection method contains a one-stage detector and a two-stage detector. The two-stage detector is divided into two steps: (1) extracting the image candidate frames and (2) making corrections for the selected regions to obtain the monitoring point results.

Two types of detectors are used in the Anchor-based object identification method: a one-stage and a two-stage. The two-stage detector comprises two stages, the first of which is the extraction of the picture candidate frames, followed by the second stage, which is making corrections for the selected regions to obtain the monitoring point results. The flow chart for it is presented in Figure 4.

The two-stage detectors mainly include R-CNN, SPPNet, etc. R-CNN is the pioneer in using neural networks to solve object detection problems. However, it has many limitations, such as individually distributed training, independent data storage, and a large number of redundant candidate regions. This can result in a significant time and space overhead [14].

For example, SPPNet [15] solves the problem of object variant or loss due to object scaling. It proposes a spatial pyramid pooling layer (SPP), which allows an image to be convolved only once, avoiding the time overhead due to repeated computations. SPP-

Net has the same limitations as RCNN. Its network cannot achieve end-to-end detection. The schematic diagram of SPPNet is shown in Figure 5.

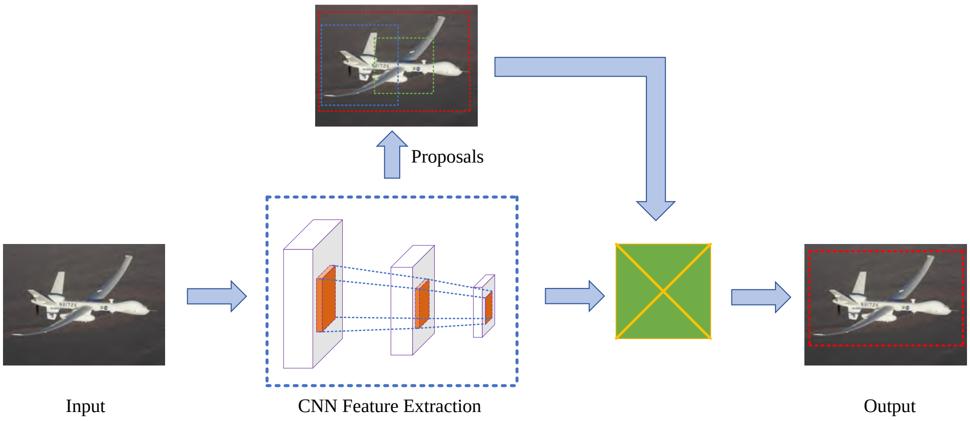


Figure 4. Framework for two-stage object detection.

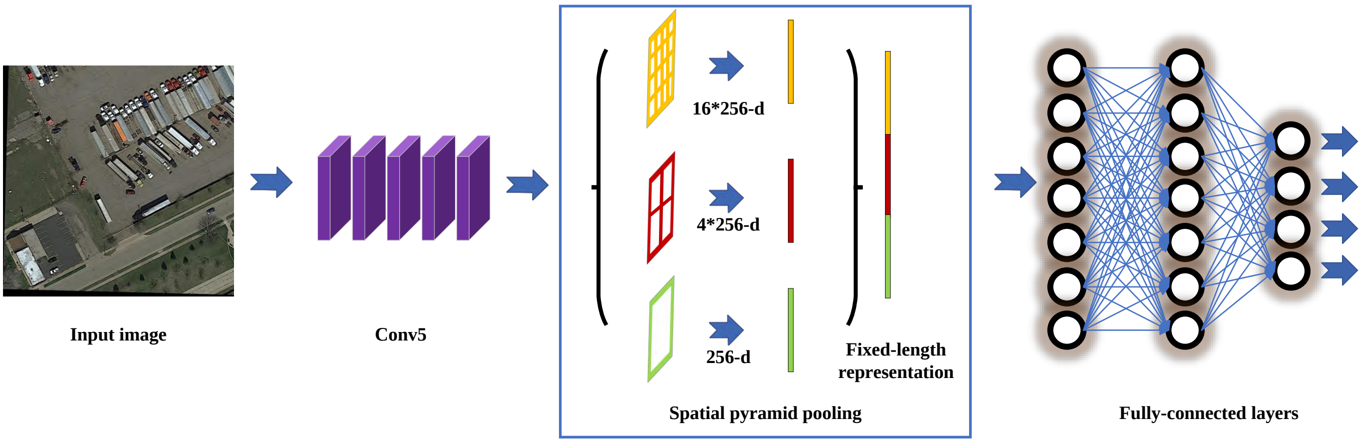


Figure 5. SPPNet Network Structure [15].

The Fast RCNN approach employs a softmax classifier as a means to address the issue of classification synchronization. Additionally, RoI layers are used to facilitate the mapping of multi-scale features, thereby addressing the challenge of scale variation. The multitask loss function of Fast RCNN enables end-to-end training for multitask purposes. Detection is slow due to the intricate algorithm employed by Fast RCNN for selecting candidate regions [16].

Faster RCNN inherits the advantages of Fast RCNN. It innovatively proposes using a region selection network to extract candidate frames, which improves the computational speed. However, it has inaccurate localization frames and cannot effectively identify small objects [17].

FPN extracts multi-scale features of images by constructing feature pyramids at different scales, which significantly improves the network accuracy. Because the network can only be trained for a specific single resolution, it can be contradictory to the multi-scale inference [18]. The Cascade RCNN approach employs a cascade detector to select thresholds merit-based. The proposed solution effectively addresses the issue of overfitting that may arise from implementing high thresholds. However, it should be noted that this approach does not facilitate real-time detection [47].

R-FCN adds a position-sensitive score map to improve the sensitivity of the convolutional network to object position. It solves the problem of object location insensitivity, but there is no improvement in computational speed [48]. Mask RCNN solves the problem of simultaneously localizing, classifying, and segmenting objects. It introduces an instance segmentation branch in order to achieve pixel-level object detection. However, the performance is also lower than the real-time performance due to the high cost of instance segmentation [19].

In extracting multi-scale objects, TridentNet differs from the multi-scale feature pyramid of FPN. It uses a multi-branch structure with different perceptual fields and shares multi-branch structure weights, improving detection accuracy. However, it cannot be monitored in real-time due to its slow detection speed [49].

The one-stage detector can obtain the final detection result directly after only one stage, which is faster than the two-stage detector. Its flow chart is shown in Figure 6. The YOLO series, which is a one-stage detector, has been evolving.

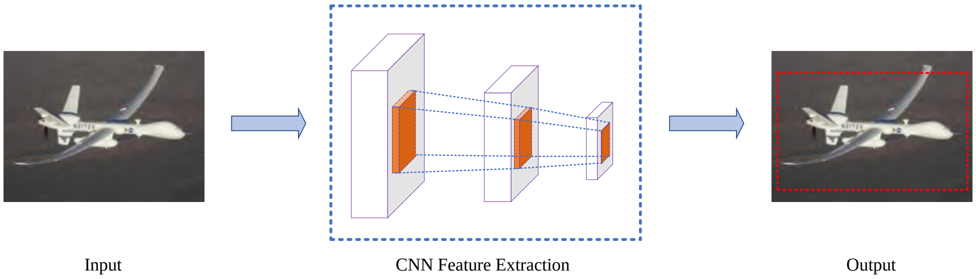


Figure 6. Framework for one-stage object detection.

YOLOv1 is the first to turn the object detection problem into a regression problem. It has a more straightforward network structure and fast detection speed. However, its accuracy of object localization could be significantly higher. When the object is small, or there are multiple objects, the detection effect of YOLOv1 is not good [20].

YOLOv2 further improves detection accuracy and detection speed. However, it does not improve the limitations of YOLOv1 [21]. To address the issue of insufficient detection of small objects, YOLOv3 employs a multi-scale feature map extraction method and an improved classification network. It is ineffective, however, at detecting medium and large objects. [22].

YOLOv4 uses Mosaic and self-adversarial training strategies for data enhancement. It integrates FPN, PAN, and so on, to improve the model performance further [23]. Yolov5 has a slightly worse performance compared to Yolov4. However, it is flexible, fast, and better at rapid model deployment. Yolov6 further improves accuracy and speed. It achieves the highest accuracy so far in real-time detection [50].

SSD [24] uses multi-scale feature map extraction and convolutional feature detection. It is faster and has higher accuracy. However, SSD relies more on manual experience and requires the manual setting of parameters for pre-selected boxes. Therefore, SSD has poor detection accuracy for small objects and multiple objects.

RetinaNet [51] uses the Focal loss function, which solves the problem of category imbalance. However, it cannot perform real-time detection and has poor detection results for small and multiple objects.

EfficientDet [52] proposes a weighted bidirectional feature pyramid network. It is simpler and faster in multi-scale feature fusion. EfficientDet proposes a composite scaling method that simultaneously scales the backbone network's resolution, depth, and width. However, it has a slower detection speed.

3.3. Anchor-Free Object Detection Methods

Anchor-free object detection methods do not require a predetermined anchor. They locate the object by multiple key points or centroids and detect it directly. Specifically, it can be further divided into centroid-based and keypoint-based Anchor-free algorithms.

The centroid-based algorithm couples classification and regression into two subgrids and directly detects the central region and boundary of the object. Specifically, it can be divided into Anchor-free algorithms based on central points and key points. The center point-based algorithm couples classification and regression into two subgrids. It directly detects the central area and boundary of the object. Representative algorithms include Centernet [53], FCOS [54], and TTFNet [55], to name a few.

For example, CenterNet [53] uses a keypoint detection algorithm. It treats detection objects as points and uses center pooling and cascaded corner point pooling. CenterNet is unsuitable for small object and multi-object detection due to the computationally intensive nature of the model.

FCOS uses a fully convolutional network to perform regression operations on the distance from each location of the feature map to the border. Similar to the principle of FCN, it treats each position of each point as a training sample. Compared with the Anchor-based algorithm, FCOS [54] saves a significant amount of memory space during training, which is suitable for instance segmentation.

TTFNetk can be seen as an improved version of Centernet. It uses an elliptical Gaussian kernel to generate negative sample supervised signals and sampling regions around the centroid. While maintaining the performance, TTFNetk [55] reduces the preprocessing operations on the data, thus improving the learning efficiency and the quality of the supervised signal.

Key point-based algorithms are also called corner point-based algorithms. At the object's top left and bottom right two-point positions, the detection frame is formed. Representative algorithms are Cornernet and Extremenet algorithms, among others. They are prone to FP due to the lack of information within the object [56].

Compared with Cornernet, Extremenet [57] uses the top, bottom, left, right, and center five points of the object as key points. Extremenet extracts local information with less noise and more robust features, enabling better detection performance.

3.4. Transformer-Based Object Detection Methods

The Transformer paradigm has experienced a noteworthy proliferation in recent years. In its early stages of development, the Transformer model was primarily subjected to testing with a focus on its application in the field of Natural Language Processing (NLP). The utilisation of the method in the rapidly expanding field of computer vision in modern times is commonly known as Vision Transformer (ViT).

The main Transformer-based object detection algorithms are DETR and YOLOS. The idea of DETR is similar to the traditional object detection methods, but the presentation has significant differences. The traditional Anchor-based method classifies the predefined anchors and regresses the edge coefficients. In contrast, DETR treats object detection as an ensemble to predict. That is, an image sequence is transformed into an ensemble sequence [58].

YOLOS redesigned the detector by combining the encoder–decoder part of DETR with the encoder-only backbone of ViT. YOLOS aims to demonstrate the powerful migration capability of ViT precisely. With only minor modifications, ViT demonstrates excellent flexibility and generalization [59].

For instance, the Swin transformer offers a remedy to the traditionally widespread issue of excessive computational complexity. In contrast to the 16× downsampling offered by ViT, the Swin transformer offers three different feature sizes: 4, 8, and 16×.

In terms of computational effort, the Swin transformer introduces the W-MSA concept. The 4× and 8× downsampling techniques divide the feature map into multiple regions that do not intersect. The MSA operation is carried out within the window so that data can be transferred between windows [60].

4. Remote Sensing Images

The problems concerning object detection of remote sensing images can be divided into two parts: (1) object detection methods and (2) remote sensing image processing.

In Section 3, we thoroughly discussed object detection methods. This section discusses object detection in remote sensing images and related problems.

4.1. Remote Sensing Images Processing

4.1.1. Limitations of Remote Sensing Images

Affected by many factors, the objects in remote sensing images have certain defects in shape, size, occlusion, resolution, and pixels, which can affect the object detection results. Some of the major limitations are as follows:

(1) The resolution of the images in the dataset is different. The object frequently changes at different scales, which affects the object detection effect.

