sustainability

MDPI

Article

A YOLO-Based Target Detection Model for Offshore Unmanned Aerial Vehicle Data

Zhenhua Wang [1], Xinyue Zhang [1], Jing Li [1] and Kuifeng Luan [2,*]

[1] College of Information Technology, Shanghai Ocean University, Shanghai 203106, China; zh-wang@shou.edu.cn (Z.W.); ankh_zhang@163.com (X.Z.); m200701393@st.shou.edu.cn (J.L.)
[2] College of Marine Sciences, Shanghai Ocean University, Shanghai 203106, China
* Correspondence: kfluan@shou.edu.cn; Tel.: +86-21-6190-0623

Abstract: Target detection in offshore unmanned aerial vehicle data is still a challenge due to the complex characteristics of targets, such as multi-sizes, alterable orientation, and complex backgrounds. Herein, a YOLO-based detection model (YOLO-D) was proposed for target detection in offshore unmanned aerial vehicle data. Based on the YOLOv3 network, the residual module was improved by establishing dense connections and adding a dual-attention mechanism (CBAM) to enhance the use of features and global information. Then, the loss function of the YOLO-D model was added to the weight coefficients to increase detection accuracy for small-size targets. Finally, the feature pyramid network (FPN) was replaced by the secondary recursive feature pyramid network to reduce the impacts of a complicated environment. Taking the car, boat, and deposit near the coastline as the targets, the proposed YOLO-D model was compared against other models, including the faster R-CNN, SSD, YOLOv3, and YOLOv5, to evaluate its detection performance. The results showed that the evaluation metrics of the YOLO-D model, including precision (*Pr*), recall (*Re*), average precision (*AP*), and the mean of average precision (*mAP*), had the highest values. The mAP of the YOLO-D model increased by 37.95%, 39.44%, 28.46%, and 5.08% compared to the faster R-CNN, SSD, YOLOv3, and YOLOv5, respectively. The AP of the car, boat, and deposit reached 96.24%, 93.70%, and 96.79% respectively. Moreover, the YOLO-D model had a higher detection accuracy than other models, especially in the detection of small-size targets. Collectively, the proposed YOLO-D model is a suitable model for target detection in offshore unmanned aerial vehicle data.

Keywords: offshore monitoring; target detection; deep learning; YOLO; unmanned aerial vehicle

Citation: Wang, Z.; Zhang, X.; Li, J.; Luan, K. A YOLO-Based Target Detection Model for Offshore Unmanned Aerial Vehicle Data. *Sustainability* **2021**, *13*, 12980. https://doi.org/10.3390/su132312980

Academic Editor: Eben Broadbent

Received: 12 October 2021
Accepted: 18 November 2021
Published: 24 November 2021

1. Introduction

Unmanned aerial vehicle (UAV) imagery shows great potential for offshore monitoring due to real-time collection of temporal/spatial data [1]. Different targets can be identified by the detection models in UAV data. However, it is still unavailable for an automatic target detection model due to the multi-sizes, alterable orientation, and complex backgrounds of the target objects. Currently, deep learning has been widely used for extracting features and detecting targets. Deep learning has great potential for improving the accuracy and efficiency of target detection in offshore unmanned aerial vehicle data.

In general, target detection based on deep learning can be divided into two major types, a two-stage detection model and a one-stage detection model [2]. In a two-stage target detection model, different targets are detected based on the series of candidate boxes [3]. The model based on a region with a CNN feature (RCNN) is a typical two-stage target detection model, showing great advantage in detection accuracy and positioning accuracy [4–7]. In a one-stage target detection model, different targets are directly detected, where target detection is abstracted as a regression problem [8]. Models based on a you-only-look-once (YOLO) or a single-shot multibox detector (SSD) are typical one-stage target detection models, showing great advantage in detection efficiency [9–13]. In addition, there are some detection methods deigned based on machine learning, such as

sparse target detection [14–16], sub-pixel target detection [17,18], and visual saliency target detection [19,20]. In recent years, there also have been efforts to apply and improve these target detection models in UAVs for offshore monitoring [21–26]. However, these models are still not the optimal option for target detection in offshore unmanned aerial vehicle data, due to the complex characteristics of targets, such as multi-sizes, alterable orientation, and complex backgrounds.

In this study, we proposed a target detection model for target detection in offshore unmanned aerial vehicle data based on the improved YOLOv3 (YOLO-D) network.

The residual module was improved to enhance the use of features and global information. The loss function of YOLO-D was then improved to enhance the detection accuracy for small targets. The feature pyramid network (FPN) was finally replaced by the secondary recursive feature pyramid network to reduce the impacts of a complicated environment.

2. Dataset and Model

2.1. Dataset

UAV data were acquired from 2019 to 2020, which were collected over Jinshan District, Fengxian District, and Pudong New District in Shanghai (China), and 504 images with a size of 1920 × 1080 pixels were extracted. The labeling software LabelImg v1.8.5 was used to label the offshore monitoring targets, including a car, a boat, and a deposit (Figure 1).

Figure 1. Offshore monitoring targets: (**a**) car, (**b**) boat, and (**c**) deposit.

The images were preprocessed to balance the ratio of positive and negative samples and strengthen the learning efficiency of small-size targets. The images were then augmented by rotating, trimming, horizontal flipping, and splicing. The expanded dataset contained 1010 images, included 12,747 marked cars, 1247 marked boats, and 1431 marked deposits. According to the standard of the COCO dataset, a target with pixels smaller than 32 × 32 was classified as a small-size target. A target with pixels greater than 96 × 96 was classified as a large-size target [27]. Here, the number of small-size targets exceeded half of the total number of targets.

