Article

Modified Deep Reinforcement Learning with Efficient Convolution Feature for Small Target Detection in VHR Remote Sensing Imagery

Shuai Liu * and Jialan Tang

School of Software Engineering, Xi'an Jiaotong University, Xi'an 710049, China; jialanT@stu.xjtu.edu.cn

* Correspondence: sh_liu@mail.xjtu.edu.cn

Abstract: Small object detection in very-high-resolution (VHR) optical remote sensing images is a fundamental but challenging problem due to the latent complexities. To tackle this problem, the MdrEcf model is proposed by modifying deep reinforcement learning (DRL) and extracting the efficient convolution feature. Firstly, an efficient attention network is constructed by introducing the local attention into the convolutional neural network. Combining the shallow low-level features with rich detail descriptions and high-level features with more semantic meanings effectively, efficient convolution features can be obtained. By this, the attention network can effectively enhance the ability to extract small target features and suppressing useless features. Secondly, the efficient feature map is sent to the region proposal network constructed by modified DRL. Using the modified reward function, this model can accumulate more rewards to conduct the search process, and potentially generate effective subsequent proposals and classification scores. It also can increase the effectiveness of object locations and classifications for small targets. Quantitative and qualitative experiments are conducted to verify the detection performance of different models. The results show that the proposed MdrEcf can effectively and accurately locate and identify related small objects.

Keywords: deep reinforcement learning; attention networks; object detection; VHR remote sensing image

Citation: Liu, S.; Tang, J. Modified Deep Reinforcement Learning with Efficient Convolution Feature for Small Target Detection in VHR Remote Sensing Imagery. *ISPRS Int. J. Geo-Inf.* **2021**, *10*, 170. <https://doi.org/10.3390/ijgi10030170>

Academic Editors: Wolfgang Kainz and Amr Abd-Elrahman

Received: 22 January 2021

Accepted: 14 March 2021

Published: 16 March 2021

Publisher's Note: MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Copyright: © 2021 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

The VHR remote sensing imagery (RSI) develops quickly due to the wide exploration of sensor technologies and aerospace research. Its typical resolution is 3–4-m ground sample distance (GSD) and the objects in VHR images are usually of diverse shapes in arbitrary orientations. With the advantages of large-scale images and multi-angle data, the VHR remote sensing images have supported an increasingly wide range of applications including resource exploration, urban planning, natural disaster assessment, military target detection, and recognition. Nowadays its application field [1,2] is still expanding. Object detection in RSI aims to determine whether a given aerial or satellite image contains one or more objects belonging to the class of interest, and determine the positions of each predicted object in RSI. Different from natural images, the objects in VHR RSI such as cars have a relatively smaller spatial extent (usually smaller than 15 pixels [3]) than the other large satellite objects. The much smaller objects [4] and complex background content [5] greatly limit its detection performance and pose some severe challenges for the applications [6,7].

In literature, various models have been proposed to effectively detect the objects of interest. Traditional methods mainly deploy the handcrafted features and shallow machine learning models, which are easy to overfit and usually require a large number of calculations. Convolutional neural networks (CNN) can automatically and powerfully learn and extract the features from data, as well as better robustness and higher detection accuracies [8–11]. They have provided great improvements and achieved much better accuracies

for object detection compared with traditional approaches [12–20]. Currently, traditional detection models have been gradually replaced by deep learning-based methods.

Considering the feature maps obtained by CNN, the deep high-level maps are of greatly lower resolutions which may harm their capacity for high-quality object localization due to the loss of detail information; the shallow low-level maps have high-resolution features that reduce the representational capacity for object recognition. Thus, most CNN-based detectors show poor performance when detecting small objects, mainly due to the coarseness of the obtained deep feature maps [21,22]. The issue of ignoring the low-level features has greatly limited the performance of CNN-based detectors. Nowadays, the attention mechanism drawing on human attention thinking has been proven to be a potential means of enhancing the network performance [23,24], which can efficiently benefit from the human brain visual mechanism by quickly filtering out the high-value knowledge from a large amount of information. Constructed by integrating the attention into the deep neural network, the attention network has shown satisfactory performance. As presented in FPN [23] and PANet [24], the performance of object detection models has been effectively enhanced by introducing attention into the CNN and generating the integrated features. Both of them inspire us that the high-level and low-level maps are complementary and the attention network can be utilized to learn complementary features for object detection [23,24]. Therefore, it is very important and natural of developing the attention network to effectively extract the object features and significantly enhance the detection performance.

Generally speaking, the object detection models need to accomplish two tasks including classification and localization. If they are not properly balanced, the performance will be suboptimal because one task may be compromised. The imbalance between classification and location becomes an increasingly important issue that limits the detection performance. Integrate deep learning's strong understanding of visual perception problems and the decision-making ability of reinforcement learning, deep reinforcement learning is becoming a promising framework for object detection with satisfying performance [25–27]. Its success can be contributed to the balance between the classification and localization for object detection. Additionally, DRL can enhance the accuracy and reduce various costs associated with the usage of VHR images at the same time. Although DRL can effectively explore the strong understanding ability of deep learning and the decision-making ability of reinforcement learning, there are still problems for the detection of remote sensing images, such as premature termination of search and sparse rewards obtained from the search.

