applied sciences

MDPI

Article

SP-YOLOv8s: An Improved YOLOv8s Model for Remote Sensing Image Tiny Object Detection

Mingyang Ma and Huanli Pang *

School of Computer Science and Engineering, Changchun University of Technology, Changchun 130012, China; 2202103032@stu.ccut.edu.cn

* Correspondence: panghuanli@ccut.edu.cn; Tel.: +86-130-8913-4093

Abstract: An improved YOLOv8s-based method is proposed to address the challenge of accurately recognizing tiny objects in remote sensing images during practical human-computer interaction. In detecting tiny targets, the accuracy of YOLOv8s is low because the downsampling module of the original YOLOv8s algorithm causes the network to lose fine-grained feature information, and the neck network feature information needs to be sufficiently fused. In this method, the strided convolution module in YOLOv8s is replaced with the SPD-Conv module. By doing so, the feature map undergoes downsampling while preserving fine-grained feature information, thereby improving the learning and expressive capabilities of the network and enhancing recognition accuracy. Meanwhile, the path aggregation network is substituted with the SPANet structure, which facilitates the acquisition of more prosperous gradient paths. This substitution enhances the fusion of feature maps at various scales, reduces model parameters, and further improves detection accuracy. Additionally, it enhances the network's robustness to complex backgrounds. Experimental verification is conducted on the following two intricate datasets containing tiny objects: AI-TOD and TinyPerson. A comparative analysis with the original YOLOv8s algorithm reveals notable enhancements in recognition accuracy. Specifically, under real-time performance constraints, the proposed method yields a 4.9% and 9.1% improvement in mAP0.5 recognition accuracy for AI-TOD and TinyPerson datasets, respectively. Moreover, the recognition accuracy for mAP0.5:0.95 is enhanced by 3.4% and 3.2% for the same datasets, respectively. The results indicate that the proposed method enables rapid and accurate recognition of tiny objects in complex backgrounds. Furthermore, it demonstrates better recognition precision and stability than other algorithms, such as YOLOv5s and YOLOv8s.

Keywords: tiny object detection; remote sensing image; SPD-Conv; SPANet; YOLOv8s

check for updates

Citation: Ma, M.; Pang, H. SP-YOLOv8s: An Improved YOLOv8s Model for Remote Sensing Image Tiny Object Detection. *Appl. Sci.* **2023**, *13*, 8161. https://doi.org/10.3390/app13148161

Academic Editor: Andrea Prati

Received: 3 June 2023
Revised: 12 July 2023
Accepted: 12 July 2023
Published: 13 July 2023

Copyright: © 2023 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (https://creativecommons.org/licenses/by/4.0/).

1. Introduction

Remote sensing images play an essential role in various fields, such as real-time aviation and navigation monitoring, ecological resource and environmental monitoring, military object detection, and geological disaster detection [1,2]. However, due to limitations in resolution and the distance between the remote sensing equipment and the objects being observed, the captured images often depict the target objects in a tiny form. These tiny objects, characterized by their small scale and weak features, present significant challenges for accurate object detection in remote sensing images.

Consequently, the detection of tiny objects in remote-sensing images has emerged as a major research problem in the field of computer vision. Specifically, this problem revolves around detecting and recognizing small-sized objects in images or videos. For instance, in the commonly used COCO dataset [3], objects with resolutions smaller than 32×32 pixels are classified as tiny objects.

In early object detection research, most methods are based on traditional feature extraction methods (e.g., shape, color, texture, etc.), determining object candidate regions through sliding windows of different scales and inputting object features such as Haar features [4],

HOG features [5], and SIFT [6] features into the classifier to classify the candidate objects within the regions. Nevertheless, conventional object detection algorithms have various limitations, such as inefficient candidate region selection and window redundancy. These drawbacks hinder the detection of tiny targets in remote sensing images [7,8], and the traditional object detection algorithms are only applicable to images with simple backgrounds and significant features, which have great limitations in practical applications.

With the rapid development of computers and deep learning [9,10], Many researchers have employed sophisticated deep learning models to implement object detection in remotely sensed images. Compared with earlier object detection methods, object detection using deep learning methods has features such as feature self-extraction, high adaptability and robustness, high accuracy, and higher efficiency in processing large-scale data, such as R-CNN [11], Fast R-CNN [12], Faster R-CNN [13], SSD [14], YOLO [15], RetinaNet [16], CenterNet [17], etc. These algorithms usually use convolutional neural networks (CNNs) to extract image features and perform detection operations on the feature maps. These methodologies can be categorized as two-stage approaches, relying on the extraction of regions as their foundation. (e.g., R-CNN, Fast R-CNN, Faster R-CNN, etc.) and one-stage methods based on regression (e.g., YOLO, SSD, RetinaNet, etc.), as well as keypoint-based methods (e.g., CenterNet, etc.). One-stage detection algorithms straightforwardly predict the object's location and class from the input image, eliminating the need for an explicit candidate region extraction phase. While computationally smaller and faster, they tend to be less accurate than two-stage detection algorithms. In two-stage algorithms, the candidate region extraction phase involves techniques such as Selective Search [18] and RPN [19] to generate regions potentially containing objects. Although more computationally demanding and slower due to explicit region extraction, two-stage detection algorithms improve object recognition accuracy, especially for objects with complex backgrounds. Unlike the previous two algorithms, the keypoint-based approach uses the idea of anchor-free, eliminating the need for anchors when training the model, thus simplifying the training process, but still does not provide a remarkable trade-off between precision and speed.

At present, there remains significant room for improvement in the accuracy of detecting tiny objects in remote sensing images. The challenges of detecting small objects arise from their low resolution, which often confuses the surrounding background. Moreover, downsampling during the image processing may cause a loss of object information, resulting in a sparse representation of object features within the high-level feature map. Consequently, these features are prone to be disregarded or misinterpreted as background, leading to false and missed detection issues. In addition, complex environmental sets introduce further complexities, including object occlusion, small object aggregation, varying light intensities, noise, and diverse object poses, all impacting the accuracy of tiny object detection. To address these challenges, this paper proposes an improved YOLOv8s-based algorithm for tiny object recognition, using YOLOv8s as a basic network. The improved path aggregation network SPANet is also used to reinforce the fusion of feature maps at various scales and reduce the model parameters. In addition, we incorporate the SPD-Conv module to reinforce the network's capability to extract features from tiny objects. The main contributions of this paper are the following:

A tiny object detection algorithm based on improved YOLOv8s is proposed for complex backgrounds.

- Using the SPD-Conv module, the benchmark network YOLOv8s enhances the complex background tiny object feature extraction capability. It can also effectively retain fine-grained feature information and improve network recognition accuracy;
- The SPANet path aggregation network is used to enhance the fusion effect of different scale feature maps, reduce the model parameters, fully fuse the contextual information, and reinforce the stability of the network to complex backgrounds.

Overall, this proposed method addresses the challenges associated with detecting tiny objects in remote sensing images, offering improved recognition accuracy and computational efficiency in complex backgrounds.

2. Related Work

Enhancing the recognition accuracy of tiny objects is a significant research area within object detection. Numerous researchers have devoted themselves to developing tiny object detection systems.

