




Article

D-ATR for SAR Images Based on Deep Neural Networks

Zongyong Cui ¹, Cui Tang ¹, Zongjie Cao ^{1,2,*} and Nengyuan Liu ¹

¹ School of Information and Communication Engineering, University of Electronic Science and Technology of China (UESTC), Chengdu 611731, China; zycui@uestc.edu.cn (Z.C.); appletreetc@sina.com (C.T.); nengyuanliu@outlook.com (N.L.)

² Center for Information Geoscience, University of Electronic Science and Technology of China (UESTC), Chengdu 611731, China

* Correspondence: zjcao@uestc.edu.cn

Received: 20 March 2019; Accepted: 27 March 2019; Published: 13 April 2019



Abstract: Automatic target recognition (ATR) can obtain important information for target surveillance from Synthetic Aperture Radar (SAR) images. Thus, a direct automatic target recognition (D-ATR) method, based on a deep neural network (DNN), is proposed in this paper. To recognize targets in large-scene SAR images, the traditional methods of SAR ATR are comprised of four major steps: detection, discrimination, feature extraction, and classification. However, the recognition performance is sensitive to each step, as the processing result from each step will affect the following step. Meanwhile, these processes are independent, which means that there is still room for processing speed improvement. The proposed D-ATR method can integrate these steps as a whole system and directly recognize targets in large-scene SAR images, by encapsulating all of the computation in a single deep convolutional neural network (DCNN). Before the DCNN, a fast sliding method is proposed to partition the large image into sub-images, to avoid information loss when resizing the input images, and to avoid the target being divided into several parts. After the DCNN, non-maximum suppression between sub-images (NMSS) is performed on the results of the sub-images, to obtain an accurate result of the large-scene SAR image. Experiments on the MSTAR dataset and large-scene SAR images (with resolution 1478×1784) show that the proposed method can obtain a high accuracy and fast processing speed, and out-performs other methods, such as CFAR+SVM, Region-based CNN, and YOLOv2.

Keywords: D-ATR; SAR images; deep neural network; non-maximum suppression

1. Introduction

Synthetic aperture radar (SAR) is capable of working every day, in all weather conditions, and all the time, to provide high resolution images, and so it plays a significant role in surveillance and battlefield reconnaissance [1,2]. Automatic target recognition (ATR) is the process of automatic target acquisition and classification, which is capable of recognizing targets or other objects, based on data obtained from the sensors, which has good application prospects in both military and civilian areas [3]. The process of SAR ATR can be summarized as finding regions of interest (ROIs) in the observed SAR image and classifying the category of each ROI (e.g., T72 or BTR70) [4]. Some earlier methods of SAR ATR can be found in [5–9].

Traditional SAR ATR techniques mainly include four steps: detection, discrimination, feature extraction, and target recognition/classification [10]. For target detection, potential ROIs are extracted from the input SAR image according to the local brightness or the shape of targets; CFAR [11] is a classical algorithm used to detect targets against a background of noise, cluster, and conduct

interference from SAR images by detecting every pixel. In the discrimination phase, the ROIs obtained from the previous step of detection are processed to remove false alarms, with the purpose of reducing classification cost. The feature extractor is specific to particular tasks in the interpretation of SAR images, which can suppress the dimension of the feature space to interpret the SAR imagery. Some researchers use a feature-based approach to deal with the problem of SAR ATR [12,13]. After detection and discrimination, the remaining ROIs are input into the recognition/classification stage to obtain the type of target (i.e., armored personnel carrier, howitzer, or tank). There are mainly two traditional methods, the most common one is based on template-matching methods. The second is based on classifier models, such as support vector machines (SVM) [5] and adaptive boosting [14]. However, traditional SAR ATR methods depend heavily on handcrafted features and have a large computational burden or poor generalization performance [15]. The accuracy will also decrease significantly if any stage of the SAR ATR is not well designed or not suitable for the current operating conditions [16].

Recently, deep learning (DL) algorithms have been significantly developed. Girshick proposed regions with CNN features (R-CNN) [17] in 2014, and object detection based on deep learning began to come into favor. Subsequently, many improved algorithms based on R-CNN have been proposed, such as Fast R-CNN [18] and Faster R-CNN [19], which have achieved high accuracies in recognizing targets in optical images. However, these methods have been too computationally intensive for embedded systems and, even with high-end hardware, too slow for real-time applications.

For the sake of speeding up computation, some researchers proposed methods based on a single network, which predicts bounding boxes directly without region proposals. Redmon proposed You Only Look Once (YOLO) [20], a regression-based method which directly recognizes different kinds of objects with different sizes in optical images and gives confidence ratios. However, it had a problem with inaccurate positioning. Liu proposed a Single Shot MultiBox Detector (SSD) [21], which showed a compromise of accuracy and speed in the field of optical object detection.

Inspired by the successful application of deep learning methods in optical areas, some researchers introduced DL methods for dealing with problems in the processing of SAR images. Ref. [22,23] effectively extracted a high-level feature representation for SAR images by using a Deep Convolutional Neural Network (DCNN) which learned high-level features automatically, rather than requiring handcrafted features. Ref. [24] proposed an efficient feature extraction and classification algorithm, based on a visual saliency model. Ref. [25] proposed a target detection and discrimination method, based on a visual attention model, and the experimental results on synthetic images and the miniSAR image data set demonstrated that the proposed target detection and discrimination method could detect and discriminate the targets from complex background clutter with a high accuracy and fast speed for high-resolution SAR images, which provides an effective way to overcome the drawbacks in target detection and discrimination in SAR images with large, complex scenes. Ref. [15] used CNN to recognize SAR targets, and achieved a competitive classification performance with existing methods considered to be state-of-the-art [22]. Their work proved that deep learning methods can be used in every process of SAR ATR. However, these methods mainly just focus on one of the four steps of SAR ATR.

To date, Wang [26] used faster R-CNN which achieved detection and recognition integration in the field of optical target detection, to realize the integration of detection and recognition in the field of SAR ATR, and obtained a system dealing with large-scene SAR images. Ref. [27] proposed a region-based convolutional neural network to process the problem of SAR target recognition in large-scene images. However, the processing time of these systems can be further decreased.

For the sake of integrating the traditional four steps of SAR ATR as a whole system, we were encouraged by the previous works in adopting deep learning methods for target detection in optical images to the field of SAR images. By encapsulating all computation in a single deep neural network, the integration of target detection and recognition of large scene SAR images can be realized.

The proposed D-ATR system can directly recognize targets from complex background clutter with a high accuracy and fast speed in large-scene SAR images. Transfer learning and data augmentation

methods, such as horizontal flip and random crop, are used in this paper, at the stage where the available SAR images are limited for training. To meet the requirement of input size of the neural network, a method of fast sliding is used to cut the large-scene SAR images into sub-images with a suitable size for the input of the neural network, and to guarantee that every target exists completely in one of the sub-images. Finally, non-maximum suppression between sub-images (NMSS) is proposed to suppress the predicted boxes among the sub-images, for more accurate recognition performance.

The organization of this paper is as follows. Section 2 introduces the structure and components of the deep convolutional neural network. Section 3 provides experimental results, by several experiments, to compare the performance of the proposed method. Finally, Section 4 makes a conclusion of this paper and prospects for the future work.

2. Structure of The D-ATR

The flowchart of the proposed D-ATR is shown in Figure 1, which can realize the integration of target detection and recognition in large-scene SAR images. It contains three main parts. The first part is a fast sliding method for cutting the large-scene SAR into sub-images with a suitable size. The second part is the network for feature extraction, target detection, and recognition. The third part is the proposed NMSS for retaining the best bounding box and obtaining the best recognition result.