(2) In high-resolution images, the relative size of the detected object is small. After the sample has been reduced several times, it may lead to the absence of crucial information for small objects. This makes it impossible for the model to detect small objects.

(3) From Section 2.1, it can be seen that the dataset currently accessible contains a limited amount of labeled data. The available methods are not sufficient to demonstrate optimal object detection performance. To achieve optimal outcomes, gathering additional images and providing annotations necessitates a significant investment of time.

4.1.2. Image Enhancement Methods

Objects with irregular scales and irregular shapes reduce detector performance. Therefore, how to use geometric changes to process training images has become an urgent problem. To date, there have been many constructive schemes for image enhancement [61–64]. However, based on the specified requirements, they do not meet the necessary criteria. Presently, a novel mode of cognitive processing has emerged. The compression of training images can reduce storage space for high-resolution images while also decreasing the encoding time. The utilization of this technique enhances the performance of object detection.

The best image enhancement methods are image denoising [65], image filtering [66], edge sharpening [67], image rotation [68], image scaling [69], and image compression [70].

4.2. Irregular Object Detection in Remote Sensing Images

In remote sensing images, many objects are closely arranged and irregularly oriented, which affects the object detection performance to different degrees. This section discusses the solutions for the object irregularity problem in remote sensing images.

4.2.1. Directional Object Detection

Recent object detection methods mainly introduce directional regression tasks in classical object detectors. Among them, SCRDet [71], CADNet [72], DRN [73], R3Det [74], ReDet [75], and Oriented RCNN [76] improve the performance by predicting the rotation angle of the border. GlidingVertex [77] and RSDet [78] improved performance by returning to the quadrilateral.

For example, Yang et al. [79] used an angle classification task [80] to solve the problem of boundary discontinuity in angle-direction estimation. All of the above methods improve the estimation of orientation based on the rotation angle representation, thus improving the detection performance of rotating objects.

4.2.2. Non-Axial Feature Learning

Object detection methods such as YOLOv1 [20], Faster RCNN [17], FCOS [54], RepPoints [81], APD [82], FAN [83], CenterNet [53], etc. are oriented to upright or axially aligned objects. They are poor at detecting densely distributed objects and are not axially aligned.

To solve this problem, Han et al. [84] designed a feature alignment module to alleviate the misalignment between axially symmetric and non-axially symmetric objects.

Ding et al. [85] took a spatial transformation of the axially symmetric RoIs. It enables the model to learn non-axially symmetric representations with supervision under a rotating border. SCRDet++ [86] obtained higher object responses in the training network by enhancing non-axisymmetric features.

Guo et al. [87] introduced a convex hull representation to train the perception of the shape and distribution of irregular objects. Learnable feature adaptation is also used to avoid feature confounding.

DRN [73] uses a feature selection module to aggregate non-axisymmetric object information of different shapes, directions, and core sizes. It also uses a dynamic filter generator to regress this information. The above methods are aimed at improving the detection performance of non-axisymmetric features.

4.2.3. Sample Allocation for Object Detection

When setting the IoU threshold, most assays opt to establish a threshold that enables the selection of positive samples. The reliability of training samples is not guaranteed because of variables such as noise. The object directions in remote sensing images are diverse and densely dispersed. Therefore, the selection of high-quality samples is essential for training directed detectors. Recently, several illustrative assignment strategies have been proposed for this issue.

ATSS [46], FreeAnchor [88], PAA [89], OTA [90], and other methods use matching optimization strategies to select the best samples.

For instance, Ming et al. [91] proposed a matching measure method using matching sensitivity loss to evaluate spatial object alignment. This measurement method can effectively enhance the correlation between directional objecting and classification. All of the above techniques can optimize the sample allocation for object detection, which helps solve the problem of object irregularities.

4.3. Small Object Detection in Remote Sensing Images

With the continued development of deep learning technology, the evolution of object detection technology has accelerated. Research on the detection of medium and large objects has made significant advances. Among them, the most popular detectors have outstanding medium- and large-object detection performance.

The term “small objects” lacks a universally accepted definition; however, existing academic definitions can be categorized into two overarching classes. The first classification pertains to the comparative magnitude. The definition is established based on the relative proportion between the object and the image. In accordance with the definition provided by Chen et al. [92], small objects are characterized by a relative area that falls within the range of 0.08% to 0.58%. This relative area is determined by calculating the median of the ratio between the bounding box area and the image area for all object instances belonging to the same category. Additionally, there exist alternative definitions. For example, the width–height ratio of the object bounding box is less than 0.1, and the open square of the ratio of the object bounding box’s area to the image’s area is less than 0.03. The second category is defined by the absolute scale, which determines the size of small objects based on the absolute pixel size of the object. The general dataset MS COCO for object detection defines a small object as an object with a resolution of less than 32 pixels × 32 pixels. The experimental dataset DOTA in this paper defines objects with pixel values in the range of [10, 50] as small objects. However, due to the small number of pixels, small scale, and easy-to-occlude shadows, small object detection still faces significant challenges.

This is mainly reflected in the few available features, high requirements for positioning accuracy, a small number of small objects in popular datasets, unbalanced samples, and object aggregation, which are particularly serious in remote sensing images. In recent years, many excellent small object detection methods have been proposed. They have made significant progress in remote sensing image datasets. In this section, we discuss the excellent small object detection methods available.

4.3.1. Multi-Scale Prediction

Multi-scale prediction aims to predict the coordinates and classes of objects on feature maps at different scales. In the machine learning era, image pyramids are the representative method of constructing multi-scale features. The technique aims to scale the images to different resolutions and then extract the features separately. It uses a sliding window-based approach to detect objects, detecting small objects at the bottom of the pyramid.

For example, MTCNN [93] uses this idea and better recognizes small objects. However, its detection time is relatively long due to the need to extract features of multiple resolutions. With the development of deep learning techniques, CNN multi-scale feature extraction replaced image pyramids, and SSD [24] was proposed. However, it was found that SSD is not effective for small objects during detection.

Aiming at the problem of a single small object feature layer in SSD, DSSD [94] uses Resnet-101 as the backbone network for extracting features, which combines the semantic information of the higher-level features with the bottom-level information. This results in richer semantic features and better detection in the small object layer.

FPN is similar to the idea of DSSD. Its bottom-up and top-down branching fully integrates the high-level and bottom features, making each layer feature rich in semantic information, which is beneficial for small object detection. PANet [95] improves on the FPN by using fewer convolutional layers to build the path enhancement module, which can retain more information on the underlying layers. It adds an adaptive feature pooling module to make the region of interest contain multiple layers of features, further improving the performance of small object detection.

FPN introduces information from other layers, causing conflicts when detecting in a single layer. To address this problem, ASFF proposes an adaptive spatial feature fusion approach. It uses a learning weight approach to fuse the features of each layer for the final detection, which further improves the small object detection performance.

Therefore, the authors of AugFPN [96] argue that FPN does not take into account the semantic differences between features at different levels. This makes the top-down feature fusion process lose features at higher levels, resulting in regions of interest in each layer without feature information from other layers. To this end, the AugFPN proposer reduces semantic differences by adding the same supervision information to each layer before feature fusion.

A residual structure combines other layer features with the top-level features, which enhances contextual information. In addition, by fusing the elements of the candidate boxes pooled in different layers, it is ensured that the area of interest of each layer has the feature information of the other layers. Its small object detection performance is further improved. The current backbone networks used for feature extraction are trained on the ImageNet dataset, while the COCO dataset is used for testing.

The authors of SNIP [97] concluded that the difference between the two datasets affects the small object detection performance. During training, SNIP only calculates the gradients of regions of interest close to the object scale in the ImageNet dataset. In this way, the scale differences between different datasets are reduced.

For the problem of scale variation in object detection, the authors of TridentNet [49] found that the perceptual field is positively correlated with the object scale. The larger the perceptual field, the better the detection of large objects; the smaller the perceptual field, the better the detection of small objects. The algorithm controls the perceptual field by controlling the parameters of the null convolution. It generates three parallel convolutional layers to detect objects at different scales and improves the small object detection performance.

RTMDet [98] comprehensively improves the current single-stage object detector. It uses CSPDarkNet as a baseline and performs multi-scale feature fusion using CSPPAFPN. In terms of training strategy optimization, it uses a dynamic soft label assignment strategy to make the matching results of classification cost more stable and accurate. In the data enhancement stage, RTMDet introduces a caching mechanism, significantly improving operation efficiency.

Hang et al. [99] improved small object detection performance by modifying the first-level detector YOLOv5. They added new feature fusion layers and detector heads from shallow layers to maximize the retention of feature information. In addition, they replaced the original convolutional prediction heads with Swin transformer prediction heads SPHs to reduce the computational complexity. Finally, the normalization-based attention module HAM was integrated into YOLOv5 to improve attention performance in a normalized manner.

Guan et al. [100] proposed a deep neural network DNN based on high-quality object locations. Small object detection performance is improved by computing multiple layered segments with superpixels to derive gap-quality object locations and perform classification.

Fang et al. [101] proposed an improved method S2ANet-SR based on S2ANet. The model sends both the original image and the restored image to the detection network and then designs a super-resolution enhancement module for the restored image to enhance the feature extraction of small objects and proposes perceptual loss function and matching texture loss as supervision. The feature network design of part of the method is shown in Figure 7.

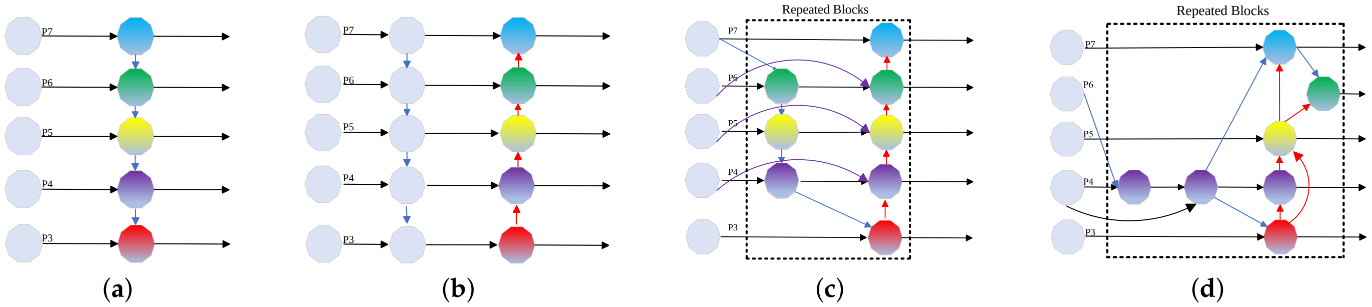


Figure 7. Feature network design. (P3–P7) Multi-scale features from level 3 to level 7. The different color dot represents different feature layers. (**a**) FPN [18] uses top-down paths to fuse multi-scale features. (**b**) PANet [95] adds additional bottom-up paths to the FPN. (**c**) NAS-FPN [102] uses neural architecture search to obtain irregular feature network topologies, then applies the same blocks repeatedly. (**d**) BiFPN [52] introduces a feature fusion mechanism with weights to extract features, then uses the same blocks repeatedly.

4.3.2. Enhanced Feature Resolution

The method can promote detection accuracy by increasing the accuracy of high-level feature maps or transforming the feature representation of the small object into a middle or big object representation approximately. STDN [103] applies this idea using a scale transfer module to increase resolution.

GAN-based PGAN [104] and SOD-MTGAN [105] inherit the generator and discriminator. Firstly, features containing enough small object information after the first convolution layer are fed to the generator and are then enhanced by adding residual representation. Secondly, the discriminative network has an adversarial branch and a perceptual branch. The network is trained with instances of large objects first. The generator and discriminator are trained in an iterative manner using a set of instances of both large and small objects, to enhance the detection accuracy of small objects.

The GAN adversarial network framework diagram is shown in Figure 8. ViTAE-B+RVSA_ORCN [106] uses the MAE [107] generative self-supervised pre-training method. It extracts the image features of non-masked regions and predicts the image contents of masked areas by an asymmetric network structure. The algorithm uses ViTAE as the backbone network and replaces the MHSA module in Plain ViT with RVSA to adapt MAE pre-training to remote sensing downstream tasks. Images generated by the Enhanced Super Resolution GAN (ESRGAN) model, which is based on the Generative Adversarial Network GAN, usually miss high-frequency edge information. This can seriously affect the detection of small objects in remote-sensing images. Inspired by this, the new edge-enhanced super-resolution adversarial network (EESRGAN) [108] uses different detector networks in an end-to-end manner to propagate detector loss directions into EESRGAN as a way to improve detection performance.