2.2. Model

Figure 2 shows the flowchart of the YOLO-based detection model (YOLO-D), including the improved backbone network, the improved feature pyramid network (FPN), and improved CCR modules.

2.2.1. Backbone Network of the YOLO-D Model

Based on Darknet-53, the backbone network of the YOLO-D model was improved by establishing dense connections and adding a dual-attention mechanism (CBAM). Figure 3 shows the mechanism of dual attention, including channel and spatial attention, which not only considered the importance of different feature channels but also considered the importance of different positions of the same feature channel [28].

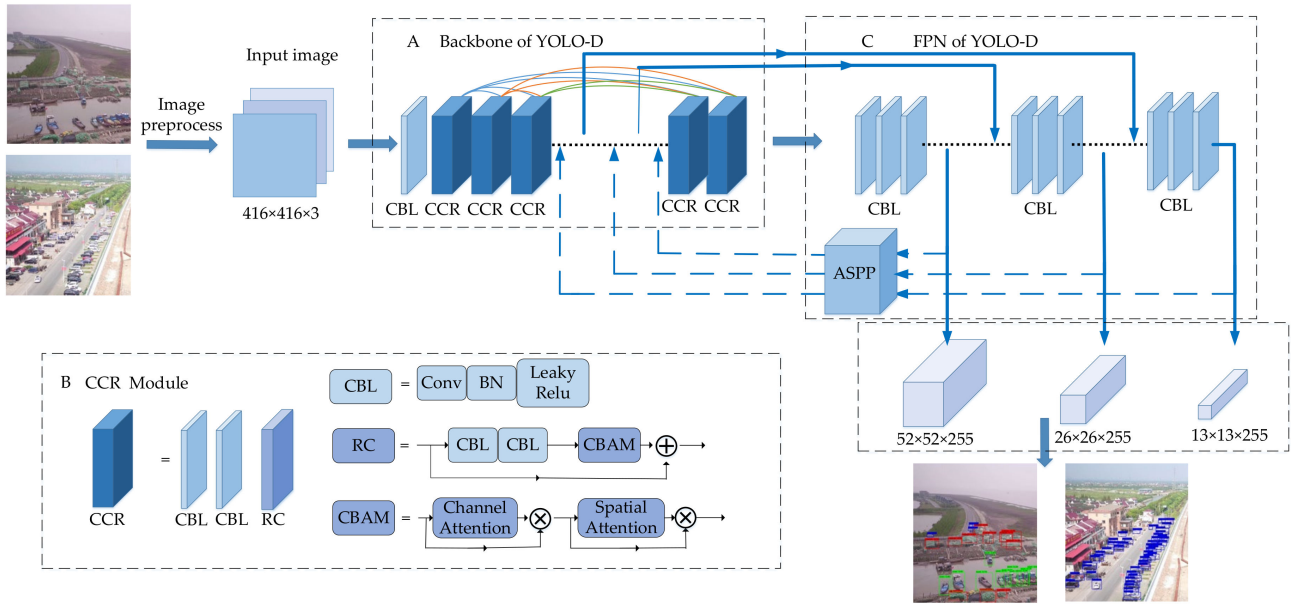

Figure 2. Flowchart of YOLO-D. (**A**) Backbone network, (**B**) CCR module, and (**C**) feature pyramid network (FPN). The dotted lines with arrows indicate the flow of the FPN's first output data.