Wang et al. [20] established a dataset for tiny-object detection in aerial images, namely, AI-TOD; in addition, a multi-center point-based learning network (M-CenterNet) was proposed to enhance the performance of tiny-object detection. Based on this, Wang et al. [21] further proposed a new NWD-based tiny-object detector, which can significantly improve the tiny-object detection performance and reach the state-of-the-art on the AI-TOD dataset. Sunkara et al. constructed a new CNN building block called SPD-Conv, representing a general and unified approach. Lin et al. [22] proposed the feature pyramid network (FPN) algorithm, incorporating a bottom-up and top-down structure that effectively addressed small object detection by independently predicting feature layers after fusing high semantic features from deep feature maps with high-resolution information from shallow layers. Liu et al. [23] proposed the path aggregation network (PAN), which pooled features from all feature layers and reduced the gap between lower and uppermost feature layers. Shuai et al. [24] utilized the scale-invariant feature transform (SIFT) as features for ship candidate regions and performed classification recognition based on the extracted features. Cheng et al. [25] extracted A multiscale histogram of oriented gradients (HOG) features for candidate regions, fused the multiscale features, and performed classification recognition. Yang et al. [26] improved feature fusion based on RetinaNet, incorporating spatial and channel attention and utilizing rotating anchor frames for loss calculation, resulting in enhanced detection accuracy for remote sensing small objects. Building upon this work, the literature [27] introduced an instance-level denoising module to achieve better feature extraction and improve detection accuracy. Ding et al. [28] addressed rotation detection by converting horizontal regions of interest into Rotation Region-of-Interest (RRoI) and using RRoI pooling for further correction. Guan et al. [29] proposed a capsule feature pyramid network named RoadCapsFPN, which extracted and integrated multi-scale capsule features to enhance resolution and contextual semantic information in remotely sensed road images. Zhang et al. [30] introduced the CoF-Net. This coarse-to-fine remote sensing image detection method progressively enhanced feature representation and training sample selection through feature adaptation and sample assignment, respectively. Based on YOLOv5s, Deng, et al. [31] proposed a lightweight aerial image object detection algorithm (LAI-YOLOv5s) with relatively less computation and parameters and relatively faster inference. Wang et al. [32] proposed a lightweight YOLO-ACG detection algorithm, which balances accuracy and speed and improves the classification errors and missed detection problems in existing steel plate defect detection algorithms. Anitha et al. [33] proposed a method that helps in producing vibrant and realistic colors by hybridizing a convolution neural network with an auto-encoder. Chen et al. [34] proposed a two-stage lightweight detection framework with extremely low computation complexity, which enables high-resolution feature maps for dense anchoring to better cover small objects, proposes a sparsely-connected convolution for computation reduction, enhances the early stage features in the backbone, and addresses the feature misalignment problem for accurate small object detection. Yang et al. [35] increased the number and size of shallow feature pyramids to enhance the detection accuracy of small objects, utilizing a densely connected structure to improve feature representation. YOLT [36] incorporated increased upsampling and connected intermediate shallow features to output features through constant mapping. Chen et al. [37] combined semantic information from the shallowest features and fused them with deeper features to enhance the detection rate of small objects. Wang et al. [38] combined shallow information and simultaneously improved the loss function to increase the training weights of small targets. Li et al. [39] used deconvolution layers to fuse shallow and deep features, further enhancing the detection of small objects. However, introducing shallow features often introduces significant background noise for small objects. Fu et al. [40] addressed this issue by introducing a balancing factor to weigh the

fusion of shallow and deep features. Still, this approach relies on the detection task's a priori knowledge, limiting its robustness across different tasks. Zhang et al. [41] utilized a two-stage Faster R-CNN approach, employing deconvolution to upsample each candidate region from the previous stage, enlarging the feature map size and improving small object detection. Schilling et al. [42] employed deconvolution layers to scale up in-depth features and fuse them with shallow features, completing the detection process jointly. Nevertheless, deconvolution-based operations introduce additional parameters. To mitigate this, Liu et al. [43] employed expanded convolution operations as a parameter-efficient alternative, reducing parameters while maintaining the same perceptual field. However, expanded convolution may lead to loss of local information. To address this limitation, Ying et al. [44] introduced a pixel attention mechanism for local information fusion, compensating for the drawbacks of expanded convolution and improving small object detection. Nevertheless, upsampling operations remain meaningful only if small objects are still distinguishable in the deep features. Up-sampling does not recover the lost feature information if small objects are "lost" due to downsampling in deep features. If small objects are "lost" due to downsampling in deep features, upsampling does not recover the lost feature information. Therefore, some works [42] combine the introduction of shallow features with a deep feature upsampling process to leverage complementary advantages. However, as the feature map scale increases, computational complexity also rises, leading to increased time consumption during the detection process, despite the enhanced small object detection capability. To mitigate this issue, this research paper introduces an end-to-end algorithm designed for detecting tiny objects within remote sensing images. The proposed approach leverages a YOLOv8s architecture as its foundation. The algorithm incorporates the efficient path aggregation network SPANet and SPD-Conv modules, enabling stable, quick, and accurate detection of tiny objects in images with complex backgrounds.

3. Methodology

YOLOv8, a one-stage object recognition algorithm, effectively converts the detection task into a regression problem. Compared to alternative algorithms, YOLOv8 exhibits swifter detection speed and improved accuracy, fulfilling the requirement for real-time detection and recognition of tiny objects. To cater to diverse application demands, the YOLOv8 network can be scaled to generate five distinct network models of varying sizes.

In this paper, we balance the speed and precision of tiny object detection and choose the YOLOv8s network as the basic network model.

3.1. YOLOv8s

As an extensively employed network in object detection, YOLOv8 surpasses numerous preceding models with its faster detection speed and enhanced detection accuracy. The YOLOv8 network has been scaled according to different usage requirements to acquire some network models of various sizes. In this study, the YOLOv8s network is selected as the foundational network model to strike a balance between the speed and accuracy of recognizing tiny objects amidst complex backgrounds.

The fundamental components of YOLOv8s encompass the CBS (Convolution, Batch Normalization, SiLU activation) module, SPPF (Spatial Pyramid Pooling Fusion) module, and C2F (C3-inspired lightweight module with ideas from ELAN) module. The CBS module comprises a 3×3 convolutional layer, a BN (Batch Normalization) [45] layer, and a SiLU (Sigmoid-weighted Linear Unit) [46] activation function. This arrangement enables the selection of models characterized by high efficiency and accuracy. By means of feature reuse, the module mitigates the risk of gradient dispersion while preserving a significant portion of the original information. The SPPF module, inspired by the SPP structure from SPPNet [47], improves classification accuracy by extracting and fusing high-level features. Multiple maximum pooling operations are employed during the fusion process to extract a broad range of high-level semantic features. The C2F module, inspired by the C3 module and ideas from ELAN [48], facilitates lightweight implementation in YOLOv8 while achieving

optimal performance. Notably, YOLOv8 diverges from previous YOLO architectures through its detection head, which adopts the favored decoupling head approach [49]. yolov8 uses BCE Loss for classification loss and CIOU Loss + DFL for regression loss, and VFL proposes an asymmetric weighting operation [50]. DFL (Distribution-based Localization): The spatial coordinates of the bounding box are represented as a generalized distribution. Let the network quickly focus on the location distribution near the object location and make the probability density near that location as prominent as possible, as shown in Equation (1). s_i is the sigmoid output of the network, y_i and y_{i+1} are the interval orders, and y is the label. Compared with the previous YOLO algorithm, YOLOv8 is very scalable.

$$DFL_{(s_i,s_{i+1})} = -((y_{i+1} - y)\log(s_i) + (y - y_i)\log(s_{i+1})) \quad (1)$$

YOLOv8 distinguishes itself from its predecessors by employing an Anchor-Free approach instead of the traditional Anchor-Based method. The utilization of a dynamic TaskAlignedAssigner for matching policies is another notable enhancement in YOLOv8. To compute the Anchor-level alignment for each instance, Equation (2) is employed, wherein the classification score is denoted by 's', the IOU value by 'u', and the weight hyperparameters by 'α' and 'β.' Positive samples are selected by choosing the top 'm' anchors with the highest value (t) for each instance, while the remaining anchors serve as negative samples. Subsequently, the model is trained using an appropriate loss function. These improvements have resulted in a 1% increase in accuracy compared to YOLOv5, establishing YOLOv8 as the most accurate detector thus far.

$$t = s^{\alpha} \times u^{\beta} \quad (2)$$

The object detection process employing the YOLOv8s algorithm closely aligns with other network models within the YOLO series. Initially, the input image is resized to ensure a consistent dimension of 640 × 640 for all inputs. Subsequently, the network utilizes the Backbone and Head modules to extract features from the input image, yielding a feature map with dimensions B × B × C. In this context, the first two dimensions (B × B) denote the dimensions of the extracted feature map, while the final dimension, C = n × (5 + N), represents the number of bounding boxes predicted per grid as per the YOLO series algorithm. In YOLOv8s, 'n' denotes the number of bounding boxes, which is set to 3, and 'N' indicates the number of categories to be detected. The value '5' represents four location coordinate information and one confidence information within each predicted bounding box.