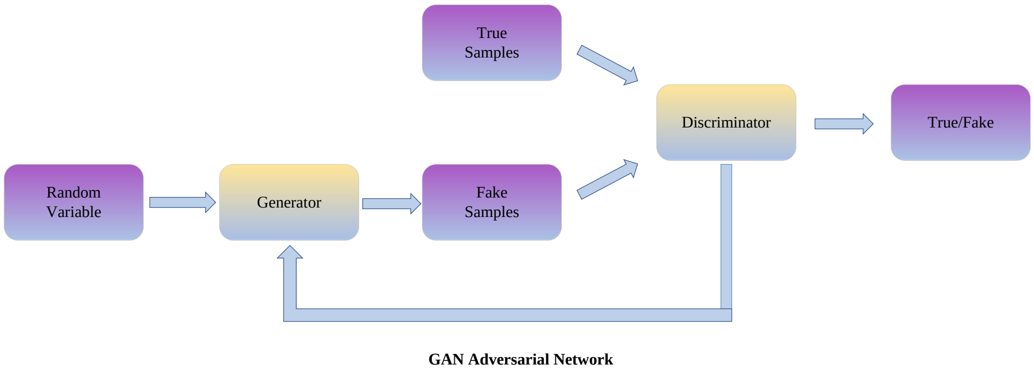


Figure 8. The GAN adversarial network framework.

4.3.3. Contextual Information

This method aims to exploit the relationship between objects in the image, i.e., contextual relationship, to improve the accuracy of small object detection.

Tang et al. [109] used a priori frame-based context-assisted methods to detect faces at different scales (mainly small-scale faces). During detection, many priori boxes related to faces will be helpful as auxiliary information to learn the supervision information of the context characteristics of small-scale and blocked faces.

Hu et al. [110] proposed a module for extracting association relationships between objects to leverage object correlations. The module partitions the characteristics of individual objects into two categories: shape and geometric features. It then combines the features of multiple objects to create detection features.

At the same time, Chen et al. [111] introduced image-level and object-level contextual information to describe the object-to-whole and object-to-object relationships, respectively. Different from the description method, CoupleNet [112] obtains contextual information related to the object by expanding the feature map of the region of interest to reduce the chance of false identification.

4.3.4. Data Enhancement

Data enhancement increases the training samples by panning, rotating, and resampling. Kisantal et al. [113] increased the training samples by resampling images containing small objects and copying and pasting small objects.

Zoph et al. [114] applied Neural Architecture Search (NAS) [115] to data augmentation to search for optimal strategies to improve RetinaNet performance.

4.3.5. Novel Backbone Network and Training Strategy

However, differences between classification and detection datasets can interfere with small object detection. Researchers have proposed novel backbone networks and training strategies for dealing with this problem. He et al. proposed a scratch-trained detection model for precise localization.

Guan et al. [116] proposed an efficient regionalized network. They treated object detection as a dual problem, divided into object proposal generation and object classification. One of the frameworks aims to generate high-quality proposals and then import the proposals and input images into the network to learn convolutional features.

Wang et al. [117] combined a pre-trained model with a training-from-scratch approach. The SSD network is used as the backbone network, and the LSN auxiliary network is used to compensate for the loss in feature extraction from the backbone network. This extracted the mid-bottom feature information more efficiently and helped detect small objects.

Li et al. proposed DetNet-59 [118], a backbone network dedicated to object detection. It designs the number of feature layers used for prediction according to the task characteristics.

Compared with ResNet-50 [119], its small object detection performance is better. Qiao et al. [120] proposed the DetectoRS algorithm, which feeds information from the FPN layer to the backbone network. The recursive structural feature reuses the information twice, greatly improving the small object detection performance.

DEA-net [121] proposed a dynamically improved anchor network to solve the issue of small object labels being easily lost or mislabeled. In order to provide qualifying samples, the network employs sample discriminators to carry out interactive sample screening between anchored and unanchored units.

GGHL [122] is suitable for object detection in arbitrary directions. It uses an adaptive label assignment strategy (OLA) for unanchored objects based on a two-dimensional oriented Gaussian heat map to define positive candidate objects. This enables the adaptive fitting of features of unused objects after feeding to the neural network CNN learning.

APE adaptive period embedding is a method for representing oriented objects in remote sensing images. The process is based on the angular periodicity of the oriented object. The angle is represented by two two-dimensional feature vectors with different periods. The vectors are continuous during the change of shape.

The CFC-Net [123] key feature capture network focuses on feature representation, pre-defined anchor points, and label assignment. The network constructs robust key features suitable for the respective tasks by polarizing the attention module. It also extracts discriminative regression features to refine the pre-defined anchor points and uses a dynamic anchor learning strategy to select high-quality anchor points adaptively.

Li et al. proposed a novel backbone network Large Selective Nuclear Network (LSKNet) [124]. It can dynamically adjust the spatial receptive field to better simulate the distance environment of various objects in the remote sensing scene.

Pang et al. [125] proposed a unified self-reinforcement network R2CNN. The network consists of a backbone Tiny-Net, an intermediate global attention module, and classifiers and detectors. As a lightweight residual structure, the Tiny-Net allows fast extraction of rich features from the input. The global attention module is used to suppress false positives. The classifier predicts the targets in each PATCH. If the object is available, the classifier tracks the detector to locate the object. The classifier and detector are trained end-to-end to speed up the detection process further and avoid false positives.

The TRD proposed by Li et al. [126] is a combination of CNN and a multilayer transformer with an encoder and decoder. To detect objects in remote sensing images, they designed an improved converter to aggregate multi-scale features and model the interaction between instances. Considering the difference between the remote sensing image dataset and the source dataset (ImageNet), they proposed the TRD with transmitted CNN (T-TRD) based on the attention mechanism due to the limited samples in the remotely sensed images and the large number of training samples required by the transformer. To avoid overfitting, data enhancement in the model is combined with the transformer to improve the detection performance.

4.3.6. Boundary Discontinuity Problem

The boundary discontinuity problem affects the object detection effect to some extent. PP-YOLOE-R [127] significantly improved the object rotation recognition. It introduces ProbloU loss to avoid the boundary discontinuity problem. PP-YOLOE-R also uses rotation task alignment learning for rotating object detection. It obtains more accurate predicted angles by angle prediction head and DFL loss.

For the boundary discontinuity problem, DCL is designed to replace the existing sparse-coded labels with densely coded labels (DCL). It achieves great improvement in training speed and detection accuracy. The angular distance and aspect ratio sensitive weighting method in DCL makes the detector more sensitive to these two aspects of the object. This improves the detection performance and makes the DCL particularly suitable for detecting square objects.

5. Comparison and Analysis of State-of-the-Art Models

5.1. *Experimentation and Analysis of Typical Algorithms*

The YOLO series is an example of a regression algorithm based on deep learning. The YOLO series have developed into big representative algorithms in object detection. On the other hand, the YOLO series are hardly ever employed in the research being done now to test remote sensing images. As a result, we began by selecting four YOLO series to be trained on the DOTA dataset so that we could evaluate how well they performed.

The results are presented in Figure 9. As a result of analyzing the available data, we have concluded that YOLOv3 has the highest mAP up to 0.495 when judged by the criterion of having 100 iterations of training. However, we did not anticipate this result at all. As a result, the traditional YOLO series performs poorly when applied to remote-sensing images.

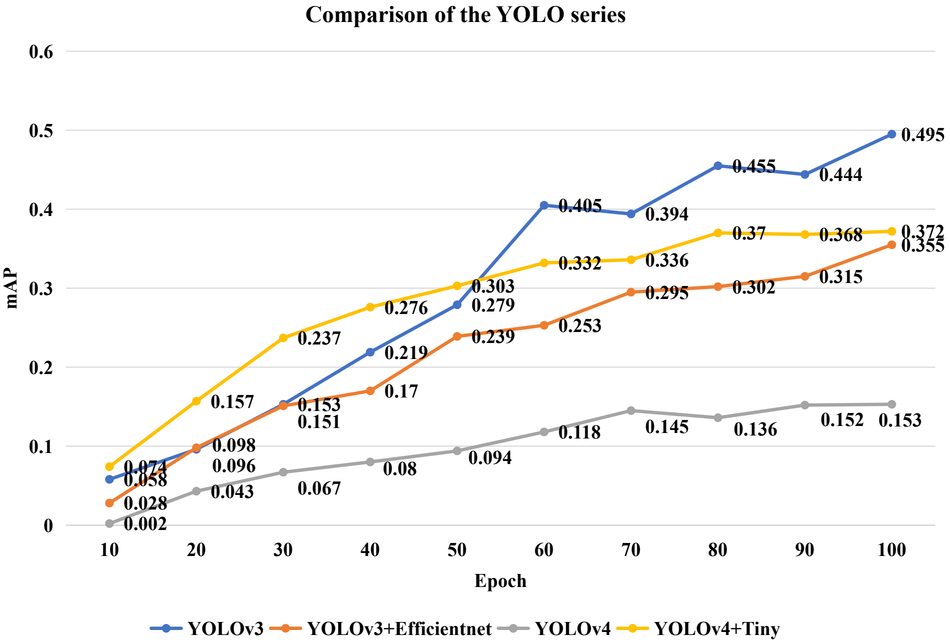


Figure 9. Comparison of the YOLO series.

5.2. *Comparison of Advanced Object Detection Methods in Remote Sensing Images*

In order to make the effect of the object detection algorithm on remote sensing images meet our expectations, we compared the more recent excellent algorithms on the DOTA dataset. The results are shown in Table 2.

In this paper, we further classified the DOTA dataset according to its object categories. Large objects include baseball diamonds, tennis courts, basketball courts, ground track fields, roundabouts, and soccer ball fields. Medium objects include planes, storage tanks, harbors, bridges, and swimming pools. Small objects include large vehicles, small vehicles, helicopters, and ships. The mAP of each method on the DOTA dataset for objects with different sizes are shown in Table 3.

As shown in Table 3, the mAP, AP_L, AP_M, and AP_S of AO2-DETR are the highest among similar models in the first-stage detectors. Among the two-stage detectors, LSKNet-S* has the highest AP_L and AP_S, and its detection performance is the best for large and small objects. The AP_M of CAD-Net is the highest, and its detection performance is the best for medium targets. Taken together, LSKNet-S* also has the highest mAP among the two-stage detectors, with up to 81.85%. For Anchor-free Methods, the Oriented RepPoints with Swin-T-FPN backbone network has the best detection performance for large and small objects, with high AP_L and AP_S of 77.90% and 75.43%. For medium-sized objects, Oriented RepPoints with R-101-FPN have the best performance with AP_M as high as 76.72%. Collectively, the Oriented RepPoints with the Swin-T-FPN backbone network has the highest mAP and the best results. Traversing the whole table and comparing the three types of methods together, LSKNet-S* has higher mAP, AP_L, AP_M, and AP_S than AO2-DETR and Oriented RepPoints using R-101-FPN. Therefore, LSKNet-S* has the best detection performance.

Table 2. Comparison of state-of-the-art methods on DOTA dataset.