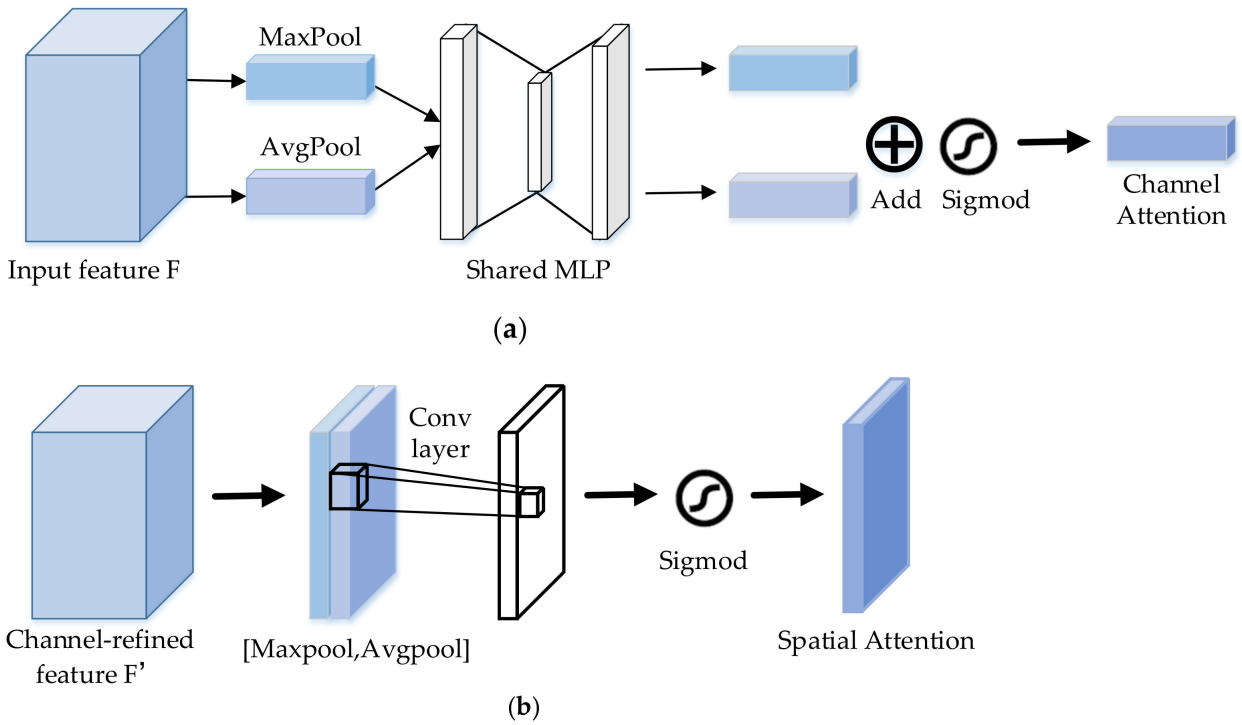

Figure 3. Attention module schematic: (**a**) channel attention and (**b**) spatial attention.

In the YOLO-D model, dual attention (CBAM) was added into the residual module as the RC module (the main components of the CCR module) to enhance feature extraction (Figure 4).

Dense connections between CCR modules were established to strengthen the transmission and use of features and improve the use of features (Figure 5).

X_i represents the output of the i-th layer. X_n = RC(CBL(CBL([X_0, X_1, X_2, . . . , X_{n-1}]))), and [X_0, X_1, X_2, . . . , X_{n-1}] represents the splicing of the output features from layer 0 to layer n−1. CBL(x) represents the passing x through the CBL module. RC(x) represents the

passing x through the RC module. The CBL module (Figure 2) includes a convolutional layer, a regularization layer, and an activation layer. The RC module (Figure 4) is a residual module that includes two CBL modules and a CBAM module [29].

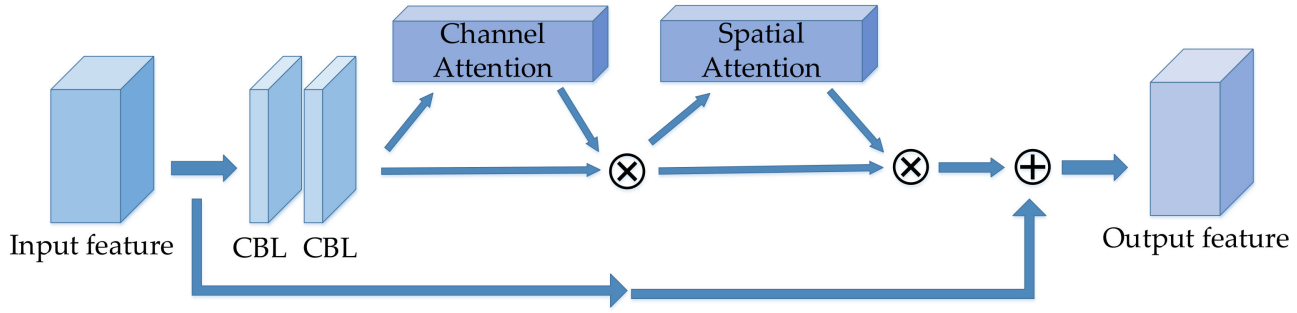

Figure 4. Schematic diagram of the RC module.

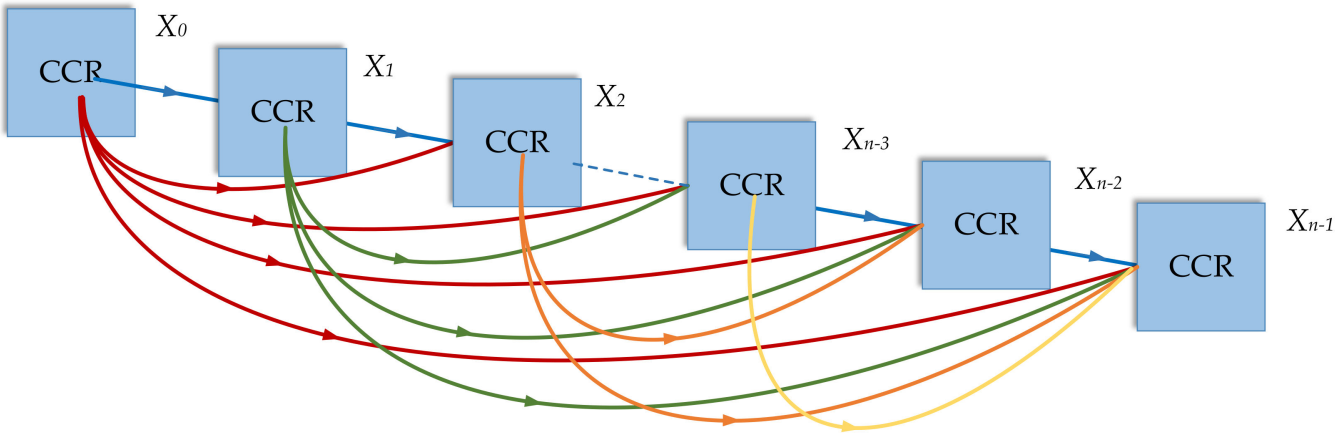

Figure 5. Data transfer diagram in dense connections.

2.2.2. Feature Pyramid Network of the YOLO-D Model

A feature pyramid network (FPN) can make use of the feature information about the bottom layer and the high layer at the same time and construct multi-size feature images [30]. Here, the FPN of YOLOv3 was improved by building a secondary recursive feature pyramid. The first-output features (f_n^{out}) of the FPN were concatenated with the first-input features (f_n^{in}) of the backbone network. Then, the first-output features passed through the atrous spatial pyramid pooling (ASPP) [31] as the second-input features (f_n) of the backbone network, ($f_n = ASPP(concat(f_n^{in}, f_n^{out}))$). Finally, the second-output features of the feature pyramid network were used as the final features and were outputted to the detection layer. Figure 6 shows the network structure of the YOLO-D model, including the structure of ASPP (Figure 6A), the second recursive model of the YOLO-D model (Figure 6B), and the network structure of the YOLO-D model (Figure 6C).

