

# Ship Classification in SAR Images Using a New Hybrid CNN–MLP Classifier

Foroogh Sharifzadeh<sup>1</sup> · Gholamreza Akbarizadeh<sup>1</sup>  · Yousef Seifi Kaviani<sup>1</sup>

Received: 30 August 2018 / Accepted: 23 October 2018 / Published online: 29 October 2018  
© Indian Society of Remote Sensing 2018

## Abstract

Ship detection on the SAR images for marine monitoring has a wide usage. SAR technology helps us to have a better monitoring over intended sections, without considering atmospheric conditions, or image shooting time. In recent years, with advancements in convolutional neural network (CNN), which is one of the well-known ways of deep learning, using image deep features has increased. Recently, usage of CNN for SAR image segmentation has been increased. Existence of clutter edge, multiple interfering targets, speckle and sea-level clutters makes false alarms and false detections on detector algorithms. In this letter, constant false alarm rate is used for object recognition. This algorithm, processes the image pixel by pixel, and based on statistical information of its neighbor pixels, detects the targeted pixels. Afterward, a neural network with hybrid algorithm of CNN and multilayer perceptron (CNN–MLP) is suggested for image classification. In this proposal, the algorithm is trained with real SAR images from Sentinel-1 and RADARSAT-2 satellites, and has a better performance on object classification than state of the art.

**Keywords** SAR image processing · Synthetic aperture radar (SAR) · Classification · Ship classification · Neural network · Convolution neural network (CNN) · Multilayer perceptron (MLP) · Hybrid CNN–MLP

## Introduction

Target classification using images taken from different sensors is one of the problems of radar systems known as automatic target recognition (ATR) (Srinivas et al. 2014). Among all the targets, the man-made targets are of particular importance; in recent years, ship monitoring has been very much considered (Lang et al. 2016). Because of night-and-day imaging and the lack of sensitivity to weather conditions, SAR images are very suitable for vessel monitoring.

CFAR detector is suggested with fixed false alarm rate, with environment conditions like clutter edge or multiple interfering targets (Frost et al. 2015; Gao et al. 2009). In a simpler explanation about this algorithm, it compares the threshold of the cell under test, and fixates the probability

of false rate alarm. Accuracy of this method mostly depends on the pixels around then cell under test; because in SAR images, often target is in a complex background. Therefore, statistical modeling of neighbor pixels is very important. Gaussian distribution is often used for modeling of neighbor pixels; but it's functionality in high-resolution SAR images has been confronted with limitations. Some examples of these limitations are noise speckle existed on images and also nonhomogenous clutter of the sea surface on the sea SAR images (Lombardo and Sciotti 2001; Modava and Akbarizadeh 2017). Other models that are suggested for distribution of probability on neighbor pixels are Weibull model and  $K$  distribution model (Jakeman and Pusey 1976).  $K$  distribution is one of the most suited models on high-resolution SAR images for sea level (Watts 1987; Yueh et al. 1989; Weiss 1982). On high-resolution images, CFAR algorithm functionality for SAR image process is discussable. In high-resolution images, the functionality of the CFAR algorithm for pixel-by-pixel SAR image processing is very complex and time-consuming. In the CFAR algorithm, the modeling of the neighboring pixels of the cell under test is used to

---

✉ Gholamreza Akbarizadeh  
g.akbari@scu.ac.ir

<sup>1</sup> Department of Electrical Engineering, Faculty of Engineering, Shahid Chamran University of Ahvaz, Ahvaz, Iran

determine the CFAR type. Regarding the performance of the CFAR algorithm, it is clear that the presence of unwanted elements in the SAR image increases the probability of false alarms. Therefore, eliminating unwanted items in the background can be effective in reducing false alarms. One of the most effective ways to remove these unwanted items is truncated statistics (Tao et al. 2016a). In this method, the data are eliminated based on the probability of having useful information. In simpler explanation, data that are less likely to have information will be deleted. Therefore, in this paper, the CFAR algorithm is used by reduced data using the TS algorithm. In the following, another approach is proposed to reduce the size of the SAR images

After the initial processing of the image and applying the CFAR algorithm, the objectives identified are passed to a hybrid neural network (CNN–MLP) for classification. The convolutional neural network (CNN) is one of the deep learning methods that has been widely used in the image processing, especially optics images. But there are fewer works in the field of SAR imagery. Recent computational developments in deep learning and optimization calculations have caused deep networks trained with a more efficient method (Bengio 2009; Hinton et al. 2006), which has led to increased use of deep neural networks (DNNs). This type of neural network has been developed in many fields, including object classifications (Chen et al. 2016; Tang et al. 2015), image recognition (Zhao and Du 2016; Girshick et al. 2014) and robotics (Bezak et al. 2014). DNNs are able to process raw data without the need for manual extraction of features (Lv et al. 2015). The convolutional neural network is one of the kinds of deep neural network that has a content-based function. This type of neural network has great results in computer vision and pattern recognition (Schmidhuber 2015) and image recovery (Yang et al. 2015). Bentes et al (2018) used a convolutional neural network for classifying ships in SAR images. Convolutional neural networks, having received images as inputs, do not need to extract features for network training and have better performance compared to other neural networks for classifying images (Ding et al. 2016; Bentes et al. 2015, 2016; Ampe et al. 2012; Hou et al. 2016).

Multilayer perceptron is a shallow structure neural network, which is trained based on the features of the image. Neural networks with shallow structure architectural design have been used extensively in SAR programs (Wu et al. 2015; Makedonas et al. 2015). Zhang et al. (2017) proposed a neural network combining a convolutional neural network and multilayer perceptron for remote sensing images. In a multilayered perceptron neural network, to train the network, it is necessary to extract the features of the images. Image texture features have good performance

for segmentation (Akbarizadeh and Tirandaz 2015), recognition (Tan and Triggs 2010) and classification (Wang et al. 2018). In this study, we used feature properties with GLCM (gray-level co-occurrence matrix) to train neural network. GLCM is a common technique, with its main features of average, variance, contrast, homogeneity, etc (Geng et al. 2015).

In this paper, 9576 medium-resolution images were extracted from the data obtained from the SAR images taken by the RADARSAT-2 and Sentinel-1 satellites for the training of the convolutional neural network (Schwegmann et al. 2017). These data are divided into three classes, ship\_positives, sea and other things (true\_negatives) and similar items (false\_positives). The first class belongs to pictures of the ships; in this dataset, 17% of the pictures are in this class. The second class specifically includes images of the surface of the ocean and all items that are not ships; this class also includes 17% of the data. In the third class, there are data that are false to be a ship, such as the shadow of the ships, rocks. This class has the most members of the collection and contains 66% of the data. The data in this dataset are matched with the automatic identification system (AIS) sender. AIS is an automatic position reporting system for locating coordinates for ships and other maritime vehicles (Mazzarella et al. 2015). Matching data from SAR images and AIS data, the result is more reliable. In this paper, a hybrid neural network is used to classify SAR images, and the neural network can compensate for the neural network defects in the detection of the edges. Here, GLCM texture features of all images are extracted, and 60% of the data is randomly used to train multilayer perceptron. To test the neural network, 20% of the data is used. Twenty percent of the data is not used in the MLP, and these data are used for the convolutional neural network as validation data. In the following, the proposed hybrid neural network is examined. This output network examines both networks described, and, based on the output vector, determines the reliability of the output. Then, it introduces the output with more accuracy as the network output. In this way, this network uses the positive features of both networks simultaneously, and improves the results of the two previous networks.