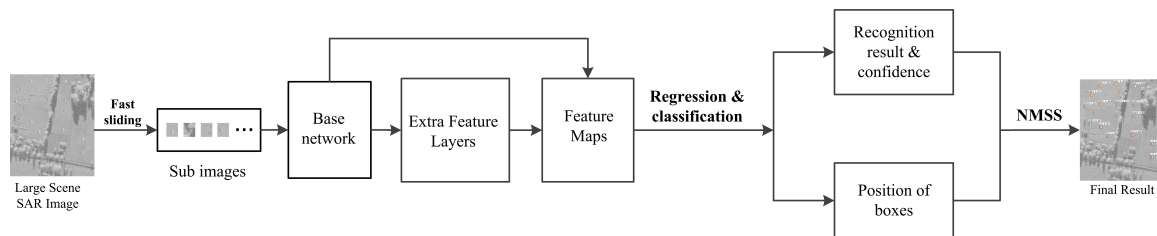


Figure 1. The flow chart of the proposed D-ATR for large-scene SAR images.

2.1. Base Network

The most common obstacle when applying deep learning methods to solve problems lies in the necessary large amount of data. The reason why so much training data is needed is that there are a large number of parameters to be determined during the training process. For example, ImageNet is a large-scale labeled image dataset, organized according to the WordNet architecture, and contains about 2.2 million categories and 15 million images, which are strictly selected and labeled by human curators. AlexNet [28] showed surprising performance on the object classification of 1000 categories in ImageNet in 2012. Subsequently, VGG-16-Net [29] and GoogLeNet [30] were proposed, with better recognition rates. The latest Res-Net [31] achieved extraordinary performance when recognizing targets in ImageNet. Deep learning has made great progress in object recognition, and is also expected to solve the problems of SAR target recognition. However, compared to ImageNet, there are insufficient annotated SAR data, as it is expensive to capture SAR images and annotate them.

Transfer learning is a method in the area of machine learning. Given a source domain $D_s = \{X_s, f_s(X)\}$ and a learning task T_s , as well as a target domain $D_T = \{X_T, f_T(X)\}$ and a learning task T_T , transfer learning aims to help improve the learning of the target predictive function $f_T(\cdot)$ in D_T , using the knowledge in D_s and T_s , where $D_s \neq D_T$ or $T_s \neq T_T$. The key point is to store the knowledge acquired in solving one problem and applying it to a different, but related, problem. For instance, the knowledge acquired by learning to recognize a dog may be suitable in applying to recognize a cat. The method of transfer learning provides an effective way to train a large network with limited training data without overfitting.

Researchers have successfully applied the method of transfer learning to the field of SAR classification. Ref. [32] proposed a method based on transfer learning, which transformed the knowledge learned from sufficient unlabeled SAR scene images to labeled SAR target data. Ref. [33] introduced transfer learning into the classification of a small number of SAR images with a limited

quantity of SAR imagery training data, and the parameters from the model trained on CIFAR10 have successfully applied to TerraSAR-X data.

In this paper, VGG-16-Net is selected as the base network, which is pre-trained on the ILSVRC CLS-LOC dataset [34].

2.2. Additional Feature Layers

Additional convolutional layers are added after the basic network. As shown in Figure 2, there are five convolutional layers (from CONV1 to CONV5) in the structure of the additional feature layer; the products in the semicircle box indicate the number of convolutional kernel and its size (e.g., $3 \times 3 \times 1024$ represents that there are 1024 convolutional kernels with size of 3×3). The number in parentheses, next to the arrow, represents the resulting feature maps from the related convolutional layer (e.g., (1024, 19×19) indicates that CONV1 generates 1024 feature maps with size of 19×19). The size of these convolutional layers decreases layer by layer, which can generate multi-scale feature maps for detection.

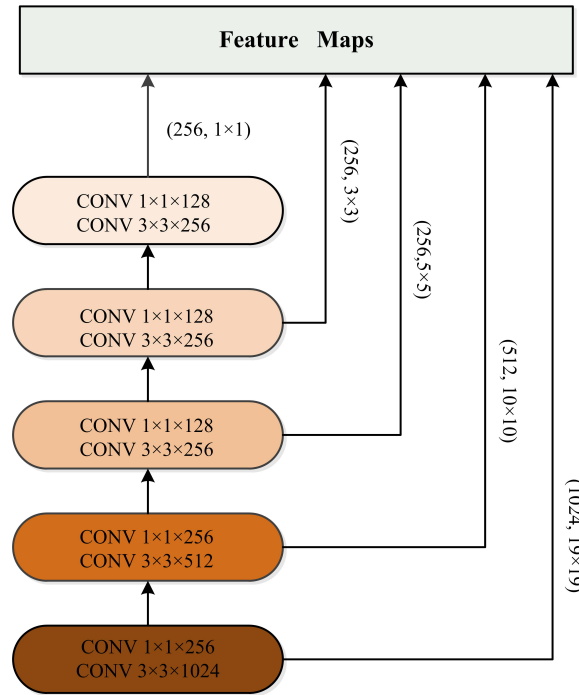


Figure 2. Additional feature layers with five convolution layers.

2.2.1. Convolutional Layer

Different features of SAR images are extracted by convolutional layers with a lot of convolution kernels. The previous layer's input feature maps $O_i^{(l-1)}$ ($i = 1, \dots, I$) are connected to all the output feature maps $O_j^{(l)}$ ($j = 1, \dots, J$), where $O_i^{(l-1)}(x, y)$ is the unit of the i th input feature map at position (x, y) ; $k_{ji}^{(l)}(u, v)$ represents the convolution kernel which connects the i th and j th input and output feature maps, respectively; and $b_j^{(l)}$ is the trainable bias of the j th output feature map. The calculation of convolution is illustrated as follows:

$$O_j^l(x, y) = f(G_j^l(x, y)) \tag{1}$$

$$G_j^l(x, y) = \sum_{i=1}^I \sum_{u,v=0}^{K-1} k_{ji}^{(l)}(u, v) \cdot O_i^{l-1}(x-u, y-v) + b_j^{(l)}, \tag{2}$$

where $f(\cdot)$ is the nonlinear activation function, $G_j^{(l)}(x, y)$ represents the weighted sum of inputs to the output feature map at position (x, y) , I is the number of input feature maps, $K \times K$ is size of the convolution kernels, and P and S are the zero padding and convolution stride, respectively.

The size of kernels of the convolutional layer is $K \times K$, and if there are J feature maps with size $W_1 \times H_1$ as input, the J output feature maps are $W_2 \times H_2$, and the computation of W_2 and H_2 is as follows, respectively:

$$W_2 = \left\lfloor \frac{W_1 - K + 2P}{S} \right\rfloor + 1 \quad (3)$$

$$H_2 = \left\lfloor \frac{H_1 - K + 2P}{S} \right\rfloor + 1. \quad (4)$$

Some researchers introduced a novel visualization technique which gives insight into the function of intermediate feature layers and the operation of the classifier, which proved that a smaller stride (2 versus 4) and filter size (7×7 versus 11×11) resulted in more distinctive features and performed better [35]. In this paper, we comply with certain guidelines, so the hyperparameters, such as convolution stride and filter size in the convolution layer, are as shown in Table 1, where the size of feature maps are calculated by Equations (5) and (6).

Table 1. Hyperparameters in the convolution layer.