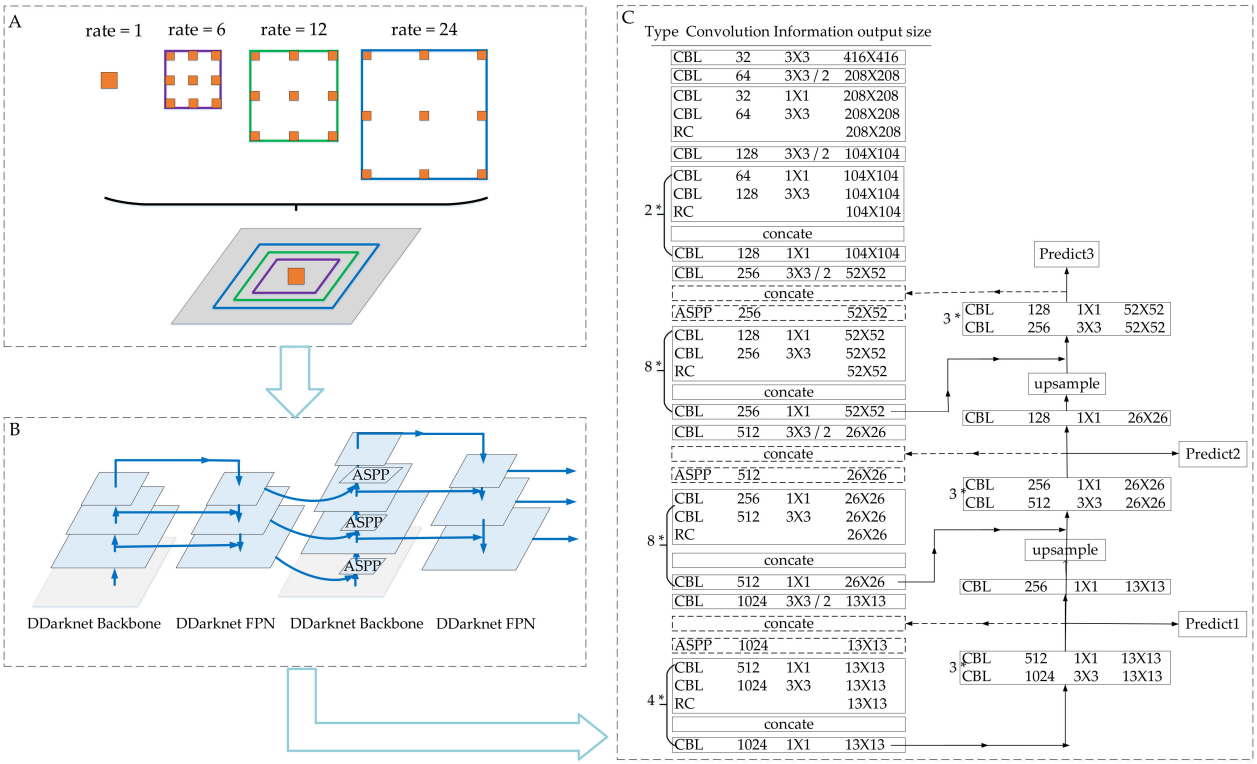

Figure 6. Network structure of the YOLO-D model. (**A**) Atrous spatial pyramid pooling, (**B**) the second recursive model of YOLO-D, and (**C**) the network structure of the YOLO-D model. The dotted line is the recursive data flow. The dashed box is the module that is executed only when recursive data are used.

2.2.3. Loss Function of the YOLO-D Model

The loss function of YOLOv3 is the cross-entropy loss function, which is composed of center coordinate loss, confidence loss, and classification loss. In the YOLO-D model, the center coordinate loss function was replaced by the *GIOU* loss function [32] and the confidence and classification loss functions were improved by adding weight coefficients [33], which enhanced the learning efficiency of confusing samples and difficult samples.

The YOLO-D loss function is defined as:

$$Loss = L_{GIOU} + L_{Conf\_fl} + L_{Class\_fl} \tag{1}$$

L_{GIOU} is the *GIOU* loss function, defined as:

$$L_{GIOU} = 1 - GIOU \tag{2}$$

$$GIOU = IOU - \frac{|C - (A \cup B)|}{|C|} \tag{3}$$

$$IOU = \frac{|A \cap B|}{|A \cup B|} \tag{4}$$

A is the true frame. B is the predicted frame.$A \cup B$ is the area of the union of A and B. $A \cap B$ is the area of the intersection of A and B. C is the area of the smallest bounding box including A and B.