3.2. SPD-Conv Module

The accuracy of object detection models is generally high when recognizing medium and large objects. However, the model's accuracy diminishes rapidly when it comes to identifying tiny objects in remote-sensing images. This decline can be attributed to the strided convolution module in the YOLOv8s backbone network. Although this module enlarges the receptive field and reduces parameter computation through downsampling, it inadvertently results in the loss of fine-grained information and less efficient feature representations. In contrast, SPD-Conv [51], instead of employing stride convolution layers and pooling layers, uses a space-to-depth (SPD) layer and a non-stride convolution layer. The SPD layer downsamples the feature map while preserving all information in the channel dimension, effectively eliminating information loss. This design strategy enhances the model's training performance by improving the network's learning capability.

Processing of feature maps by SPD-Conv module when scale = 2, as depicted in Figure 1, the unprocessed feature map is shown in Figure 1a, Figure 1b indicates that the feature map is divided equally into four categories in the dimension of space. One color indicates one category, the vectors of the same color are concatenated together in the dimension of the space; that is, the feature map of arbitrary size S× S × C_1 is downsampled to generate four feature maps of size S/2 × S/2 × C_1, as illustrated in Figure 1c. Subsequently,

these four feature maps are concatenated along the C_1 dimension, resulting in a feature map of size $S/2 \times S/2 \times 4C_1$, as illustrated in Figure 1d.

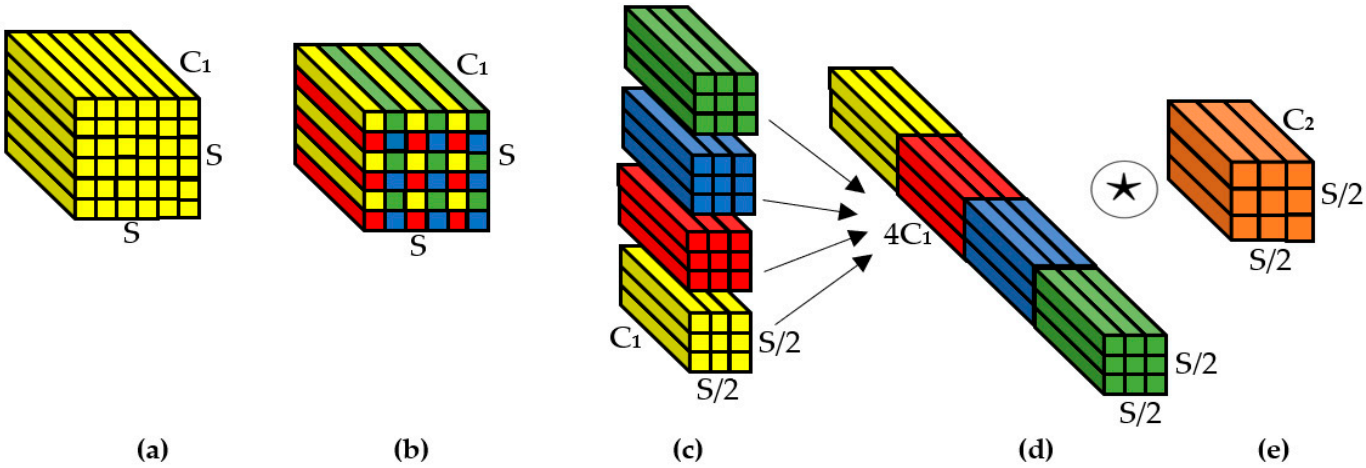

Figure 1. Processing of feature maps by SPD-Conv module when scale = 2; (**a**) unprocessed feature maps; (**b**–**d**) space-to-depth on the feature map; (**e**) the output feature map.

Following the SPD feature transformation layer, a feature map of size $S/2 \times S/2 \times C_2$, possessing the necessary number of channels C_2 for the subsequent module, is obtained using nonstrided convolution, as illustrated in Figure 1e. This process effectively preserves all discriminative feature information.

3.3. Path Aggregation Network SPANet

In deep learning, the low feature layer has a smaller receptive field than the high feature layer, and the backbone network obtains more low feature information, which helps to improve the accuracy of tiny object detection. However, the original YOLOv8s neck network feature information is not sufficiently fused, making the accuracy rate low. Therefore, building upon the original PANet architecture, this study incorporates the shallow feature layer P2, which is outputted from the backbone network, as an additional input to PANet. Introducing the detailed information from lower layers into PANet facilitates the fusion of deep semantic and shallow detail information. Like feature layers P3 and P4, P2 is concatenated with the up-sampled deep feature layer, enabling the fusion process to integrate deep semantic and fine-grained details information. The resulting feature map is then downsampled to match the size of P3′, followed by feature concatenation and convolution operations. Moreover, the feature maps concatenated with P3 and P4 include not only the down-sampled feature maps from the preceding layer but also the feature maps (P3′ and P4′) obtained through previous up-sampling, concatenation, and convolution operations. This enrichment of gradient information renders the network more robust and effective.

Compared with the original PANet, the improved model further shortens the distance between the top and lower layers and enhances the model's ability to predict tiny objects. To avoid adding P2 into the path aggregation network to make the model parameters too much, P4″ downsampling and feature concatenating and convolution operations are removed from the original PANet to achieve the purpose of reducing parameters and balancing the training speed. Finally, the output feature layers P2′, P3″ and P4″ are taken as the input into yolo head to obtain the prediction box, class, and other information. The architecture of the SPANet (Small Path Aggregation Network) is shown in Figure 2.

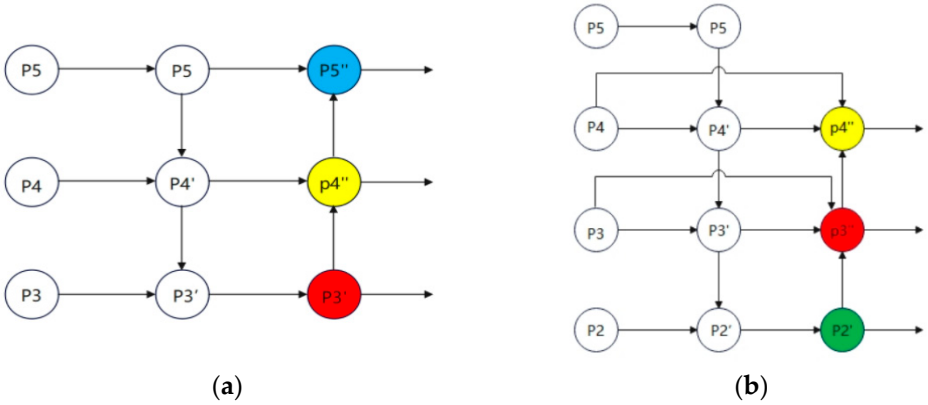

Figure 2. Path aggregation network architecture; (**a**) PANet; (**b**) SPANet.