CONV Layer	CONV1	CONV2	CONV3	CONV4	CONV5
CONV stride	1	2	2	1	1
filter size	3×3	3×3	3×3	3×3	3×3
Feature map	19×19	10×10	5×5	3×3	1×1

2.2.2. Receptive Field

After determining the hyperparameters of the CNN network, the corresponding theoretical receptive fields in each layer are also determined. The receptive field of a neuron in one of the lower layers encompasses only a small area of the image, while the receptive field of a neuron in subsequent (higher) layers involves a combination of receptive fields from several (but not all) neurons in the layer before (i.e., a neuron in a higher layer “looks” at a larger portion of the image than a neuron in a lower layer does). In this way, each successive layer is capable of learning increasingly abstract features of the original image. Assuming that each receptive domain is R_i ($i = 1, 2, \dots, n$), where R_i denotes the receptive domain of the i th layer, the formula for calculating the perceptual domain is as follows:

$$R_{i-1} = s_i (R_i - 1) + k_i \quad (5)$$

$$s_i = s_1 \times s_2 \times \dots \times s_{i-1}, \quad (6)$$

where s_i and k_i represent the convolution stride and the size of the convolution kernel in each layer, respectively. R_{i-1} and R_i are the receptive fields of the $(i - 1)$ th and the i th convolutional layers. Obviously, with a larger number of network layers, the size of the receptive field gradually increases, thus allowing simultaneous detection of targets of different sizes. The large receptive field is mainly responsible for the detection of large targets, and the small receptive field is responsible for the detection of small targets.

2.2.3. Detector and Classifier

As shown in Figure 3, the detector and classifier are mainly comprised of three parts. The first part is to generate the default bounding box, the second part is for positioning or localization, and the third part is responsible for generating the confidence of the category. In detail, the $m \times n$ feature maps obtained from the additional convolutional layers and base network will be convoluted with two different 3×3 convolutional kernels, one for producing a score or confidence for a category, and the

other generating a shape offset relative to the default box coordinates. For a feature map of size $m \times n$, there are $m * n$ feature map cells, in total. The number of default bounding boxes of each cell and the number of objects to be detected and recognized are denoted by K and C , respectively. Each cell requires a total of $K * (C + 4)$ predictions, and so all cells need a total of $K * (C + 4) * m * n$ predictions.

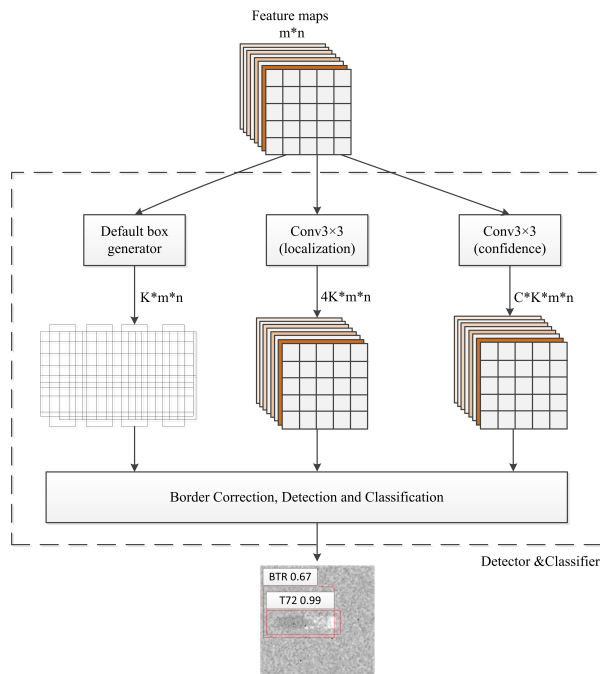


Figure 3. Detector and classifier integrated into one network.

2.2.4. Overall Training Process

The training process includes the following five steps:

- Step 1: Obtain the basic features of the input image by forward propagation;
- Step 2: Extract multi-scale feature maps and select candidate regions with different scales and different aspect ratios in these feature maps;
- Step 3: Calculate the coordinate position offset and category score of each candidate area;
- Step 4: Calculate the final region, according to the offsets of the candidate region and the coordinate position, and then calculate the loss function of the candidate region according to the category score and accumulate the final loss function; and
- Step 5: The weight of each layer is modified by the last loss function by a back-propagation algorithm.

The center (cx, cy) , width (w) , and height (h) of the default bounding box are regressed to offsets. The overall loss function is similar to [19], as shown in Equation (7), which contains two parts including localization loss (loc) and confidence loss (conf).

$$L(x, c, l, g) = \frac{1}{N} \left(L_{conf}(x, c) + aL_{loc}(x, l, g) \right). \tag{7}$$

The localization loss is shown as follows

$$L_{loc}(x, l, g) = \sum_{i \in Pos} \sum_{m \in \{cx, cy, w, h\}} x_{ij}^k smooth_{L1}(l_i^m - \hat{g}_j^m), \tag{8}$$

and the confidence loss is

$$L_{conf}(x, c) = - \sum_{i \in Pos} x_{ij}^p \log(\hat{c}_i^p) - \sum_{i \in Neg} \log(\hat{c}_i^0), \quad (9)$$

where $\hat{c}_i^p = \frac{\exp(c_i^p)}{\sum_p \exp(c_i^p)}$,

$$\hat{g}_j^{cx} = (g_j^{cx} - d_i^{cx}) / d_i^w, \hat{g}_j^{cy} = (g_j^{cy} - d_i^{cy}) / d_i^h, \quad (10)$$

$$g_j^w = \log\left(\frac{g_j^w}{d_i^w}\right), \hat{g}_j^h = \log\left(\frac{g_j^h}{d_i^h}\right). \quad (11)$$

The $(\hat{g}_j^{cx}, \hat{g}_j^{cy}, \hat{g}_j^w, \hat{g}_j^h)$ and $(d_i^{cx}, d_i^{cy}, d_i^w, d_i^h)$ in Equations (10) and (11) represent the ground truth box and default bounding box, respectively. During the training process, the default bounding boxes are matched to the ground truth boxes. If two default boxes with T72 and BMP2 have been matched, then they will be treated as positives while the rest are treated as negatives.

2.3. Before and After Operation for DCNN

Before DCNN, a fast sliding method is proposed to partition the large image into sub-images to keep the information integrity. After DCNN, NMSS is performed to eliminate false alarms.

2.3.1. Fast Sliding

In conventional cases, images are resized so that all images have the same size. Take AlexNet for example, during the CNN training and testing stages, all images are resized to a same size of 227×227 , before being fed into the network and for feature extraction and classification, respectively. Generally, the size of a large-scene SAR image is several times larger than the resized size of the CNN. However, when resizing a SAR image with a large scene, it may suffer from substantial information loss and object distortion, which may compromise image matching between query and database images. This problem is significant for target recognition, as the object of interest may take up only a small region in the target image (however, in an image with a large size), the details can be more clearly observed; and, in the recognition stage, keeping the aspect ratio of an image will also help to preserve the shape of the objects/scene, thus making the classification more accurate [36].

In order to avoid the situation discussed above, it is necessary to partition the large-scene images into sub-images with a suitable size. During the cutting process, the target in the scene is likely to be divided into several parts, which will lead to a terrible recognition result. Therefore, it is significant to design a strategy to cut the image into a suitable size to match the input of the convolutional network and ensure that every target will exist in one sub-image completely.