The focal loss function L_{focal} is defined as [27]:

$$L_{focal} = \begin{cases} -\alpha(1-y')^{\gamma} log y', & y = 1 \\ -(1-\alpha)y'^{\gamma} log(1-y'), & y = 0 \end{cases} \tag{5}$$

$\alpha \in (0,1]$ and $\gamma \in (0,2]$ are self-defined constants. y' is the predicted output.y is the label of the real sample. When $\gamma > 1$, the network focuses on samples with learning difficulties. Weight coefficients are added to the original confidence loss. The new confidence loss function L_{Conf_fl} is defined as:

$$\begin{aligned} L_{Conf\_fl} = (|y - y'|)^{\gamma} \times (\sum_{i=0}^{s^2} \sum_{j=0}^{B} I_{ij}^{obj} [\hat{C}_i^j log(C_i^j) + (1 - \hat{C}_i^j) log(1 - C_i^j)] \\ + \lambda_{noobj} \sum_{i=0}^{s^2} \sum_{j=0}^{B} I_{ij}^{noobj} [\hat{C}_i^j log(C_i^j) + (1 - \hat{C}_i^j) log(1 - C_i^j)]) \end{aligned} \quad (6)$$

If the predicted box contains an object, $I_{ij}^{obj} = 1$, $I_{ij}^{noobj} = 0$, otherwise $I_{ij}^{obj} = 0$, $I_{ij}^{noobj} = 1$. C_i^j is the predicted value. $\hat{C}_i^j$ is the true value. If the predicted box is responsible for predicting an object, $\hat{C}_i^j = 1$, otherwise $\hat{C}_i^j = 0$. y' is the predicted output. y is the label of the real sample. We set $\gamma = 2$. If one sample is difficult to detect, y' will tend to 0 and the confidence loss value will increase.

L_{Class_fl} is the classification loss function, which is defined as:

$$L_{Class\_fl} = |a + y - 1| \times (\sum_{i=0}^{s^2} I_{ij}^{obj} \sum_{cIclass} [\hat{p}_i(c) \log(p_i(c)) + (1 - \hat{p}_i(c)) \log(1 - p_i(c))]) \quad (7)$$

$\hat{p}_i(c)$ is the probability that the anchor is predicted to be class c. $p_i(c)$ is the true value. If the predicted box contains an object and is difficult to be classified, the correct classification loss will be smaller and the wrong classification loss will be greater.

3. Experiment

To evaluate the detection performance of the YOLO-D model, two comparison experiments were conducted. In the ablative experiments, the improved efficiency of the backbone network, feature pyramid network, and loss function was analyzed. The detection performance of the YOLO-D model was evaluated by comparing it with other end-to-end models, including the faster R-CNN [7], SSD [12], YOLOv3 [11], and YOLOv5.

3.1. Experiment Metrics

Five different metrics, that is, average Precision (*AP*), the mean of average precision (*mAP*), precision (*Pr*), recall (*Re*), and frame per second (*FPS*), were calculated to estimate the target detection performance [34]:

$$AP = \int_0^1 P(r)dr \quad (8)$$

$$mAP = \frac{1}{N} \sum_{i=1}^{N} AP_i \quad (9)$$

$$Pr = \frac{TP}{TP + FP} \quad (10)$$

$$Re = \frac{TP}{TP + FN} \quad (11)$$

$$FPS = \frac{1}{t} \quad (12)$$

where the true positive (*TP*) represents the number of positive samples that are predicted to be positive, the false positive (*FP*) represents the number of samples that are predicted to be positive but are actually negative, and the false negative (*FN*) represents the number of samples predicted to be negative but actually positive. *FPS* represents the number of

pictures that can be processed per second, and t represents the time required to process a picture.

The experiment was based on the tensor flow deep learning framework, and we built a virtual environment of Python 3.6 and TensorFlow-gpu 2.0 on Anaconda. The training and accuracy tests were carried out on a Ubuntu 16.04.4 system, NVIDIA Tesla P100 16 GB graphics card, and CUDA 10.0. The intersection over union (*IOU*) threshold was set as 0.3 and the score threshold as 0.45.

3.2. Results

Figure 7 showed the images and detection results obtained by the YOLO-D model. Taking *Re* and *Pr* as the abscissa and the ordinate axis, respectively, the P-R curve of each class is shown in Figure 8.

Figure 7. The original images and experimental results of YOLO-D.

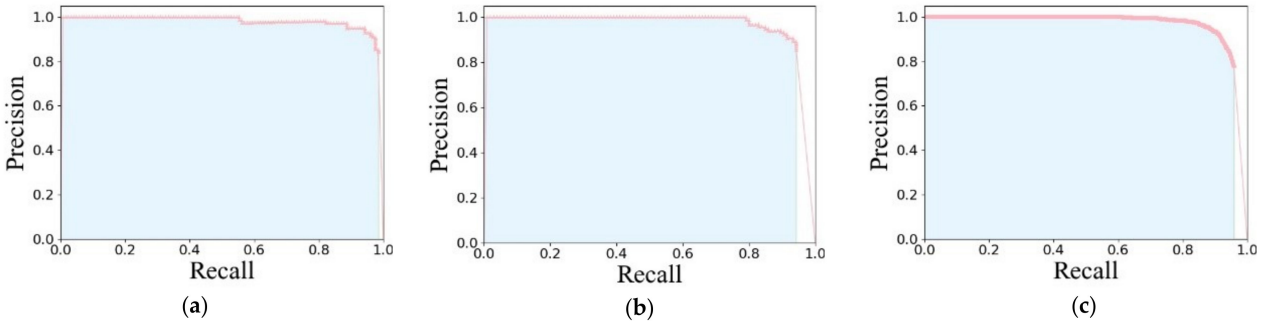

Figure 8. P-R curve of each class of the YOLO-D model: (**a**) P-R curve of the boat, (**b**) P-R curve of the car, and (**c**) P-R curve of the deposit.