Methods	Backbone	PL	BD	BR	GTF	SV	LV	SH	TC	BC	ST	SBF	RA	HA	SP	HC	mAP
One-stage Methods																	
R3Det-DCL [80]	R-152-FPN	89.78	83.95	52.63	69.70	76.84	81.26	87.30	90.81	84.67	85.27	63.50	64.16	68.96	68.79	65.45	75.54
R3Det [74]	R-152-FPN	89.49	81.17	50.53	66.10	70.92	78.66	78.21	90.81	85.26	84.23	61.81	63.77	68.16	69.83	67.17	73.74
S2A-Net [84]	R-50-FPN	89.11	82.84	48.37	71.11	78.11	78.39	87.25	90.83	84.90	85.64	60.36	62.60	65.20	69.31	57.94	74.12
RetinaNet-O [51]	R-50-FPN	88.67	77.62	41.81	58.71	74.58	71.64	79.11	90.29	82.18	74.32	54.75	60.60	62.57	69.67	60.64	68.43
RSDet [78]	R-152-FPN	90.10	82.00	53.80	68.50	70.20	78.70	73.60	91.20	87.10	84.70	64.30	68.20	66.10	69.30	63.70	74.10
DAL [91]	R-101-FPN	88.61	79.69	46.27	70.31	65.89	76.10	78.53	90.84	79.98	78.41	58.71	62.02	69.23	71.32	60.65	71.78
CFA [87]	R-152	89.08	83.20	54.37	66.87	81.23	80.96	87.17	90.21	84.32	86.09	52.34	69.94	75.52	80.76	67.96	76.67
DAFNet [128]	R-101	89.40	86.27	53.70	60.51	82.04	81.17	88.66	90.37	83.81	87.27	53.93	69.38	75.61	81.26	70.86	76.95
SASM [129]	RX-101	89.54	85.94	57.73	78.41	79.78	84.19	89.25	90.87	58.80	87.27	63.82	67.81	78.67	79.35	69.37	79.17
AO2-DETR [130]	R-50	89.95	84.52	56.90	74.83	80.86	83.47	88.47	90.87	86.12	88.55	63.24	65.09	79.09	82.88	73.46	79.22
Two-stage Methods																	
Oriented R-CNN [76]	R-101-FPN	88.86	83.48	55.27	76.92	74.27	82.10	87.52	90.90	85.56	85.33	65.51	66.82	74.36	70.15	57.28	76.28
ReDet [75]	ReR-50-ReFPN	88.79	82.64	53.97	74.00	78.10	84.06	88.04	90.89	87.78	85.75	61.76	60.39	75.96	68.07	63.59	76.25
CenterMap [131]	R-50-FPN	88.88	81.24	53.15	60.65	78.62	66.55	78.10	88.83	77.80	83.61	49.36	66.19	72.10	72.36	58.70	71.74
MaskOBB [132]	R-50-FPN	89.61	85.09	51.85	72.90	75.28	73.23	85.57	90.37	82.08	85.05	55.73	68.39	71.61	69.87	66.33	74.86
Gliding Vertex [77]	R-101-FPN	89.64	85.00	52.26	77.34	73.01	73.14	86.82	90.74	79.02	86.81	59.55	70.91	72.64	70.86	57.32	75.02
RoI-Trans [85]	R-101-FPN	88.65	82.60	52.53	70.87	77.93	76.67	86.87	90.71	83.83	82.51	53.95	67.61	74.67	68.75	61.03	74.61
FAOD [133]	R-101-FPN	90.21	79.58	45.49	76.41	73.18	68.27	79.56	90.83	83.40	84.68	53.40	65.42	74.17	69.69	64.86	73.28
SCRDet [71]	R-101-FPN	89.98	80.65	52.09	68.36	68.36	60.32	72.41	90.85	87.94	86.86	65.02	66.68	66.25	68.24	65.21	72.61
CAD-Net [72]	R-101-FPN	87.80	82.40	49.40	73.50	71.10	63.50	76.60	90.90	79.20	73.30	48.40	60.90	62.00	67.00	62.20	69.90
Faster RCNN-O [17]	R-50-FPN	88.44	73.06	44.86	59.09	73.25	71.49	77.11	90.84	78.94	83.90	48.59	62.95	62.18	64.91	56.18	69.50
CSL [79]	R-152	90.25	85.53	54.64	75.31	70.44	73.51	77.62	90.84	86.15	86.69	69.60	68.04	73.83	71.10	68.93	76.17
DODet [134]	R-50-FPN	89.96	85.52	58.01	81.22	78.71	85.46	88.59	90.89	87.12	87.80	70.50	71.54	82.06	77.43	74.47	80.62
AOPG [135]	R-50-FPN	89.88	85.57	60.90	81.51	78.70	85.29	88.85	90.89	87.60	87.65	71.66	68.69	82.31	77.32	73.10	80.66
LSKNet-S * [124]	LSKNet	89.69	85.70	61.47	83.23	81.37	86.05	88.64	90.88	88.49	87.40	71.67	71.35	79.19	81.77	80.86	81.85
LSKNet-S [124]	LSKNet	89.57	86.34	63.13	83.67	82.20	86.10	88.66	90.89	88.41	87.42	71.72	69.58	78.88	81.77	76.52	81.64
Anchor-free Methods																	
Oriented RepPoints [136]	R-50-FPN	87.02	83.17	54.13	71.16	80.81	78.40	87.28	90.90	85.97	86.25	59.90	70.49	73.53	72.27	58.97	75.97
Oriented RepPoints [136]	R-101-FPN	89.53	84.07	59.86	71.76	79.95	80.03	87.33	90.84	87.54	85.23	59.15	66.37	75.23	73.75	57.23	76.52
Oriented RepPoints [136]	Swin-T-FPN	89.11	82.32	56.71	74.95	80.70	83.73	87.67	90.81	87.11	85.85	63.60	68.60	75.95	73.54	63.76	77.63
DRN [73]	H-104	89.71	82.34	47.22	64.10	76.22	74.43	85.84	90.57	86.18	84.89	57.65	61.93	69.30	69.63	58.48	73.23
PIoU [137]	DLA-34	80.90	69.70	24.10	60.20	38.30	64.40	64.80	90.90	77.20	70.40	46.50	37.10	57.10	61.90	64.00	60.50
CenterNet-O [53]	DLA-34	81.00	64.00	22.60	56.60	38.60	64.00	64.90	90.80	78.00	72.50	44.00	41.10	55.50	55.00	57.40	59.10

* O indicates a detection effect with a oriented bounding box. The extreme values of AP and mAP for each type of object are marked in red.

Following are some of the conclusions that can be reached through comparison: (1) At the moment, Resnet-FPN serves as the backbone network for most of the object detection methods used on remote sensing images. By analyzing their mAP, it can be found that the performance of these methods is more stable, i.e., medium level. (2) The overall performance level of the first method is better than that of the second method when comparing the Anchor-based method with the Anchor-free method. (3) The newly proposed LSKNet backbone network shows significant advantages on the DOTA dataset, including various categories of accuracy (AP) and mAP.

By analyzing the above-related information, we can observe that, on the one hand, object detection methods on remotely sensing images are constantly evolving and improving in performance. On the other hand, the proposal of new backbone networks helps to improve object detection performance significantly. Therefore, the design of backbone networks can be a major focus of future research.

Table 3. Average precision for different types of objects on DOTA dataset.

Methods	Backbone	mAP	AP_L	AP_M	AP_S
One-stage Methods					
R3Det-DCL [80]	R-152-FPN	75.54	76.13	73.09	73.09
R3Det [74]	R-152-FPN	73.74	74.82	72.45	71.65
S2A-Net [84]	R-50-FPN	74.12	75.44	71.53	70.94
RetinaNet-O [51]	R-50-FPN	68.43	70.69	67.41	69.13
RSDet [78]	R-152-FPN	74.10	76.88	72.80	70.48
DAL [91]	R-101-FPN	71.78	73.59	70.77	68.49
CFA [87]	R-152	76.67	74.48	77.16	77.73
DAFNet [128]	R-101	76.95	74.05	77.45	78.83
SASM [129]	RX-101	79.17	74.28	78.51	78.17
AO2-DETR [130]	R-50	79.22	77.45	79.47	80.16
Two-stage Methods					
Oriented R-CNN [76]	R-101-FPN	76.28	78.20	74.79	70.95
ReDet [75]	ReR-50-ReFPN	76.25	76.24	74.51	73.46
CenterMap [131]	R-50-FPN	71.74	70.68	74.02	69.06
MaskOBB [132]	R-50-FPN	74.86	75.76	73.60	71.18
Gliding Vertex [77]	R-101-FPN	75.02	77.09	74.44	68.58
RoI-Trans [85]	R-101-FPN	74.61	74.93	73.42	74.10
FAOD [133]	R-101-FPN	73.28	74.84	72.85	69
SCRDet [71]	R-101-FPN	72.61	76.58	72.68	65.53
CAD-Net [72]	R-101-FPN	69.90	72.55	85.46	65.95
Faster RCNN-O [17]	R-50-FPN	69.50	68.91	68.86	66.46
CSL [79]	R-152	76.17	79.21	75.30	72.22
DODet [134]	R-50-FPN	80.62	81.13	79.05	79.02
AOPG [135]	R-50-FPN	80.66	80.99	79.61	78.60
LSKNet-S * [124]	LSKNet	81.85	81.89	79.9	82.50
LSKNet-S [124]	LSKNet	81.64	81.77	80.15	81.65
Anchor-free Methods					
Oriented RepPoints [136]	R-50-FPN	75.97	76.93	74.64	72.61
Oriented RepPoints [136]	R-101-FPN	76.52	76.62	76.72	72.74
Oriented RepPoints [136]	Swin-T-FPN	77.63	77.90	76.23	75.43
DRN [73]	H-104	73.23	73.80	72.15	69.69
PIoU [137]	DLA-34	60.50	63.60	58.88	57.15
CenterNet-O [53]	DLA-34	59.10	62.42	57.32	53.75

The mAP, AP_L, AP_M, AP_S extremes of the three types of methods are marked in blue. *: With EMA finetune.

5.3. Results and Discussion

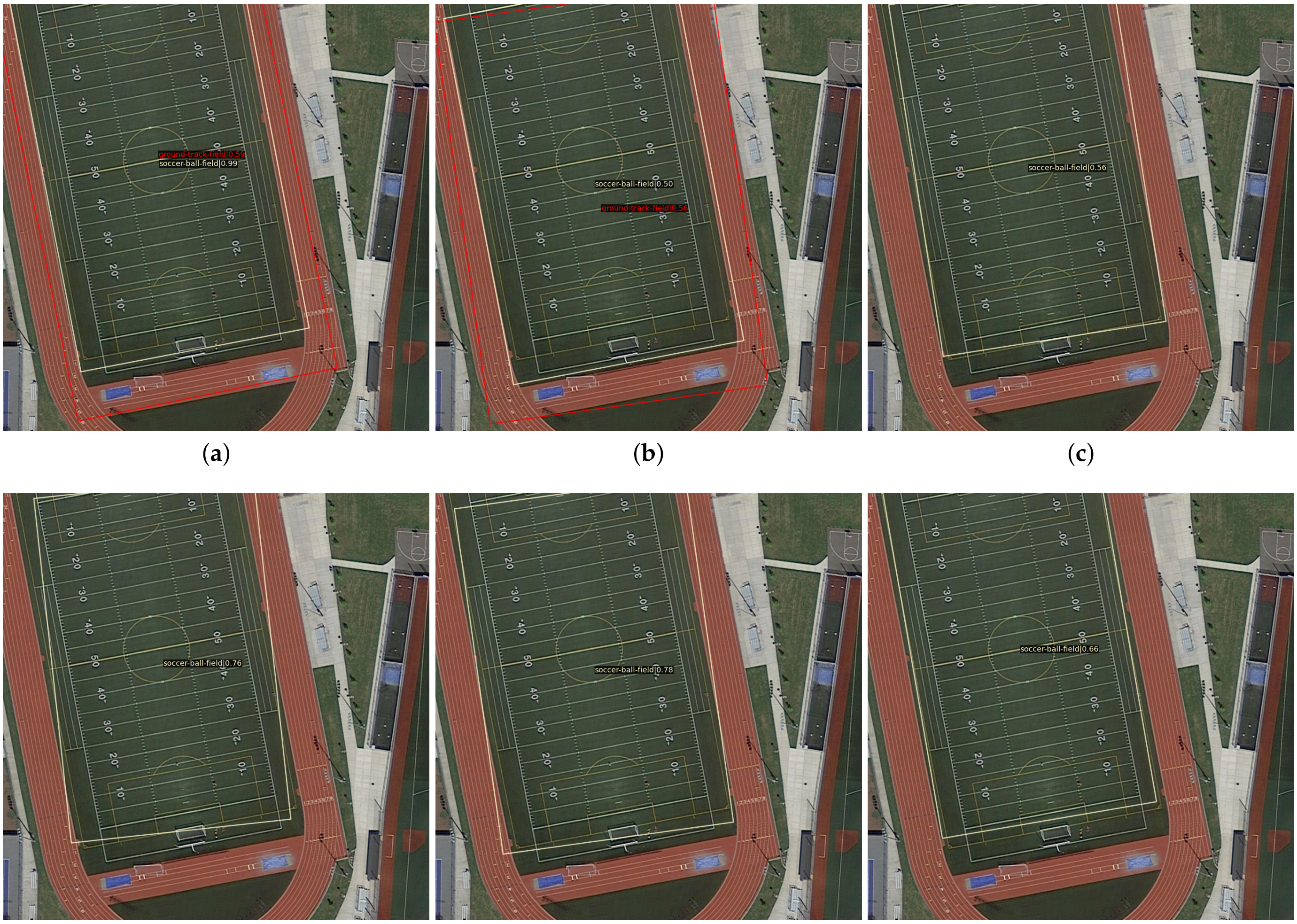
We selected six efficient object detection methods to visualize our experimental results on the remote sensing image datasets. We presented the visualization results to visualize and comprehensively represent their applications on remote sensing images.

