Detection of ship targets in photoelectric images based on an improved recurrent attention convolutional neural network

Zhijing Xu¹, Yuhao Huo¹, Kun Liu¹ and Sidong Liu²

Abstract
Deep learning algorithms have been increasingly used in ship image detection and classification. To improve the ship detection and classification in photoelectric images, an improved recurrent attention convolutional neural network is proposed. The proposed network has a multi-scale architecture and consists of three cascading sub-networks, each with a VGG19 network for image feature extraction and an attention proposal network for locating feature area. A scale-dependent pooling algorithm is designed to select an appropriate convolution in the VGG19 network for classification, and a multi-feature mechanism is introduced in attention proposal network to describe the feature regions. The VGG19 and attention proposal network are cross-trained to accelerate convergence and to improve detection accuracy. The proposed method is trained and validated on a self-built ship database and effectively improve the detection accuracy to 86.7% outperforming the baseline VGG19 and recurrent attention convolutional neural network methods.

Keywords
Ship detection, fine-grained image classification, recurrent attention convolutional neural network, scale-dependent pooling, cross-training

Date received: 7 June 2019; accepted: 24 February 2020

Handling Editor: Janez Perš
Recently, deep learning algorithms have been increasingly used to improve the ship image detection and classification accuracy. Most of these methods use convolutional neural networks (CNNs) to extract the image features, to locate the target position, and to identify the types of ships. These methods can be divided into two categories: strong supervision and weak supervision. In the strong supervision mode, the model not only uses image labels, but also relies on the target bounding boxes and component points to assist feature learning. Z Ning et al. proposed a part R-CNN algorithm, which locates targets through a fully connected network (FCN) and divides them into two masks for joint discrimination. In the weak supervision mode, the classification model relies on image labels alone, possible with a compromised performance compared with strong supervised learning. However, no bounding box of the target is needed, and thus it is feasible to collect large-scale training data and easier to train the network. Simon and Rodner proposed the weak supervision convolutional neural network (DCNN), which uses smaller convolution and pooling kernels, making it easier to adjust and apply.

In 2017, Fu et al. proposed a recurrent attention CNN to solve the classification problem of fine-grained image databases. The network achieves the optimal level on three mainstream fine-grained image sets: CUB Birds, Stanford Dogs, and Stanford Cars, achieving 3.3%, 3.7%, and 3.8% improvement in accuracy, respectively. However, the network cannot make good use of the global information of the image, and the shape of the feature area is always the same. Therefore, in this article, an improved recurrent attention convolutional neural network (RA-CNN) is proposed to detect the photoelectric ship targets. The scale-dependent pooling (SDP) and multi-feature attention proposal network (MF-APN) algorithms are used to improve the network’s recognition accuracy for fine-grained images. At the same time, the use of the MF-APN algorithm will increase the amount of network calculations and reduce the real-time performance of detection.

### Related works

#### CNN development

CNN has been the cornerstone in the deep learning field. Through a series of convolution and pooling operations in CNN, the network can learn to generate the target location box and to classify the target category given a training dataset. In 1998, The LeNet-5 model was proposed by LeCun et al., which established the modern structure of CNN. In 2006, Hinton and Salakhutdinov proposed a multi-hidden-layer neural network demonstrating better feature learning ability, and its training complexity can be effectively alleviated by initializing each layer. In 2012, AlexNet won the ImageNet competition, proving the practicability of deeper CNNs. Nowadays, a variety of complex CNN-based networks are designed to improve the detection accuracy and shorten the detection time, such as faster R-CNN, Yolo, and SSD (single-shot detector). The VGG19 network is a deep convolutional neural network (DCNN), which uses smaller convolution and pooling kernels, making it easier to adjust and apply.

#### Ship detection

Due to the development of CNN, many CNN-based ship identification methods have emerged in recent years. In 2017, Kang et al. proposed a contextual region-based CNN, which can effectively detect ship targets in synthetic aperture radar (SAR) images. In 2018, Zhao et al. proposed a coupled CNN network for small and dense ship identification, which demonstrates the ability of CNN to detect surface targets. In the same year, Oliveau and Sahbi proposed semi-supervised deep attribute networks, which enhanced the recognition effect of CNN on fine-grained images through shallow and deep attributes. In 2019, Zhang et al. improved faster R-CNN, reduced the impact of environmental factors on target detection, and achieved good results in real-time surface target detection. In the same year, Lin et al. proposed the squeeze and excitation mechanism to further improve the detection capability of faster R-CNN for ships in SAR images.