3.4. SP-YOLOv8s

For the recognition task of tiny objects, the original YOLOv8s algorithm has an insufficient ability to extract features of tiny objects because tiny objects occupy very tiny pixels of the image. This leads to a low accuracy rate when detecting tiny objects and is, therefore, not suitable for practical applications. To solve these problems, this paper proposes to use an efficient path aggregation network, SPANet, to replace the path aggregation network, PANet, in the YOLOv8s network. This aims to improve the network's ability to fuse feature maps at different scales while reducing the network parameters. In addition, except for the first downsampling, which retains the original CBS, all other downsampled CBS modules are replaced with SPD-Conv modules to make the network focus on the tiny object features and reduce the complex background interference. Specifically, the improvement lies in replacing the original downsampling modules in layers 3, 6, 9, and 12 of the YOLOv8s backbone network and layers 25 and 29 of the neck network with SPD-Conv modules, and replacing the original neck network with SPANet. It is true that the SP-YOLOv8s backbone network consists of layers 1–14, the neck network consists of layers 15–31, and the last layer is the decoupling detection head, and the above three parts constitute the complete SP-YOLOv8s. In short, this paper proposes the SP-YOLOv8s algorithm (YOLOv8s incorporating SPD-Conv and SPANet). The network structure of the proposed algorithm is shown in Figure 3. The SP-YOLOv8s algorithm is executed as follows:

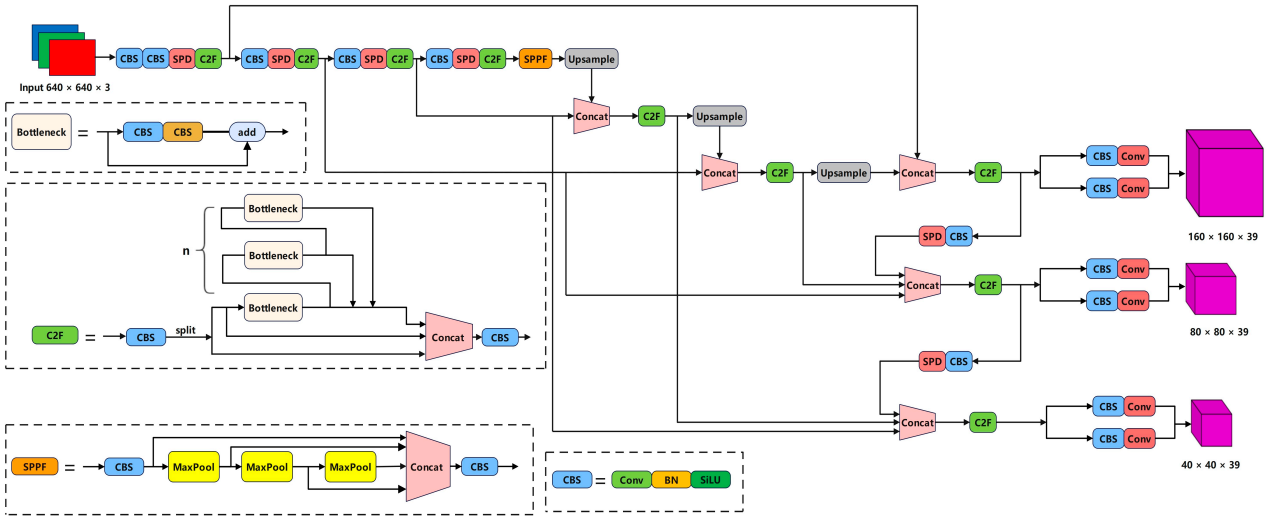

Figure 3. Network structure of the improved YOLOv8s: SP-YOLOv8s.

Input: images of size 640 × 640 × 3.

1. Image feature extraction using backbone network;
2. Feature fusion using neck network;
3. The feature maps of layers 23, 27, and 31 are input to the decoupling detection head.

Output: feature maps of size 160 × 160 × 39, 80 × 80 × 39, 40 × 40 × 39.

4. Experimental Evaluation

This section begins by introducing the dataset employed in the experiments and the evaluation metrics adopted to assess the performance. Subsequently, a series of comparative experiments are conducted between the proposed method and other object detection algorithms using the AI-TOD aerial image dataset, specially focusing on the recognition tiny objects. These experiments aim to showcase the effectiveness and superiority of the proposed approach over existing methods. Following this, ablation experiments are carried out to evaluate the impact and efficacy of the proposed improvements. Additionally, experiments are conducted on the Tinyperson dataset [52] to verify the generalization capability of the proposed method and affirm its robustness across different datasets and scenarios.

4.1. Experimental Environment and Datasets

All experimental evaluations presented in this research paper were performed within a consistent experimental environment, employing Ubuntu 20.04 as the operating system, an Intel(R) Xeon(R) Silver 4214R CPU operating at 2.40 GHz, an NVIDIA Tesla T4 GPU with 16 GB of memory. The stochastic gradient descent algorithm was utilized to optimize the loss function. During training, an initial learning rate of 0.01 was employed, and the learning rate adjustment strategy followed a cosine annealing algorithm.

The proposed approach was evaluated using the recently introduced AI-TOD aerial image micro-object detection dataset. The images in this dataset possess dimensions of 800 × 800 pixels, while the dimensions of the network's input image were modified to a resolution of 640 × 640 pixels. The dataset encompasses eight distinct object categories, namely, aircraft, bridge, storage tank, ship, swimming pool, car, pedestrian, and windmill, totaling 700,621 object instances. To ensure comprehensive evaluation, the dataset was partitioned into training and test sets, following an 8:2 ratio. Consequently, the training set consists of 11,214 samples, while the test set contains 2804 samples. It should be noted that the training and test data are strictly independent, ensuring unbiased evaluation. Notably, the AI-TOD dataset differs from other benchmark datasets, as its largest object measures less than 64 pixels. Furthermore, approximately 86% of the objects in the dataset are smaller than 16 pixels, with an average object size of approximately 12.8 pixels. It is worth mentioning that certain object classes, such as swimming pools and windmills, exhibit significantly fewer instances compared to more prevalent classes like vehicles and boats. This class imbalance is commonly observed in aerial image datasets, such as DOTA [53] and DIOR [54].

Figure 4 shows some images of the AI-TOD dataset (Figure 4a) and some images of the TinyPerson dataset (Figure 4b).

4.2. Evaluation Indicators

The primary aim of this research paper is to present an effective algorithm for detecting and recognizing tiny objects in complex backgrounds, effectively balancing the trade-off between model accuracy and speed. Accordingly, three metrics, namely, accuracy, recall, and mean average precision (mAP), are employed to assess the model's accuracy. Additionally, frames per second (FPS), model size, and billion floating point operations (GFLOPs) are utilized as performance measures to evaluate the model's speed.