For the dataset, we re-cropped part of the DOTA dataset and generated 217 images of 1204 × 1024, which were tested using a pre-trained model. The mAP, floating point operations (FLOPs), number of parameters (Params), and frames per second (FPS) of these methods were measured using a correlation evaluation method. This is used to evaluate the performance of the six methods. For the experimental environment, we chose Python 3.8, PyTorch 1.8.0, and CUDA 11.3. All models were tested with two Nvidia 3090 graphics cards with 24G of video memory each.

There are six object detection methods shown in Figure 10 for large object detection on remote sensing images. In facing the large object detection task, the difference between the six methods is not significant. However, the detection performance of LSKNet-S and Oriented-RepPoints is higher than the other four methods in facing the significant object inclusion problem.

There are six object detection methods shown in Figure 11 for large and small object detection on remote sensing images. The performance of Oriented-RepPoints is poor when facing the detection images with the coexistence of large and small objects. Among them, the larger object features mask the features of small objects, making detecting small objects less effective. In contrast, the detection results of the other five methods do not differ much and are better.

There are six object detection methods shown in Figure 12 for the small object detection of remote sensing images. In the face of dense small objects, the detection effect of the six methods is not much different. However, it is worth noting that the objects at the boundary of image segmentation cause missing detection due to incomplete objects. Thus, it can be seen that the object incompleteness problem caused by image segmentation of images can significantly affect object detection performance.



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

(**d**) (**e**) (**f**)

Figure 10. Visualization results of different object detection methods on the DOTA dataset: (**a**) LSKNet-S [124], (**b**) Oriented-RepPoints [136], (**c**) R3Det [74], (**d**) S2A-Net [84], (**e**) CSL [79], (**f**) CFA [87].

In order to compare the above six methods more comprehensively, we measured the FLOP, Params, FPS, and mAP evaluation metrics of the six methods on our own partial DOTA dataset. The specific data are shown in Table 4. In terms of the number of floating-point operations, LSKNet-S [124] is the smallest, followed by Oriented-RepPoints [136], CFA [87], S2A-Net [84], R3Det [74], and CSL [79], and in terms of the number of parameters, LSKNet-S [124] is the smallest, followed by Oriented-RepPoints [136], CFA [87], R3Det [74], CSL [79], and S2A-Net [84].

It can be seen from these two metrics that LSKNet-S [124] is the lightest method among the six methods with its smaller number of parameters and computations. In terms of the number of frames per second transmitted, Oriented-RepPoints [136] is the largest, followed by LSKNet-S [124], R3Det [74], CFA [87], CSL [79], and S2A-Net [84]. This metric shows that Oriented-RepPoints [136] has the fastest computation speed, followed by LSKNet-S [124]. In terms of average precision mean value, LSKNet-S [124] is the largest, followed by CFA [87], S2A-Net [84], Oriented-RepPoints [136], CSL [79], and R3Det [74], in that order. LSKNet-S [124] tops the list with a very high average precision value.

By comparison, we conclude that LSKNet-S [124] has the best performance, followed by Oriented-RepPoints [136], CFA [87], S2A-Net [84], R3Det [74], and CSL [79], in that order.

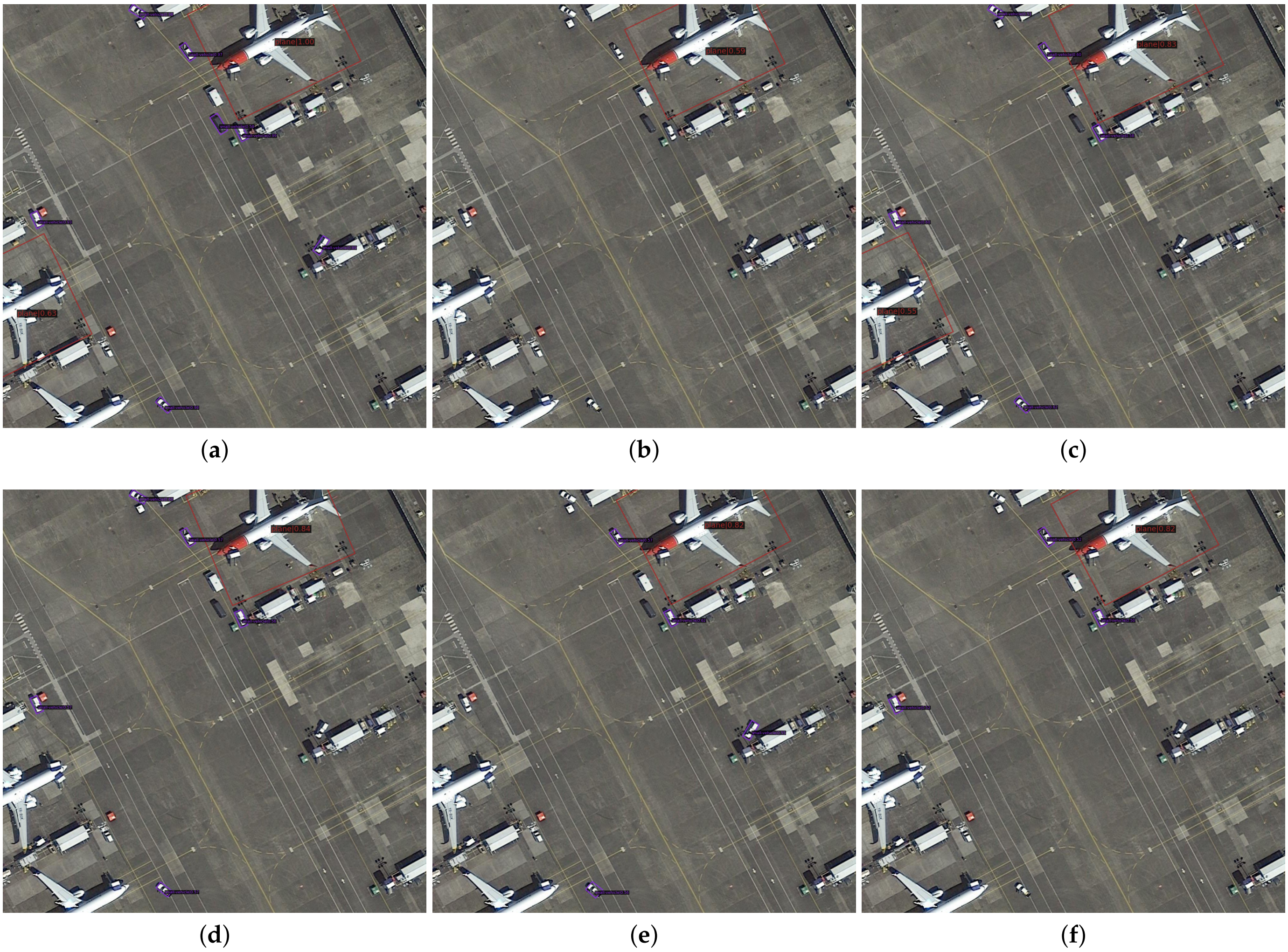
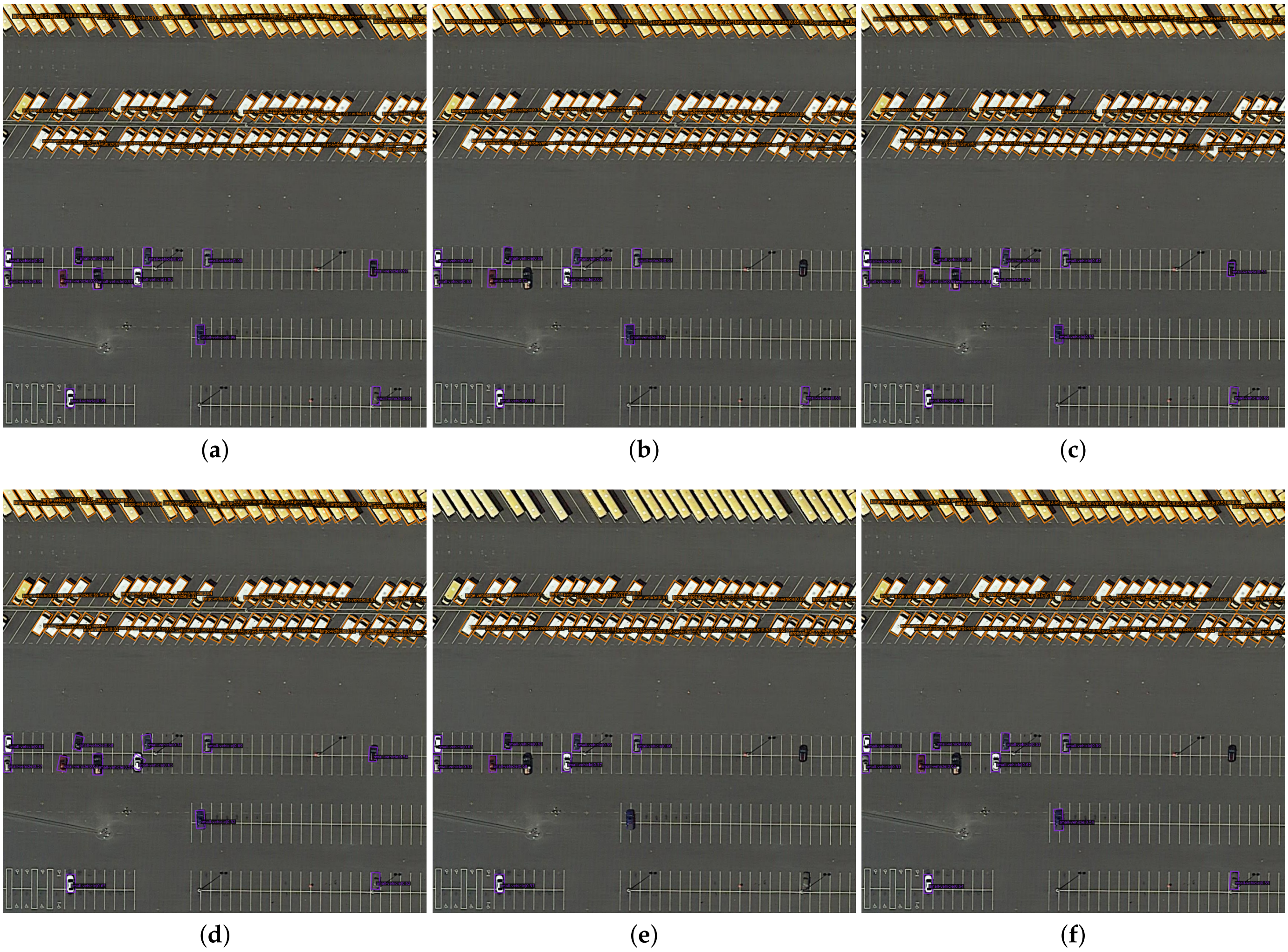


Figure 11. Visualization results of different object detection methods on the DOTA dataset: (**a**) LSKNet-S [124], (**b**) Oriented-RepPoints [136], (**c**) R3Det [74], (**d**) S2A-Net [84], (**e**) CSL [79], (**f**) CFA [87].

Table 4. Performance comparison of object detection methods.

Method	FLOPs	Params	FPS	mAP
LSKNet-S [124]	173.59 G	30.98 M	17.70	94.36
S2A-Net [84]	197.62 G	38.60 M	6.85	81.61
R3Det [74]	232.67 G	37.18 M	16.30	73.41
Oriented-RepPoints [136]	194.32 G	36.61 M	21.93	79.25
CSL [79]	236.29 G	37.35 M	8.45	74.77
CFA [87]	194.32 G	36.61 M	10.45	86.23


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

(**d**) (**e**) (**f**)

Figure 12. Visualization results of different object detection methods on the DOTA dataset: (**a**) LSKNet-S [124], (**b**) Oriented-RepPoints [136], (**c**) R3Det [74], (**d**) S2A-Net [84], (**e**) CSL [79], (**f**) CFA [87].

6. Current Challenges and Future Directions

This paper presents models and methods for achieving outstanding results in remote sensing image object detection. These models and methodologies have contributed to the development of object detection techniques for remote sensing images, as demonstrated by their experimental outcomes. Furthermore, the exposition of the latest object detection methodologies illustrates that this is a promising field of research.

Despite significant progress, the field of image object recognition continues to face numerous issues and obstacles. Remote sensing images pose unique challenges for applications, particularly in comparison to natural images. These challenges include diverse application scenarios, a large number of objects, and diverse directions that can be difficult to locate. Additionally, external factors such as weather and illumination can also significantly impact remote sensing images.

This section outlines some promising future approaches for advancing object identification applications, particularly small object identification remote sensing images. We have a strong conviction that these paths will entice additional outstanding academics to focus their attention on the study of remote sensing picture object detection and contribute their efforts.