In this paper, a fast sliding method is proposed to deal with the problem of partitioning the large scene into sub-images with a suitable size by sliding a rectangular window with a fixed size on the original large-scene SAR images; sliding the window on the original image in a certain step, such that the latter slice overlaps the previous slice with a certain area [37]. Assuming that the largest bounding box of the target is $w_t * h_t$, the size of the sliding window is $w_s * h_s$ and the size of the large-scene SAR image is $w_o * h_o$.

Figure 4 shows the process of fast sliding, where λ_h and λ_v denote horizontal and vertical sliding, respectively. If the sliding window slides to the edge of the image but exceeds the boundary of the image, the sliding window moves forward until the right side of the sliding window coincides with the right side of the image. As shown in Figure 5, the SAR image is divided into four parts by the method. Different colors of the rectangular boxes indicate different positions of the sliding windows, of which the target in the green box is split. However, the purple box contains the target completely.

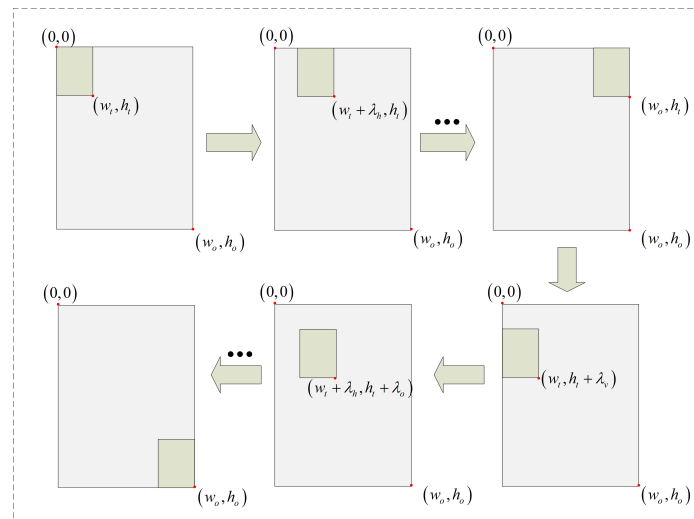


Figure 4. The process of fast sliding.

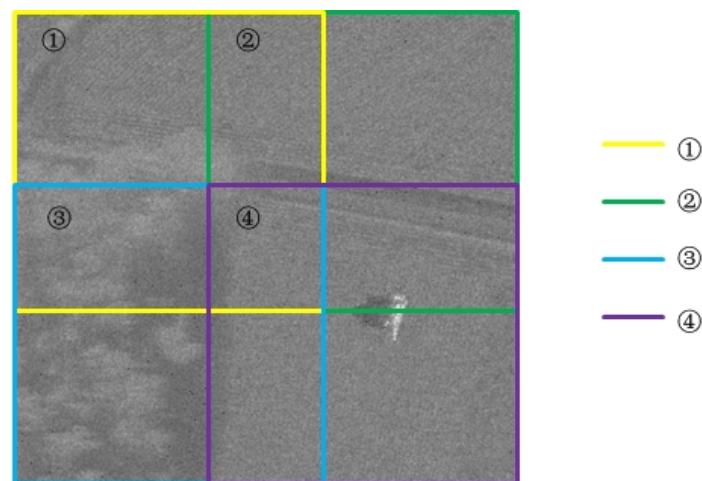


Figure 5. Sliding windows on the SAR image. It can be seen that the complete target will be in one of the sub-images.

To ensure every target in the large-scene image will exist in at least one sub-image completely, the relationship between the parameters is as shown in Equation (12). In this paper, $w_s = h_s = 258$, $\lambda_h = \lambda_v = 128$, and the overlap is set to 0.5.

$$overlap \geq \frac{\max(w_t, h_t)}{\min(w_s, h_s)}. \quad (12)$$

2.3.2. NMSS

By the method of fast sliding, the large-scene SAR image was divided into sub-images with a suitable size for the input of the network, which will then be sent into the network, sequentially, to detect and recognize targets. When the recognition result and confidence of the targets and position of bounding boxes are generated by bounding box regression and classification, the primary task in this stage is to analyze the results on the sub-images and select the appropriate results to display on the original large-scene SAR images.

As the task of object detection is to map an image to a set of boxes—a box for each object of interest in the image, with each box surrounding an object. This means that the detectors ought to return only one detect result per object. Non-maximum suppression (NMS) is a post-processing

algorithm responsible for merging all detections that belong to the same object and removing redundant detections [38]. For every sub-image, NMS has been adopted to ignore bounding boxes that significantly overlap each other. However, different sub-images may contain a same target, because there exists an overlap between some of them by the fast sliding method. If the detection results of the sub-images are directly displayed on the original picture, it may cause multiple confused detection results for some targets in the figure.

To solve this problem, non-maximum suppression between sub-images (NMSS) is proposed in this paper. The specific process of this method is as follows:

- Step 1: Coordinate transformation, mapping the coordinates of the sub-images to the original image;
- Step 2: Retain the bounding box with highest category confidence for the current target;
- Step 3: Retain the bounding boxes which are independent in the image;
- Step 4: Calculate the intersection over union (IoU) between the rest of the boxes with the box from Step 2, and delete the bounding boxes which have an IoU exceeding the set threshold;
- Step 5: Continue to choose a box from the category with highest confidence from the unprocessed box and repeat Steps 1 and 2; and
- Step 6: Repeat the previous four steps, until the N bounding boxes with a highest category confidence of the targets are found.

3. Experiments

3.1. Dataset Generation

In this paper, the training dataset and test dataset are generated from the MSTAR dataset, provided by the Air Force Research Laboratory and the Defence Advanced Research Projects Agency (AFRL/DARPA) [4]. The dataset serves as a standard data set for the research of SAR ATR. The sensor that collected the dataset is a spotlight SAR, with a high resolution of 0.3×0.3 in both range and azimuth. There are thousands of SAR images, including ten categories of ground military vehicles (armored personnel carrier: BMP2, BRDM2, BTR60, and BTR70; tank: T62 and T72; rocket launcher: 2S1; air defense unit: ZSU234; truck: ZIL131; and bulldozer: D7), which are publicly released. Examples of SAR images of ten types of targets at similar aspect angles and their corresponding optical images are depicted in Figure 6.

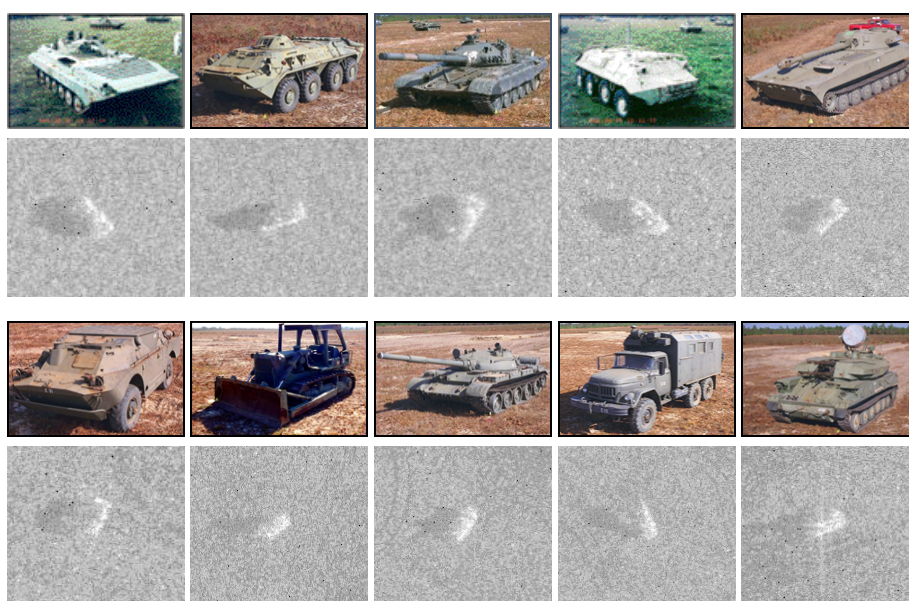


Figure 6. Types of military targets: (top) optical images versus (bottom) SAR images. According to the order: BMP2, BTR70, T72, BTR60, 2S1, BRDM2, D7, T62, ZIL131, and ZSU234.