The general ideas of this paper are summarized in the following: 1. Using the TS volume reduction method, which is suggested in this paper, data that are statistically less valuable are excluded. This will reduce the computational volume in the CFAR algorithm and significantly improves the processing speed. 2. The detection of images using the CFAR algorithm, which uses the slider window technique, is used to create an adaptive threshold. 3. Using the hybrid neural network proposed in this paper for SAR images, the accuracy of classification has been increased, because in this classification, in addition to the benefits of

CNN as a deep neural network in extracting the features of the image internally, which has led to features that should be hand fed to older neural network, the MLP trained by the texture features is used too. The result is chosen from the output of the two networks discussed in this article, which makes the result the most reliable answer possible.

## Methodology

The general work of paper is shown in Fig. 1. First, in “MLP Multilayer Perceptron” section, a short version of the MLP method is given, and then in “CNN Convolutional Neural Network” section, the CNN method will be described. “Preprocessing” section describes the operation of the first block as preprocessor; then, in “Detector” section, how the CFAR detection algorithm is used. In “Classification” section, which is the last part of this paper, the operations of the CNN convolutional neural network and the MLP multilayer perceptron are examined separately, and then the operation of the CNN–MLP hybrid algorithm is described.

### MLP Multilayer Perceptron

Multilayer perceptron is one of the classical types of neural networks, in which each set of input outputs particular vectors. The MLP multilayered perceptron structure

consists of three layers, an input layer, a hidden layer, and an output layer. The input layer is represented by the following formula:

$$a^1 = x \quad (1)$$

which indicates  $x$  as the input and  $a^1$  as the first layer of the network; and the input of each layer is the weighted output of the previous layer (Pacifi et al. 2009) and corresponds to:

$$a^{(l+1)} = \sigma(w^{(l)}a^{(l)} + b^{(l)}) \quad (2)$$

where  $l$  is a specific layer,  $w^{(l)}$  indicates the layer weight, and  $b^{(l)}$  as the bias symbol in the layer  $l$ ,  $\sigma$  also indicates a nonlinear function, which is used in this network. This function can be a sigmoid function, a hyperbolic tangent, etc. The output layer is also shown below:

$$h_{w,b}(x) = a^{(n)} \quad (3)$$

Here,  $n$  is the number of network layers,  $w$  as weights, and  $b$  as bias. The objective function is a function that minimizes the difference between output and desired output:

$$J(w, b; x, y) = \frac{1}{2} \|h_{(w,b)}(x) - y\|^2 \quad (4)$$

### CNN Convolutional Neural Network

Convolutional neural networks are designed with the ability to extract image features for image processing, which is why their use in image classification is increasing. Convolutional neural networks contain three main layers, 1. Convolutional layers 2. Pooling layer 3. Output layers. In this type of neural network, after each convolutional layer, a pooling layer is enclosed (Romero et al. 2016), and the final convolutional layer is connected to the output layer (LeCun et al. 2015).

Different layers have different tasks. There are two stages of training in each convolutional neural network. Feed forward and back propagation or post-back phases. In the first step, the image of the input is given to the network, and this is in fact the multiplication of the point between the input and the parameters of each neuron, and finally the application of convolutional operations in each layer (Arel et al. 2010). Then, the network output is calculated; to set the network parameters or, in other words, to train the network, the output is used to calculate the network false rate. In the next step, based on the false rate, the back propagation operation begins.

In this step, the parameters change based on their effect on the network false rate. After the new parameters become available, the feed forward stage starts. After completing enough number of these steps, the network training will

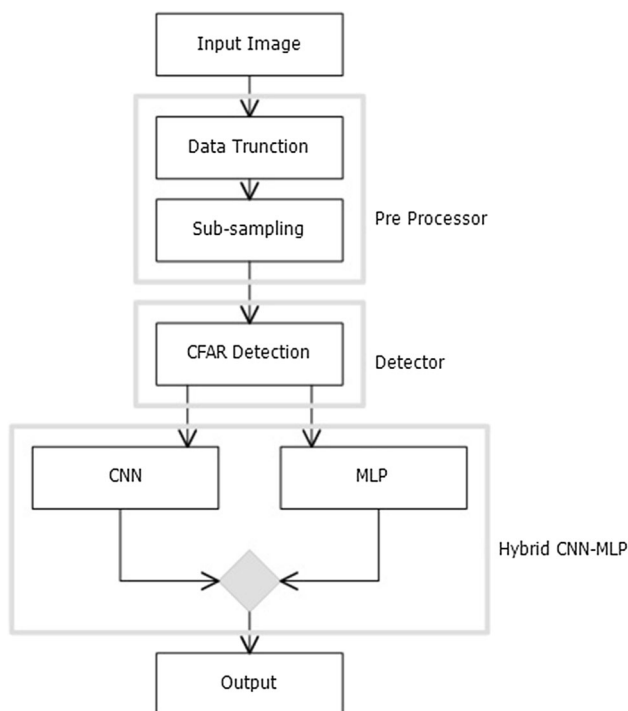


Fig. 1 Workflow of hybrid CNN–MLP classifier

end. In a convolutional neural network, the convolutional layers and pooling layers are one in between, and in the end there are several layers with a fully connected connection.

## Preprocessing

In this part of the work, it has been tried to preview the image in such a way that applying the following algorithms to the images becomes simpler and reducing the volume of computing as well as the processing time of the image. Because due to the high volumes of SAR images and pixel-by-pixel processing of CFAR algorithm, the detection of the detector will be very timely if it is not preprocessed. In addition, the presence of heterogeneous sea-level clutter is considered as a complication to obtain various features of the SAR image of the sea (Lombardo and Sciotti 2001). Also, changing the posterior aspect of the image due to ocean and ocean phenomena will create edge clutter. The existence of intrusive goals also leads to inaccuracies in the estimation of the parameters and causes the image modeling to be inadequate (Tao et al. 2016a). These cases increase the number of false alarms in the CFAR algorithm and weaken its performance (Lombardo and Sciotti 2001). As a result, excluding data that are not useful information can be helpful in solving these problems.