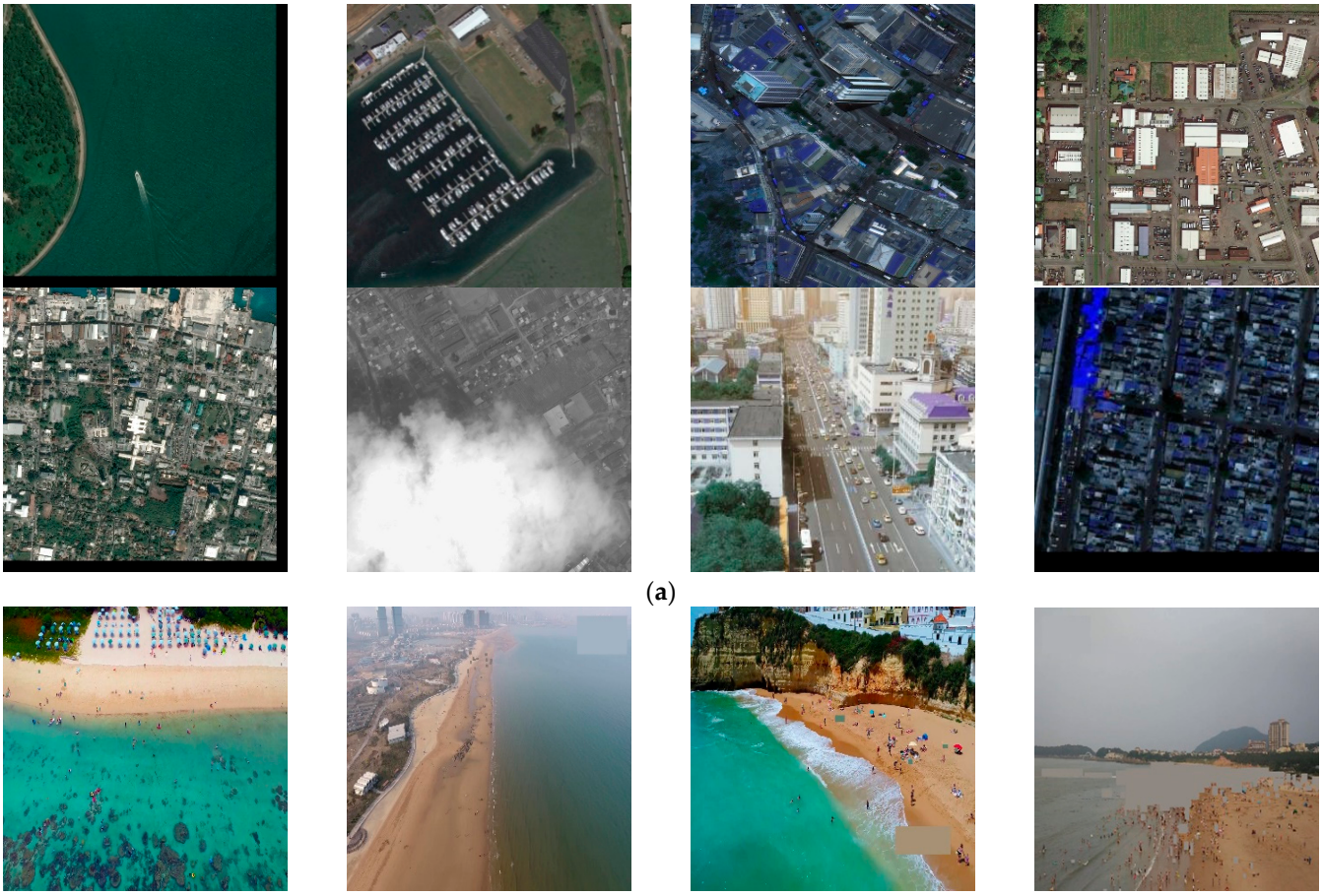

Figure 4. Datasets; (**a**) AI-TOD dataset; (**b**) TinyPerson dataset.

The formulas for precision (P) and recall (R) are the following:

$$\mathrm{P} = \frac{TP}{(TP + FP)} \tag{3}$$

$$\mathrm{R} = \frac{TP}{(TP + FN)} \tag{4}$$

In object detection evaluation, *TP* represents the count of accurately predicted bounding boxes, *FP* represents the count of incorrectly identified positive samples, and *FN* denotes the count of hidden objects.

The average precision (mAP) metric is a comprehensive measure of the model's performance by considering both precision and recall. It is computed by determining the average precision (AP) for each object category at a fixed intersection over the union (IOU) threshold and subsequently averaging these AP values across all categories. The mAP values are obtained by calculating the area under the precision-recall (P-R) curve for each category, where precision is plotted on the X-axis and recall is plotted on the Y-axis. The following formulas are utilized for the calculation of AP and mAP:

$$\mathrm{AP} = \int_0^1 P(r)dr \tag{5}$$

$$\mathrm{mAP} = \frac{\sum_{i=1}^{C} AP_i}{C} \tag{6}$$

where $p(r)$ denotes the precision-recall curve, and C represents the number of detection categories.

To ensure a comprehensive and rigorous evaluation of the proposed algorithm, the evaluation metric mAP@0.5:0.95 was employed. This metric computes the average mean average precision (mAP) at various intersections over union (IOU) thresholds, ranging from 0.5 to 0.95 in increments of 0.05 (specifically, at IOU thresholds of 0.5, 0.55, 0.6, 0.65, 0.7, 0.75, 0.8, 0.85, 0.9, and 0.95). Considering a range of IOU thresholds, this evaluation metric comprehensively assesses the algorithm's performance.

4.3. Comparison with Other Methods

In order to validate the efficacy of the proposed method in detecting tiny objects, some notable networks, including YOLOv3-spp [55], YOLOv5s, Faster R-CNN, ATSS [56], Cascade R-CNN [57], a tiny object detection method from the literature [21], and the original YOLOv8s algorithm were chosen. The AI-TOD dataset was employed as the experimental data, and the outcomes of these experiments are presented in Table 1.

Table 1. Experimental results of different methods on AI-TOD dataset.

Method	mAP0.5/%	mAP0.5:0.95/%	FPS	Model Size/MB
Faster R-CNN [13]	26.3	11.1	16	236.33
ATSS [56]	30.6	12.8	2	244.56
Cascade R-CNN [57]	30.8	13.8	1	319.45
YOLOv3-spp [55]	41.1	18.6	74	29.97
YOLOv5s	42.2	18.6	102	17.67
YOLOv8s	43.4	19.3	94	21.48
Literature [21]	49.3	20.8	8	942.92
Proposed algorithm	48.3	22.7	37	19.95

For the Faster R-CNN algorithm, this method selects candidate regions by RPN from the features extracted from the backbone network in the first stage. It extracts information from these candidate regions for detection in the second stage. mAP0.5 and mAP0.5:0.95 are both lower, and the FPS is only 16.

For ATSS and Cascade R-CNN algorithms, although both are close to mAP0.5 and mAP0.5:0.95, and both have higher accuracy than Faster R-CNN, both have extremely low FPS and more complex models with more parameters than Faster R-CNN.

For the YOLOv3-spp algorithm, this method uses the SPP structure after the backbone network of the YOLOv3 network to enhance the fusion effect of feature maps at different scales to achieve tiny object detection. Due to this method's poor feature extraction effect and feature fusion, it is not easy to extract and fuse tiny object features in complex backgrounds. Despite the FPS of 74, it has 41.1% for mAP0.5 and 18.6% for mAP0.5:0.95, which are reduced by 2.3% and 0.7%, respectively, compared with the YOLOv8s algorithm.

The YOLOv5s algorithm uses an efficient aggregation network PANet to replace the feature extraction network in the original YOLOv3 network to improve recognition accuracy and speed. Although the method achieves a recognition speed of 102 FPS, it is difficult to achieve better detection results using this algorithm due to interference in the form of a lack of fine-grained feature information of tiny objects in complex background tiny object images. Except for the Faster R-CNN and YOLOv3-spp algorithms, its mAP0.5 and mAP0.5:0.95 are lower than other comparative methods.

Although the YOLOv8s algorithm achieves 43.4% for mAP0.5 and 19.3% for mAP0.5:0.95, with a recognition speed of 94 FPS, the accuracy is lower than that of the method used in the literature [21], and the trade-off between recognition accuracy and speed is not well obtained.

The method used in the literature [21] was combined with Cascade R-CNN by replacing the original IOU metric with the NWD metric to enhance the accuracy rate of

complex background tiny object recognition. Although the mAP0.5 of this method reached 49.3% and mAP0.5:0.95 reached 20.8%, the model size of this method reached 942.92 MB, while the FPS was only 8. This method could be more favorable for practical deployment applications.

The proposed method combines the path aggregation network SPANet, redesigned YOLOv8s feature extraction network, and uses the SPD-Conv module to make the network retain more feature information. Compared with the original YOLOv8s algorithm, mAP0.5 improves by 4.9%, mAP0.5:0.95 improves by 3.4%, and model parameters are reduced, and compared with the algorithm in the literature [21], mAP0.5 differs little, mAP0.5:0.95 improves by 1.9%, FPS improves by 29, and model parameters are significantly reduced, model size reduced 46 times, thus achieving a better detection precision and speed trade-off. It provides the basis for future tiny object detection algorithms that can be deployed on edge devices for real-time detection.

4.4. Ablation Experiments

A series of experiments were conducted to facilitate a comprehensive evaluation of the improved SPANet structure's influence on both recognition precision and speed and to establish the merits of the SPANet structure employed in this research. These experiments involved utilizing the SPANet structure in conjunction with the YOLOv8s algorithm, and their respective outcomes are presented in the initial two rows of Table 2.

Table 2. Experimental results of different modules on AI-TOD dataset.

Method	mAP0.5/%	mAP0.5:0.95/%	FPS	Model Size/MB	GFLOPs
YOLOv8s	43.4	19.3	94	21.48	28.5
YOLOv8s + SPANet	45.9	21.0	47	17.49	62.9
YOLOv8s + SPD-Conv	46.1	20.9	63	24.41	45.8
YOLOv8s + SPD + SPANet	48.3	22.7	37	19.95	86.5

Using the SPANet structure as a path aggregation network increases the model size and GFLOPs, and its FPS decreases, it still manages to improve its mAP0.5 by 2.5% and mAP0.5:0.95 by 1.7% while ensuring real-time performance. The experiments show that using SPANet as a path aggregation network not only improves the accuracy of the original YOLOv8s algorithm and reduces the parameters of the model but also meets the demand of real-time recognition.