To address the above issues, we propose a novel small object detection model for VHR remote sensing images by exploiting deep reinforcement learning and efficient convolution feature learning (MdrIEcf). First, local attention is added to CNN to construct the attention network and obtain the efficient convolution features, which integrates the low-level features of content and high-level features of semantic meanings. By effectively depicting the images, more discriminative features are generated for small targets in different positions. That is, the detailed information-rich features can be selectively enhanced by the network to improve the detection accuracy. Second, modified DRL is exploited to effectively detect the small object by designing the new reward functions. It can accumulate more rewards to guide the search process, and potentially generate effective subsequent proposals and classification scores. For VHR remote sensing images, experimental results show that the proposed MdrIEcf can effectively improve the quantitative and qualitative results of small object detection.

The rest of this paper is organized as follows. Section 2 introduces specific algorithms and frameworks related to this paper. An overview of the proposed model is presented in Section 3. Section 4 briefly introduces the experimental setup and results. Section 5 is the conclusion.

2. Related Works

Object Detection. The object detection models based on deep learning can be roughly divided into two categories: anchor-based algorithms and anchor-free algorithms. Their differences are whether the anchor points are utilized to extract the region proposals. Anchor-based algorithms include the popular two-stage detection model R-CNN [12], Fast R-CNN [13], Faster R-CNN [14], etc., and one-stage detection model YOLOv2 [15], SSD [16], etc. Integrating with region proposals, R-CNN explored a high-capacity CNN trained with bottom-up candidate boxes to locate and segment objects [12]. Follow the idea of R-CNN, Girshick et al. [13] proposed the Fast R-CNN to improve the training and testing speed while increasing the detection accuracy. Ren et al. [14] firstly introduced a region proposal network (RPN) which generated high-quality region proposals to conduct the unified network where to look. It has shown a huge improvement in the development of object detection. The one-stage detection model SSD [16] predicted category scores and box offsets for a fixed set of default bounding boxes by using small convolutional filters. Instead of the anchor point, anchor-free approaches calculate the descriptions of the bounding box in other ways, e.g., YOLO [17], CornerNet [18], ExtremeNet [19], FCOS [20]. YOLO [17] was a representative anchor-free algorithm, which directly predicted bounding boxes and class probabilities from the full images in one evaluation. Law et al. [18] regarded the bounding box as the key points (the upper left corner and lower right corner of the target) and exploited a single neural network to perform the detection. Based on it, Yi et al. [19] presented a novel object detection framework by detecting the four bottom-up extreme points of the target (that is, the uppermost point, the lowermost point, the leftmost point, the rightmost point). As an anchor-free and proposal-free algorithm, FCOS [20] was proposed to solve object detection in a per-pixel prediction fashion, analog to semantic segmentation.

Attention Networks. The attention has been proven to be a potential means to enhance the performance of deep neural networks [21,22] because it can utilize multi-level features to generate discriminative feature representations. Integrating the attention into the deep neural network, the attention network is constructed to achieve satisfactory results. FPN [23] proposed lateral connections to enhance the semantic characteristics of shallow layers via a top-down pathway. It has shown a huge improvement as the generic feature extractors. After that, PANet [24] explored a bottom-up pathway to further enhance the low-level information of deep layers. Jie et al. [28] focused on channels and explicitly modelled the interdependence between channels in CNN by the attention to enhance the network performance. Based on [28], Wang et al. [29] proposed the efficient channel attention through a fast one-dimensional convolution which involves a handful of parameters while bringing clear performance gains. Currently, the attention network has been widely utilized in various types of tasks such as natural language processing (NLP) [30], image classification [31], speech recognition [32], and facial recognition [33], which achieved remarkable results.

Deep reinforcement learning in Object Detection. With the development of deep reinforcement learning, object detection becomes a new task in this field. In [34], Bellver et al. proposed a hierarchical deep reinforcement learning object detection framework, which was characterized by a top-down exploration of a hierarchy of regions guided by an intelligent agent. Utilizing the multi-agent-based algorithm, Kong et al. [35] proposed a joint search algorithm based on collaborative deep reinforcement learning to learn the optimal strategy for target localizing. To reduce the high computational and monetary cost, a reinforcement learning agent was proposed for large images [36] by adaptively selecting the spatial resolution of each image. In [37], a novel and effective detector were proposed by integrating the bottom-up single-shot convolutional neural networks and a top-down operating strategy.

3. Methods

Exploring deep reinforcement learning and attention network, the MdrIEcf model is proposed for the small object detection in VHR optical remote sensing image. First, CNN (here we use VGG16) is utilized as the main network of the attention network for feature learning. To integrate the detail features of the shallow layer into semantic features of the deep layer, local attention is introduced into VGG16. We conduct the experiments in Section 4.2 to determine where the local attention is added. By this, the efficient convolution characteristics of small targets can be fully depicted. Then, the integrated convolution features are delivered into the modified DRL with the proposed reward function. By accumulating much more rewards during the search process, the modified DRL can potentially generate effective subsequent proposals and classification scores. That is, the modified DRL can make a trade-off between the location and classification for the small object detection to a certain degree. At last, the prediction bounding boxes and classification results are output. The overall framework of the proposed MdrIEcf model is shown in Figure 1.

Figure 1. The overall framework of the proposed MdrIEcf model, including efficient convolution feature learning and modified deep reinforcement learning. The green part represents the local attention added to VGG16 and the integrated feature map. The purple rectangular is the efficient convolution feature learning, and the orange rectangular represents the modified deep reinforcement learning (DRL). The yellow line is the modified reward functions utilized in MdrIEcf.