The serial number, depression angle, and number of images available for training and testing are listed in Table 2. Images for training are acquired at a 17° depression angle, and images for testing are captured at 15° . In this paper, for a three-type target detection and recognition problem, three categories (armored personnel carrier: BMP2 and BTR70, and tank: T72) are adopted to train and test our method. For ten-type target detection and recognition problem, all types of targets in Table 2 are used for generating the training and testing dataset.

Table 2. Sample number of training and testing sets from the MSTAR dataset.

Targets	Train			Test		
	Serial No.	No. Images	Depression	Serial No.	No. Images	Depression
BMP2	9563,9566,c21	698	17°	9563,9566,c21	587	15°
BTR70	c71	233	17°	c71	196	15°
T72	132,812,s7	691	17°	132,812,s7	582	15°
BTR60	k10yt7532	256	17°	k10yt7532	195	15°
2S1	b01	299	17°	b01	274	15°
BRDM2	E71	299	17°	E71	298	15°
D7	92v13015	299	17°	92v13015	274	15°
T62	A51	299	17°	A51	273	15°
ZIL131	E12	299	17°	E12	274	15°
ZSU234	d08	299	17°	d08	274	15°

As the cost of acquiring SAR images, including ground vehicle targets in large scenes, is expensive, it is essential to adopt the large scenes and target images provided in the MSTAR dataset to generate the large-scene SAR images containing targets for research. The MSTAR dataset provides thousands of scene images without targets. Therefore, we embed many targets from the 128×128 image chips into the large scene image. This operation is reasonable, because both the targets and the scene image are captured by the same spotlight SAR with the same resolution of 0.3×0.3 . In this paper, several large-scene SAR images were made for our experiments. In Figure 7, a composite SAR image with a large scene with 15 targets randomly distributed on it is shown, and the target category and its corresponding number is shown in Table 3.

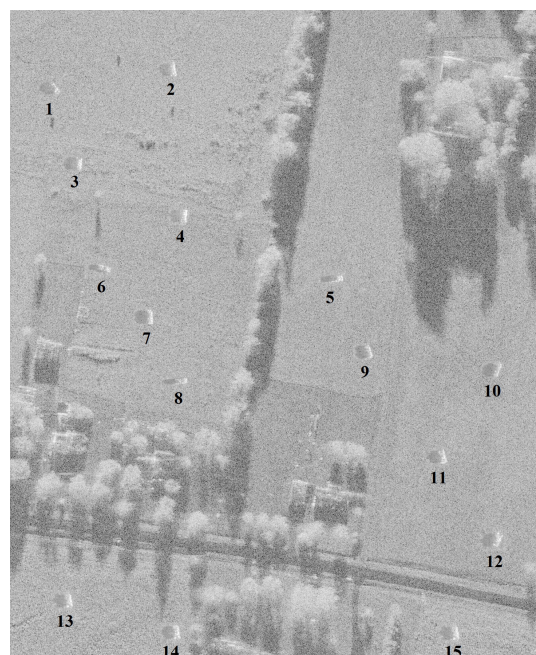


Figure 7. Composite large-scene SAR image with 15 targets.

Table 3. Targets in large scene SAR image, with their corresponding number.

Targets	Corresponding Number
BMP2	1,2,3,4,5
BTR70	6,7,8,9,10
T72	11,12,13,14,15

3.2. Accuracy of Detection and Recognition

For evaluating the performance of D-ATR, which integrates the traditional four steps of SAR ATR as a whole system, we implemented this method to solve the three-type and the ten-type target detection and recognition problems, respectively. In the three-type target problem, the three types of targets included BMP2, BTR70, and T72, and the number of images available for training and testing are listed in Table 2. As the number of available images for training is limited, some data augmentation methods, such as horizontal flip and random crop, are used in this paper. As the size of the test samples was 128×128 , which was suitable to input into the network directly, fast sliding and NMSS were not used in this part of the experiment. As shown in Figure 8, it shows a detection and recognition result of three types of targets. Every target in each chip is surrounded by box and its category with high confidence. The confusion matrix for the three-type task is shown in Table 4, and the confusion matrix for the ten-type task is shown in Table 5. The true target types are listed on the left and predicted target types are shown on top.

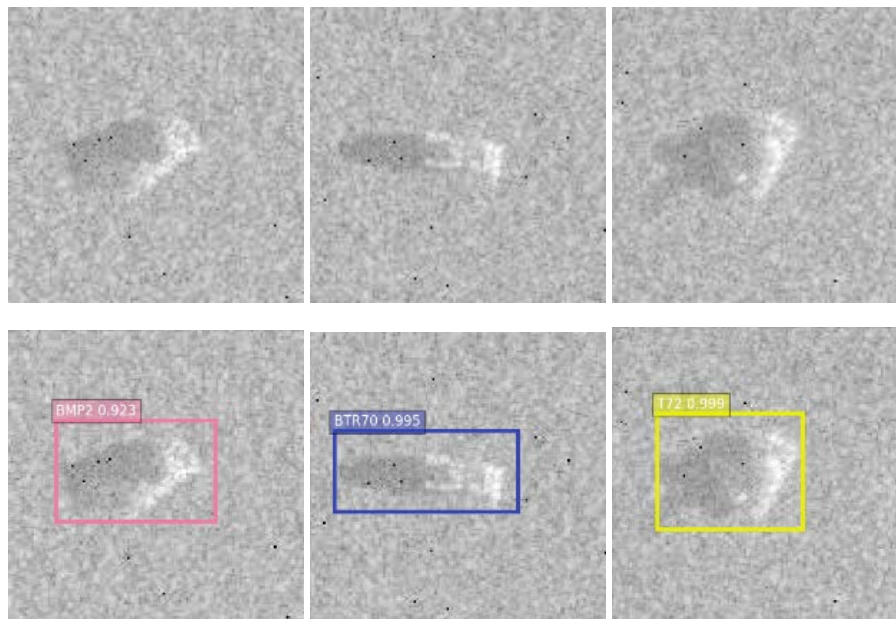


Figure 8. The result of D-ATR for target chips (bottom) and the original images, according to the order: BMP2, BTR70, T72.

Table 4. Confusion matrix for the three-type task.

Class	BMP2	BTR70	T72	Accuracy
BMP2	578	0	9	98.4%
BTR70	0	196	0	100%
T72	3	0	579	99.4%
Average	-	-	-	99.3%

Table 5. Confusion matrix for the ten-type task.