To establish the effectiveness of the employed SPD-Conv module, an experimental setup was devised wherein the SPD-Conv module was integrated with the original YOLOv8s framework, while the unmodified YOLOv8s configuration was used as a basis for comparison. The results of this investigation, as depicted in the third row of Table 2, exhibit the impact of incorporating the SPD-Conv module. As observed in Table 3, incorporating the SPD-Conv module leads to a noteworthy 2.7% improvement in mAP0.5 and a 1.6% enhancement in mAP0.5:0.95. Remarkably, these performance gains are achieved while maintaining a comparable model size, thus attaining an improved balance between accuracy and speed. The conducted experiments convincingly demonstrate that including the SPD-Conv module effectively enhances the detection and recognition capabilities of the models.

Table 3. Experimental results of different methods on the TinyPerson dataset.

Method	mAP0.5/%	mAP0.5:0.95/%	FPS	Model Size/MB	GFLOPs
YOLOv3-spp [55]	26.3	8.91	40	29.95	44.0
YOLOv5s	25.4	8.60	35	17.65	23.8
YOLOv7-tiny [58]	13.2	3.18	128	11.70	0.9
YOLOv8s	25.4	8.60	40	21.46	28.4
Proposed method	34.5	11.8	27	19.88	86.4

To ascertain the efficacy of the proposed method, the combination of YOLOv8s + SPD-Conv + SPANet is employed, with YOLOv8s + SPD-Conv serving as the comparative approach. The experimental results, presented in the final two rows of Table 2, highlight the impact of incorporating the SPANet structure. Notably, the utilization of the SPANet structure yields a 2.2% improvement in mAP0.5 and a 1.8% enhancement in mAP0.5:0.95. Moreover, this structural modification effectively reduces the number of model parameters, thereby striking a desirable balance between accuracy and speed, despite a slight decrease in FPS. The conducted experiments conclusively demonstrate that concurrently including the SPD-Conv module and adoption of the SPANet structure significantly enhance the model's detection and identification performance while simultaneously minimizing the model's parameter count.

Figures 5 and 6 show the validation results of YOLOv8s and SP-YOLOv8s on the AI-TOD dataset, respectively, with the GFLOPs of YOLOv8s and SP-YOLOv8s shown in the red boxes.

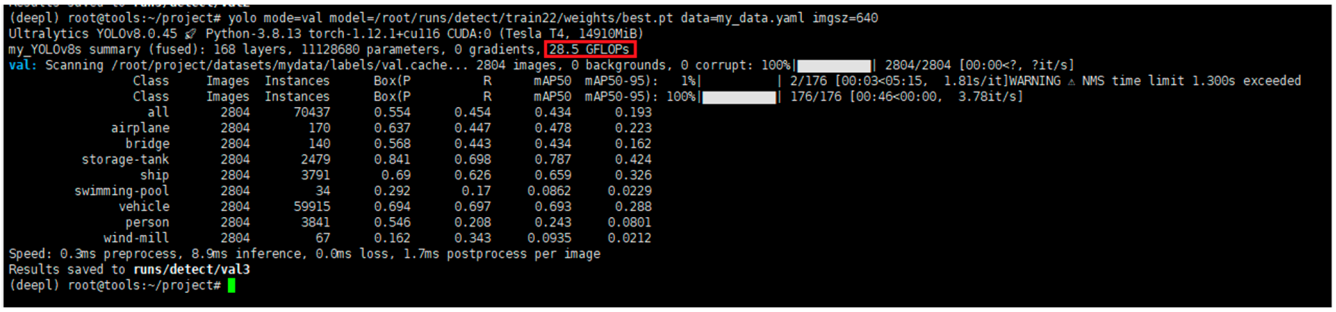

Figure 5. YOLOv8s validation results.

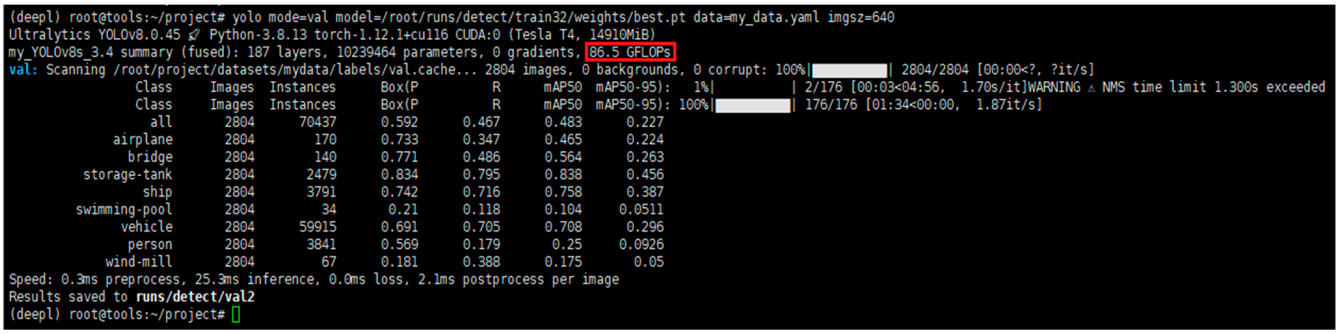

Figure 6. SP-YOLOv8s validation results.

4.5. Validation on Other Datasets

In order to fully validate the generality and robustness of the proposed method, 962 tiny object images are selected as experimental data in the public dataset TinyPerson, of which 696 images are used as the training set and 266 images are used as the validation set, and the training and validation sets are strictly independent. The proposed methodology was compared experimentally with the original YOLOv8s algorithm, YOLOv7-tiny [58] algorithm, YOLOv5s algorithm, and YOLOv3-spp algorithm. The outcomes are presented in Tables 3 and 4 for reference.

Table 4. Results of the proposed method for each category using the TinyPerson dataset.

Class	Precision/%	Recall/%	mAP0.5/%	mAP0.5:0.95/%
Earth_person	48.4	32.7	33.7	12.1
Sea_person	49.6	38.1	35.3	11.6

Table 3 evaluates the detection accuracies through comparative analysis achieved by the different methods. The proposed method outperforms YOLOv3-spp, YOLOv5s, and YOLOv7-tiny in terms of mAP0.5 and mAP0.5:0.95 while maintaining real-time performance. This indicates that the proposed method exhibits superior recognition performance and robustness across various types of tiny objects.

Table 4 compares the recognition results of the proposed method for each category. The empirical evidence substantiates that the proposed methodology exhibits commendable recognition accuracy across diverse categories.

More intuitively, four labeled images with two categories, Earth_person and Sea_person, are shown in Figure 7. The recognition results of YOLOv3-spp algorithm (Figure 7a), YOLOv5s algorithm (Figure 7b), YOLOv8s algorithm (Figure 7c), and SP-YOLOv8s (Figure 7d) are labeled in the figure. SP-YOLOv8s algorithm has a better recognition accuracy than YOLOv3-spp, YOLOv5s, and YOLOv8s.

(**a**)