This paper uses a truncated statistics method to exclude data that are not useful information (Tao et al. 2016a). In this method, the truncation ratio  $R_t$ , which represents the ratio of the excluded data to the total number of data, is used. The value  $R_t$  is obtained by the total amount of samples  $N_{ROI}$ , the size of the largest experimental object  $N_{target}$  and the number of targets in the image  $C$ , as follows:

$$\frac{N_{ROI} - (C \times N_{target})}{N_{ROI}} > R_t \geq \frac{C \times N_{target}}{N_{ROI}} \quad (5)$$

As seen in formula (5), it is better to have a large value of  $R_t$  that is true in the formula above, and also its value should not be below the percentage of the data that contain the information. It should be noted that the values given for the number of data in the image  $N_{ROI}$ , the size of the largest object  $N_{target}$  and the number of targets in the image  $C$  are all experimental values that will be different for each input image (Tao et al. 2016b), so in this article the value is considered  $R_t = 10\%$ , which has been surveyed for more than 1000 input data, and it is assumed that it is true for other input images as well.

After excluding the data in the above manner, another method is also used to reduce the image volume of the input image to preprocess the image. In the proposed method, by subsampling the input image, its volume is reduced. The volume of the image determines the  $S_{th}$  method (Doulgeris et al. 2011; Doulgeris 2015). Indeed, this criterion implies that one from each of the  $S_{th}$

prototypes remains in the image. Therefore, the image size after the subsampling operations will be equal to:

$$\text{New image size} = \frac{\text{Image size}}{S_{th}^2} \quad (6)$$

## Detector

A CFAR detector is used for detection of the image in this paper. This method estimates a threshold value for specifying the target from the background. Clearly, the greater the difference between the target gray and the background level, the simpler threshold estimation is and better target detection is performed. It should be noted that if a constant threshold is used for the whole image, one cannot expect a satisfactory result, and therefore, to obtain more accurate results, a CFAR detector with a dynamic threshold should be used. To determine the amount of dynamic thresholds, it is necessary to identify the post-graduate statistical model. With the amount of false rates and a review of the SAR model, the appropriate value for the threshold is obtained.

False rates as  $P_f$  and detection rates as  $P_d$  are shown in Eqs. (7) and (8):

$$P_f = \int_T^{\infty} P_B(x) dx \quad (7)$$

$$P_d = \int_0^T P_T(x) dx \quad (8)$$

The threshold  $T$  value is obtained from the solution of the following equation.

$$1 - P_f = \int_0^T P_B(x) dx \quad (9)$$

In this equation,  $P_f$  is the probability of a false alert and  $T$  is obtained by varying its value. If  $I$  is the image as input, the detection is performed according to the following formula:

$$I(i,j) = \begin{cases} 1; I(i,j) \geq T \\ 0; I(i,j) < T \end{cases} \quad (10)$$

To achieve the adaptive threshold, the sliding window technique is used (Leng et al. 2015; Hwang and Ouchi 2010). By setting a window of the desired size and scrolling the whole image, it introduces a local threshold for each scroll, thus eliminating the problems from the constant threshold.

## Classification

Three types of neural networks are used in this paper. Multilayer perceptron neural network MLP, CNN torsion neural network and CNN–MLP neural network are proposed in this paper.

### Multilayer Perceptron Functionality

Multilayer perceptron requires training input vectors as DATA and expected output vectors corresponding to it. In a multilayer perceptron, you cannot use the image itself as an input, and it is necessary to extract the features of the image first, and then give to network as input. In this case, the GLCM technique is used to extract the texture features of the image. As a result, vectors containing textural features are used as inputs for the MLP, and these vectors contain 19 feature of each image in the data.

Another vector is also given for network training, which is in fact the label or class of each image, and is given by the number of classes and the number of inputs to the network input. In this vector, for each image in the input vector, the corresponding row corresponds to the class array in Fig. 1.

After training the network, it is time to test it. To test the network, it is also necessary to give the image or test images as test input to the network, to determine the output vector of the test image class.

The resulting output is a vector with 3 members (the output vector is as large as the class specified in the network), the number in each array of this vector specifies the probability that the test image will be affixed to each of the classes.

### Convolutional Neural Network Functionality

To train convolutional neural network, it is not necessary to extract image features because CNNs are able to extract the image features themselves. So, in this work, 60% of the images in the data are randomly assigned directly to the CNN input. It is also possible to input the labels of the images with their own inputs to the network.

Neural network input image dimensions are  $(128 \times 128 \times 1)$  and are in three classes, ship\_positives as ship images, true\_negatives as images that are not a ship, false\_positives as images that appear to be false as a ship.

CNN-1 (Chen et al. 2016) and CNN-2 (Bentes et al. 2018) and CNN-3 (Wilmanski et al. 2016) networks have been introduced for comparison with the proposed hybrid algorithm, also learning rate = 0.001 and the training stops are adjusted by validation data. The architecture of these networks is shown in Fig. 2.

Here, the CNN-1 neural network (Chen et al. 2016) is tested for combining with the MLP multilayer perceptron; the CNN output can be both a vector, and label of the image that has been tested.

### Hybrid CNN–MLP Functionality

In the hybrid network, it is attempted to take advantage of the multilayer perceptron MLP and the CNN convolutional neural network simultaneously. To this end, both networks need to be trained with the same data and the test image should be given to the input of both networks. Then, the output of both networks is extracted; now, the output that is more valid needs to be introduced as a hybrid network output.

Here, the output of each network is represented in the form of  $Y = \{y_1.y_2.\dots.y_n\}$  in which  $n$  denotes the number of classes.  $i \in [1.n]$  The number  $y_i$  is in fact the probability that an image in the class  $i$  can be, which can be a number between zero and one. Classification models represent the highest probability of membership as predicted outcomes.

Here, a standard is given for the level of output reliability, which is in fact the same standard deviation in the output vector:

$$\sigma = \max(Y) - \text{mean}(Y) \quad (11)$$

$\max(Y)$  which indicates the largest number of set of  $Y$  or actually the highest probability of membership and the  $\text{mean}(Y)$  is value of average of  $Y$ . The criterion  $\sigma$  determines the confidence of the class existence in an image. If the input image has a homogeneous background, the diagnosis of the image class in the CNN will be more reliable and  $\sigma$  will be a larger value, but in a heterogeneous background or in the presence of unwanted cases in the background of the image, the value of the  $\sigma$  will be less.