#### RA-CNN

RA-CNN consists of three cascading scale sub-networks with the same network structure, but different and independent network parameters. Each scale sub-network has two different kinds of networks, namely, a VGG19 network for classification and an attention proposal network (APN) for target location. In each scale level, the image features are extracted from the input image and classified by VGG19; then the APN locates the feature area based on the extracted features and further cut out and enlarge the feature area as the input to the next scale network. By fusing three scales’ outputs, RA-CNN can determine the category of the fine-grained image.
The advantage of RA-CNN is that it can gradually focus on the feature area in the learning process through its multi-scale architecture. In addition, there are two loss functions designed in the RA-CNN, namely, classification loss and inter-scale loss. Through cross-training, the classification and target localization networks promote each other, leading to a quicker convergence of the model weights.

**SDP**

In 2016, F Yang et al. proposed using the SDP algorithm in their paper to improve the accuracy of object detection based on CNN. The output of the last pooling layer of CNN is usually used to determine the type of objects in the image. However, this may lead to a problem that if the target feature area is too small, multiple convolutions may diminish the discriminating power of the features and affect the final classification. The SDP algorithm extracts features from different CNN convolutional layers according to the size of candidate regions. For small candidate regions, the features are selected from the front convolutional layers, and, for large candidate regions, the features are selected from the back layers.

**Method**

Labeling fine-grained images needs domain knowledge and careful evaluation because of their high similarity, which can be very time-consuming and labor intensive. RA-CNN, through weak supervised learning, only needs the category label of the image to classify the class and does not require the corresponding bounding box information to locate the feature area. However, the shape of the feature area generated by APN in RA-CNN is fixed and thus not conducive to the learning of various special geometric appearance feature areas. In other words, APN ignores the direct connection between the feature area and VGG19 after location. Therefore, in this study, we use the SDP and multi-feature area joint discrimination method to improve the model.

**Preprocessing**

To standardize the input images to feed in the VGG-SDP network, we adjusted the train and test images to the $224 \times 224$ specification to meet the input requirement of the network. The size of the image should not be less than $80 \times 80$; otherwise, the network detection effect will decrease. Each image is an RGB three-channel map with pixel value ranging from 0 to 255. To reduce the problem of training non-convergence caused by the singular pixel value and to speed up the convergence of the initial iteration of training, it is necessary to assign the pixel values to $[-1, 1]$, as shown in equation (1)

$$x'_{i,j} = \frac{x_{i,j}}{255} \times 2 - 1$$  \hspace{1cm} (1)

where $i, j \in [0, 223]$, and $x_{i,j}$ and $x'_{i,j}$ represents the unmodified and modified pixel values, respectively.

**Network architecture**

The proposed network is designed to detect and classify ship targets using the preprocessed photoelectric images as the input. In the results, we render the target in a square box and display the ship category. The ship target recognition flowchart is shown in Figure 1:

![Ship target recognition flowchart](image)

The improved RA-CNN network still adopts the original three-scale architecture. We replace the target classification network VGG19 in each scale layer with a VGG-SDP network. Then, the feature location network APN is replaced with an MF-APN network. A generalized overview of the proposed network is shown in Figure 2. The process of feature extraction and target location is detailed as follows:

1. Feed the input image to the classification network $v_1$ in the first scale layer for feature extraction.
2. Send the output of the fifth pooling layer $P_5$ in the network $v_1$ to the location network $m_1$, and obtain the target position and size $N$.
3. Choose the most appropriate maxpooling layer’s output in $v_1$ according to the target size $N$; after the full connection operation, the softmax function is used to obtain the prediction label $Y^{(1)}$ of the first scale layer.
4. Subject to the target position and size $N$ in Step 2, the network $m_1$ will extract $M$ feature maps (in this article, $M$ is set to 3) and enlarge them to $224 \times 224$ as the input of the next scale layer.
5. Send $M$ feature maps to the classification network $v_2$ in the second scale layer for deeper feature extraction, and Steps 2 and 3 are repeated for each feature map; then, we converge the prediction probabilities of the $M$ feature maps into the prediction labels $Y^{(2)}$ of the second scale layer. The location network in the second scale...
layer only uses the normal APN, so we still have $M$ feature maps generated into the third scale layer.

6. Repeat Steps 2 and 3 to fuse the prediction probabilities of the feature maps into the prediction label $Y^{(3)}$ of the third scale layer.