3.2.1. Ablation Experiment

Table 1 shows the performance comparison between ablation studies, including YOLOv3 with an improved backbone, YOLOv3 with an improved FPN, and YOLOv3 with an improved loss function. Based on Table 1, we can see that all designs of the YOLO-D

model could prominently enhance the *AP* and *mAP* of each class target. In the YOLO-D model, the *AP* of the boat, car, and deposit reached 93.7%, 96.24%, and 96.79%, respectively. Notably, the *mAP* reached 95.58%, although the speed increase of the YOLO-D model was not obvious.

Table 1. Comparison of evaluation metrics in ablation studies.

Improved Backbone Network	Improved FPN	Improved Loss Function	*AP*_Boat	*AP*_Car	*AP*_Deposit	*mAP*@0.5	FPS
√	√	—	83.39	91.75	91.34	88.83	3
√	—	√	91.98	94.10	92.84	92.97	5
—	√	√	80.02	94.99	94.52	89.84	4.5
√	√	√	93.70	96.24	96.79	95.58	3

3.2.2. Comparison with Other Models

The detection performance of the YOLO-D model was further compared with other models, including the faster R-CNN, SSD, YOLOv3, and YOLOv5. As shown in Figure 9 and Table 2, the YOLO-D model had the highest values of *Pr*, *Re*, *AP*, and *mAP*. The *mAP* value of the YOLO-D model increased by 37.95%, 39.44%, 28.46%, and 5.08% compared to the faster R-CNN, SSD, YOLOv3, and YOLOv5, respectively. Importantly, the *AP* value of the car reached 96.24%. Notably, most of car the targets were small-size targets. The detection speed of YOLO-D was 10 times faster than that of the faster R-CNN and SSD but was slightly slower than that of YOLOv3 and YOLOv5. Collectively, the YOLO-D model reduced the rate of false detection and showed great potential for accurate detection, especially for the detection of small-size targets.

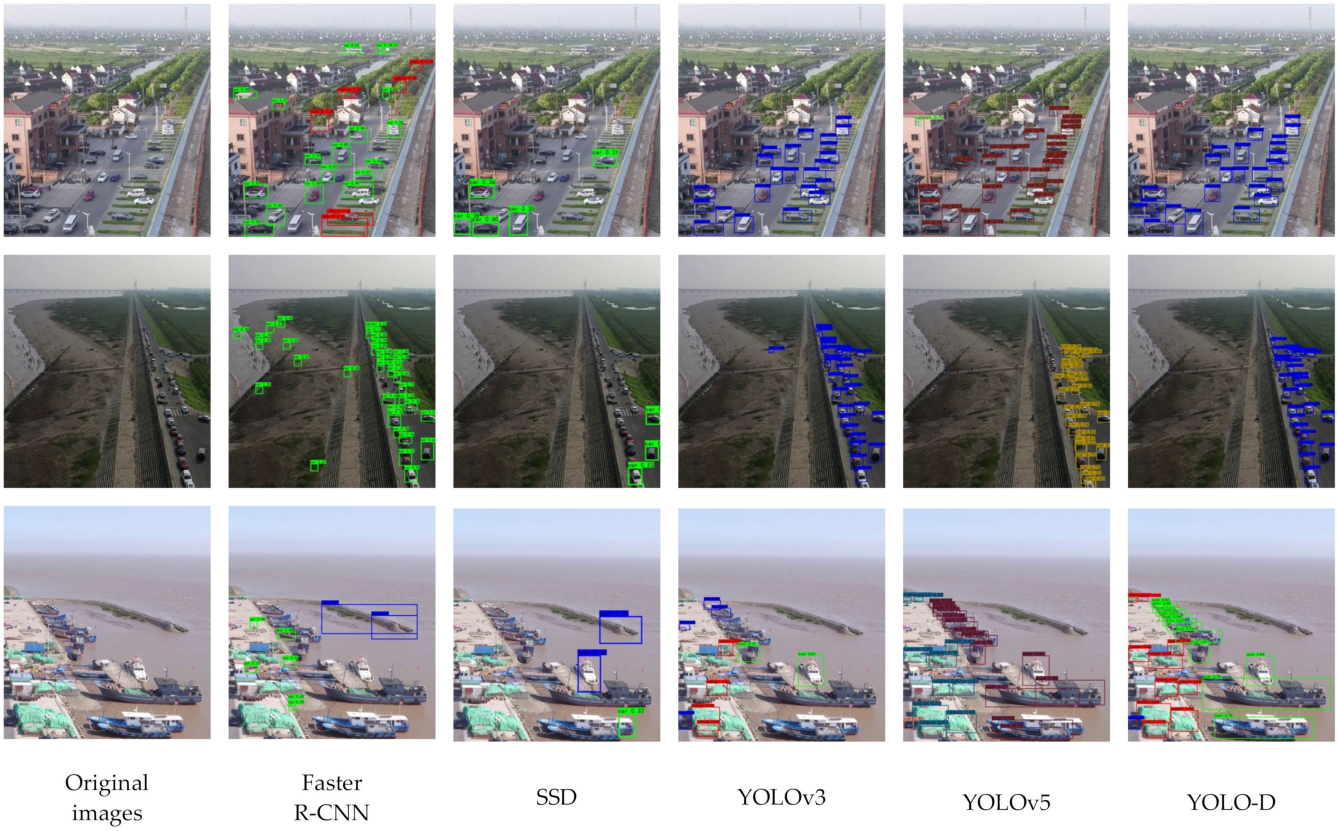

Figure 9. Comparison of detection performance using different models.

Table 2. Comparison of evaluation metrics using different models.