Class	BMP2	BTR70	T72	BTR60	2S1	BRDM2	D7	T62	ZIL131	ZSU234	Accuracy
BMP2	582	2	3	0	0	0	0	0	0	0	99%
BTR70	0	196	30	0	0	0	0	0	0	0	100%
T72	1	0	581	0	0	0	0	0	0	0	99.8%
BTR60	0	0	0	185	2	2	1	1	1	3	94.8%
2S1	0	0	0	1	208	0	0	46	19	0	75.9%
BRDM2	0	0	0	0	0	270	0	0	4	0	98.5%
D7	0	0	0	0	0	0	274	0	0	0	100%
T62	0	0	0	0	0	0	0	266	0	7	97.4%
ZIL131	0	0	0	0	0	0	1	0	273	0	99.6%
ZSU234	0	0	0	0	0	0	0	0	0	273	100%
Average	-	-	-	-	-	-	-	-	-	-	96.5%

3.3. Performance on Large Scene SAR Images

To detect and recognize targets in a SAR image with large scene, first of all, a SAR ATR system would have to detect potential targets and isolate the regions out from a complex background, such as river, sea surface and forest. Then those isolated image chips are fed to a classifier and ultimately declare the recognized target type. For the purpose of presenting such a case, [22] used two stages network, with the first one performing binary classification, i.e., detection, and the second performing recognition.

Most methods for SAR interpretation use strategy of segmentation to deal with large scene SAR images, however, many detection and recognition methods are sensitive to segmentation results, thus easily leading to a worse result.

As for our method, every target in the scene only need to be guaranteed to appear completely on no less than one sub-image. In this paper, D-ATR system is proposed for the sake of realizing the SAR ATR system for large scene SAR images, which integrate the traditional four steps, i.e., detection, discrimination, feature extraction and classification as a whole system.

In this part, several large scene SAR images were simulated from the publicly available MSTAR dataset to show the feasibility and performance of D-ATR system. As shown in Figure 9, there are 21 targets distributing in the scene randomly. These ten types of targets are surrounded with boxes in different colors, such as target surrounded with yellow is ZIL131. This 1478×1784 image is first cut into a series of 256×256 sub-images, and then these sub-images will be input to the network sequently in order to detect and recognize the potential targets.

When the NMSS is not used, the result is shown in Figure 9a. For the purpose of suppressing the redundant bounding box and retaining the suitable box with highest predict confidence for every target, NMSS is used in this part, and the result is shown in Figure 9b.

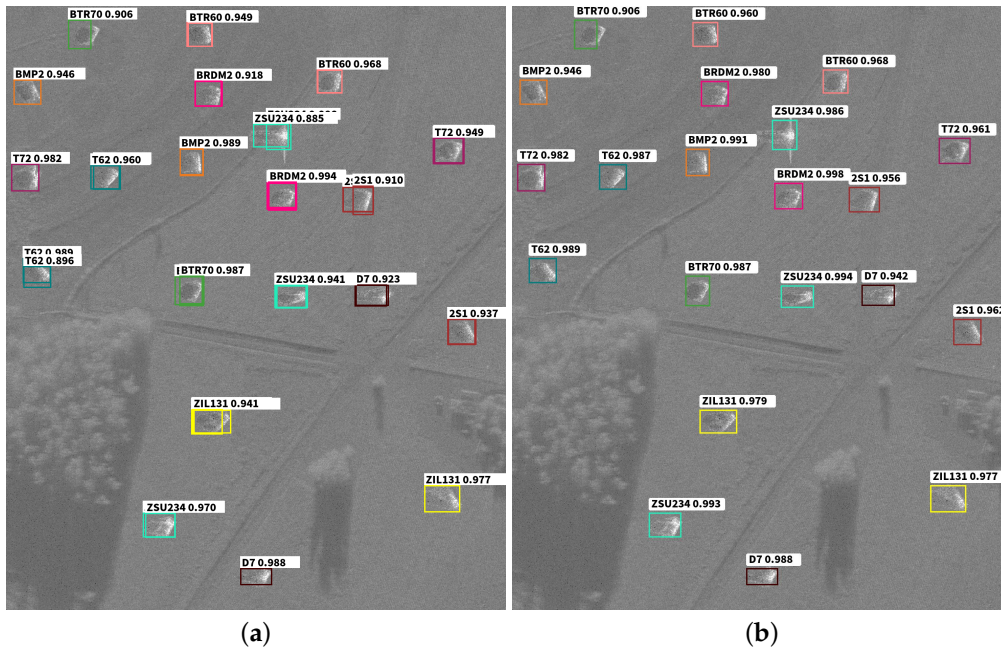


Figure 9. The detection and recognition results on large scene SAR images: (a) without NMSS; (b) with NMSS.

3.4. Comparison Experiments

To verify the feasibility and efficiency of the proposed method, several comparison experiments were conducted. As shown in Table 6, four different methods (i.e., CFAR+SVM [39], Region-based CNN [27], YOLOv2 [40], and D-ATR) were performed on one simple and one complex large-scene SAR image, respectively.

There are six parameters listed in Table 6: number of targets (No.Target) in the SAR image, number of correctly detected targets (No.Det), the proportion of targets that are correctly detected in all targets (Det Rate), number of correctly recognized targets (No. Rec), the proportion of targets that are correctly recognized in all detected targets (Rec Rate), and time consumption.

Table 6. Comparison of different methods for Figure 9 with a simple scene and Figure 7 with a complex scene.

SAR Image	Method	No. Target	No. Det	Det Rate	No. Rec	Rec Rate	Time (s)
Figure 10a	CFAR+SVM	15	14	93%	13	92.86%	20.9
	Region-based CNN	15	15	100%	15	100%	23.1
	YOLO-2	15	11	73.33%	11	100%	1.5
	Proposed method	15	15	100%	15	100%	5.8
Figure 10b	CFAR+SVM	15	9	60%	8	88.9%	21.3
	Region-based CNN	15	15	100%	15	100%	23.3
	YOLO-2	15	10	66.67%	10	100%	1.6
	Proposed method	15	15	100%	15	100%	6.8

4. Discussion

4.1. Analysis on Detection and Recognition Accuracy

As shown in Table 4, the detection and recognition result on the three-type task is inspiring. However, the result of BMP2 and T72 were a little bit worse than BTR70, and it seems that several

slices of BMP2 and T72 were inaccurately recognized as each other. The reason may be that the two kinds of targets have a similar turret and gun barrel, which makes them easily confused.

As shown in Table 5, the average accuracy of the detection and recognition results on the ten-type task is 96.5%. The accuracy of most types is higher than 94%.

Actually, when interpreting all the 1365 128×128 SAR image chips, it costed 13 seconds in total, which shows a faster speed than the method that Wang [26] proposed to realize the integration of detection and recognition in the field of SAR ATR.

4.2. Analysis on Performance of Large-Scene SAR Images

From Figure 9a, it can be seen that all of the 21 targets are surrounded by several boxes. The reason for this situation is that the method of fast sliding makes it possible for each target to appear on multiple sub-images.

From Figure 9b, it can be seen that the results of the SAR image with a relatively simple scene is exciting, with all targets correctly recognized and each of the 21 targets covered by only one rectangular box, which means the rest of the predicted boxes, with a lower confidence category, are deleted.

Comparing Figures 9a,b, it can be seen that the proposed NMSS can solve the multiple confused detection problem effectively.

Finally, to show the performance of the D-ATR system on SAR images with more complex scenes, we test our model on Figure 7, in which there are more trees and bushes and 15 targets randomly embedded in the scene. The category and the corresponding number of each target is shown in Figure 7.

As shown in Figures 10a,b, each of the 15 targets is surrounded by a bounding box with high prediction confidence, and the result illustrates that the proposed D-ATR system that realizes integration of target detection and recognition performs well. No trees or bushes are interpreted as targets, which has proved the effectiveness and usefulness of features extracted by DCNN. There is only one surrounding box on each target on this large-scene SAR image; many of these targets appeared on more than two sub-images. Thus, it was proved that NMSS is a useful strategy in dealing with prediction boxes among adjacent sub-images.