7. The final target is positioned as a square feature box in the first scale layer, and the target category is the fusion of the predicted labels of the three scale layers. The specific flowchart is shown in Figure 2.

**VGG-SDP classification**

Due to the robustness and superior feature extraction capability, VGG19 is used as the basic classification network in RA-CNN, which consists of 16 convolutional layers, 5 maxpooling layers, and 3 fully connected layers, where the convolutional layers and the maxpooling layers are divided into 5 convolutional blocks.\(^{22}\) The fully connected layers use the output of $P_3$ for the softmax operation to obtain the target’s prediction label.

Sizes of the feature areas obtained by the APN can be variable. However, the classification network always determines the category based on the output of $P_5$. When the APN output feature area is small, $P_5$ cannot describe the image features well due to the fact that multiple convolutions retain less information. Therefore, by incorporating the SDP method into the VGG19 network, the network can intelligently select the most appropriate convolution block’s output for classification and enhance the network’s learning ability for small ship features. The network architecture diagram is shown in Figure 3.

The input image is first completely extracted through the classification network. Then the MF-APN network calculates $m$ which is the size of the feature area, and the VGG-SDP network will select the best result of pooling among the three pooling layers according to $N$ to represent image $I$ for classification. The selection criteria are as follows

$$f(I) = \begin{cases} P_3(I), & 0 < N \leq 64 \\ P_4(I), & 64 < N \leq 128 \\ P_5(I), & 128 < N \leq 224 \end{cases}$$

(2)
where $P_3(I)$ and $P_4(I)$ represent the third and fourth pooling layers, respectively; $f(\cdot)$ selects the optimal pooled output according to the pixel number of target regions returned by the MF-APN network. Choosing the power of 2 as the segmentation point is helpful for network training.\textsuperscript{29} $F(\cdot/C1)$ indicates the last full connection and softmax operation. When $N$ is too large, the output of the final $P_5$ should be selected to describe the feature of the large target. When $N$ is small, $P_3$ with more information should be selected.

The classification result of the image $I$ is a fusion of the prediction labels of the three-scale VGG-SDP networks. First, normalize each prediction label $Y^{(s)}$, put it into a fully connected layer, and then use the softmax function to obtain the final prediction classification label.\textsuperscript{12} MF-APN location

The original location network APN contains two fully connected layers. The feature area is obtained by inputting $P_5$. Once the network detects the feature area, it will crop and enlarge the area to obtain the input of the next scale layer. This method uses a square box to mark the target, and it is prone to include the part that does not belong to the feature or to cut off the part that originally belongs to the feature. Especially for ships, the characteristics of the forecastle, bridge, smoke outlet, radar antenna, and so on have specific regular geometric shapes and are mostly long rectangles, so it is easier to count into the background using only square boxes. In this article, we use different a priori rectangular boxes to mark the target. Finally, classifying multiple feature regions and performing joint discrimination can identify the target more robustly. Therefore, a multi-feature map (MF-APN) method is proposed.

Assume that the APN part outputs $t_s$ and $t_f$ represent the coordinate values of the center point of the target area, and $t_l$ is half the side length of the square box. $N$ is the number of pixels in the square box, that is, the target area. $W_i$ and $H_i$, respectively, represent half the length and width of the $i$th prior rectangle, and the scale factor $k_i$ represents the length–width ratio of the $i$th rectangle, as shown in Figure 4.

Then the following equations can be derived

$$N = (2t_l)^2 = 4t_l^2, \quad k_i = \frac{W_i}{H_i}$$

Assume that the area of the a priori rectangular box is equal to the area of the output square, and there are

$$N = 2W_l \times 2H_l = 4k_lH_l^2$$

Substituting equation (4) into equation (5), we get a new expression

$$W_i = \left[ \sqrt{k_i} \times t_l \right], \quad H_i = \left[ \frac{1}{\sqrt{k_i}} \times t_l \right]$$

The $[\cdot]$ function means rounding down to prevent square root operations from generating decimals. Use the top left and bottom right corners of the a priori rectangle to represent this rectangle. The subscript $ul$ stands for the upper left corner and $br$ stands for the lower right corner; then the coordinates of the two points are
Considering that the neural network backpropagation algorithm is derivable, we cannot use ordinary interception functions. Therefore, we design a deductible intercept function \( D(\cdot) \):

\[
D(\cdot) = \left[ h(x - t_{x,UL}) - h(x - t_{x,BR}) \right] \\
\times \left[ h(y - t_{y,UL}) - h(y - t_{y,BR}) \right]
\]

where \( h(\cdot) \) represents a sigmoid function and the formula is as follows

\[
h(x) = \frac{1}{1 + e^{-ax}}
\]

When \( a \) is large enough, the value of \( h(\cdot) \) is always 1 where the pixel point in the feature area, and \( a = 10 \) is set in this article. The final intercepted target area \( M_i \) can be expressed in the following form

\[
M_i = I \otimes D(t_x, t_y, t_{x,t}, t_{y,t})
\]

where the \( \otimes \) operation represents the element point multiplication.