3.1. Efficient Convolution Feature Learning

As presented in [28], the SE module has been widely explored due to its advantages of improving network performance. Therefore, it is chosen to design the local attention of our attention network. For the SE module, the squeeze part can globally exploit contextual information, and its excitation part aims to exploit channel-wise dependencies. The purple rectangular of Figure 1 presents our attention network, where the local attention is explored

to allow VGG16 to selectively enhance low-level features with rich detail information. By this, the feature map obtained by our attention network will efficiently contain the low-level informative features and high-level semantic information simultaneously. Additionally, it has dynamic adaptability to the input and helps to enhance the general ability of the proposed model.

Let I represent an input VHR image, and the map $F(I)$ denotes the shallow features generated by the first block of VGG16. Two 3×3 convolutional layers with 64 channels and one max-pooling layer are combined to construct the first block, in which the local detail information can be well perceived by the convolutional layer.

The local attention is added between the first block and the second block according to the results in Section 4.2. The feature map $F(I)$ with the size of $W \times H \times C$ is considered as the input of the local attention. The global average pooling (GAP) is performed to well eliminate the spatial interference between features and generate a $1 \times 1 \times C$ weight w_{gap} . Here the parameter C is set to 64. Then, the full connection (FC) is utilized to incorporate learned weights and obtain the final weight w . The location attention is formulated as follows:

$$GAP(F(I)) = w_{gap}, \quad FC(w_{gap}) = w, \quad \hat{F}(I) = w \odot F(I), \quad (1)$$

where $\hat{F}(I)$ is the re-weighted feature map and $\odot$ denotes the element-wise product. The overall structure of local attention is presented in Figure 2.

Figure 2. The overall structure of local attention.

As a soft attention, the utilized local attention pays more attention to areas or channels. It means local attention has the same latent characteristics as the Modified DRL mentioned in Section 3.2, which aims to generate effective subsequent proposals and classification scores. Its parameters can be calculated by performing the gradient, forward propagation, and backward feedback algorithms.

After re-weighting, the new feature map $\hat{F}(I)$ is sent to the four remaining blocks of the attention network. In detail, the second block includes two 3×3 convolutional layers with 128 channels and 1 maxpooling layer; the third block includes three 3×3 convolutional layers with 256 channels and 1 maxpooling layer. The fourth and last blocks include three 3×3 convolutional layers with 512 channels and 1 maxpooling layer respectively. Utilizing these blocks, the low-level informative features are selectively combined and enhanced. Then, it is integrated with the deep features, which are usually obtained by the fully connected layer at the end of the network. Finally, the attention network outputs the efficient convolution feature map $\hat{F}(I)$.

3.2. Deep Reinforcement Learning with the Modified Reward Function

Exploring deep reinforcement learning is a new research direction for solving object detection problems. When performing DRL, the agent receives the input data from its environment and estimates how good or bad the taken actions according to a reward function.

Normally, the reward function is utilized to assigns a numerical value to each performed action from a given state and then taken actions aim to achieve the predetermined goal. In addition, the agent reaches the new state after performing an action. The framework of DRL in the proposed method is displayed in the orange rectangular of Figure 1.

Let S represent the state space, A is the action space and R is the reward. There are two types of actions (fixate action a_t^f and done action a_t^d) in A corresponding to the two rewards (fixate reward r_t^f and done reward r_t^d) in R . The feature map $\dot{F}(I)$ is input into the modified DRL and forms an initial state $s_t(t = 0)$. Then in each time slot t , the agent selects the best action to output by utilizing the policy $\pi(a_t|s_t)$, which is a given random strategy and represents the maps from states to actions in the policy center. The $\pi(a_t|s_t)$ is formulated as follows:

$$\begin{cases} \pi(a_t = a_t^f | s_t) = P(z_t)[1 - \sigma(w_s d_t)] \\ \pi(a_t = a_t^d | s_t) = \sigma(w_s d_t) \end{cases}, \quad (2)$$

where $P(z_t)$ represents the probability map of the new position z_t ; $\sigma(\bullet)$ is the logistic sigmoid function; w_s presents the trainable weight vector, and $d_t \in R^{625}$ is a vector generated by the first done action.

When the agent chooses the fixate action ($a_t^f = z_t$ & $a_t^d = 0$), the new location z_t is visited and the fixate reward r_t^f is obtained. Meanwhile, region of interests (RoIs) are updated with the areas centered at z_t . Let IoU_t^i represent the intersection-over-union (IoU) of the ground truth instance g_i given by the RoIs of the specific time slot t ; IoU^i is defined as the maximum IoU between the predicted bounding box and the ground truth bounding box for the i -th instance in the time slot $0 \dots t-1$. The modified fixate reward r_t^f in time slot t is formulated as

$$r_t^f = -\beta + \sum_i \frac{IoU_t^i - IoU^i}{\tau}, \quad g_i : IoU_t^i > IoU^i \geq \tau, \quad (3)$$

where $-\beta$ (set to 0.075 [26]) is a small negative reward, and τ represents the IoU threshold.