6.1. Image Processing

As an important element affecting the quality of object recognition in remote sensing images, image quality faces certain challenges. Due to the large cost of large-area high-resolution remote sensing images and the high capital cost consumed in the practical application, new schemes are needed to solve the detection of small objects in low-resolution remote sensing images.

To address this problem, remote sensing image super-resolution technology has great research potential. As a classical computer vision task, image super-resolution techniques aim to reconstruct low-resolution images into high-resolution images. In this way, the influence of external factors, such as the environment, on remote sensing images is mitigated.

Compared with low-resolution images, high-resolution images are richer in object features, which will help improve the performance of object detection models.

6.2. Learning Strategy

A reasonable learning strategy can effectively improve the object detection performance of remote-sensing images. On the one hand, most current object detection models use IoU functions. Although many new loss functions, such as GiOU and DiOU, have been proposed, their applications in small object detection in remote sensing images have yet to produce satisfactory results.

The most suitable loss function for small object detection in remote sensing images still needs to be investigated. On the other hand, batch normalization is widely used in the field of object detection in order to accelerate the model training speed. The impact of the aforementioned technique on the detection of small objects in remote-sensing images needs to be improved. Further investigation is required to determine the appropriate normalization techniques for small object detection tasks in remote sensing images.

6.3. Network Design

Excellent network frameworks not only have higher evaluation metrics, but their efficient learning requires less runtime and computational resources. Therefore, more efficient and lightweight networks are still a significant research hotspot. However, many current deep network models have excellent performance and good results on benchmark datasets.

However, network frameworks for small object detection are still very scarce. Therefore, there is a need to develop more objected, lightweight, and efficient network architectures for small object detection to improve detection performance.

In the future, network design as a major research hotspot will attract a large number of researchers to continue to innovate on the basis of the old network. Meanwhile, its application to remote sensing images will also boost the development of small object detection applied to remote sensing images.

6.4. Dataset Construction

Remote sensing images are subject to inherent constraints and require significant time and resources to generate within the dataset. In comparison to natural images, the data volume of these images is limited. Remote sensing images of various regions and scenes exhibit noticeable differences owing to the impact of topography and vegetation. This leads to a limited representativeness and generalisation of the dataset. The generation of remote-sensing images remains a significant obstacle.

6.5. Multiple Data Fusion

In addition to using remote sensing images, other types of data sources can be combined with remote sensing images, such as LiDAR, GIS, etc. The rich sample information can boost the development of small object detection and provide more reliable support for practical applications.

7. Conclusions

This study presents a comprehensive review of object detection methods, particularly methods for detecting small objects. This article discusses the use of common datasets, evaluation methodologies, various classification criteria, the limitations of remote sensing images, and challenges related to detecting irregular objects. Furthermore, we discussed the diverse applications of object detection techniques in remote sensing imagery.

Finally, although the research on object detection methods in remote sensing images has made significant progress in recent years, there are still many problems, such as low model inference efficiency and unsatisfactory object detection results. Therefore, we propose promising research directions, such as better applications of image processing techniques, more efficient and lightweight backbone networks, and more reasonable learning strategies.

We hope the review in this paper can help researchers gain a deeper understanding of object detection methods, especially the application of small object detection methods in remote sensing images. It is expected to promote the development and progress of remote-sensing image technology.

Author Contributions: Conceptualization, X.W. and A.C.; software, A.W.; investigation, A.W. and J.Y.; formal analysis, X.W.; writing—original draft preparation, X.W., A.W. and J.Y.; writing—review and editing, A.C. and Y.S.; supervision, Y.S.; funding acquisition, X.W. and Y.S. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by the Natural Science Foundation of Shandong Province (ZR2020QF108, ZR2022QF037).

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The datasets are available on Github at https://captain-whu.github.io/DOTA/dataset.html, accessed on 10 April 2023.

Acknowledgments: We would like to thank the anonymous reviewers for their supportive comments to improve our manuscript.

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

References

1. Bai, L.; Li, Y.; Cen, M.; Hu, F. 3D Instance Segmentation and Object Detection Framework Based on the Fusion of Lidar Remote Sensing and Optical Image Sensing. *Remote Sens.* **2021**, *13*, 3288. [CrossRef]
2. Wei, Z.; Liu, Y. Deep Intelligent Neural Network for Medical Geographic Small-target Intelligent Satellite Image Super-resolution. *J. Imaging Sci. Technol.* **2021**, *65*, 030406-1–030406-10. [CrossRef]
3. Kussul, N.; Lavreniuk, M.; Skakun, S.; Shelestov, A. Deep learning classification of land cover and crop types using remote sensing data. *IEEE Geosci. Remote Sens. Lett.* **2017**, *14*, 778–782. [CrossRef]
4. Pi, Y.; Nath, N.D.; Behzadan, A.H. Convolutional neural networks for object detection in aerial imagery for disaster response and recovery. *Adv. Eng. Inform.* **2020**, *43*, 101009. [CrossRef]
5. Bashir, S.M.A.; Wang, Y. Deep learning for the assisted diagnosis of movement disorders, including isolated dystonia. *Front. Neurol.* **2021**, *12*, 638266. [CrossRef]
6. Bashir, S.M.A.; Wang, Y. Small object detection in remote sensing images with residual feature aggregation-based super-resolution and object detector network. *Remote Sens.* **2021**, *13*, 1854. [CrossRef]
7. DARAL, N. Histograms of oriented gradients for human detection. In Proceedings of the 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05), San Diego, CA, USA, 20–25 June 2005; pp. 886–893.
8. Wang, X.; Han, T.X.; Yan, S. An HOG-LBP human detector with partial occlusion handling. In Proceedings of the 2009 IEEE 12th International Conference on Computer Vision, Kyoto, Japan, 29 September–2 October 2009; pp. 32–39.
9. Lin, C. Fast Human Detection Using a Cascade of histograms of Oriented Gradients. In Proceedings of the 2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'06), New York, NY, USA, 17–22 June 2005; pp. 886–893.
10. Divvala, S.K.; Efros, A.A.; Hebert, M. How important are "deformable parts" in the deformable parts model? In Proceedings of the Computer Vision–ECCV 2012—Workshops and Demonstrations: Florence, Italy, 7–13 October 2012; Proceedings, Part III 12; Springer: Berlin/Heidelberg, Germany, 2012; pp. 31–40.

11. Felzenszwalb, P.F.; Girshick, R.B.; McAllester, D.; Ramanan, D. Object detection with discriminatively trained part-based models. *IEEE Trans. Pattern Anal. Mach. Intell.* **2009**, *32*, 1627–1645. [CrossRef] [PubMed]
12. Girshick, R.; Iandola, F.; Darrell, T.; Malik, J. Deformable part models are convolutional neural networks. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Boston, MA, USA, 7–12 June 2015; pp. 437–446.
13. Ouyang, W.; Wang, X. Joint deep learning for pedestrian detection. In Proceedings of the IEEE International Conference on Computer Vision, Sydney, NSW, Australia, 1–8 December 2013; pp. 2056–2063.
14. Girshick, R.; Donahue, J.; Darrell, T.; Malik, J. Rich feature hierarchies for accurate object detection and semantic segmentation. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Columbus, OH, USA, 23–28 June 2014; pp. 580–587.
15. He, K.; Zhang, X.; Ren, S.; Sun, J. Spatial pyramid pooling in deep convolutional networks for visual recognition. *IEEE Trans. Pattern Anal. Mach. Intell.* **2015**, *37*, 1904–1916. [CrossRef]
16. Girshick, R. Fast R-CNN. In Proceedings of the IEEE International Conference on Computer Vision, Santiago, Chile, 7–13 December 2015; pp. 1440–1448.
17. Ren, S.; He, K.; Girshick, R.; Sun, J. Faster R-CNN: Towards real-time object detection with region proposal networks. *Adv. Neural Inf. Process. Syst.* **2015**, *28*. [CrossRef]
18. 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.
19. He, K.; Gkioxari, G.; Dollár, P.; Girshick, R.B. Mask R-CNN. *IEEE Trans. Pattern Anal. Mach. Intell.* **2017**, *42*, 386–397. [CrossRef]
20. 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, 26 June–1 July 2016; pp. 779–788.
21. Redmon, J.; Farhadi, A. YOLO9000: Better, faster, stronger. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July 2017; pp. 7263–7271.
22. Redmon, J.; Farhadi, A. Yolov3: An incremental improvement. *arXiv* **2018**, arXiv:1804.02767.
23. Bochkovskiy, A.; Wang, C.Y.; Liao, H.Y.M. Yolov4: Optimal speed and accuracy of object detection. *arXiv* **2020**, arXiv:2004.10934.
24. 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, Proceedings, Part I 14; Springer: Berlin/Heidelberg, Germany, 2016; pp. 21–37.
25. Li, Z.; Wang, Y.; Zhang, N.; Zhang, Y.; Zhao, Z.; Xu, D.; Ben, G.; Gao, Y. Deep Learning-Based Object Detection Techniques for Remote Sensing Images: A Survey. *Remote Sens.* **2022**, *14*, 2385. [CrossRef]
26. Li, K.; Wan, G.; Cheng, G.; Meng, L.; Han, J. Object detection in optical remote sensing images: A survey and a new benchmark. *ISPRS J. Photogramm. Remote Sens.* **2020**, *159*, 296–307. [CrossRef]
27. Liu, Y.; Sun, P.; Wergeles, N.; Shang, Y. A survey and performance evaluation of deep learning methods for small object detection. *Expert Syst. Appl.* **2021**, *172*, 114602. [CrossRef]
28. Sun, X.; Wang, P.; Yan, Z.; Xu, F.; Wang, R.; Diao, W.; Chen, J.; Li, J.; Feng, Y.; Xu, T.; et al. FAIR1M: A benchmark dataset for fine-grained object recognition in high-resolution remote sensing imagery. *ISPRS J. Photogramm. Remote Sens.* **2022**, *184*, 116–130. [CrossRef]
29. Wang, Y.; Bashir, S.M.A.; Khan, M.; Ullah, Q.; Wang, R.; Song, Y.; Guo, Z.; Niu, Y. Remote sensing image super-resolution and object detection: Benchmark and state of the art. *Expert Syst. Appl.* **2022**, *197*, 116793. [CrossRef]
30. Long, Y.; Gong, Y.; Xiao, Z.; Liu, Q. Accurate object localization in remote sensing images based on convolutional neural networks. *IEEE Trans. Geosci. Remote Sens.* **2017**, *55*, 2486–2498. [CrossRef]
31. Xiao, Z.; Liu, Q.; Tang, G.; Zhai, X. Elliptic Fourier transformation-based histograms of oriented gradients for rotationally invariant object detection in remote-sensing images. *Int. J. Remote Sens.* **2015**, *36*, 618–644. [CrossRef]
32. Cheng, G.; Zhou, P.; Han, J. Learning rotation-invariant convolutional neural networks for object detection in VHR optical remote sensing images. *IEEE Trans. Geosci. Remote Sens.* **2016**, *54*, 7405–7415. [CrossRef]
33. Dong, R.; Xu, D.; Zhao, J.; Jiao, L.; An, J. Sig-NMS-based faster R-CNN combining transfer learning for small target detection in VHR optical remote sensing imagery. *IEEE Trans. Geosci. Remote Sens.* **2019**, *57*, 8534–8545. [CrossRef]
34. Rasche, C. Land use classification with engineered features. *IEEE Geosci. Remote Sens. Lett.* **2021**, *19*, 2500805. [CrossRef]
35. Xu, K.; Huang, H.; Li, Y.; Shi, G. Multilayer feature fusion network for scene classification in remote sensing. *IEEE Geosci. Remote Sens. Lett.* **2020**, *17*, 1894–1898. [CrossRef]
36. Xue, W.; Dai, X.; Liu, L. Remote sensing scene classification based on multi-structure deep features fusion. *IEEE Access* **2020**, *8*, 28746–28755. [CrossRef]
37. Xia, G.S.; Bai, X.; Ding, J.; Zhu, Z.; Belongie, S.; Luo, J.; Datcu, M.; Pelillo, M.; Zhang, L. DOTA: A large-scale dataset for object detection in aerial images. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–23 June 2018; pp. 3974–3983.
38. Razakarivony, S.; Jurie, F. Vehicle detection in aerial imagery: A small target detection benchmark. *J. Vis. Commun. Image Represent.* **2016**, *34*, 187–203. [CrossRef]
39. Yang, M.Y.; Liao, W.; Li, X.; Rosenhahn, B. Deep learning for vehicle detection in aerial images. In Proceedings of the 2018 25th IEEE International Conference on Image Processing (ICIP), Athens, Greece, 7–10 October 2018; pp. 3079–3083.