The bilinear interpolation method is then used to enlarge the target area to generate the inputs of the next scale. If we choose plurality of the a priori rectangular boxes in each scale after the first scale, finally, the number of feature areas will become multiplicative growth. Considering the computational cost, only \( i \) rectangular boxes are extracted in the second scale layer. In addition, \( t_i \) in the new scale cannot be less than one-third of the previous scale, preventing the feature region from being too small to contain the feature portion.

### Network loss

The improved RA-CNN loss function is still divided into two parts, the same as in the original RA-CNN: namely, the intra-scale classification loss \( L_{inner} \) and the inter-scale loss \( L_{scale} \), where \( L_{inner} \) is a general cross-entropy calculation and \( L_{scale} \) is calculated as follows

\[
L_{scale}\left( p_i^{(s)}, p_i^{(s+1)} \right) = \max\left\{ 0, p_i^{(s)} - p_i^{(s+1)} + 0.05 \right\}
\]

By taking the maximum value, the network is required to update when the current scale real class probability \( p_i^{(s+1)} \) is smaller than the previous scale real class probability \( p_i^{(s)} \), which causes the network to have a higher prediction probability on a finer scale. The loss between scales is only updated when \( p_i^{(s)} - p_i^{(s+1)} + 0.05 > 0 \). The addition of 0.05 is to prevent both sides from staying in stagnation when they both are equal to zero.

Since there are \( M \) feature rectangles in the second and third scales, the final predicted category label \( Y^{(s)} \) is the weighted average of the predicted probabilities of the \( M \) feature rectangles. For the prediction probability \( p_j^{(s)} \) of the \( j \)th class in \( Y^{(s)} \), the formula is

\[
p_j^{(s)} = \frac{1}{M} \sum_{i=1}^{M} a_i \times p_{j,i}^{(s)}
\]

where \( M \) represents the number of rectangular boxes and \( a_i \) represents the weight of the \( i \)th rectangular box and has \( \sum_{i=1}^{M} a_i = 1 \). In this article, we set \( M = 3 \) (contains a square box), the values of the scale factor \( k \) are 2, 1, and 0.5, respectively, and the corresponding weight
values are 0.4, 0.2, and 0.4. With such settings, the influence of the rectangular box in the prediction is enhanced, whereas the influence of the square box is weakened.

**Experimental results**

*The hardware environment*

The experimental environment is TensorFlow 1.10.0, OpenCV3, CUDA 9.0, and CUDNN 7.0. The tasks are executed on a computer with Intel Core i7-7800X with 16 GB RAM and an NVIDIA GTX1080Ti with 11 GB memory.

*The dataset*

Many open-source fine-grained image datasets can be found on the network such as CUB Birds, Stanford Dogs, and Stanford Cars. These datasets can well test the performance of fine-grained classification networks. However, this article proposes a special algorithm for ships and basically no datasets about the ship can be found on the network. Therefore, in this article, a self-built ship dataset is used to verify the network performance. The dataset consists of eight categories, namely, container ships, sailboats, tugs, passenger ships (PS), tankers, ocean surveillance ships (OSS), engineering ships, and fishing boats (FB). There are 2635 samples in the dataset, with about 300 samples in each category. The number of images is still relatively small and we will continue to expand the amount. You can find our dataset at https://www.dropbox.com/sh/pyyfootec7f9r5l/AAA8yVwhr3N7VwndU_lOCsKLa?dl=0

The ratio of the samples used for training and testing is approximately 4:1. The specific distribution of the dataset is shown in Table 1. Each ship class includes the front view, rear view, side view, and the side-overhead view to ensure accurate identification in all directions during identification. The samples are shown in Figure 5. The image resolutions are almost between 500 × 400 and 300 × 200. The target accounts for almost 50%–70% of the picture.

|                | Container | Sailboat | Tug | PS | Tanker | OSS | Engineering | FB |
|----------------|-----------|----------|-----|----|--------|-----|-------------|----|
| **Training**   | 304       | 256      | 240 | 256| 160    | 256 | 368         | 272|
| **Test**       | 69        | 72       | 62  | 60 | 43     | 64  | 84          | 69 |

PS: passenger ships; OSS: ocean surveillance ships; FB: fishing boats.