The fixate reward r_t^f reflects the quality of the selected location z_t . After the fixate reward r_t^f is obtained, all the corresponding RoIs will be sent to the RoI pooling module, and then do the classification and bounding box offset prediction of the specific class. The predictions are mapped to certain locations and are added to the history of a specific class h_t . The r_t^f and the h_t will be combined with the original state s_t to form a new state s_{t+1} . At the time $t+1$, the agent redetermines whether taking the new actions or not in the policy center according to the new state s_{t+1} .

If the agent decides to perform the done action ($a_t^d = 1$), it will stop searching the feature map $\dot{F}(I)$ and collect all selected predictions in the entire trajectory for prediction and classification. Also, the agent gains a done reward r_t^d reflecting the quality of the search, which can be utilized to guide the next search process. The modified done reward r_t^d is formulated as

$$r_t^d = \sum_i \frac{IoU^i - \tau}{\tau}, \quad g_i : IoU^i \geq \tau. \quad (4)$$

According to the rewards obtained during the searching process, the agent can effectively optimize the searching process of the next feature map. At last, the prediction and classification results will be output. The overall pseudo code of the modified deep reinforcement learning is shown in Algorithm 1.

Algorithm 1 Modified deep reinforcement learning**Input:** feature map $\dot{F}(I)$ **Output:** the final classification results y

```

1:   initialize the state space and give the agent an initial state  $s_0$ 
2:   for time slot  $t$ :
3:     derive actions based on strategy  $\pi(a_t|s_t)$ 
4:     IF  $a_t^f = z_t$  &&  $a_t^d = 0$ :
5:       visit new location  $z_t$  and update RoIs
6:       calculate the maximum IoU between each object instance and the ground truth at time slot  $0 \dots t-1$ , denoted as  $IoU^i$ 
7:       in each time slot  $t$ , calculate the IoU between the RoIs and the object instance, denoted as  $IoU_t^i$ 
8:       IF  $IoU_t^i > IoU^i \geq \tau$ 
9:         calculate  $r_t^f$  based on (3), let  $IoU^i = IoU_t^i$ 
          do the classification and bounding box offset prediction of the specific class
          insert the prediction results into the history of specific class  $h_t$ 
          combine  $h_t$  and  $r_t^f$  with  $s_t$  to form a new state  $s_{t+1}$ 
10:      Jump to Step 2
11:     ELSE IF done action is chosen ( $a_t^d = 1$ )
12:       calculate  $IoU^i$  and  $r_t^d$  based on (4)
13:       generate the region proposals
14:     Stop the agent
15:   calculate and draw the prediction boxes based on the regional proposals; obtain the classification result  $y$ .
```

4. Experiments

In this section, we first describe the datasets, comparison methods, experiment settings, and evaluation metrics. Then, we compare and analyze the results obtained by the proposed MdrIEcf and six compared approaches on the experimental datasets.

4.1. Experimental Setup

(1) Datasets: To verify the performance and effectiveness of the proposed model, experiments are carried out on three public VHR datasets. The details are listed in Table 1.

Table 1. The experimental datasets.

Dataset	Number of Categories	Number of Pictures	Supplements
NWPU VHR-10 [38]	10	800	A 10-level geographic remote sensing dataset for space object detection.
SAR-Ship-Dataset [39]	1	43,819	This dataset labeled by SAR experts was created using 102 Chinese Gaofen-3 images and 108 Sentinel-1 images.
RSOD [40]	4	976	Spatial resolution from 0.3 m to 3 m, and contains 6950 instances

(2) Comparison Methods: Different state-of-the-art detection approaches are chosen to compare with the proposed MdrIEcf, including RICNN [2], Faster R-CNN [14], DRL-Fr [26], MDRL, SSD [16], and YOLO [17]. MDRL is proposed as an ablation experiment method to verify the effectiveness of convolution feature learning described in Section 3.1, which is a DRL model with the modified reward functions only. Comparing the DRL-Fr and MDRL, the necessary and effectiveness of the modified reward functions can be verified; comparing MDRL and MdrIEcf, those of the proposed efficient convolution feature learning can be evaluated.

(3) Experimental Settings: We retrained the approaches (DRL-Fr [26], MDRL, MdrIEcf) using the datasets in Section 4.1. To be fair, all the ablation experiments choose VGG16 as the backbone and all hyperparameters are consistent. In the experiment, the parameter settings (DRL-Fr [26], MDRL, MdrIEcf) are presented as follows. The number of iterations is 110,000, and the batch size is 256. The learning rate is 0.00025, which is automatically adjusted every 80,000 times to decrease it by 0.1 times. The input size of the image is 600 on the smallest side and 1000 on the longest side. IoU threshold is set to 0.45. As for other methods (Faster R-CNN [14], RICNN [2], SSD [16], and YOLO [17]), their pretrained weights and the related setting are utilized to conduct our experiments.

The conditions of the experiments are presented as follows. TensorFlow and python are utilized to build the target detection environment. The experimental environment is Linux, and the platform is equipped with an 8GB GPU (Tesla P100), 14-core CPU (Intel(R) Xeon(R) Gold 5117 CPU @ 2.00 GHz). GPU and CPU are used for joint training.