40. Lin, T.Y.; Maire, M.; Belongie, S.; Hays, J.; Perona, P.; Ramanan, D.; Dollár, P.; Zitnick, C.L. Microsoft coco: Common objects in context. In Proceedings of the Computer Vision—ECCV 2014: 13th European Conference, Zurich, Switzerland, 6–12 September 2014, Proceedings, Part V 13; Springer: Berlin/Heidelberg, Germany, 2014; pp. 740–755.
41. Zhu, H.; Chen, X.; Dai, W.; Fu, K.; Ye, Q.; Jiao, J. Orientation robust object detection in aerial images using deep convolutional neural network. In Proceedings of the 2015 IEEE International Conference on Image Processing (ICIP), Quebec City, QC, Canada, 27–30 September 2015; pp. 3735–3739.
42. Zhao, L.; Tang, P.; Huo, L. Feature significance-based multibag-of-visual-words model for remote sensing image scene classification. *J. Appl. Remote Sens.* **2016**, *10*, 035004. [CrossRef]
43. Dimitrov, D.; Knauer, C.; Kriegel, K.; Rote, G. Bounds on the quality of the PCA bounding boxes. *Comput. Geom.* **2009**, *42*, 772–789. [CrossRef]
44. Ming, Q.; Miao, L.; Zhou, Z.; Song, J.; Yang, X. Sparse Label Assignment for Oriented Object Detection in Aerial Images. *Remote Sens.* **2021**, *13*, 2664. [CrossRef]
45. Cheng, G.; Han, J. A survey on object detection in optical remote sensing images. *ISPRS J. Photogramm. Remote Sens.* **2016**, *117*, 11–28. [CrossRef]
46. Zhang, S.; Chi, C.; Yao, Y.; Lei, Z.; Li, S.Z. Bridging the gap between anchor-based and anchor-free detection via adaptive training sample selection. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Seattle, WA, USA, 13–19 June 2020; pp. 9759–9768.
47. 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.
48. Dai, J.; Li, Y.; He, K.; Sun, J. R-FCN: Object detection via region-based fully convolutional networks. *Adv. Neural Inf. Process. Syst.* **2016**, *29*.
49. Li, Y.; Chen, Y.; Wang, N.; Zhang, Z. Scale-aware trident networks for object detection. In Proceedings of the IEEE/CVF International Conference on Computer Vision, Seoul, Republic of Korea, 27 October–2 November 2019; pp. 6054–6063.
50. Li, C.; Li, L.; Geng, Y.; Jiang, H.; Cheng, M.; Zhang, B.; Ke, Z.; Xu, X.; Chu, X. YOLOv6 v3. 0: A Full-Scale Reloading. *arXiv* **2023**, arXiv:2301.05586.
51. Lin, T.Y.; Goyal, P.; Girshick, R.; He, K.; Dollár, P. Focal loss for dense object detection. In Proceedings of the IEEE International Conference on Computer Vision, Venice, Italy, 22–29 October 2017; pp. 2980–2988.
52. Tan, M.; Pang, R.; Le, Q.V. Efficientdet: Scalable and efficient object detection. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Seattle, WA, USA, 13–19 June 2020; pp. 10781–10790.
53. Zhou, X.; Wang, D.; Krähenbühl, P. Objects as points. *arXiv* **2019**, arXiv:1904.07850.
54. Tian, Z.; Shen, C.; Chen, H.; He, T. Fcos: Fully convolutional one-stage object detection. In Proceedings of the IEEE/CVF International Conference on Computer Vision, Seoul, Republic of Korea, 27 October–2 November 2019; pp. 9627–9636.
55. Liu, Z.; Zheng, T.; Xu, G.; Yang, Z.; Liu, H.; Cai, D. Training-Time-Friendly Network for Real-Time Object Detection. In Proceedings of the AAAI Conference on Artificial Intelligence, Honolulu, HI, USA, 27 January–1 February 2019.
56. 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; pp. 734–750.
57. Zhou, X.; Zhuo, J.; Krahenbuhl, P. Bottom-up object detection by grouping extreme and center points. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Long Beach, CA, USA, 15–20 June 2019; pp. 850–859.
58. Carion, N.; Massa, F.; Synnaeve, G.; Usunier, N.; Kirillov, A.; Zagoruyko, S. End-to-end object detection with transformers. In Proceedings of the Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, 23–28 August 2020, Proceedings, Part I 16; Springer: Berlin/Heidelberg, Germany, 2020; pp. 213–229.
59. 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. *Adv. Neural Inf. Process. Syst.* **2021**, *34*, 26183–26197.
60. Liu, Z.; Lin, Y.; Cao, Y.; Hu, H.; Wei, Y.; Zhang, Z.; Lin, S.; Guo, B. Swin transformer: Hierarchical vision transformer using shifted windows. In Proceedings of the IEEE/CVF International Conference on Computer Vision, Nashville, TN, USA, 11–27 October 2021; pp. 10012–10022.
61. Kuang, X.; Sui, X.; Liu, Y.; Chen, Q.; Gu, G. Single infrared image enhancement using a deep convolutional neural network. *Neurocomputing* **2019**, *332*, 119–128. [CrossRef]
62. Suzuki, K.; Horiba, I.; Sugie, N. Neural edge enhancer for supervised edge enhancement from noisy images. *IEEE Trans. Pattern Anal. Mach. Intell.* **2003**, *25*, 1582–1596. [CrossRef]
63. Sreedhar, K.; Panlal, B. Enhancement of images using morphological transformation. *arXiv* **2012**, arXiv:1203.2514.
64. Piao, Y.; Shin, I.; Park, H. Image resolution enhancement using inter-subband correlation in wavelet domain. In Proceedings of the 2007 IEEE International Conference on Image Processing, San Antonio, TX, USA, 16 September–19 October 2007; Volume 1, pp. 1–445.
65. Wu, X.; Liu, M.; Cao, Y.; Ren, D.; Zuo, W. Unpaired learning of deep image denoising. In Proceedings of the Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, 23–28 August 2020; Proceedings, Part IV; Springer: Berlin/Heidelberg, Germany, 2020; pp. 352–368.
66. He, K.; Sun, J.; Tang, X. Guided image filtering. *IEEE Trans. Pattern Anal. Mach. Intell.* **2012**, *35*, 1397–1409. [CrossRef]
67. Lev, B. Sharpening the intangibles edge. *Harv. Bus. Rev.* **2004**, *6*, 109–116.

68. Lin, C.Y.; Wu, M.; Bloom, J.A.; Cox, I.J.; Miller, M.L.; Lui, Y.M. Rotation, scale, and translation resilient watermarking for images. *IEEE Trans. Image Process.* **2001**, *10*, 767–782. [CrossRef]
69. Lin, X.; Ma, Y.l.; Ma, L.z.; Zhang, R.l. A survey for image resizing. *J. Zhejiang Univ. Sci. C* **2014**, *15*, 697–716. [CrossRef]
70. Dhawan, S. A review of image compression and comparison of its algorithms. *Int. J. Electron. Commun. Technol.* **2011**, *2*, 22–26.
71. Yang, X.; Yang, J.; Yan, J.; Zhang, Y.; Zhang, T.; Guo, Z.; Sun, X.; Fu, K. Scrdet: Towards more robust detection for small, cluttered and rotated objects. In Proceedings of the IEEE/CVF International Conference on Computer Vision, Seoul, Republic of Korea, 27 October–2 November 2019; pp. 8232–8241.
72. Zhang, G.; Lu, S.; Zhang, W. CAD-Net: A Context-Aware Detection Network for Objects in Remote Sensing Imagery. *IEEE Trans. Geosci. Remote Sens.* **2019**, *57*, 10015–10024. [CrossRef]
73. Pan, X.; Ren, Y.; Sheng, K.; Dong, W.; Yuan, H.; Guo, X.W.; Ma, C.; Xu, C. Dynamic Refinement Network for Oriented and Densely Packed Object Detection. In Proceedings of the 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Seattle, WA, USA, 13–19 June 2020; pp. 11204–11213.
74. Yang, X.; Liu, Q.; Yan, J.; Li, A. R3Det: Refined Single-Stage Detector with Feature Refinement for Rotating Object. In Proceedings of the AAAI Conference on Artificial Intelligence, Honolulu, HI, USA, 27 January–1 February 2019.
75. Han, J.; Ding, J.; Xue, N.; Xia, G. ReDet: A Rotation-equivariant Detector for Aerial Object Detection. In Proceedings of the 2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Nashville, TN, USA, 20–25 June 2021; pp. 2785–2794.
76. Xie, X.; Cheng, G.; Wang, J.; Yao, X.; Han, J. Oriented R-CNN for Object Detection. In Proceedings of the 2021 IEEE/CVF International Conference on Computer Vision (ICCV), Montreal, BC, Canada, 11–17 October 2021; pp. 3500–3509.
77. Xu, Y.; Fu, M.; Wang, Q.; Wang, Y.; Chen, K.; Xia, G.; Bai, X. Gliding Vertex on the Horizontal Bounding Box for Multi-Oriented Object Detection. *IEEE Trans. Pattern Anal. Mach. Intell.* **2019**, *43*, 1452–1459. [CrossRef]
78. Qian, W.; Yang, X.; Peng, S.; Yan, J.; Guo, Y. Learning modulated loss for rotated object detection. In Proceedings of the AAAI Conference on Artificial Intelligence, Virtually, 2–9 February 2021; Volume 35, pp. 2458–2466.
79. Yang, X.; Yan, J. Arbitrary-Oriented Object Detection with Circular Smooth Label. In Proceedings of the European Conference on Computer Vision, Glasgow, UK, 23–28 August 2020.
80. Yang, X.; Hou, L.; Zhou, Y.; Wang, W.; Yan, J. Dense Label Encoding for Boundary Discontinuity Free Rotation Detection. In Proceedings of the 2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Seattle, WA, USA, 13–19 June 2020; pp. 15814–15824.
81. Yang, Z.; Liu, S.; Hu, H.; Wang, L.; Lin, S. RepPoints: Point Set Representation for Object Detection. In Proceedings of the 2019 IEEE/CVF International Conference on Computer Vision (ICCV), Seoul, Republic of Korea, 27 October–2 November 2019; pp. 9656–9665.
82. Zhang, J.; Lin, L.; Li, Y.; chen Chen, Y.; Zhu, J.; Hu, Y.; Hoi, S.C.H. Attribute-Aware Pedestrian Detection in a Crowd. *IEEE Trans. Multimed.* **2019**, *23*, 3085–3097. [CrossRef]
83. Zhang, J.; Wu, X.; Zhu, J.; Hoi, S.C.H. Feature Agglomeration Networks for Single Stage Face Detection. *arXiv* **2017**, arXiv:1712.00721.
84. Han, J.; Ding, J.; Li, J.; Xia, G. Align Deep Features for Oriented Object Detection. *IEEE Trans. Geosci. Remote Sens.* **2020**, *60*, 5602511. [CrossRef]
85. Ding, J.; Xue, N.; Long, Y.; Xia, G.; Lu, Q. Learning RoI Transformer for Oriented Object Detection in Aerial Images. In Proceedings of the 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Long Beach, CA, USA, 16–17 June 2019; pp. 2844–2853.
86. Yang, X.; Yan, J.; Yang, X.; Tang, J.; Liao, W.; He, T. SCRDet++: Detecting Small, Cluttered and Rotated Objects via Instance-Level Feature Denoising and Rotation Loss Smoothing. *IEEE Trans. Pattern Anal. Mach. Intell.* **2020**, *45*, 2384–2399. [CrossRef]
87. Guo, Z.; Liu, C.; Zhang, X.; Jiao, J.; Ji, X.; Ye, Q. Beyond Bounding-Box: Convex-hull Feature Adaptation for Oriented and Densely Packed Object Detection. In Proceedings of the 2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Nashville, TN, USA, 20–25 June 2021; pp. 8788–8797.
88. Zhang, X.; Wan, F.; Liu, C.; Ji, X.; Ye, Q. Learning to Match Anchors for Visual Object Detection. *IEEE Trans. Pattern Anal. Mach. Intell.* **2019**, *44*, 3096–3109. [CrossRef]
89. Kim, K.; Lee, H.S. Probabilistic Anchor Assignment with IoU Prediction for Object Detection. In Proceedings of the European Conference on Computer Vision, Glasgow, UK, 23–28 August 2020.
90. Ge, Z.; Liu, S.; Li, Z.; Yoshie, O.; Sun, J. OTA: Optimal Transport Assignment for Object Detection. In Proceedings of the 2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Nashville, TN, USA, 20–25 June 2021; pp. 303–312.
91. Ming, Q.; Zhou, Z.; Miao, L.; Zhang, H.; Li, L. Dynamic Anchor Learning for Arbitrary-Oriented Object Detection. *arXiv* **2020**, arXiv:2012.04150.
92. Chen, C.; Liu, M.Y.; Tuzel, O.; Xiao, J. R-CNN for small object detection. In Proceedings of the Computer Vision–ACCV 2016: 13th Asian Conference on Computer Vision, Taipei, Taiwan, 20–24 November 2016; Revised Selected Papers, Part V 13; Springer: Berlin/Heidelberg, Germany, 2017; pp. 214–230.
93. Zhang, K.; Zhang, Z.; Li, Z.; Qiao, Y. Joint Face Detection and Alignment Using Multitask Cascaded Convolutional Networks. *IEEE Signal Process. Lett.* **2016**, *23*, 1499–1503. [CrossRef]
94. Fu, C.Y.; Liu, W.; Ranga, A.; Tyagi, A.; Berg, A.C. DSSD : Deconvolutional Single Shot Detector. *arXiv* **2017**, arXiv:1701.06659.