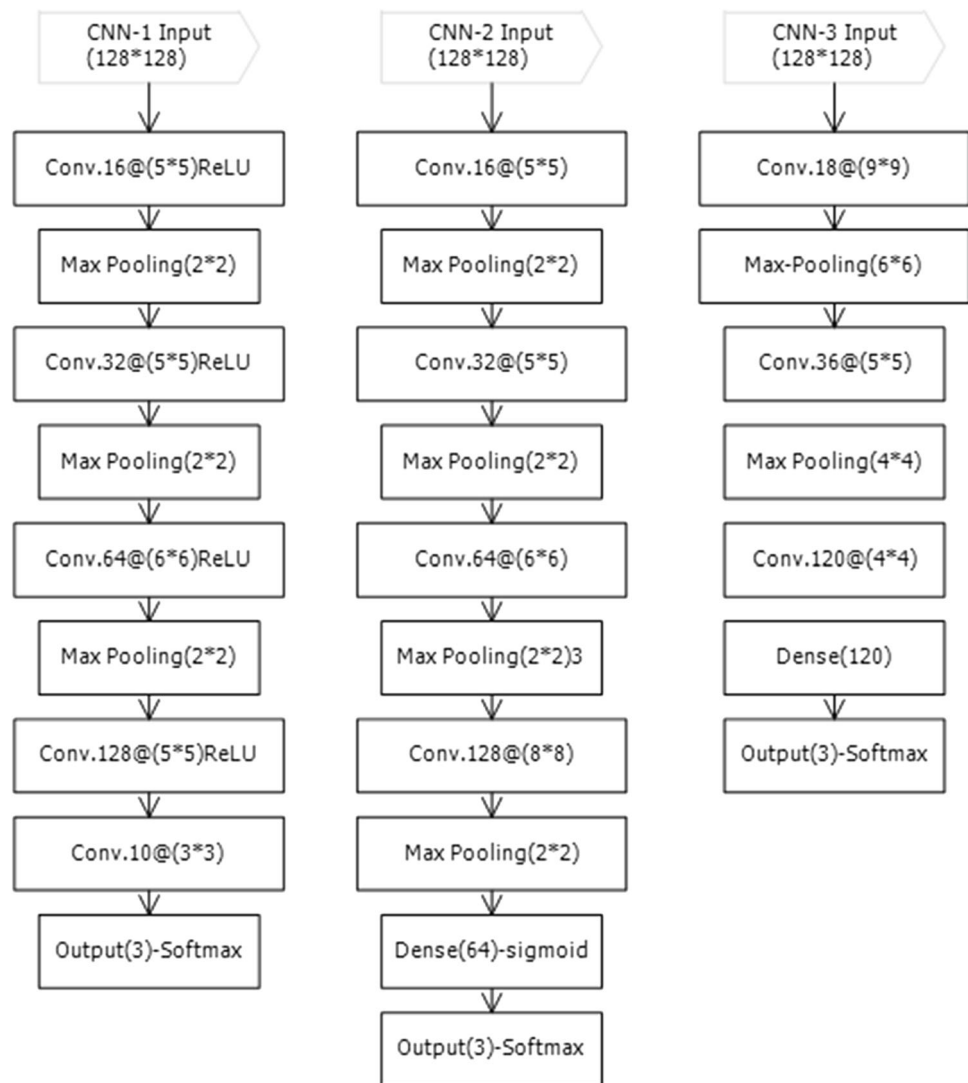
Because the CNN network has a better overall performance than MLP and is more suitable for classification of images, the value  $\sigma_{\text{CNN}}$  in most cases will be a larger number. Therefore,  $\sigma_{\text{CNN}}$  is used as a benchmark. Two thresholds  $\alpha$  and  $\beta$  are also for comparison of  $\sigma$ , which amount  $\alpha \in [0.1, 0.4]$  and amount  $\beta \in [0.6, 0.9]$  can be changed. Now, according to  $\sigma$  the output class criteria is specified as follows:

$$\text{class}_{\text{CNN-MLP}} = \begin{cases} \text{class}_{\text{CNN}} & \sigma_{\text{CNN}} > \beta \\ \text{class}_{\text{CNN}} & \alpha < \sigma_{\text{CNN}} < \beta, \sigma_{\text{CNN}} > \sigma_{\text{MLP}} \\ \text{class}_{\text{MLP}} & \alpha < \sigma_{\text{CNN}} < \beta, \sigma_{\text{CNN}} < \sigma_{\text{MLP}} \\ \text{class}_{\text{MLP}} & \sigma_{\text{CNN}} < \alpha \end{cases} \quad (12)$$

In which,  $\text{class}_{\text{MLP}}$  and  $\text{class}_{\text{CNN}}$ , respectively, are the result of the classifications derived from MLP and CNN.



**Fig. 2** The models of CNN-1 (Chen et al. 2016), CNN-2 (Bentes et al. 2018) & CNN-3 (Wilmanski et al. 2016), with their parameters



To get the appropriate value for  $\alpha$  and  $\beta$  initially  $\beta = 0.9$  and  $\alpha$  starts at the value of 0.1 and increment by 0.05, and the value for which the highest accuracy is obtained as the final number is considered. The number  $\beta$  will be obtained in the same way.

## Results

In this section, all three multilayer perceptron neural network, convolutional neural network and CNN–MLP hybrid network with actual SAR data have been tested (Schwegmann et al. 2017), and the average results are displayed for 100 trials in Table 1. The value of  $\alpha$  and  $\beta$  is obtained  $\alpha = 0.45$  and  $\beta = 0.65$  using the mentioned method. According to the information in this table, the criterion *precision* (indicating the ratio of the number of correct diagnosis of the target to the total number of goals) has

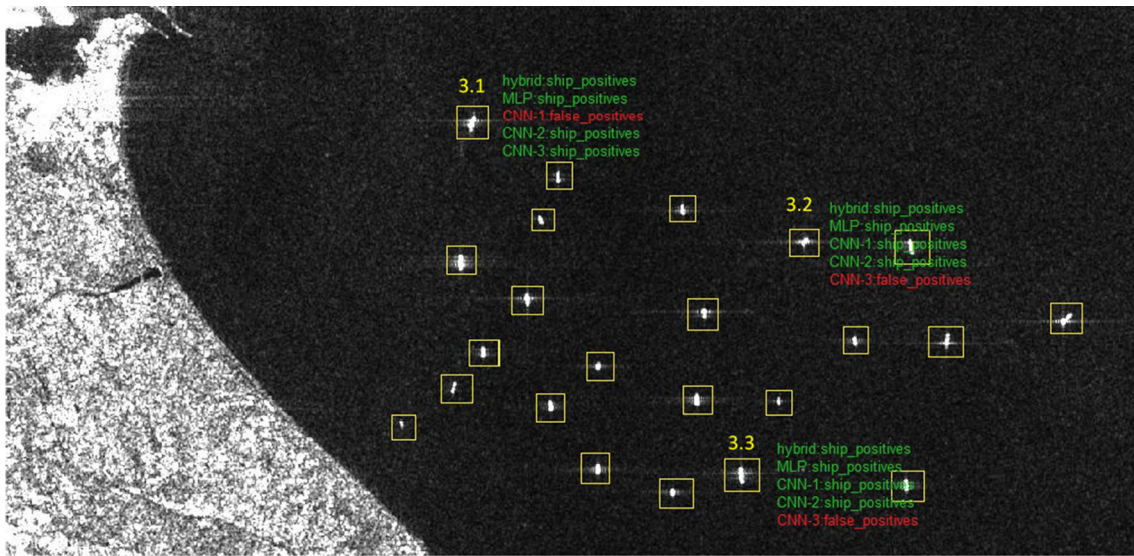
improved. The amount *recall* that also indicates the ratio of the correct target detection value to the total number of target detection (both true and false) is also better in the hybrid network than the other two networks (Tables 2, 3, 4, 5, 6, 7).