(4) Evaluation Metrics: The Average Precision (AP) and mean Average Precision (mAP) are utilized as the evaluation indicators. AP is utilized to measure the performance of the detector in each category; mAP is utilized to estimate the detector performance in all categories. The higher the values of AP and Map, the better the detection performance. Let $p(r)$ represents the P-R curve; N is the number of target types in the test set. Both of them can be formulated as follows:

$$AP = \int_0^1 p(r) dr, \quad (5)$$

$$mAP = \frac{\sum_{n=1}^N AP(n)}{N}. \quad (6)$$

4.2. Experimental Results and Analysis

4.2.1. Estimating Locations of the Added Local Attention

The quantitative experiments are conducted to determine where to add the local attention module. We choose the SE module as a comparison and utilize the VGG16 as the network. The SE module and local attention share the same experimental settings. The following descriptions of the SE module can also apply to the local attention, just replace the SE module with the local attention.

First, the SE module (or local attention) is added for training only after the first block. Then the test results and output matrix table can be calculated. After that, we increase the number of SE modules, such as adding SE modules after the first and second blocks, adding SE modules after the first three blocks, adding SE modules after the second and third blocks, and adding SE modules after all five blocks. In addition, the scaling parameters are set to 16, 32, and 64 for debugging. The experiments for local attention are basically the same.

Table 2 presents the detection results of different approaches on the NWPU VHR-10 dataset. In Table 2, “Add layer” indicates the position where the SE module (or local attention) is added, and the number indicates the detailed block of VGG16. The values of mAP are very different when the adding locations change. From Table 2, it can be easily observed that the best result is obtained when the local attention is added after the first block, which can show the effectiveness of integrating the low-level detailed characteristics into deep features to a certain degree.

Based on the above results, local attention is added after the first block of VGG16 for the proposed approach. The final structure of efficient convolution feature learning is displayed in Figure 3.

Table 2. Experiments on the locations of the added SE module and local attention.

Module	Add Layer	Scaling Parameters	mAP
local attention	1	×	0.834
SE module	1,2	16	0.670
SE module	1,2	32	0.780
SE module	1,2	64	0.748
local attention	1,2	×	0.810
SE module	1,2,3	16	0.608
SE module	1,2,3	32	0.663
local attention	1,2,3	×	0.742
SE module	2,3	16	0.588
local attention	2,3	×	0.809
SE module	1,2,3,4,5	16	0.527

Figure 3. The final structure of the efficient convolution feature learning.

4.2.2. Quantitative Analysis

RICNN [2], Faster R-CNN [14], DRL-Fr [26], SSD [16], YOLO [17]), MDRL, and MdrIEcf are performed on three popular datasets, and the results are shown in Tables 3–5. All the bold numbers in the tables indicate the best results.

Table 3. Comparison of detection accuracy of different models on the NWPU VHR-10 dataset.

	RICNN	Faster R-CNN	DRL-Fr	MDRL	MdrIEcf	SSD	YOLO
airplane	0.884	0.903	0.907	0.905	0.927	0.932	0.874
ship	0.773	0.800	0.814	0.853	0.844	0.857	0.847
storage tank	0.853	0.711	0.629	0.631	0.775	0.617	0.658
baseball diamond	0.881	0.894	0.897	0.872	0.922	0.904	0.931
tennis court	0.408	0.815	0.813	0.818	0.838	0.850	0.658
baseball court	0.585	0.808	0.816	0.823	0.824	0.870	0.872
ground track field	0.867	0.909	0.995	0.999	0.988	0.985	0.976
harbor	0.686	0.798	0.786	0.797	0.789	0.812	0.832
bridge	0.615	0.706	0.706	0.734	0.722	0.731	0.793
vehicle	0.711	0.714	0.704	0.715	0.715	0.762	0.742
mAP	0.726	0.806	0.807	0.814	0.834	0.832	0.818
Training Time(s)	8.77	0.242	0.267	0.271	0.264	0.186	0.182

Table 4. Comparison of detection accuracy of different models on the SAR-Ship-Dataset dataset.

	RICNN	Faster R-CNN	DRL-Fr	MDRL	MdrIEcf	SSD	YOLO
ship	0.803	0.907	0.900	0.901	0.917	0.901	0.746
mAP	0.803	0.907	0.900	0.901	0.917	0.901	0.746
Training Time(s)	7.13	0.213	0.225	0.214	0.198	0.189	0.196

Table 5. Comparison of detection accuracy of different models on the RSOD dataset.

	RICNN	Faster R-CNN	DRL-Fr	MDRL	MdrIEcf	SSD	YOLO
oiltank	0.721	0.901	0.901	0.906	0.909	0.692	0.743
playground	0.678	0.885	0.885	0.884	0.889	0.712	0.738
aircraft	0.763	0.804	0.805	0.806	0.810	0.702	0.751
overpass	0.844	0.732	0.694	0.692	0.737	0.823	0.851
mAP	0.751	0.831	0.821	0.822	0.836	0.732	0.771
Training Time(s)	8.25	0.236	0.241	0.231	0.227	0.214	0.217

Table 3 presents the comparisons of detection accuracies on the NWPU VHR-10 dataset. It can be easily found that the mAP of MdrIEcf is higher than the other compared methods, which reaches 83.4% as shown in Table 3. Analyzing the AP values of each category, we found that SSD and YOLO show better results in some categories, but their overall evaluation is not as good as MdrIEcf. In the ablation experiment, MDRL presents better values of mAP and training time than DRL-Fr. Additionally, the mAP values and the training time of MdrIEcf are much better than those of MDRL, and most AP values of MdrIEcf are superior to those of MDRL.