Model	*Pr*	*Re*	*AP*_Boat	*AP*_Car	*AP*_Deposit	*mAP*@0.5	FPS
Faster R-CNN	79.97	39.61	60.81	55.34	56.74	57.63	0.35
SSD	91.44	16.32	68.14	52.11	47.25	55.84	0.38
YOLOv3	90.70	55.08	60.34	78.67	63.48	67.50	9
YOLOv5	92.20	87.40	93.10	88.70	89.90	90.50	14
YOLO-D	92.70	92.06	93.70	96.24	96.79	95.58	3

4. Discussion

Unmanned aerial vehicle (UAV) obtains increased real-time datasets for offshore monitoring. However, the automatic detection of UAV data is still a tricky problem due to the multi-sizes, alterable orientation, and complex backgrounds of the target objects. It is particularly difficult to detect small-size targets. In this study, a YOLO-based target detection model (YOLO-D) was proposed for offshore unmanned aerial vehicle data.

Compared with other detection models, such as the faster R-CNN, SSD, YOLOv3, and YOLOv5, the proposed YOLO-D model can significantly enhance detection accuracy. The evaluation metrics of the YOLO-D model, including precision (*Pr*), recall (*Re*), average precision (*AP*), and the mean of average precision (*mAP*), had the highest score. The *mAP* value of detection targets increased by 37.95%, 39.44%, 28.46%, and 5.08% compared to the faster R-CNN, SSD, YOLOv3, and YOLOv5, respectively. The result suggested that the YOLO-D model has great potential for accurate detection of offshore UAV data.

In addition, the YOLO-D model can efficiently and accurately detect targets in offshore UAV data. However, there are still some limitations. The YOLO-D model is designed based on the spatial information in offshore UAV data. However, it ignores the contextual information, which is also important for target detection. The YOLO-D model shows great potential for accurate detection, but the speed is slightly slower than YOLOv3 and YOLOv5. In the future, we will further improve the YOLO-D model to enhance the accuracy and efficiency of target detection in offshore UAV data.

Taken together, this study proposed a YOLO-D model for the detection of offshore UAV data. In this model, the residual module is improved by establishing dense connections and adding a dual-attention mechanism (CBAM), which can enhance the use of features and global information. The loss function of YOLO-D is improved by adding weight coefficients, which can enhance the detection accuracy for small targets. The feature pyramid network (FPN) is replaced by a secondary recursive feature pyramid network to reduce the impacts of a complicated environment. The YOLO-D model shows great potential for accurate detection of offshore UAV data, especially for the detection of small-size targets.

Author Contributions: Conceptualization, Z.W. and X.Z.; methodology, X.Z. and J.L.; validation, K.L. and Z.W.; formal analysis, Z.W. and K.L.; investigation, X.Z. and J.L.; writing—original draft preparation, X.Z. and Z.W.; writing—review and editing, Z.W. and K.L.; supervision, Z.W.; funding acquisition, Z.W. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by the Capacity Development for Local College Project (grant no. 19050502100).

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Acknowledgments: We would like to thank the anonymous reviewers for their valuable suggestions.