Figure 7. *Cont.*

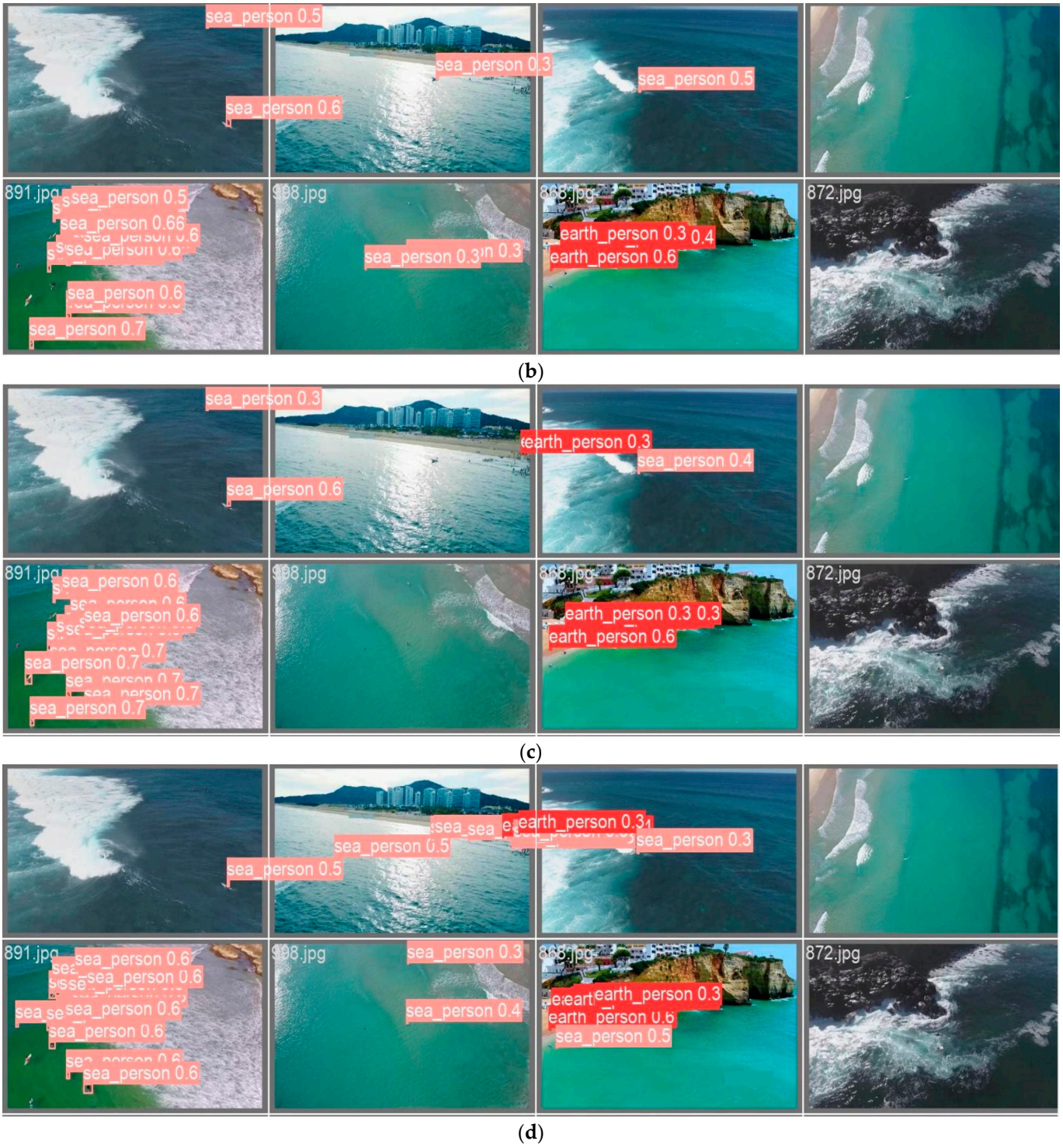

Figure 7. Comparison of detection results of different methods; (**a**) results of YOLOv3-spp; (**b**) results of YOLOv5s; (**c**) results of YOLOv8s; (**d**) results of the proposed method.

5. Conclusions

This paper proposes SP-YOLOv8s, a novel algorithm for detecting and recognizing tiny objects with complex backgrounds. Based on the improved YOLOv8s method, SP-YOLOv8s achieves fast, accurate, stable detection and recognition of tiny objects. The algorithm contains the following two key components: the SPD-Conv module and the SPANet path aggregation network. Since SP-YOLOv8s incorporates these two components lead to a lower FPS than the original YOLOv8s. However, these components reinforce the

feature extraction capability of the baseline network and the fusion effect of feature maps at different scales, thus improving the accuracy of detecting tiny objects with complex backgrounds while maintaining real-time performance. Our proposed method exhibits better performance than existing methods for tiny object detection and recognition. Numerous experiments have verified its effectiveness and superiority. In the future, our improvement direction will focus on achieving higher recognition accuracy while reducing computational complexity. We will strive to optimize the trade-off between speed and precision further. In addition, our future research will focus on exploring the application of our model in detecting and recognizing other complex scenes.

Author Contributions: Conceptualization, H.P. and M.M.; methodology, H.P. and M.M.; software, M.M.; validation, H.P. and M.M.; formal analysis, H.P. and M.M.; investigation, H.P. and M.M.; resources, H.P. and M.M.; data curation, H.P. and M.M.; writing—original draft preparation, M.M.; writing—review and editing, H.P. and M.M.; visualization, H.P. and M.M.; supervision, H.P. and M.M.; project administration, H.P. and M.M.; funding acquisition, H.P. All authors have read and agreed to the published version of the manuscript.

Funding: This research was supported by the Science and Technology Department of Jilin Province under Grant No. 20220201096GX.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Not applicable.

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

References

1. Zhang, Z. *A Study on Harbor Target Recognition in High Resolution Optical Remote Sensing Image*; University of Science and Technology of China: Hefei, China, 2005.
2. Li, W. *Detection of Ship in Optical Remote Sensing Image of Median-Low Resolution*; National University of Defense Science and Technology: Changsha, China, 2008.
3. 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; Part V 13. pp. 740–755.
4. Viola, P.; Jones, M. Rapid object detection using a boosted cascade of simple features. In Proceedings of the 2001 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, CVPR 2001, Kauai, HI, USA, 8–14 December 2001.
5. Dalal, N.; Triggs, B. 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.
6. Lowe, D.G. Distinctive image features from scale-invariant keypoints. *Int. J. Comput. Vis.* **2004**, *60*, 91–110. [CrossRef]
7. Tong, X.-Y.; Xia, G.-S.; Lu, Q.; Shen, H.; Li, S.; You, S.; Zhang, L. Land-cover classification with high-resolution remote sensing images using transferable deep models. *Remote Sens. Environ.* **2020**, *237*, 111322. [CrossRef]
8. Zhu, M.; He, Y.; He, Q. A review of researches on deep learning in remote sensing application. *Int. J. Geosci.* **2019**, *10*, 1–11. [CrossRef]
9. Cha, Y.J.; Choi, W.; Büyüköztürk, O. Deep learning-based crack damage detection using convolutional neural networks. *Comput. Aided Civ. Infrastruct. Eng.* **2017**, *32*, 361–378. [CrossRef]
10. Xiao, Y.; Tian, Z.; Yu, J.; Zhang, Y.; Liu, S.; Du, S.; Lan, X. A review of object detection based on deep learning. *Multimed. Tools Appl.* **2020**, *79*, 23729–23791. [CrossRef]
11. 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, 24–27 June 2014; pp. 580–587.
12. Girshick, R. Fast r-cnn. In Proceedings of the IEEE International Conference on Computer Vision, Washington, DC, USA, 7–13 December 2015; pp. 1440–1448.
13. Ren, S.; He, K.; Girshick, R.; Sun, J. Faster r-cnn: Towards real-time object detection with region proposal networks. In Proceedings of the Neural Information Processing Systems, Montreal, QC, Canada, 7–12 December 2015; Volume 28.
14. 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; Part I 14; pp. 21–37.