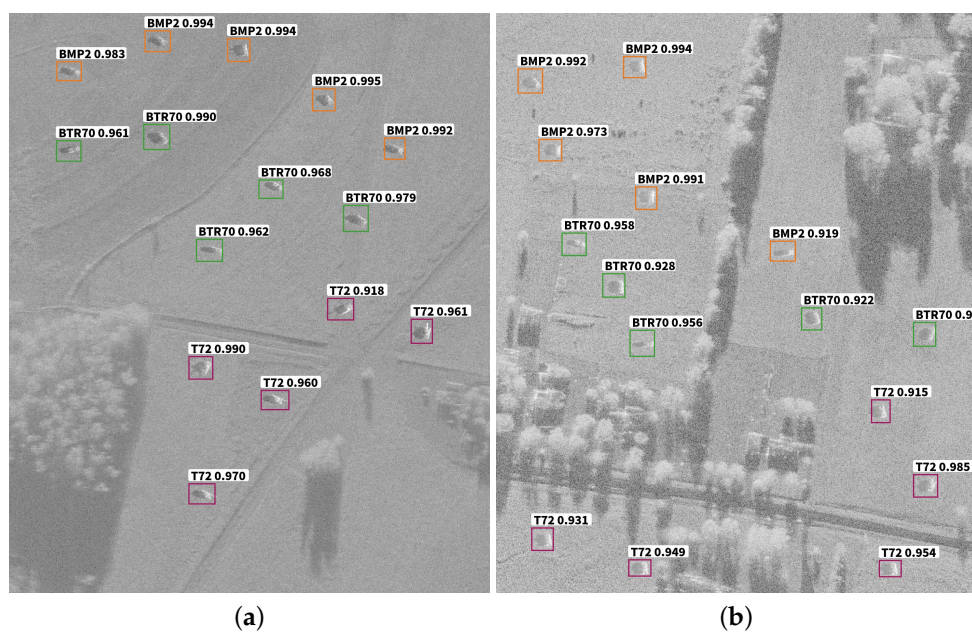


Figure 10. The detection and recognition results on large-scene SAR images: (a) simple scene; (b) complex scene.

4.3. Analysis on Comparison Experiments

As shown in Table 6, it can be seen that the result of detection and recognition rate for targets in large-scene SAR image by D-ATR was 100%, and with a relatively low time consumption.

For comparison, the proposed method outperforms CFAR+SVM, not only in the accuracy of detection and recognition, but also in time consumption. As for the comparison between the proposed method and Region-based CNN, it is obvious that our proposed method had the same accuracy as the Region-based CNN when detection and recognition for SAR images with large scene. However, the proposed D-ATR consumed 17.3 s and 16.5 s less than Region-based CNN on large-scene SAR images with simple scenes and complex scenes, respectively.

Additionally, it can be seen, from Table 6 that the time consumption of YOLO-2 was 1.5–1.6 s, which was the lowest. However, some targets were missing by YOLO-2, with a detection rate less than 73.33%. A possible reason is that YOLO-2 divides the images into many 7×7 or 13×13 sub-images, which may cause the target to be partitioned into several parts. When recognizing, these parts cannot be recognized as a target, which will cause targets to be missing.

In a word, the proposed D-ATR can detect and recognize all targets in a large-scene SAR image accurately, and performed better than other methods listed in the table.

5. Conclusions

The traditional SAR ATR mainly includes four steps: detection, discrimination, feature extraction, and target recognition/classification. However, these processes are independent, and the processing result from each step will affect the following one. Inspired by the recent success of deep learning methods in optical image processing, these problems in SAR ATR can be solved by encapsulating all computation into a single deep convolutional neural network. Whenever a large-scene SAR image is directly input into the network, it may suffer from substantial information loss and object distortion when resizing. The fast sliding method is proposed to cut a large image into a series of sub-images, which can guarantee that every target will be contained in one of the chips completely. NMSS is proposed to retain the best bounding box with the highest confidence for every target. Experimental results on simulated large-scene SAR images (with size 1478×1784) show that the recognition rate can reach to 100%, with a time consumption less than 7 s.

Author Contributions: Methodology, Z.C. and C.T.; Validation, Z.C. and N.L. All authors contributed to analysing experimental results and writing the paper.

Funding: This work was supported by the National Natural Science Foundation of China under Grant 61801098 and by the Fundamental Research Funds for the Central Universities under Grant 2672018ZYGX2018J013.

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

References

1. El-Darymli, K. Automatic Target Recognition in Synthetic Aperture Radar Imagery: A State-of-the-Art Review. *IEEE Access* **2017**, *4*, 6014–6058. [[CrossRef](#)]
2. Wu, J.; Pu, W.; Huang, Y.; Yang, J.; Yang, H. Bistatic Forward-Looking SAR Focusing Using $\omega - k$ Based on Spectrum Modeling and Optimization. *IEEE J. Sel. Top. Appl. Earth Obs. Remote Sens.* **2018**, *11*, 4500–4512. [[CrossRef](#)]
3. Bhanu, B.; Jones, T.L. Image understanding research for automatic target recognition. *IEEE Aerosp. Electron. Syst. Mag.* **1993**, *8*, 15–23. [[CrossRef](#)]
4. Karine, A.; Toumi, A.; Khenchaf, A.; El Hassouni, M. Radar Target Recognition Using Salient Keypoint Descriptors and Multitask Sparse Representation. *Remote Sens.* **2018**, *10*, 843. [[CrossRef](#)]
5. Zhao, Q.; Principe, J.C. Support vector machines for SAR automatic target recognition. *IEEE Trans. Aerosp. Electron. Syst.* **2001**, *37*, 643–654. [[CrossRef](#)]