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

References

1. Zang, H.; Xu, X. Application of UAV Remote Sensing Technology in Marine Resources Supervision. *Inf. Technol. Informatiz.* **2020**, *25*, 231–233. [CrossRef]
2. Xu, D.; Wang, Y.; Li, F. A Survey of Research on Typical Target Detection Algorithms for Deep Learning. *Comput. Eng. Appl.* **2021**, *57*, 10–25. [CrossRef]
3. Li, P.; Yu, H. Survey of object detection algorithms based on two classification standards. *Appl. Res. Comput.* **2021**, *38*, 2582–2589. [CrossRef]
4. Girshick, R.; Donahue, J.; Darrell, T. 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*; IEEE: Piscataway, NJ, USA, 2014; pp. 580–587. [CrossRef]
5. He, K.; Zhang, X.; Ren, S. Spatial pyramid pooling in deep convolutional networks for visual recognition. *IEEE Trans. Pattern Anal. Mach. Intell.* **2015**, *37*, 1904–1916. [CrossRef] [PubMed]
6. Girshick, R. Fast R-CNN. In *Proceedings of the IEEE International Conference on Computer Vision, Santiago, Chile, 7–13 December 2015*; IEEE: Piscataway, NJ, USA, 2015; pp. 1440–1448. [CrossRef]
7. Ren, S.; He, K.; Girshick, R. Faster R-CNN: Towards real-time object detection with region proposal networks. *IEEE Trans. Pattern Anal. Mach. Intell.* **2017**, *39*, 1137–1149. [CrossRef] [PubMed]
8. Cheng, X.; Song, C.; Shi, J. A Survey of Generic Object Detection Methods Based on Deep Learning. *Acta Electron. Sin.* **2021**, *49*, 1428–1438.
9. Redmon, J.; Divvala, S.; Girshick, R. You only look once: Unified, real-time object detection. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 27–30 June 2016*; IEEE: Piscataway, NJ, USA, 2016; pp. 779–788. [CrossRef]
10. Redmon, J.; Farhadi, A. YOLO9000: Better, Faster, Stronger. In *Proceedings of the 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Honolulu, HI, USA, 22–25 July 2017*; IEEE: Piscataway, NJ, USA, 2017; pp. 6517–6525. [CrossRef]
11. Redmon, J.; Farhadi, A. Yolov3: An incremental improvement. *arXiv* **2018**, arXiv:1804.02767.
12. Liu, W.; Anguelov, D.; Erhan, D. SSD: Single Shot MultiBox Detector. In *Proceedings of the 2016 European Conference on Computer Vision, Amsterdam, The Netherlands, 11–14 October 2016*; Springer International Publishing: Cham, Switzerland, 2016; pp. 21–37. [CrossRef]
13. Fu, C.-Y.; Liu, W.; Ranga, A. Dssd: Deconvolutional single shot detector. *arXiv* **2017**, arXiv:1701.06659.
14. Kong, X.; Yang, C.; Cao, S. Infrared Small Target Detection via Nonconvex Tensor Fibered Rank Approximation. In *IEEE Transactions on Geoscience and Remote Sensing*; IEEE: Piscataway, NJ, USA, 2021; pp. 1–21. [CrossRef]
15. Zhao, C.; Yao, X.; Zhang, L. Target Detection Sparse Algorithm by Recursive Dictionary Updating and GPU Implementation. *Acta Opt. Sin.* **2016**, *36*, 279–287.
16. Zhang, L.; Peng, L.; Zhang, T. Infrared Small Target Detection via Non-Convex Rank Approximation Minimization Joint l2,1 Norm. *Remote Sens.* **2018**, *10*, 1821. [CrossRef]
17. Wang, P.; Wang, L.; Leung, H. Super-Resolution Mapping Based on Spatial-Spectral Correlation for Spectral Imagery. *IEEE Trans. Geosci. Remote Sens.* **2021**, *59*, 2256–2268. [CrossRef]
18. Li, R.; Latifi, S. Improving Hyperspectral Subpixel Target Detection Using Hybrid Detection Space. *J. Appl. Remote Sens.* **2017**, *12*, 015022. [CrossRef]
19. Fang, X.; Liu, J.; Zeng, D. Detection and identification of unsupervised ships and warships on sea surface based on visual saliency. *Opt. Precis. Eng.* **2017**, *25*, 1300–1311. [CrossRef]
20. Chen, Y.; Song, B.; Du, X. Infrared Small Target Detection Through Multiple Feature Analysis Based on Visual Saliency. *IEEE Access* **2019**, *7*, 38996–39004. [CrossRef]
21. Li, S.; Zhang, W.; Li, G. Vehicle detection in uav traffic video based on convolution neural network. In *Proceedings of the 2018 IEEE Conference on Multimedia Information Processing and Retrieval (MIPR), Miami, FL, USA, 10–12 April 2018*; IEEE: Piscataway, NJ, USA, 2018; pp. 1–6. [CrossRef]
22. Luo, X.; Tian, X.; Zhang, H. Fast Automatic Vehicle Detection in UAV Images Using Convolutional Neural Networks. *Remote Sens.* **2020**, *12*, 1994. [CrossRef]
23. Kellenberger, B.; Volpi, M.; Tuia, D. Fast animal detection in UAV images using convolutional neural networks. In *Proceedings of the 2017 IEEE International Geoscience and Remote Sensing Symposium (IGARSS), Fort Worth, TX, USA, 23–28 July 2017*; IEEE: Piscataway, NJ, USA, 2017; pp. 866–869. [CrossRef]
24. Qu, T.; Zhang, Q.; Sun, S. Vehicle detection from high-resolution aerial images using spatial pyramid pooling-based deep convolutional neural networks. *Multimed. Tools Appl.* **2017**, *76*, 21651–21663. [CrossRef]
25. Kerkech, M.; Hafiane, A.; Canals, R. Vine disease detection in UAV multispectral images using optimized image registration and deep learning segmentation approach. *Comput. Electron. Agric.* **2020**, *174*, 105446. [CrossRef]
26. Wang, X.; Cheng, P.; Liu, X. Fast and accurate, convolutional neural network based approach for object detection from UAV. In *Proceedings of the IECON 2018-44th Annual Conference of the IEEE Industrial Electronics Society, Washington, DC, USA, 21–23 October 2018*; IEEE: Piscataway, NJ, USA, 2018; pp. 3171–3175. [CrossRef]
27. Lin, T.; Maire, M.; Belongie, S. Microsoft coco: Common objects in context. In *European Conference on Computer Vision*; Springer: Cham, Switzerland, 2014; pp. 740–755. [CrossRef]

28. Woo, S.; Park, J.; Lee, J. CBAM: Convolutional Block Attention Module. In *Proceedings of the European conference on computer vision (ECCV), Florence, Italy, 7–13 October 2012*; Springer International Publishing: Cham, Switzerland, 2018; pp. 3–19. [CrossRef]
29. Huang, G.; Liu, Z.; Laurens, V. Densely connected convolutional networks. In Proceedings of the IEEE conference on computer vision and pattern recognition, Honolulu, HI, USA, 21–26 July 2017; pp. 4700–4708. [CrossRef]
30. Qiao, S.; Chen, L.; Yuille, A. DetectoRS: Detecting Objects with Recursive Feature Pyramid and Switchable Atrous Convolution. *arXiv* **2020**, arXiv:2006.02334.
31. Chen, L.; Papandreou, G.; Schroff, F. Rethinking Atrous Convolution for Semantic Image Segmentation. *arXiv* **2017**, arXiv:1706.05587.
32. Rezatofighi, H.; Tsoi, N.; Gwak, J. Generalized intersection over union: A metric and a loss for bounding box regression. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Long Beach, CA, USA, 15–20 June 2019; pp. 658–666. [CrossRef]
33. Lin, T.-Y.; Goyal, P.; Girshick, R. Focal loss for dense object detection. *IEEE Trans. Pattern Anal. Mach. Intell.* **2018**, *42*, 318–327. [CrossRef] [PubMed]
34. Keshan, C.; Yu, H.; Hongbo, H. Research on detection algorithm of aeroengine installation position based on SSD model. *J. Beijing Univ. Aeronaut. Astronaut.* **2021**, *47*, 682–689. [CrossRef]