The hybrid CNN–MLP network, *f1 – score* which is a combination of two criteria *precision* and *recall* obtained by the following formula, has been shown to be better.

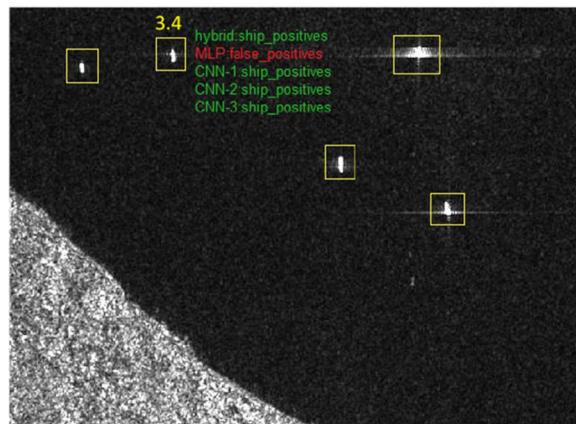
$$f1 - score = 2 \times \frac{precision \times recall}{(precision + recall)} \quad (13)$$

The numerical value *accuracy*, which is precision of the measurement, is improved compared to the other two networks.

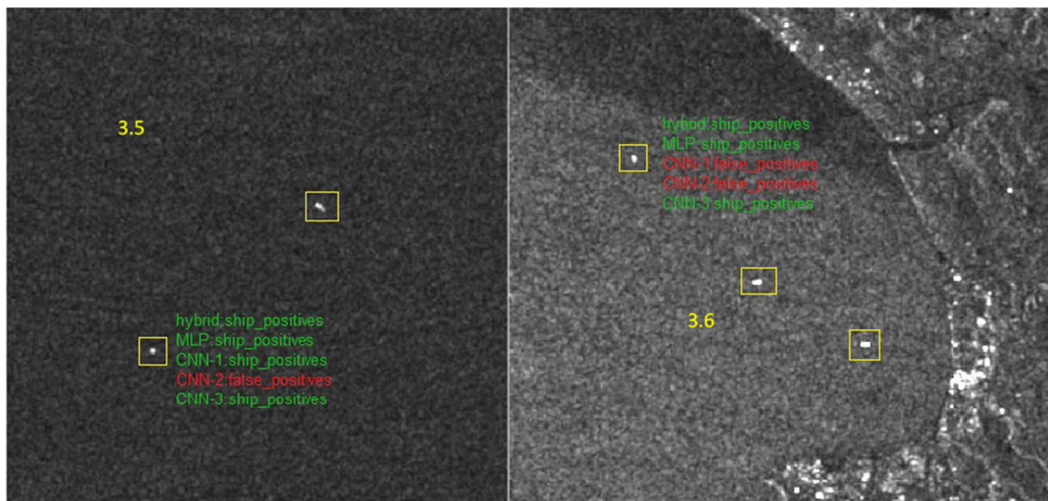
Neural networks have also been tested for the actual SAR images (3), (4) and (5). As can be seen in the image, the hybrid neural network in the targets where the detection of CNN-1 is incorrect and the MLP is correctly detected, is correct, in cases where CNN-1 detection is correct, it has



(a)



(b)



(c)

(d)

**Fig. 3** a, b, c, d Real SAR images that are given to the input of all the neural networks introduced in this article. Items that a network detected wrong, are in red, and items that detected correctly is in

green, and all other items marked with a yellow rectangle are properly detected by all networks

**Table 1** Average results for 100 trials

	<i>Precision</i>	<i>Recall</i>	<i>f1-score</i>	<i>Accuracy</i>
CNN-1 (Chen et al. 2016)	0.92	0.91	0.91	0.91
CNN-2 (Bentes et al. 2018)	0.8766	0.8464	0.8612	0.8934
CNN-3 (Wilmanski et al. 2016)	0.7577	0.8527	0.8024	0.8495
MLP	0.7563	0.8838	0.8150	0.8243
Hybrid CNN-MLP	0.9319	0.9067	0.9191	0.92

**Table 2** The probability of membership in each class for Fig. 3a, image 3.1

	CNN-1	CNN-2	CNN-3	MLP
False_positives	0.57	0.14	$3 \times 10^{-10}$	$2 \times 10^{-4}$
Ship_positives	0.46	0.86	1	0.99
True_negatives	$3 \times 10^{-8}$	$8 \times 10^{-7}$	0	$2.2 \times 10^{-6}$

been similar to that of CNN-1. The CNN-1 convolutional neural network has poor performance for targets where the boundary of objects is not well defined, while the CNN-MLP hybrid network has solved this problem using MLP results. As shown in the figure, all targets marked with a yellow rectangle are given to the input of all three neural networks; in cases where the neural network has an incorrect answer, the response is marked in red and if the correct answer is given it is marked in green. In all other

**Table 3** The probability of membership in each class for Fig. 3a, image 3.2

	CNN-1	CNN-2	CNN-3	MLP
False_positives	0.34	0.41	0.99	0.002
Ship_positives	0.66	0.58	$9.6 \times 10^{-4}$	0.99
True_negatives	$3.3 \times 10^{-8}$	$4.2 \times 10^{-7}$	0	$1.64 \times 10^{-5}$

**Table 4** The probability of membership in each class for Fig. 3a, image 3.3

	CNN-1	CNN-2	CNN-3	MLP
False_positives	0.31	0.11	0.99	$7.8 \times 10^{-4}$
Ship_positives	0.69	0.89	$8.46 \times 10^{-5}$	0.99
True_negatives	$1.29 \times 10^{-8}$	$4.6 \times 10^{-9}$	0	$3.6 \times 10^{-6}$

**Table 5** The probability of membership in each class for Fig. 3b, image 3.4

	CNN-1	CNN-2	CNN-3	MLP
False_positives	$2.2 \times 10^{-4}$	$1.6 \times 10^{-4}$	$1.1 \times 10^{-19}$	$2.9 \times 10^{-4}$
Ship_positives	0.99	0.99	1	0.99
True_negatives	$6.3 \times 10^{-12}$	$6.6 \times 10^{-14}$	0	$3.01 \times 10^{-7}$

**Table 6** The probability of membership in each class for Fig. 3c, image 3.5

	CNN-1	CNN-2	CNN-3	MLP
False_positives	0.74	0.5	$1.09 \times 10^{-7}$	0.14
Ship_positives	0.25	0.49	1	0.95
True_negatives	$3.9 \times 10^{-5}$	$1.7 \times 10^{-10}$	0	$1.2 \times 10^{-3}$

**Table 7** The probability of membership in each class for Fig. 3d, image 3.6

	CNN-1	CNN-2	CNN-3	MLP
False_positives	0.51	0.69	$2.38 \times 10^{-5}$	0.35
Ship_positives	0.49	0.31	1	0.61
True_negatives	$1.1 \times 10^{-4}$	$2.1 \times 10^{-10}$	0	$2.7 \times 10^{-5}$



cases where the answer is not mentioned, all three networks provide the correct answer.