**Parameter settings and evaluation metrics**

In this article, we take several neural network acceleration methods to promote network convergence and accelerate network training. The drop rate of the neural network is set to 0.2, and the initial learning rate is set to 0.1. After 100,000 iterations, the learning rate is reduced to one-tenth of the original. To speed up convergence, the network will be trained 10 rounds on the ImageNet dataset in advance.

Use Accuracy as an evaluation metric to test neural network performance; the accuracy calculation formula is as follows

\[
\text{Accuracy} = \frac{TP + TN}{TP + TN + FP + FN} \tag{13}
\]

where \(TP\) is true positive (predict the positive sample correctly), \(TN\) is true negative (correctly predict but the negative sample), \(FP\) is false positive (predict the negative sample as the positive sample), and \(FN\) is false negative (predict the positive sample as the negative sample).

**Experiment 1: calculating the amount and time**

For deep learning networks, computational complexity is a very important factor to evaluate the efficiency of the network. The RA-CNN is effectively designed on the network structure to improve the fine-grained
detection performance. However, due to the adoption of multiple scale layers, the computational complexity of the network is increased. Meanwhile, the MF method in this article improves network accuracy by adding three feature regions at the second and third scale layers. That is, the network parameters of the second and third scale layers are increased by nearly three times, and the overall parameter amount and calculation time should be correspondingly increased. So it is necessary to test the impact of the network complexity. The overall network structure and parameters are shown in Table 2.

It is worth mentioning that the parameter values in the APN-FC1 layer are uncertain. This is because the input may come from one of the three different pooling layers, so the MF-APN part prepares three input entries. The computational costs of different models are illustrated in Figure 6. The Flops in the figure are used to measure the amount of network calculation, and the calculation formulas for the convolutional layer and the fully connected layer are

\[
\text{Flops} = \left\{ \frac{(H \times W \times C_{\text{in}} \times C_{\text{out}} + C_{\text{out}}) \times (H \times W)}{N_{\text{in}} \times N_{\text{out}} + N_{\text{out}}} \right\}
\]

where \( H, W \) represents the size of the convolution kernel, \( C_{\text{in}} \) represents the number of channels input, \( C_{\text{out}} \) represents the number of channels output, and \( N_{\text{in}} \) and \( N_{\text{out}} \) represent the numbers of inputs and outputs of the fully connected layer, respectively.

From Figure 6, it can be seen that RA-CNN is nearly 0.7 s longer than the VGG19 in single-frame image processing time due to its three-layer progressive structure. The figure size in Figure 6 is proportional to the amount of network parameters. Therefore, it can

![Figure 6. Performance analysis of the RA-CNN + SDP + MF algorithm.](image)

| Part | Name | Type | Stride | Output | Parameters |
|------|------|------|--------|--------|------------|
| VGG19 | Conv 1 | 3×3 conv,64 | 1×1 | 224×224×64 | 1.7k |
| | Conv 1 | 3×3 conv,64 | 1×1 | 224×224×64 | 36.8k |
| | Pool1 | 2×2 max pooling | 2×2 | 112×112×64 | 73.7k |
| | Conv 2 | 3×3 conv,128 | 1×1 | 112×112×128 | 147.4k |
| | Pool2 | 2×2 max pooling | 2×2 | 56×56×128 | 147.4k |
| | Conv 3 | 3×3 conv,256 | 1×1 | 56×56×256 | 294.9k |
| | Conv 4 | 3×3 conv,512 | 1×1 | 28×28×512 | 1179.6k |
| | Pool4 | 2×2 max pooling | 2×2 | 14×14×512 | 2359.2k |
| | Conv 5 | 3×3 conv,512 | 1×1 | 14×14×512 | 2359.2k |
| | Conv 6 | 3×3 conv,512 | 1×1 | 14×14×512 | 2359.2k |
| | Conv 7 | 3×3 conv,512 | 1×1 | 14×14×512 | 2359.2k |
| | Conv 8 | 3×3 conv,512 | 1×1 | 14×14×512 | 2359.2k |
| | Pool5 | 2×2 max pooling | 2×2 | 7×7×512 | 2359.2k |
| MF-APN | VGG-FC1 | FC | 1×1×4096 | 10,2760.4k |
| | VGG-FC2 | FC | 1×1×4096 | 16,777.2k |
| | VGG-FC3 | FC | 1×1×8 | 32.7k |
| | APN-FC1 | FC | 1×1×1024 | * |
| | APN-FC2 | FC | 1×1×3 | 3k |

MF-APN: multi-feature attention proposal network; FC: fully connected layer.