For the SAR-ship-dataset presented in Table 4, MdrIEcf achieves the best mAP (91.7%). This dataset has only one category of the ship, and the ratio of the ship's length or width to the image size is in the range of 0.04 to 0.24, which is much smaller than the PASCAL VOC's 0.2 to 0.9. From Table 4, the superior values of MdrIEcf fully show its effectiveness in detecting small targets.

For the RSOD dataset presented in Table 5, our method shows obvious improvements in detecting the small and medium-sized targets (like oil tanks, playgrounds, and aircraft). For the overpass, it only has shape information compared to other classes and YOLO achieves the best mAP (85.1%). This is because overpass belongs to large-scale objects in remote sensing images, and the standard YOLO is usually good for detecting large-scale objects. For the overpass, the AP obtained by MdrIEcf increases by almost 4% compared with those of DRL-Fr and MDRL, which greatly demonstrates that the added local attention and the modified reward function can well enhance the detection performance.

From Tables 3–5, although the proposed MdrIEcf does not show the superior AP values in some categories, its overall performance is better than other compared methods, which means MdrIEcf can stably detect the small targets for VHR remote sensing datasets. In ablation experiments, MdrIEcf has superior values of mAP and takes much less training time, which verifies the effectiveness of the proposed modules in Section 3. Additionally, the proposed MdrIEcf takes a longer time to train compared with SSD or YOLO, and its training time grows with the increasing number of images and IoUs. In the future, we will try to reduce the training time.

4.3. Visualization Results and Analysis

The visualization results of different VHR datasets are shown in Figures 4–7. The red or white rectangular in Figures 4–7 represent the predicted bounding box, and the number in each blue label represents the value of IoU.

Figure 4 shows a few sample detection results of the proposed MdrIEcf on the NWPU VHR-10 dataset. It can be clearly observed that the proposed MdrIEcf can accurately detect the objects that belong to different classes with small or medium size, such as airplanes, vehicles, ships, and playgrounds. Additionally, the objects that densely stand together are also clearly detected.

SAR-ship-Dataset is a two-dimensional ship dataset, which contains a large amount of small objects. As shown in Figure 5, the proposed MdrIEcf can effectively detect ships with different sizes and angles; however, some ships fail to be detected in Figure 5. This can be attributed to the fact that similar backscattering mechanisms are shared by the targets in different backgrounds such as buildings, harbors, and islands. Therefore, there is still a lot of room for improving the proposed method.

Figure 4. The results of some detected objects from NWPU VHR10.

Figure 5. The results of some detected objects from SAR-ship-Dataset.

Figure 6. The results of some detected objects from RSOD.

Figure 7. Evaluation of the accuracies of prediction location results (IoU) of different methods: (a) Faster R-CNN; (b) DRL-Fr; (c) MDRL; (d) MdrlEcf.

The detection results of the RSOD dataset are shown in Figure 6. Obviously, the proposed MdrIEcf not only can accurately detect small and medium-sized targets (aircrafts, oil tanks, and playgrounds) but can also detect large-sized objects that overpass complex backgrounds correctly. Noticing that, the targets of RSOD are much smaller than those of the NWPU VHR-10 datasets. This fact further demonstrates the detection effectiveness of MdrIEcf for detecting small objects.

To compare the effectiveness of locating the objects, we design an experiment to evaluate the accuracies of prediction bounding boxes obtained by Faster R-CNN, DRL-Fr, MDRL, and MdrIEcf. Figure 7 presents the numerical results of the IoU. Compared to the numbers in the blue labels, it can be clearly seen that the average IoU results of MdrIEcf are better than the other Faster R-CNN, DRL-Fr, MDRL, and the related prediction bounding boxes are more accurate. In Figure 7, the Faster R-CNN, DRL-Fr, and MDRL models are not well fitted, so multiple prediction bounding boxes appear in the results. To further evaluate the accuracies of MdrIEcf in locating the objects, Figure 8 shows the comparisons between the prediction bounding boxes and the related ground truth, where the red boxes represent the prediction bounding boxes and the green boxes are the ground truth. Comparing the bounding boxes and the values of IOU, it can be observed that the prediction bounding boxes of MdrIEcf can well match those of the ground truth, which can be also found by the values of IOU in the blue labels. From the visualization analysis in Figures 4–8, it can further prove that the proposed method presents superior effectiveness and robustness to other compared methods.

Figure 8. Evaluation of the accuracies of MdrIEcf in locating the objects.

5. Conclusions

By modifying deep reinforcement learning and enhancing convolution features, the MdrIEcf model is proposed for small object detection of VHR remote sensing images. Using the local attention mechanism and CNN, the attention network is constructed to integrate low-level features into high-level features. The complementary feature map can also be obtained, which can effectively extract the remarkable characteristics of small targets. After that, the DRL with an improved reward function is explored to utilize the feature map and achieve small object detection. Using the redesigned reward function, the modified DRL can greatly increase the searching rewards, and efficiently guide the agent to search the informative features. After that, the small objects can be well located and classified. Three popular VHR remote sensing images are applied to evaluate the performance of MdrIEcf and six compared models, whose categories are quite different. The experimental results efficiently verify the effectiveness and feasibility of the proposed MdrIEcf and suggest that MdrIEcf can achieve better results in quality and quantity.

There is no doubt that it still has room to improve the proposed approach. Based on the deep reinforcement learning, MdrIEcf takes a longer time in training. For further research, we will try to improve the speed of predicting the bounding boxes in deep reinforcement

learning and we are also interested in designing the lightweight detector and transplanting it to portable hardware.