In the following, the performance of all three networks has been studied on similar images.

In Fig. 4a, b are shown two images of the class false\_ positive which are given to the input of each of the three networks. The CNN-1 neural network places these images correctly in the false\_positives class. But the class is marked by true\_negatives multilayer perceptron, and the value of  $\sigma$  in these images is as follows:

$$\begin{aligned} \sigma_{\text{CNN-1}} &= 0.66, \sigma_{\text{MLP}} = 0.19 \xrightarrow{(12)} \text{class} \\ &= \text{class}_{\text{CNN-1}} = \text{false\_positives} \end{aligned}$$

$$\begin{aligned} \sigma_{\text{CNN-1}} &= 0.499, \sigma_{\text{MLP}} = 0.3 \xrightarrow{(12)} \text{class} \\ &= \text{class}_{\text{CNN-1}} = \text{false\_positives} \end{aligned}$$

Given the high values and placement of them in formula (12), the class specified for both images by the hybrid network of the same class is CNN-1. It can be seen that value of  $\sigma_{\text{MLP}}$  for these images is small, indicating a

low level of confidence in the MLP response to these images.

Images in Fig. 4c, d are also in the false\_positives class, the CNN-1 response to these ship\_positives images and the MLP response is false\_positive, and the value of  $\sigma$  for these images is as follows:

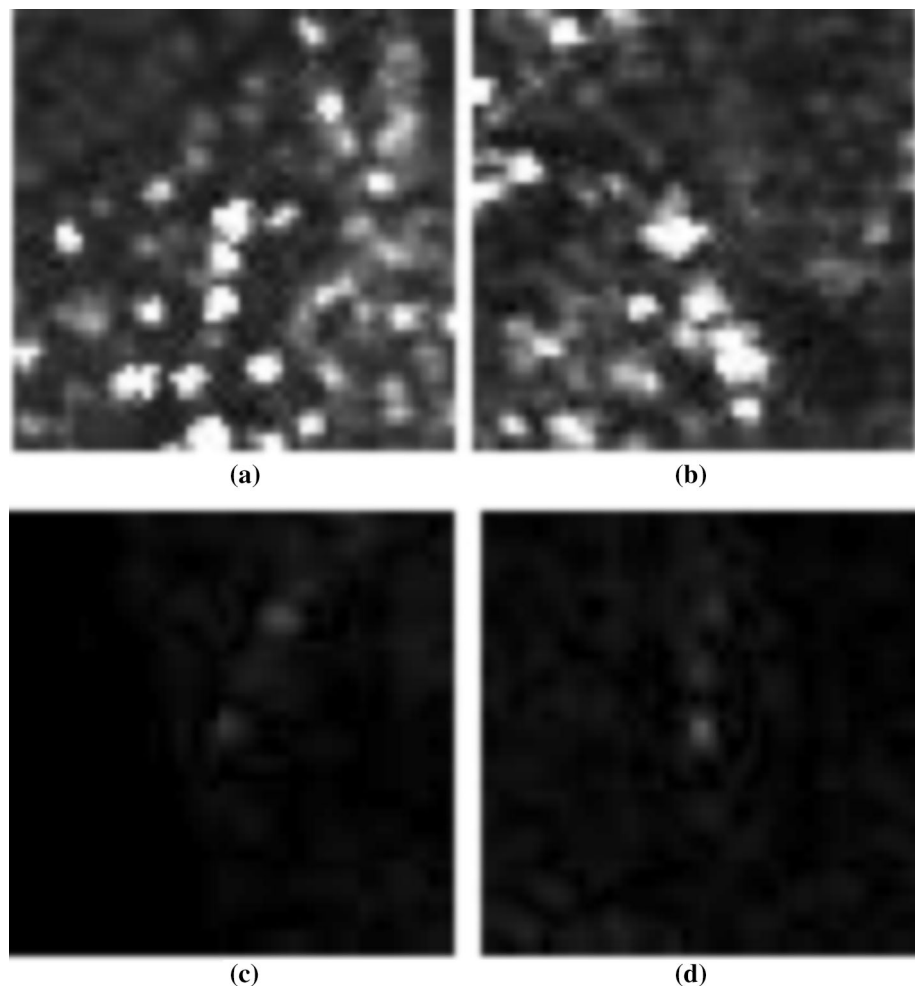
$$\begin{aligned} \sigma_{\text{CNN-1}} &= 0.56, \sigma_{\text{MLP}} = 0.66 \xrightarrow{(12)} \text{class} \\ &= \text{class}_{\text{MLP}} = \text{false\_positives} \end{aligned}$$

$$\begin{aligned} \sigma_{\text{CNN-1}} &= 0.31, \sigma_{\text{MLP}} = 0.67 \xrightarrow{(12)} \text{class} \\ &= \text{class}_{\text{MLP}} = \text{false\_positives} \end{aligned}$$

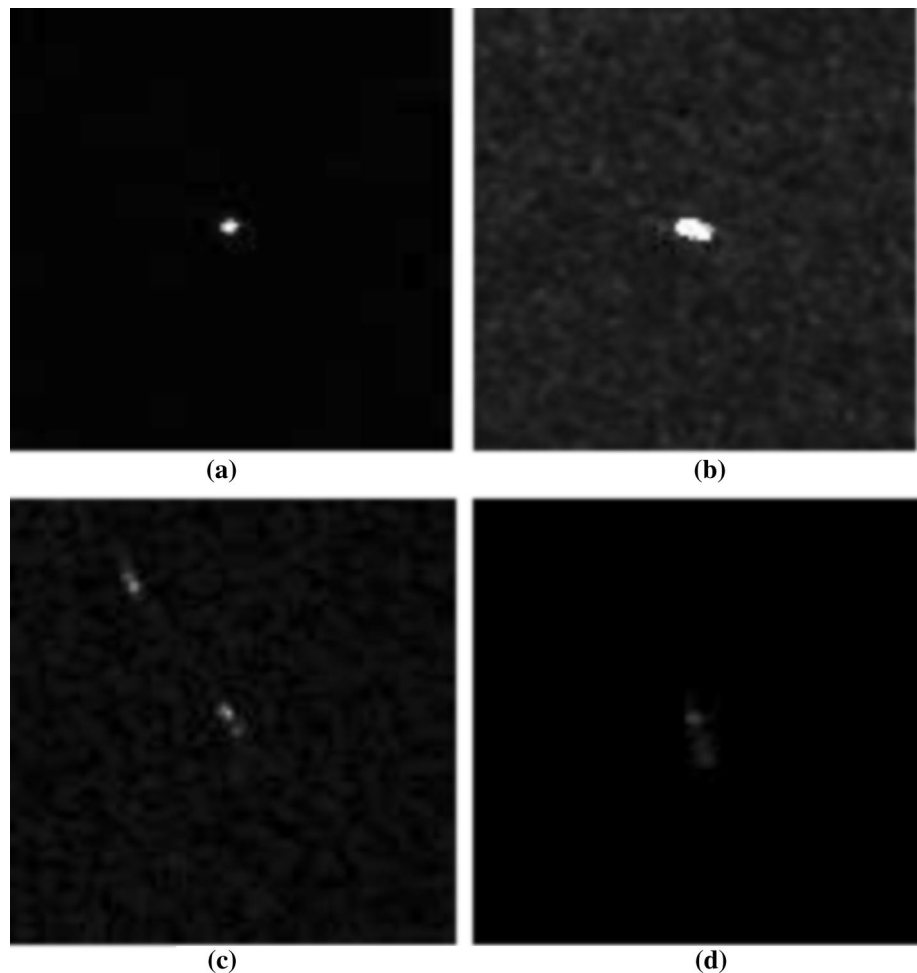
On the right side image, value of  $\sigma_{\text{CNN-1}}$  is 0.56, which is not a low value, but according to the formula (12), since  $\alpha < \sigma_{\text{CNN-1}} < \beta$ ,  $\sigma_{\text{CNN-1}} < \sigma_{\text{MLP}}$ , MLP result is considered as the output. The image on the left,  $\sigma_{\text{CNN-1}} = 0.31$ , which shows low level of confidence for CNN-1 about the class of this image.