15. 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, 24–30 June 2016; pp. 779–788.
16. 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.
17. Zhou, X.; Wang, D.; Krähenbühl, P. Objects as points. *arXiv* **2019**, arXiv:1904.07850.
18. Fang, Z.; Cao, Z.; Xiao, Y.; Gong, K.; Yuan, J. MAT: Multianchor visual tracking with selective Search region. *IEEE Trans. Cybern.* **2020**, *52*, 7136–7150. [CrossRef] [PubMed]
19. Wang, R.; Jiao, L.; Xie, C.; Chen, P.; Du, J.; Li, R. S-RPN: Sampling-balanced region proposal network for small crop pest detection. *Comput. Electron. Agric.* **2021**, *187*, 106290. [CrossRef]
20. Wang, J.; Yang, W.; Guo, H.; Zhang, R.; Xia, G.-S. Tiny object detection in aerial images. In Proceedings of the 2020 25th International Conference on Pattern Recognition (ICPR), Milan, Italy, 10–15 January 2021; pp. 3791–3798.
21. Wang, J.; Xu, C.; Yang, W.; Yu, L. A normalized Gaussian Wasserstein distance for tiny object detection. *arXiv* **2021**, arXiv:2110.13389.
22. Liu, S.; Huang, D.; Wang, Y. Learning spatial fusion for single-shot object detection. *arXiv* **2019**, arXiv:1911.09516.
23. Liu, S.; Qi, L.; Qin, H.; Shi, J.; Jia, J. Path aggregation network for instance segmentation. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–22 June 2018; pp. 8759–8768.
24. Shuai, T.; Sun, K.; Shi, B.; Chen, J. A ship target automatic recognition method for sub-meter remote sensing images. In Proceedings of the 2016 4th International Workshop on Earth Observation and Remote Sensing Applications (EORSA), Guangzhou, China, 4–6 July 2016; pp. 153–156.
25. Cheng, G.; Han, J.; Guo, L.; Qian, X.; Zhou, P.; Yao, X.; Hu, X. Object detection in remote sensing imagery using a discriminatively trained mixture model. *ISPRS J. Photogramm. Remote Sens.* **2013**, *85*, 32–43. [CrossRef]
26. 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.
27. Yang, X.; Yan, J.; Liao, W.; Yang, X.; Tang, J.; He, T. Scrdet++: Detecting small, cluttered and rotated objects via instance-level feature denoising and rotation loss smoothing. *IEEE Trans. Pattern Anal. Mach. Intell.* **2022**, *45*, 2384–2399. [CrossRef]
28. Ding, J.; Xue, N.; Long, Y.; Xia, G.-S.; Lu, Q. Learning roi transformer for oriented object detection in aerial images. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Long Beach, CA, USA, 15–20 June 2019; pp. 2849–2858.
29. Guan, H.; Yu, Y.; Li, D.; Wang, H. RoadCapsFPN: Capsule feature pyramid network for road extraction from VHR optical remote sensing imagery. *IEEE Trans. Intell. Transp. Syst.* **2021**, *23*, 11041–11051. [CrossRef]
30. Zhang, C.; Lam, K.-M.; Wang, Q. CoF-Net: A Progressive Coarse-to-Fine Framework for Object Detection in Remote-Sensing Imagery. *IEEE Trans. Geosci. Remote Sens.* **2023**, *61*, 5600617. [CrossRef]
31. Deng, L.; Bi, L.; Li, H.; Chen, H.; Duan, X.; Lou, H.; Zhang, H.; Bi, J.; Liu, H. Lightweight aerial image object detection algorithm based on improved YOLOv5s. *Sci. Rep.* **2023**, *13*, 7817. [CrossRef] [PubMed]
32. Wang, C.; Sun, M.; Cao, Y.; He, K.; Zhang, B.; Cao, Z.; Wang, M. Lightweight Network-Based Surface Defect Detection Method for Steel Plates. *Sustainability* **2023**, *15*, 3733. [CrossRef]
33. Anitha, A.; Shivakumara, P.; Jain, S.; Agarwal, V. Convolution Neural Network and Auto-encoder Hybrid Scheme for Automatic Colorization of Grayscale Images. In *Smart Computer Vision*; Springer: Berlin/Heidelberg, Germany, 2023; pp. 253–271.
34. Chen, S.; Cheng, T.; Fang, J.; Zhang, Q.; Li, Y.; Liu, W.; Wang, X. TinyDet: Accurate Small Object Detection in Lightweight Generic Detectors. *arXiv* **2023**, arXiv:2304.0342.
35. Yang, X.; Sun, H.; Sun, X.; Yan, M.; Guo, Z.; Fu, K. Position detection and direction prediction for arbitrary-oriented ships via multitask rotation region convolutional neural network. *IEEE Access* **2018**, *6*, 50839–50849. [CrossRef]
36. Van Etten, A. You only look twice: Rapid multi-scale object detection in satellite imagery. *arXiv* **2018**, arXiv:1805.09512.
37. Chen, S.; Zhan, R.; Zhang, J. Geospatial object detection in remote sensing imagery based on multiscale single-shot detector with activated semantics. *Remote Sens.* **2018**, *10*, 820. [CrossRef]
38. Wang, P.; Sun, X.; Diao, W.; Fu, K. FMSSD: Feature-merged single-shot detection for multiscale objects in large-scale remote sensing imagery. *IEEE Trans. Geosci. Remote Sens.* **2019**, *58*, 3377–3390. [CrossRef]
39. Li, M.; Guo, W.; Zhang, Z.; Yu, W.; Zhang, T. Rotated region based fully convolutional network for ship detection. In Proceedings of the IGARSS 2018-2018 IEEE International Geoscience and Remote Sensing Symposium, Valencia, Spain, 22–27 July 2018; pp. 673–676.
40. Fu, Y.; Wu, F.; Zhao, J. Context-aware and depthwise-based detection on orbit for remote sensing image. In Proceedings of the 2018 24th International Conference on Pattern Recognition (ICPR), Beijing, China, 20–24 August 2018; pp. 1725–1730.
41. Zhang, W.; Wang, S.; Thachan, S.; Chen, J.; Qian, Y. Deconv R-CNN for small object detection on remote sensing images. In Proceedings of the IGARSS 2018—2018 IEEE International Geoscience and Remote Sensing Symposium, Valencia, Spain, 22–27 July 2018; pp. 2483–2486.
42. Schilling, H.; Bulatov, D.; Niessner, R.; Middelmann, W.; Soergel, U. Detection of vehicles in multisensor data via multibranch convolutional neural networks. *IEEE J. Sel. Top. Appl. Earth Obs. Remote Sens.* **2018**, *11*, 4299–4316. [CrossRef]
43. Liu, W.; Ma, L.; Wang, J. Detection of multiclass objects in optical remote sensing images. *IEEE Geosci. Remote Sens. Lett.* **2018**, *16*, 791–795. [CrossRef]

44. Ying, X.; Wang, Q.; Li, X.; Yu, M.; Jiang, H.; Gao, J.; Liu, Z.; Yu, R. Multi-attention object detection model in remote sensing images based on multi-scale. *IEEE Access* **2019**, *7*, 94508–94519. [CrossRef]
45. Ioffe, S.; Szegedy, C. Batch normalization: Accelerating deep network training by reducing internal covariate shift. In Proceedings of the International Conference on Machine Learning, Lille, France, 6–11 July 2015; pp. 448–456.
46. Klambauer, G.; Unterthiner, T.; Mayr, A.; Hochreiter, S. Self-normalizing neural networks. In Proceedings of the 31stConferenceonNeuralInformationProcessingSystems(NIPS2017), LongBeach, CA, USA, 4–9 December 2017; Volume 30.
47. 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]
48. Wang, C.-Y.; Liao, H.-Y.M.; Yeh, I.-H. Designing Network Design Strategies Through Gradient Path Analysis. *arXiv* **2022**, arXiv:2211.04800.
49. Ge, Z.; Liu, S.; Wang, F.; Li, Z.; Sun, J. Yolox: Exceeding yolo series in 2021. *arXiv* **2021**, arXiv:2107.08430.
50. Cao, Y.; Chen, K.; Loy, C.C.; Lin, D. Prime sample attention in object detection. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Seattle, WA, USA, 13–19 June 2020; pp. 11583–11591.
51. Sunkara, R.; Luo, T. No more strided convolutions or pooling: A new CNN building block for low-resolution images and small objects. In Proceedings of the Machine Learning and Knowledge Discovery in Databases: European Conference, ECML PKDD 2022, Grenoble, France, 19–23 September 2022; Part III; pp. 443–459.
52. Yu, X.; Gong, Y.; Jiang, N.; Ye, Q.; Han, Z. Scale match for tiny person detection. In Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision, Snowmass Village, CO, USA, 1–5 March 2020; pp. 1257–1265.
53. 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, 18–22 June 2018; pp. 3974–3983.
54. 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]
55. Redmon, J.; Farhadi, A. Yolov3: An incremental improvement. *arXiv* **2018**, arXiv:1804.02767.
56. 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, 14–19 June 2020; pp. 9759–9768.
57. 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, 18–22 June 2018; pp. 6154–6162.
58. Wang, C.-Y.; Bochkovskiy, A.; Liao, H.-Y.M. YOLOv7: Trainable bag-of-freebies sets new state-of-the-art for real-time object detectors. *arXiv* **2022**, arXiv:2207.02696.

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.