Author Contributions: Conceptualization, Shuai Liu; formal analysis, Shuai Liu; methodology, Shuai Liu; software, Jialan Tang; validation, Jialan Tang; writing—original draft, Jialan Tang. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by the National Natural Science Foundation of China (No. 61703328), the China Postdoctoral Science Foundation-funded project (No. 2018M631165), Shaanxi Province Postdoctoral Science Foundation (No. 2018BSHYDZZ23), and the Fundamental Research Funds for the Central Universities (No. XJJ2018253).

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The datasets presented in this study are available through: https://github.com/chaozhong2010/VHR-10_dataset_coco (accessed on 16 March 2021), <https://github.com/CAESAR-Radi/SAR-Ship-Dataset> (accessed on 16 March 2021), <https://github.com/RSIA-LIESMARS-WHU/RSOD-Dataset-> (accessed on 16 March 2021).

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

References

- Sharma, V.; Mir, R.N. A comprehensive and systematic look up into deep learning based object detection techniques: A review. *Comput. Sci. Rev.* **2020**, *38*, 100301. [\[CrossRef\]](#)
- Cheng, G.; Zhou, P.C.; Han, J.W. Leaning rotation-invariant convolutional neural networks for object detection in VHR optical remote sensing images. *IEEE Trans. Geosci. Remote Sens.* **2016**, *54*, 7405–7415. [\[CrossRef\]](#)
- 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. [\[CrossRef\]](#)
- Yang, S.; Tian, L.; Zhou, B.; Chen, D.; Zhang, D.; Xu, Z.; Liu, J. Inception Parallel Attention Network for Small Object Detection in Remote Sensing Images. In Proceedings of the 3rd Chinese Conference on Pattern Recognition and Computer Vision (PRCV), Nanjing, China, 16–18 October 2020; pp. 469–480.
- Bosquet, B.; Mucientes, M.; Brea, V.M. STDnet: Exploiting high resolution feature maps for small object detection. *Eng. Appl. Artif. Intell.* **2020**, *91*, 103615. [\[CrossRef\]](#)
- Deng, Z.; Sun, H.; Zhou, S.; Zhao, J.; Lei, L.; Zou, H. Multi-scale object detection in remote sensing imagery with convolutional neural networks. *ISPRS J. Photogramm. Remote Sens.* **2018**, *145*, 3–22. [\[CrossRef\]](#)
- Sun, C.; Ai, Y.; Wang, S.; Zhang, W. Mask-guided SSD for small-object detection. *Appl. Intell.* **2020**, 1–12. [\[CrossRef\]](#)
- Agarwal, S.; Terrail, J.O.D.; Jurie, F. Recent advances in object detection in the age of deep convolutional neural networks. *arXiv* **2018**, arXiv:1809.03193.
- Liu, L.; Yang, W.O.; Wang, X. Deep Learning for Generic Object Detection: A Survey. *Int. J. Comput.* **2020**, *128*, 261–318. [\[CrossRef\]](#)
- Zou, Z.; Shi, Z. Random Access Memories: A New Paradigm for Target Detection in High Resolution Aerial Remote Sensing Images. *IEEE Trans. Image Process.* **2018**, *27*, 1100–1111. [\[CrossRef\]](#) [\[PubMed\]](#)
- Zhu, X.X.; Tuia, D.; Mou, L.; Xia, G.S.; Zhang, L.; Xu, F.; Fraundorfer, F. Deep Learning in Remote Sensing: A Comprehensive Review and List of Resources. *Remote Sens.* **2017**, *5*, 8–36. [\[CrossRef\]](#)
- Girshick, R.; Donahue, J.; Darrell, T.; Malik, J. Rich feature hierarchies for accurate object detection and semantic segmentation. In Proceedings of the 2014 IEEE Conference on Computer Vision and Pattern Recognition, Columbus, OH, USA, 23–28 June 2014; pp. 580–587.
- Girshick, R. Fast R-CNN. In Proceedings of the 2015 IEEE International Conference on Computer Vision, Santiago, MN, USA, 7–13 December 2015; pp. 1440–1448.
- Ren, S.; He, K.; Girshick, R.; Sun, J. Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks. *IEEE Trans. Image Process.* **2017**, *39*, 1137–1149. [\[CrossRef\]](#) [\[PubMed\]](#)
- Redmon, J.; Farhadi, A. YOLO9000: Better, Faster, Stronger. In Proceedings of the 2017 IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July 2017; pp. 6517–6525.
- Liu, W.; Anguelov, D.; Erhan, D.; Szegedy, C.; Reed, S.E.; Fu, C.-Y.; Berg, A.C. SSD: Single Shot MultiBox Detector. In Proceedings of the European Conference on Computer Vision, Amsterdam, The Netherlands, 8–16 October 2016; pp. 21–37.
- Redmon, J.; Divvala, S.; Girshick, R.; Farhadi, A. You only look once: Unified, real-time object detection. In Proceedings of the 2016 IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 27–30 June 2016; pp. 779–788.
- Law, H.; Deng, J. CornerNet: Detecting Objects as Paired Keypoints. In Proceedings of the European Conference on Computer Vision, Munich, Germany, 8–14 September 2018; pp. 734–750.