95. Liu, S.; Qi, L.; Qin, H.; Shi, J.; Jia, J. Path Aggregation Network for Instance Segmentation. In Proceedings of the 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–22 June 2018; pp. 8759–8768.
96. Guo, C.; Fan, B.; Zhang, Q.; Xiang, S.; Pan, C. AugFPN: Improving Multi-Scale Feature Learning for Object Detection. In Proceedings of the 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Seattle, WA, USA, 13–19 June 2020; pp. 12592–12601.
97. Singh, B.; Davis, L.S. An Analysis of Scale Invariance in Object Detection—SNIP. In Proceedings of the 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–23 June 2018; pp. 3578–3587.
98. Lyu, C.; Zhang, W.; Huang, H.; Zhou, Y.; Wang, Y.; Liu, Y.; Zhang, S.; Chen, K. RTMDet: An Empirical Study of Designing Real-Time Object Detectors. *arXiv* **2022**, arXiv:2212.07784.
99. Gong, H.; Mu, T.; Li, Q.; Dai, H.; Li, C.; He, Z.; Wang, W.; Han, F.; Tuniyazi, A.; Li, H.; et al. Swin-Transformer-Enabled YOLOv5 with Attention Mechanism for Small Object Detection on Satellite Images. *Remote Sens.* **2022**, *14*, 2861. [CrossRef]
100. Guan, Y.; Aamir, M.; Hu, Z.; Dayo, Z.A.; Rahman, Z.; Abro, W.A.; Soothar, P. An Object Detection Framework Based on Deep Features and High-Quality Object Locations. *Trait. Signal* **2021**, *38*, 719–730. [CrossRef]
101. Xiaolin, F.; Fan, H.; Ming, Y.; Tongxin, Z.; Ran, B.; Zenghui, Z.; Zhiyuan, G. Small object detection in remote sensing images based on super-resolution. *Pattern Recognit. Lett.* **2022**, *153*, 107–112. [CrossRef]
102. Ghiasi, G.; Lin, T.Y.; Le, Q.V. Nas-fpn: Learning scalable feature pyramid architecture for object detection. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Long Beach, CA, USA, 15–20 June 2019; pp. 7036–7045.
103. Zhou, P.; Ni, B.; Geng, C.; Hu, J.; Xu, Y. Scale-Transferrable Object Detection. In Proceedings of the 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–23 June 2018; pp. 528–537.
104. Li, J.; Liang, X.; Wei, Y.; Xu, T.; Feng, J.; Yan, S. Perceptual Generative Adversarial Networks for Small Object Detection. In Proceedings of the 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Honolulu, HI, USA, 21–26 July 2017; pp. 1951–1959.
105. Bai, Y.; Zhang, Y.; Ding, M.; Ghanem, B. SOD-MTGAN: Small Object Detection via Multi-Task Generative Adversarial Network. In Proceedings of the European Conference on Computer Vision, Munich, Germany, 8–14 September 2018.
106. Wang, D.; Zhang, Q.; Xu, Y.; Zhang, J.; Du, B.; Tao, D.; Zhang, L. Advancing Plain Vision Transformer Towards Remote Sensing Foundation Model. *arXiv* **2022**, arXiv:2208.03987.
107. He, K.; Chen, X.; Xie, S.; Li, Y.; Doll'ar, P.; Girshick, R.B. Masked Autoencoders Are Scalable Vision Learners. In Proceedings of the 2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), New Orleans, LA, USA, 18–24 June 2022; pp. 15979–15988.
108. Rabbi, J.; Ray, N.; Schubert, M.; Chowdhury, S.; Chao, D. Small-object detection in remote sensing images with end-to-end edge-enhanced GAN and object detector network. *Remote Sens.* **2020**, *12*, 1432. [CrossRef]
109. Tang, X.; Du, D.K.; He, Z.; Liu, J. PyramidBox: A Context-assisted Single Shot Face Detector. *arXiv* **2018**, arXiv:1803.07737.
110. Hu, H.; Gu, J.; Zhang, Z.; Dai, J.; Wei, Y. Relation Networks for Object Detection. In Proceedings of the 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–23 June 2018; pp. 3588–3597.
111. Chen, X.; Gupta, A.K. Spatial Memory for Context Reasoning in Object Detection. In Proceedings of the 2017 IEEE International Conference on Computer Vision (ICCV), Venice, Italy, 22–29 October 2017; pp. 4106–4116.
112. Zhu, Y.; Zhao, C.; Wang, J.; Zhao, X.; Wu, Y.; Lu, H. CoupleNet: Coupling Global Structure with Local Parts for Object Detection. In Proceedings of the 2017 IEEE International Conference on Computer Vision (ICCV), Venice, Italy, 22–29 October 2017; pp. 4146–4154.
113. Kisantal, M.; Wojna, Z.; Murawski, J.; Naruniec, J.; Cho, K. Augmentation for small object detection. *arXiv* **2019**, arXiv:1902.07296.
114. Zoph, B.; Cubuk, E.D.; Ghiasi, G.; Lin, T.Y.; Shlens, J.; Le, Q.V. Learning Data Augmentation Strategies for Object Detection. In Proceedings of the European Conference on Computer Vision, Thessaloniki, Greece, 23-25 September 2019.
115. Wang, N.; Gao, Y.; Chen, H.; Wang, P.; Tian, Z.; Shen, C. NAS-FCOS: Fast Neural Architecture Search for Object Detection. In Proceedings of the 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Seattle, WA, USA, 13–19 June 2020; pp. 11940–11948.
116. Guan, Y.; Aamir, M.; Hu, Z.; Abro, W.A.; Rahman, Z.; Dayo, Z.A.; Akram, S. A Region-Based Efficient Network for Accurate Object Detection. *Trait. Signal* **2021**, *38*, 481–494. [CrossRef]
117. Wang, T.; Anwer, R.M.; Cholakkal, H.; Khan, F.S.; Pang, Y.; Shao, L. Learning Rich Features at High-Speed for Single-Shot Object Detection. In Proceedings of the 2019 IEEE/CVF International Conference on Computer Vision (ICCV), Seoul, Republic of Korea, 27 October–2 November 2019; pp. 1971–1980.
118. Li, Z.; Peng, C.; Yu, G.; Zhang, X.; Deng, Y.; Sun, J. DetNet: A Backbone network for Object Detection. *arXiv* **2018**, arXiv:1804.06215.
119. Li, H.; Wu, X. Infrared and Visible Image Fusion with ResNet and zero-phase component analysis. *arXiv* **2018**, arXiv:1806.07119.
120. Qiao, S.; Chen, L.C.; Yuille, A.L. DetectoRS: Detecting Objects with Recursive Feature Pyramid and Switchable Atrous Convolution. In Proceedings of the 2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Nashville, TN, USA, 20–25 June 2021; pp. 10208–10219.
121. Liang, D.; Geng, Q.; Wei, Z.; Vorontsov, D.A.; Kim, E.L.; Wei, M.; Zhou, H. Anchor Retouching via Model Interaction for Robust Object Detection in Aerial Images. *IEEE Trans. Geosci. Remote Sens.* **2021**, *PP*, 5619213. [CrossRef]
122. Huang, Z.; Li, W.; Xia, X.G.; Tao, R. A General Gaussian Heatmap Label Assignment for Arbitrary-Oriented Object Detection. *IEEE Trans. Image Process.* **2021**, *31*, 1895–1910. [CrossRef]

123. Ming, Q.; Miao, L.; Zhou, Z.; Dong, Y. CFC-Net: A Critical Feature Capturing Network for Arbitrary-Oriented Object Detection in Remote-Sensing Images. *IEEE Trans. Geosci. Remote Sens.* **2021**, *60*, 5605814. [CrossRef]
124. Li, Y.; Hou, Q.; Zheng, Z.; Cheng, M.M.; Yang, J.; Li, X. Large Selective Kernel Network for Remote Sensing Object Detection. *arXiv* **2023**, arXiv:2303.09030.
125. Pang, J.; Li, C.; Shi, J.; Xu, Z.; Feng, H. $\mathcal{R}^2$-CNN: fast Tiny object detection in large-scale remote sensing images. *IEEE Trans. Geosci. Remote Sens.* **2019**, *57*, 5512–5524. [CrossRef]
126. Li, Q.; Chen, Y.; Zeng, Y. Transformer with transfer CNN for remote-sensing-image object detection. *Remote Sens.* **2022**, *14*, 984. [CrossRef]
127. Wang, X.; Wang, G.; Dang, Q.; Liu, Y.; Hu, X.; Yu, D. PP-YOLOE-R: An Efficient Anchor-Free Rotated Object Detector. *arXiv* **2022**, arXiv:2211.02386.
128. Lang, S.; Ventola, F.; Kersting, K. Dafne: A one-stage anchor-free deep model for oriented object detection. *arXiv* **2021**, arXiv:2109.06148.
129. Hou, L.; Lu, K.; Xue, J.; Li, Y. Shape-adaptive selection and measurement for oriented object detection. In Proceedings of the AAAI Conference on Artificial Intelligence, Virtually, 22 February–1 March 2022; Volume 36, pp. 923–932.
130. Dai, L.; Liu, H.; Tang, H.; Wu, Z.; Song, P. Ao2-detr: Arbitrary-oriented object detection transformer. *IEEE Trans. Circuits Syst. Video Technol.* **2022**, *33*, 2342–2356. [CrossRef]
131. Wang, J.; Yang, W.; Li, H.C.; Zhang, H.; Xia, G.S. Learning center probability map for detecting objects in aerial images. *IEEE Trans. Geosci. Remote Sens.* **2020**, *59*, 4307–4323. [CrossRef]
132. Wang, J.; Ding, J.; Guo, H.; Cheng, W.; Pan, T.; Yang, W. Mask OBB: A semantic attention-based mask oriented bounding box representation for multi-category object detection in aerial images. *Remote Sens.* **2019**, *11*, 2930. [CrossRef]
133. Li, C.; Xu, C.; Cui, Z.; Wang, D.; Zhang, T.; Yang, J. Feature-attentioned object detection in remote sensing imagery. In Proceedings of the 2019 IEEE International Conference on Image Processing (ICIP), Taipei, Taiwan, 22–25 September 2019; pp. 3886–3890.
134. Cheng, G.; Yao, Y.; Li, S.; Li, K.; Xie, X.; Wang, J.; Yao, X.; Han, J. Dual-aligned oriented detector. *IEEE Trans. Geosci. Remote Sens.* **2022**, *60*, 1–11. [CrossRef]
135. Cheng, G.; Wang, J.; Li, K.; Xie, X.; Lang, C.; Yao, Y.; Han, J. Anchor-free oriented proposal generator for object detection. *IEEE Trans. Geosci. Remote Sens.* **2022**, *60*, 5618111. [CrossRef]
136. Li, W.; Chen, Y.; Hu, K.; Zhu, J. Oriented reppoints for aerial object detection. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, New Orleans, LA, USA, 18–24 June 2022; pp. 1829–1838.
137. Chen, Z.; Chen, K.; Lin, W.; See, J.; Yu, H.; Ke, Y.; Yang, C. Piou loss: Towards accurate oriented object detection in complex environments. In Proceedings of the Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, 23–28 August 2020; Proceedings, Part V 16; Springer: Berlin/Heidelberg, Germany, 2020; pp. 195–211.

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.