In Fig. 5a, b are shown two images of class ship\_positives which are given as input to all three networks, CNN-1

**Fig. 4** a, b, c, d SAR images of false\_positives class that are tested by the hybrid neural network proposed in this article



**Fig. 5** **a, b, c, d** SAR images of ship\_positives class that are tested by the hybrid neural network proposed in this article



result for these images is false\_positives and MLP result is ship\_positives, and value of  $\sigma$  is as follows:

$$\begin{aligned} \sigma_{\text{CNN-1}} &= 0.49, \sigma_{\text{MLP}} = 0.65 \xrightarrow{(12)} \text{class} \\ &= \text{class}_{\text{MLP}} = \text{ship\_positives} \end{aligned}$$

$$\begin{aligned} \sigma_{\text{CNN-1}} &= 0.12, \sigma_{\text{MLP}} = 0.61 \xrightarrow{(12)} \text{class} \\ &= \text{class}_{\text{MLP}} = \text{ship\_positives} \end{aligned}$$

In Fig. 5c, d are from ship\_positives class, CNN-1 results in ship\_positives class, and MLP results in false\_positive class, and the value of  $\sigma$  is as follows:

$$\begin{aligned} \sigma_{\text{CNN-1}} &= 0.52, \sigma_{\text{MLP}} = 0.41 \xrightarrow{(12)} \text{class} \\ &= \text{class}_{\text{CNN-1}} = \text{ship\_positives} \end{aligned}$$

$$\begin{aligned} \sigma_{\text{CNN-1}} &= 0.65, \sigma_{\text{MLP}} = 0.67 \xrightarrow{(12)} \text{class} \\ &= \text{class}_{\text{CNN-1}} = \text{ship\_positives} \end{aligned}$$

According to the above calculations on the right image  $\sigma_{\text{CNN-1}} = 0.65$ ,  $\sigma_{\text{MLP}} = 0.67$ , both CNN-1 and MLP networks have a high confidence level, but given the

CNN-1 preference, the output response is the same as the CNN-1 response.

## Conclusion

In this paper, a method of working from two neural networks, a CNN-1 convolutional network and a multilayer MLP perceptron, was presented, and then a hybrid method of these two networks to take advantage of the positive characteristics of each of them has been suggested. With respect to the accuracy parameters, it is clear that the multilayered perceptron has a weaker performance than CNN-1; however, when it is hybrid with CNN-1, it will improve the accuracy parameters. The convolutional neural network also does not work well in cases where the boundary of the object is not well defined; in which case, the hybrid of CNN with multilayer perceptron based on the characteristics of the trained image has solved this problem.