19. Yi, Z.X.; Cheng, Z.J.; Philipp, K. Bottom-up Object Detection by Grouping Extreme and Center Points. In Proceedings of the 2019 IEEE Conference on Computer Vision and Pattern Recognition, Long Beach, CA, USA, 15–20 June 2019; pp. 850–859.
20. Zhi, T.; Chunhua, S.; Hao, C.; Tong, H. FCOS: Fully Convolutional One-Stage Object Detection. In Proceedings of the 2019 IEEE International Conference on Computer Vision, Seoul, Korea, 27 October–2 November 2019; pp. 9626–9635.
21. Pang, J.; Chen, K.; Shi, J.; Feng, H.; Ouyang, W.; Lin, D. Libra R-CNN: Towards balanced learning for object detection. In Proceedings of the 2019 IEEE Conference on Computer Vision and Pattern Recognition, Long Beach, CA, USA, 15–20 June 2019; pp. 821–830.
22. Tong, K.; Wu, Y.; Zhou, F. Recent advances in small object detection based on deep learning: A review. *Image Vis. Comput.* **2020**, *97*, 103910. [[CrossRef](#)]
23. Lin, T.Y.; Dollár, P.; Girshick, R.; He, K.; Hariharan, B.; Belongie, S. Feature pyramid networks for object detection. In Proceedings of the 2017 IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July 2017; pp. 936–944.
24. Liu, S.; Qi, L.; Qin, H.; Shi, J.; Jia, J. Path aggregation network for instance segmentation. In Proceedings of the 2018 IEEE Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–23 June 2018; pp. 8759–8768.
25. Caicedo, J.C.; Lazebnik, S. Active object localization with deep reinforcement learning. In Proceedings of the IEEE International Conference on Computer Vision, Las Condes, Chile, 11–18 December 2015; pp. 2488–2496.
26. Pirinen, A.; Sminchisescu, C. Deep Reinforcement Learning of Region Proposal Networks for Object Detection. In Proceedings of the 2018 IEEE Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–23 June 2018; pp. 6945–6954.
27. Al, W.A.; Yun, I.D. Partial policy-based reinforcement learning for anatomical landmark localization in 3d medical images. *IEEE Trans. Med. Imaging* **2019**, *39*, 1245–1255. [[CrossRef](#)] [[PubMed](#)]
28. Jie, H.; Shen, L.; Sun, G. Squeeze-and-Excitation Networks. In Proceedings of the 2018 IEEE Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–23 June 2018; pp. 7132–7141.
29. Wang, Q.; Wu, B.; Zhu, P.; Li, P.; Zuo, W.; Hu, Q. ECA-net: Efficient channel attention for deep convolutional neural networks. In Proceedings of the 2020 IEEE Conference on Computer Vision and Pattern Recognition, Seattle, WA, USA, 13–19 June 2020; pp. 11534–11542.
30. Hu, D. An Introductory Survey on Attention Mechanisms in NLP Problems. *arXiv* **2018**, arXiv:1811.05544.
31. Mnih, V.; Heess, N.; Graves, A.; Kavukcuoglu, K. Recurrent Models of Visual Attention. In Proceedings of the 27th International Conference on Neural Information Processing Systems, Montreal, QC, Canada, 8–13 December 2014; pp. 2204–2212.
32. Shan, C.; Zhang, J.; Wang, Y.; Xie, L. Attention-Based End-to-End Speech Recognition on Voice Search. In Proceedings of the 2018 IEEE International Conference on Acoustics, Speech and Signal Processing, Calgary, AB, Canada, 15–20 April 2018; pp. 4764–4768.
33. Li, Y.; Bei, Z.J.; Shan, S.; Chen, X. Occlusion Aware Facial Expression Recognition Using CNN with Attention Mechanism. *IEEE Trans. Image Process* **2019**, *28*, 2439–2450. [[CrossRef](#)]
34. Bellver, M.; Giro-i-Nieto, X.; Marques, F.; Torres, J. Hierarchical object detection with deep reinforcement learning. In Proceedings of the Conference on Neural Information Processing Systems, Barcelona, Spain, 5–20 December 2016.
35. Kong, X.; Xin, B.; Wang, Y.; Hua, G. Collaborative deep reinforcement learning for joint object search. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, San Francisco, CA, USA, 21–26 July 2017; pp. 7072–7081.
36. Uzket, B.; Yeh, C.; Ermon, S. Efficient object detection in large images using deep reinforcement learning. In Proceedings of the 2020 IEEE Winter Conference on Applications of Computer Vision, Snowmass Village, CO, USA, 1–5 March 2020; pp. 1824–1833.
37. Liu, S.; Huang, D.; Wang, Y. Pay Attention to Them: Deep Reinforcement Learning-Based Cascade Object Detection. *IEEE Trans. Neural Netw. Learn. Syst.* **2020**, *31*, 2544–2556. [[CrossRef](#)] [[PubMed](#)]
38. Yao, Q.; Hu, X.; Lei, H. Multiscale Convolutional Neural Networks for Geospatial Object Detection in VHR Satellite Images. *IEEE Geosci. Remote Sens. Lett.* **2021**, *18*, 23–27. [[CrossRef](#)]
39. Wang, Y.; Wang, C.; Zhang, H.; Dong, Y.; Wei, S. A SAR Dataset of Ship Detection for Deep Learning under Complex Backgrounds. *Remote Sens.* **2019**, *11*, 765. [[CrossRef](#)]
40. 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](#)]