*represents the uncertain parameters’ after this sentence.

International Journal of Distributed Sensor Networks
also be seen from Figure 6 that the SDP algorithm hardly increases the parameters and calculation time, while the MF algorithm increases nearly 0.6 s and the floating-point calculation amount is also greatly improved.

It takes about 1.5 s for the new network to process a single frame, which is not feasible for real-time video detection. Considering that each scale's input is calculated from the previous scale, this serial structure is difficult to optimize. However, three feature areas can be calculated independently at the same time, so the MF portion can use the distributed computing method to shorten the calculation time to 0.92 s. From the experimental results, the network studied in this article can be used for real-time recognition of single-frame images. However, when applied to video recognition, a high frame rate cannot be guaranteed. It is still necessary to optimize the reduction of the overall floating-point calculation from the network parameters.

Experiment 2: accuracy of different baseline models

The model used in this article is based on RA-CNN, while RA-CNN is based on VGG19. We propose two improvement methods, which are SDP for choosing pooling results flexibly and MF-APN for more precise category discrimination.

In this experiment, we combine the RA-CNN and two proposed methods one by one to test the effect of a single method. Finally, we compare the method of this article with baseline models. The results are shown in Table 3. The overall accuracy curve is shown in Figure 7.

It can be seen from Table 3 that the accuracy of RA-CNN is 5.8% higher than that of VGG19 as the basic detection network, indicating that RA-CNN is a better network for fine-grained image detection and classification. The results of using SDP and MF alone were 0.5% and 0.9% higher than the original RA-CNN network, respectively. When using both SDP and MF, the accuracy is improved by up to 2%. This implies that the use of adaptive feature pooling and the use of multiple feature maps can effectively improve the accuracy and the combination of the two methods can achieve better results. It can be seen from Figure 7 that the unmodified VGG19 can achieve a certain accuracy faster. In contrast, the network based on RA-CNN will take more time to train on MF-APN, but the final result will be much better.

In Figure 8, the effect of the network detection ship is demonstrated. In order to better display the type of ship, the feature area generated by MF-APN is used. It must be acknowledged that, during the testing process, the network will still classify some ships into the wrong category, so in the future work it is still necessary to expand the dataset and optimize the network.

Experiment 3: accuracy and attention region of different scales

The original RA-CNN model includes three scale layers to local feature map step by step. However, the accuracy of the third scale layer can be less than those of the first two layers. In this experiment, we will test the impact of new methods on the accuracy of different scales. The experimental conditions are the same as those in Experiment 1. The difference is that only the different scale layers of RA-CNN and RA-CNN + SDP + MF are tested. The results are shown in Table 4.

It can be seen from Table 4 that the accuracy of using only the second scale in the original RA-CNN is 0.9% higher than that when using only the third scale. But this gap is narrowed to 0.2% in the method proposed in this article. This proves that the new method can compensate for the accuracy impact of global information loss on a finer scale.

We also test the attention region on the trained model. Figure 9 shows the attention region of each scale of the network. Figure 9(a) shows the original picture and Figure 9(b) shows the input for the first scale.

| Approach         | Accuracy (%) |
|------------------|--------------|
| VGG19            | 78.4         |
| RA-CNN           | 84.2         |
| RA-CNN + SDP     | 84.7         |
| RA-CNN + MF      | 85.1         |
| RA-CNN + SDP + MF| 86.7         |

RA-CNN: recurrent attention convolutional neural network; SDP: scale-dependent pooling; MF: multi-feature.

Figure 7. Overall accuracy curve of different baseline models.
after preprocessing. From Figure 9(c), we can see that, after MF-APN, the first scale selects three attention areas. These areas will be intercepted once and the corresponding pictures in Figure 9(d) will be generated. It can be seen from the results that, after training, the network can accurately locate the target’s attention region in the second scale.