6. Novak, L.M.; Benitz, G.R.; Owirka, G.J.; Bessette, L.A. ATR performance using enhanced resolution SAR. In Proceedings of the Algorithms for Synthetic Aperture Radar Imagery III. International Society for Optics and Photonics, Orlando, FL, USA, 8–12 April 1996; Volume 2757, pp. 332–338.
7. Bhatnagar, V.; Shaw, A.K.; Williams, R.W. Improved automatic target recognition using singular value decomposition. In Proceedings of the 1998 IEEE International Conference on Acoustics, Speech and Signal Processing, Seattle, WA, USA, 15 May 1998; Volume 5, pp. 2717–2720. [[CrossRef](#)]
8. Tison, C.; Pourthie, N.; Souyris, J.C. Target recognition in SAR images with Support Vector Machines (SVM). In Proceedings of the 2007 IEEE International Geoscience and Remote Sensing Symposium, Barcelona, Spain, 23–28 July 2007; pp. 456–459. [[CrossRef](#)]
9. Kaplan, L.M. Analysis of multiplicative speckle models for template-based SAR ATR. *IEEE Trans. Aerosp. Electron. Syst.* **2001**, *37*, 1424–1432. [[CrossRef](#)]
10. Novak, L.M.; Owirka, G.J.; Brower, W.S.; Weaver, A.L. The automatic target-recognition system in SAIP. *Linc. Lab. J.* **1997**, *10*, 187–202.
11. Robey, F.C.; Fuhrmann, D.R.; Kelly, E.J.; Nitzberg, R. A CFAR adaptive matched filter detector. *IEEE Trans. Aerosp. Electron. Syst.* **1992**, *28*, 208–216. [[CrossRef](#)]
12. Clemente, C.; Pallotta, L.; Proudler, I.; De Maio, A.; Soraghan, J.J.; Farina, A. Pseudo-Zernike Based Multi-Pass Automatic Target Recognition From Multi-Channel SAR. *IET Radar Sonar Navig.* **2015**, *9*, 457–466. [[CrossRef](#)]
13. Clemente, C.; Pallotta, L.; Gaglione, D.; De Maio, A.; Soraghan, J.J. Automatic target recognition of military vehicles with Krawtchouk moments. *IEEE Trans. Aerosp. Electron. Syst.* **2017**, *53*, 493–500. [[CrossRef](#)]
14. Sun, Y.; Liu, Z.; Todorovic, S.; Li, J. Adaptive boosting for SAR automatic target recognition. *IEEE Trans. Aerosp. Electron. Syst.* **2007**, *43*, 112–125. [[CrossRef](#)]
15. Morgan, D.A. Deep convolutional neural networks for ATR from SAR imagery. In Proceedings of the Algorithms for Synthetic Aperture Radar Imagery XXII. International Society for Optics and Photonics, Baltimore, MD, USA, 20–24 April 2015; Volume 9475, p. 94750F.
16. Huang, Y.; Pei, J.; Yang, J.; Wang, B.; Liu, X. Neighborhood geometric center scaling embedding for SAR ATR. *IEEE Trans. Aerosp. Electron. Syst.* **2014**, *50*, 180–192. [[CrossRef](#)]
17. 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. [[CrossRef](#)]
18. Girshick, R. Fast R-CNN. *Comput. Sci.* **2015**, 1440–1448.
19. Ren, S.; He, K.; Girshick, R.; Sun, J. Faster R-CNN: Towards real-time object detection with region proposal networks. In Proceedings of the International Conference on Neural Information Processing Systems, Montreal, QC, Canada, 7–12 December 2015; pp. 91–99.
20. Redmon, J.; Divvala, S.; Girshick, R.; Farhadi, A. You only look once: Unified, real-time object detection. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 27–30 June 2016; pp. 779–788.
21. Liu, W.; Anguelov, D.; Erhan, D.; Szegedy, C.; Reed, S.; Fu, C.Y.; Berg, A.C. Ssd: Single shot multibox detector. In *European Conference on Computer Vision*; Springer: Berlin, Germany, 2016; pp. 21–37.
22. Chen, S.; Wang, H.; Xu, F.; Jin, Y.Q. Target classification using the deep convolutional networks for SAR images. *IEEE Trans. Geosci. Remote Sens.* **2016**, *54*, 4806–4817. [[CrossRef](#)]
23. Kechagias-Stamatis, O.; Aouf, N. Fusing Deep Learning and Sparse Coding for SAR ATR. *IEEE Trans. Aerosp. Electron. Syst.* **2018**. [[CrossRef](#)]
24. Amrani, M.; Jiang, F.; Xu, Y.; Liu, S.; Zhang, S. SAR-Oriented Visual Saliency Model and Directed Acyclic Graph Support Vector Metric Based Target Classification. *IEEE J. Sel. Top. Appl. Earth Obs. Remote Sens.* **2018**, *11*, 3794–3810. [[CrossRef](#)]
25. Wang, Z.; Du, L.; Zhang, P.; Li, L.; Wang, F.; Xu, S.; Su, H. Visual Attention-Based Target Detection and Discrimination for High-Resolution SAR Images in Complex Scenes. *IEEE Trans. Geosci. Remote Sens.* **2018**, *56*, 1855–1872. [[CrossRef](#)]
26. Wang, S.; Cui, Z.; Cao, Z. Target recognition in large scene SAR images based on region proposal regression. In Proceedings of the 2017 IEEE International Geoscience and Remote Sensing Symposium (IGARSS), Fort Worth, TX, USA, 23–28 July 2017; pp. 3297–3300.

27. Cui, Z.; Dang, S.; Cao, Z.; Wang, S.; Liu, N. SAR Target Recognition in Large Scene Images via Region-Based Convolutional Neural Networks. *Remote Sens.* **2018**, *10*, 776. [[CrossRef](#)]
28. Krizhevsky, A.; Sutskever, I.; Hinton, G.E. ImageNet classification with deep convolutional neural networks. In Proceedings of the International Conference on Neural Information Processing Systems, Lake Tahoe, NV, USA, 3–6 December 2012; pp. 1097–1105.
29. Simonyan, K.; Zisserman, A. Very Deep Convolutional Networks for Large-Scale Image Recognition. *Comput. Sci.* **2014**, arXiv:1409.
30. Szegedy, C.; Liu, W.; Jia, Y.; Sermanet, P.; Reed, S.; Anguelov, D.; Erhan, D.; Vanhoucke, V.; Rabinovich, A. Going deeper with convolutions. In Proceedings of the 2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Boston, MA, USA, 7–12 June 2015; pp. 1–9. [[CrossRef](#)]
31. He, K.; Zhang, X.; Ren, S.; Sun, J. Deep Residual Learning for Image Recognition. *Comput. Vis. Pattern Recognit.* **2016**, 770–778, arXiv:1512.03385.
32. Huang, Z.; Pan, Z.; Lei, B. Transfer Learning with Deep Convolutional Neural Network for SAR Target Classification with Limited Labeled Data. *Remote Sens.* **2017**, *9*, 907. [[CrossRef](#)]
33. Kang, C.; He, C. SAR image classification based on the multi-layer network and transfer learning of mid-level representations. In Proceedings of the 2016 IEEE International Geoscience and Remote Sensing Symposium, Beijing, China, 10–15 July 2016.
34. Russakovsky, O.; Deng, J.; Su, H.; Krause, J.; Satheesh, S.; Ma, S.; Huang, Z.; Karpathy, A.; Khosla, A.; Bernstein, M. ImageNet Large Scale Visual Recognition Challenge. *Int. J. Comput. Vis.* **2014**, *115*, 211–252. [[CrossRef](#)]
35. Zeiler, M.D.; Fergus, R. *Visualizing and Understanding Convolutional Networks*; Springer: Cham, Switzerland, 2014; pp. 818–833.
36. Zheng, L.; Zhao, Y.; Wang, S.; Wang, J.; Tian, Q. Good Practice in CNN Feature Transfer. *arXiv* **2016**, arXiv:1604.00133.
37. Zhang, T.; Liang, J.; Yang, Y.; Cui, G.; Kong, L.; Yang, X. Antenna Deployment Method for Multistatic Radar under the Situation of Multiple Regions for Interference. *Signal Process.* **2018**, *143*, 292–297. [[CrossRef](#)]
38. Hosang, J.; Benenson, R.; Schiele, B. Learning Non-maximum Suppression. In Proceedings of the 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Honolulu, HI, USA, 21–26 July 2017; pp. 6469–6477. [[CrossRef](#)]
39. An, W.; Xie, C.; Yuan, X. An Improved Iterative Censoring Scheme for CFAR Ship Detection With SAR Imagery. *IEEE Trans. Geosci. Remote Sens.* **2014**, *52*, 4585–4595.
40. Redmon, J.; Farhadi, A. YOLO9000: Better, Faster, Stronger. In Proceedings of the IEEE Conference on Computer Vision & Pattern Recognition (CVPR), Honolulu, HI, USA, 21–26 July 2017; pp. 7263–7271.



© 2019 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 (<http://creativecommons.org/licenses/by/4.0/>).