**Acknowledgements** Funding was provided by Shahid Chamran University of Ahvaz (Grant No. 96/3/02/16670).

## References

- Akbarizadeh, G., & Tirandaz, Z. (2015). Segmentation parameter estimation algorithm based on curvelet transform coefficients energy for feature extraction and texture description of SAR images. In *2015 7th conference on information and knowledge technology (IKT)* (pp. 1–4).
- Ampe, E. M., Vanhamel, I., Salvatore, E., Dams, J., Bashir, I., Demarchi, L., et al. (2012). Impact of urban land-cover classification on groundwater recharge uncertainty. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, 5(6), 1859–1867.
- Arel, I., Rose, D. C., & Karnowski, T. P. (2010). Deep machine learning—A new frontier in artificial intelligence research. *IEEE Computational Intelligence Magazine*, 5(4), 13–18.
- Bengio, Y. (2009). Learning deep architectures for AI. *Foundations and trends® in Machine Learning*, 2(1), 1–127.
- Bentes, C., Frost, A., Velotto, D., & Tings, B. (2016). Ship-iceberg discrimination with convolutional neural networks in high resolution SAR images. In *Proceedings of EUSAR 2016: 11th European conference on synthetic aperture radar* (pp. 1–4).
- Bentes, C., Velotto, D., & Lehner, S. (2015). Target classification in oceanographic SAR images with deep neural networks: Architecture and initial results. In *2015 IEEE International on geoscience and remote sensing symposium (IGARSS)* (pp. 3703–3706).
- Bentes, C., Velotto, D., & Tings, B. (2018). Ship classification in terrasars-x images with convolutional neural networks. *IEEE Journal of Oceanic Engineering*, 43(1), 258–266.
- Bezak, P., Bozek, P., & Nikitin, Y. (2014). Advanced robotic grasping system using deep learning. *Procedia Engineering*, 96, 10–20.
- Chen, S., Wang, H., Xu, F., & Jin, Y.-Q. (2016). Target classification using the deep convolutional networks for SAR images. *IEEE Transactions on Geoscience and Remote Sensing*, 54(8), 4806–4817.
- Ding, J., Chen, B., Liu, H., & Huang, M. (2016). Convolutional neural network with data augmentation for SAR target recognition. *IEEE Geoscience and Remote Sensing Letters*, 13(3), 364–368.
- Doulgeris, A. P. (2015). An automatic U-distribution and Markov random field segmentation algorithm for PolSAR images. *IEEE Transactions on Geoscience and Remote Sensing*, 53(4), 1819–1827.
- Doulgeris, A. P., Anfinsen, S. N., & Eltoft, T. (2011). Automated non-Gaussian clustering of polarimetric synthetic aperture radar images. *IEEE Transactions on Geoscience and Remote Sensing*, 49(10), 3665–3676.
- Frost, A., Ressel, R., & Lehner, S. (2015). Iceberg detection over northern latitudes using high resolution TerraSAR-X images. In *36th Canadian symposium of remote sensing-abstracts* (pp. 1–8).
- Gao, G., Liu, L., Zhao, L., Shi, G., & Kuang, G. (2009). An adaptive and fast CFAR algorithm based on automatic censoring for target detection in high-resolution SAR images. *IEEE Transactions on Geoscience and Remote Sensing*, 47(6), 1685–1697.
- Geng, J., Fan, J., Wang, H., Ma, X., Li, B., & Chen, F. (2015). High-resolution SAR image classification via deep convolutional autoencoders. *IEEE Geoscience and Remote Sensing Letters*, 12(11), 2351–2355.
- Girshick, R., Donahue, J., Darrell, T., & Malik, J. (2014). Rich feature hierarchies for accurate object detection and semantic segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 580–587).
- Hinton, G. E., Osindero, S., & Teh, Y.-W. (2006). A fast learning algorithm for deep belief nets. *Neural Computation*, 18(7), 1527–1554.
- Hou, B., Kou, H., & Jiao, L. (2016). Classification of polarimetric SAR images using multilayer autoencoders and superpixels. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, 9(7), 3072–3081.
- Hwang, S.-I., & Ouchi, K. (2010). On a novel approach using MLCC and CFAR for the improvement of ship detection by synthetic aperture radar. *IEEE Geoscience and Remote Sensing Letters*, 7(2), 391–395.
- Jakeman, E., & Pusey, P. (1976). A model for non-Rayleigh sea echo. *IEEE Transactions on Antennas and Propagation*, 24(6), 806–814.
- Lang, H., Zhang, J., Zhang, X., & Meng, J. (2016). Ship classification in SAR image by joint feature and classifier selection. *IEEE Geoscience and Remote Sensing Letters*, 13(2), 212–216.
- LeCun, Y., Bengio, Y., & Hinton, G. (2015). Deep learning. *Nature*, 521(7553), 436.
- Leng, X., Ji, K., Yang, K., & Zou, H. (2015). A bilateral CFAR algorithm for ship detection in SAR images. *IEEE Geoscience and Remote Sensing Letters*, 12(7), 1536–1540.
- Lombardo, P., & Sciotti, M. (2001). Segmentation-based technique for ship detection in SAR images. *IEEE Proceedings—Radar, Sonar and Navigation*, 148(3), 147–159.
- Lv, Q., Dou, Y., Niu, X., Xu, J., Xu, J., & Xia, F. (2015). Urban land use and land cover classification using remotely sensed SAR data through deep belief networks. *Journal of Sensors*, 2015, 1–10.
- Makedonas, A., Theoharatos, C., Tsagaris, V., Anastasopoulos, V., & Costicoglou, S. (2015). Vessel classification in Cosmo-SkyMed SAR data using hierarchical feature selection. *The International Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences*, 40, 975.
- Mazzarella, F., Vespe, M., & Santamaria, C. (2015). SAR ship detection and self-reporting data fusion based on traffic knowledge. *IEEE Geoscience and Remote Sensing Letters*, 12(8), 1685–1689.
- Modava, M., & Akbarizadeh, G. (2017). Coastline extraction from SAR images using spatial fuzzy clustering and the active contour method. *International Journal of Remote Sensing*, 38(2), 355–370.
- Pacifici, F., Chini, M., & Emery, W. J. (2009). A neural network approach using multi-scale textural metrics from very high-resolution panchromatic imagery for urban land-use classification. *Remote Sensing of Environment*, 113(6), 1276–1292.
- Romero, A., Gatta, C., & Camps-Valls, G. (2016). Unsupervised deep feature extraction for remote sensing image classification. *IEEE Transactions on Geoscience and Remote Sensing*, 54(3), 1349–1362.
- Schmidhuber, J. (2015). Deep learning in neural networks: An overview. *Neural Networks*, 61, 85–117.
- Schwegmann, C. P., Kleynhans, W., Salmon, B. P., Mdakane, L. W., & Meyer, R. G. V. (2017). A SAR ship dataset for detection, discrimination and analysis. *IEEE Dataport*. <https://doi.org/10.21227/H2RK82>.
- Srinivas, U., Monga, V., & Raj, R. G. (2014). SAR automatic target recognition using discriminative graphical models. *IEEE Transactions on Aerospace and Electronic Systems*, 50(1), 591–606.
- Tan, X., & Triggs, B. (2010). Enhanced local texture feature sets for face recognition under difficult lighting conditions. *IEEE Transactions on Image Processing*, 19(6), 1635–1650.
- Tang, J., Deng, C., Huang, G.-B., & Zhao, B. (2015). Compressed-domain ship detection on spaceborne optical image using deep

- neural network and extreme learning machine. *IEEE Transactions on Geoscience and Remote Sensing*, 53(3), 1174–1185.
- Tao, D., Anfinsen, S. N., & Brekke, C. (2016a). Robust CFAR detector based on truncated statistics in multiple-target situations. *IEEE Transactions on Geoscience and Remote Sensing*, 54(1), 117–134.
- Tao, D., Doulgeris, A. P., & Brekke, C. (2016b). A segmentation-based CFAR detection algorithm using truncated statistics. *IEEE Transactions on Geoscience and Remote Sensing*, 54(5), 2887–2898.
- Wang, M., Fei, X., Zhang, Y., Chen, Z., Wang, X., Tsou, J. Y., et al. (2018). Assessing texture features to classify coastal wetland vegetation from high spatial resolution imagery using completed local binary patterns (CLBP). *Remote Sensing*, 10(5), 778.
- Watts, S. (1987). Radar detection prediction in  $K$ -distributed sea clutter and thermal noise. *IEEE Transactions on Aerospace and Electronic Systems*, 23, 40–45.
- Weiss, M. (1982). Analysis of some modified cell-averaging CFAR processors in multiple-target situations. *IEEE Transactions on Aerospace and Electronic Systems*, 18, 102–114.
- Wilmanski, M., Kreucher, C., & Lauer, J. (2016). Modern approaches in deep learning for SAR ATR. In *Algorithms for synthetic aperture radar imagery XXIII* (p. 98430 N).
- Wu, F., Wang, C., Jiang, S., Zhang, H., & Zhang, B. (2015). Classification of vessels in single-pol COSMO-SkyMed images based on statistical and structural features. *Remote Sensing*, 7(5), 5511–5533.
- Yang, X., Qian, X., & Mei, T. (2015). Learning salient visual word for scalable mobile image retrieval. *Pattern Recognition*, 48(10), 3093–3101.
- Yueh, S. H., Kong, J. A., Jao, J. K., Shin, R. T., & Novak, L. M. (1989).  $K$ -distribution and polarimetric terrain radar clutter. *Journal of Electromagnetic Waves and Applications*, 3(8), 747–768.
- Zhang, C., Pan, X., Li, H., Gardiner, A., Sargent, I., Hare, J., et al. (2017). A hybrid MLP-CNN classifier for very fine resolution remotely sensed image classification. *ISPRS Journal of Photogrammetry and Remote Sensing*, 140, 133–144.
- Zhao, W., & Du, S. (2016). Spectral–spatial feature extraction for hyperspectral image classification: A dimension reduction and deep learning approach. *IEEE Transactions on Geoscience and Remote Sensing*, 54(8), 4544–4554.