**Conclusion**

Based on the fine-grained detection network RA-CNN, this article proposes an improved identification method for intelligently distinguishing fine-grained ships. As an end-to-end weak surveillance detection network, training can be performed with no bounding boxes provided. The SDP algorithm allows the classification network to adaptively select the most representative pooling layer output, which enhances the network’s ability to learn local features. The MF method is proposed for the special geometric appearance characteristics of ships. By extracting multiple prior feature boxes in the image to comprehensively describe the feature regions, the algorithm avoids situations where the single feature area may contain too much unrelated background. At the same time, the weighted average decision prediction label is used to enhance the influence of rectangular boxes. Using batch normalization and dropout, changing the learning rate and using pre-training weights can optimize the network model and accelerate the convergence of network training. Our experiments show that this method can quickly and accurately classify and detect ship photoelectric targets in complex backgrounds and provide early warning against collisions.

**Declaration of conflicting interests**

The author(s) declared no potential conflicts of interest with respect to the research, authorship, and/or publication of this article.

**Funding**

The author(s) disclosed receipt of the following financial support for the research, authorship, and/or publication of this article: This research was funded by the National Science Foundation of China (Grant No. 61673259).

**ORCID iD**

Yuhao Huo https://orcid.org/0000-0002-5205-9755

**References**

1. Lang H, Zhang J, Zhang X, et al. Ship classification in SAR image by joint feature and classifier selection. *IEEE Geosci Remote Sens Lett* 2016; 13(2): 212–216.

**Table 4.** Accuracy of different scales.

| Approach                        | Accuracy (%) |
|---------------------------------|--------------|
| RA-CNN                          | 84.2         |
| RA-CNN (Scale 2)                | 83.7         |
| RA-CNN (Scale 3)                | 82.8         |
| RA-CNN (Scales 1 + 2)           | 83.9         |
| RA-CNN + SDP + MF               | 86.7         |
| RA-CNN + SDP + MF (Scale 2)     | 85.7         |
| RA-CNN + SDP + MF (Scale 3)     | 85.5         |
| RA-CNN + SDP + MF (Scales 1 + 2)| 86.1         |

RA-CNN: recurrent attention convolutional neural network; SDP: scale-dependent pooling; MF: multi-feature.

**Figure 8.** Examples of network classification. The prediction box in the figure comes from the prediction of the APN network.

**Figure 9.** Attention region of different scales: (a) original picture, (b) first scale, (c) second scale, and (d) third scale.
2. Mazzarella F, Vespe M and Santamaria C. SAR ship detection and self-reporting data fusion based on traffic knowledge. *IEEE Geosci Remote Sens Lett* 2015; 12(8): 1685–1689.

3. Jiang M, Yang X, Dong Z, et al. Ship classification based on superstructure scattering features in SAR images. *IEEE Geosci Remote Sens Lett* 2016; 13(5): 616–620.

4. Leng X, Ji K, Yang K, et al. A bilateral CFAR algorithm for ship detection in SAR images. *IEEE Geosci Remote Sens Lett* 2015; 12(7): 1536–1540.

5. Zhang X, Xiong H, Zhou W, et al. Picking deep filter responses for fine-grained image recognition. In: *Proceedings of the IEEE conference on computer vision and pattern recognition*, Las Vegas, NV, 27–30 June 2016, pp.1134–1142. New York: IEEE.

6. Ioffe S and Szegedy C. Batch normalization: accelerating deep network training by reducing internal covariate shift, 2015, arXiv preprint arXiv:1502.03167.

7. Radenovic F, Tolias G and Chum O. CNN image retrieval learns from bow: unsupervised fine-tuning with hard examples. In: *European conference on computer vision*, Amsterdam, 11–14 October 2016, pp.3–20. Berlin: Springer.

8. Ning Z, Donahue J, Girshick R, et al. Part-based R-CNNs for fine-grained category detection. In: *European conference on computer vision*, Zurich, 6–12 September 2014. Berlin: Springer.

9. Wei XS, Xie CW, Wu J, et al. Mask-CNN: localizing parts and selecting descriptors for fine-grained bird species categorization. *Pattern Recognit* 2018; 76: 704–714.

10. Simon M and Rodner E. Neural activation constellations: unsupervised part model discovery with convolutional networks. In: *Proceedings of the IEEE international conference on computer vision*, Santiago, Chile, 7–13 December 2015, pp.1143–1151. New York: IEEE.

11. Lin TY, Roy Chowdhury A and Maji S. Bilinear CNN models for fine-grained visual recognition. In: *Proceedings of the IEEE international conference on computer vision*, Santiago, Chile, 7–13 December 2015, pp.1449–1457. New York: IEEE.

12. Fu J, Zheng H and Mei T. Look closer to see better: recurrent attention convolutional neural network for fine-grained image recognition. In: *Proceedings of the IEEE conference on computer vision and pattern recognition*, Honolulu, HI, 21–26 July 2017, pp.4438–4446. New York: IEEE.

13. LeCun Y, Bengio Y and Hinton G. Deep learning. *Nature* 2015; 521(7553): 436–444.

14. LeCun Y, Bottou L, Bengio Y, et al. Gradient-based learning applied to document recognition. *Proc IEEE* 1998; 86(11): 2278–2324.

15. Hinton GE and Salakhutdinov RR. Reducing the dimensionality of data with neural networks. *Science* 2006; 313(5786): 504–507.

16. Krizhevsky A, Sutskever I and Hinton GE. ImageNet classification with deep convolutional neural networks. In: *Advances in neural information processing systems*, Lake Tahoe, NV, 3–6 December 2012, pp.1097–1105. New York: ACM.

17. Ren S, He K, Girshick R, et al. Faster R-CNN: towards real-time object detection with region proposal networks. *IEEE Trans Pattern Anal Mach Intell* 2017; 39(6): 1137–1149.

18. Redmon J, Divvala S, Girshick R, et al. You only look once: unified, real-time object detection. In: *Proceedings of the IEEE conference on computer vision and pattern recognition*, Las Vegas, NV, 27–30 June 2016, pp.779–788. New York: IEEE.

19. Redmon J and Farhadi A. YOLO9000: better, faster, stronger. In: *Proceedings of the IEEE conference on computer vision and pattern recognition*, Honolulu, HI, 21–26 July 2017, pp.7263–7271. New York: IEEE.

20. Redmon J and Farhadi A. YOLOv3: An incremental improvement, 2018, arXiv preprint arXiv:180402767.

21. Liu W, Anguelov D, Erhan D, et al. SSD: Single shot multibox detector. In: *European conference on computer vision*, Amsterdam, 8–16 October 2016, pp.21–37. Berlin: Springer.

22. Simonyan K and Zisserman A. Very deep convolutional networks for large-scale image recognition, 2014, arXiv preprint arXiv:14091556.

23. Kang M, Ji K, Leng X, et al. Contextual region-based convolutional neural network with multilayer fusion for SAR ship detection. *Remote Sens* 2017; 9(8): 860.

24. Zhao J, Guo W, Zhang Z, et al. A coupled convolutional neural network for small and densely clustered ship detection in SAR images. *Sci China Inform Sci* 2019; 62(4): 42301.

25. Oliveau Q and Sabhi H. Semi-supervised deep attribute networks for fine-grained ship category recognition. In: *2018 IEEE international geoscience and remote sensing symposium (IGARSS 2018)*, Valencia, 22–27 July 2018, pp.6871–6874. New York: IEEE.

26. Zhang L, Zhang Y, Zhang Z, et al. Real-time water surface object detection based on improved faster R-CNN. *Sensors* 2019; 19(16): 3523.

27. Lin Z, Ji K, Leng X, et al. Squeeze and excitation rank faster R-CNN for ship detection in SAR images. *IEEE Geosci Remote Sens Lett* 2019; 16(5): 751–755.

28. Du C and Gao S. Image segmentation-based multi-focus image fusion through multi-scale convolutional neural network. *IEEE Access* 2017; 5: 15750–15761.

29. Yang F, Choi W and Lin Y. Exploit all the layers: fast and accurate CNN object detector with scale dependent pooling and cascaded rejection classifiers. In: *Proceedings of the IEEE conference on computer vision and pattern recognition*, Las Vegas, NV, 27–30 June 2016, pp.2129–2137. New York: IEEE.

30. Girshick R, Donahue J, Darrell T, et al. Rich feature hierarchies for accurate object detection and semantic segmentation. In: *Proceedings of the IEEE conference on computer vision and pattern recognition*, Columbus, OH, 23–28 June 2014, pp.580–587. New York: IEEE.

31. Klambauer G, Unterthiner T, Mayr A, et al. Self-normalizing neural networks. In: *Advances in neural information processing systems*, Long Beach, CA, 4–9 December 2017, pp.971–980. New York: ACM.

32. Fawcett T. An introduction to ROC analysis. *Pattern Recognit Lett* 2006; 27(8): 861–874.

33. Molchanov P, Tyree S, Karras T, et al. Pruning convolutional neural networks for resource efficient inference, 2016, arXiv preprint arXiv:161106440.