Compressed-Domain Ship Detection on Spaceborne Optical Image Using Deep Neural Network and Extreme Learning Machine

Jiexiong Tang, *Student Member, IEEE*, Chenwei Deng, *Member, IEEE*,
Guang-Bin Huang, *Senior Member, IEEE*, and Baojun Zhao

***Abstract*—Ship detection on spaceborne images has attracted great interest in the applications of maritime security and traffic control. Optical images stand out from other remote sensing images in object detection due to their higher resolution and more visualized contents. However, most of the popular techniques for ship detection from optical spaceborne images have two shortcomings: 1) Compared with infrared and synthetic aperture radar images, their results are affected by weather conditions, like clouds and ocean waves, and 2) the higher resolution results in larger data volume, which makes processing more difficult. Most of the previous works mainly focus on solving the first problem by improving segmentation or classification with complicated algorithms. These methods face difficulty in efficiently balancing performance and complexity. In this paper, we propose a ship detection approach to solving the aforementioned two issues using wavelet coefficients extracted from JPEG2000 compressed domain combined with deep neural network (DNN) and extreme learning machine (ELM). Compressed domain is adopted for fast ship candidate extraction, DNN is exploited for high-level feature representation and classification, and ELM is used for efficient feature pooling and decision making. Extensive experiments demonstrate that, in comparison with the existing relevant state-of-the-art approaches, the proposed method requires less detection time and achieves higher detection accuracy.**

***Index Terms*—Compressed domain, deep neural network (DNN), extreme learning machine (ELM), JPEG2000, optical spaceborne image, remote sensing, ship detection.**

I. Introduction

SHIP detection in spaceborne remote sensing images is of vital importance for maritime security and other applications, e.g., traffic surveillance, protection against illegal fisheries, oil discharge control, and sea pollution monitoring [1]. Vessel monitoring from satellite images provides a wide visual field and covers large sea area and thus achieves a continuous monitoring of vessels' locations and movements [2]. It is also known that optical spaceborne images have higher resolution and more visualized contents than other remote sensing images, which is more suitable for ship detection or recognition in the aforementioned applications.

However, optical spaceborne images usually suffer from two main issues: 1) weather conditions like clouds, mists, and ocean waves result in more pseudotargets for ship detection, and 2) optical spaceborne images with higher resolution naturally lead to larger data quantity than other remote sensing images, and thus, optical spaceborne images are more difficult to be tackled for real-time applications.

Previous works have established some basic ship detection frameworks. Lure and Rau [3] and Weiss *et al.* [4] proposed hybrid detection systems based on ship tracks in Advanced Very High Radiometer (AVHR) imagery. The works in [5]–[7] presented several approaches in ship detection and recognition from airborne infrared images with sky–sea backgrounds. Burgess [8] introduced vessels detecting algorithm in Satellite Pour l'Observation de la Terre (SPOT) Multispectral and Landsat Thematic Mapper images. Also, Wu *et al.* [9] analyzed the characteristics of different satellite remote sensing images. These works cannot solve the first issue well even though they have low complexity.

On the other hand, some other methods achieve better ship detection performance with the price of high computational complexity. Corbane *et al.* used a neural network to classify small ship candidates from SPOT-5 High Resolution Geometric 5-m images [10] and presented a complete processing chain for ship detection [2]. Morphological filtering is combined with wavelet analysis and radon transform to better distinguish ships from surrounding turbulence. Bi *et al.* [11] presented a statistical salient-region-based algorithm, and multiresource regions of interest are extracted to improve the ship–sea segmentation. Zhu *et al.* [1] adopted an ensemble learning algorithm, in which multiple high-dimensional local features (679 dimension) are extracted from potential targets using a support vector machine (SVM). Each of these techniques improves a specific procedure in either preprocessing or classification and achieves better performance than classical methods. However, the aforementioned second issue has not been fully addressed. In real-time applications with limited resources, e.g., satellite-based object detection/recognition, the performance and computational complexity should be equally taken into account.

Unlike the previous works which usually focus on one of the two issues, our approach aims to solve both issues. As for the

Manuscript received November 18, 2013; revised April 20, 2014; accepted June 29, 2014. This work was supported in part by the National Natural Science Foundation of China under Grant 61301090, by the Beijing Excellent Talent Fund under Grant 2013D009011000001, and by the Excellent Young Scholars Research Fund of Beijing Institute of Technology under Grant 2013YR0508.

J. Tang, C. Deng, and B. Zhao are with the School of Information and Electronics, Beijing Institute of Technology, Beijing 100081, China (e-mail: cwdeng@bit.edu.cn).

G.-B. Huang is with the School of Electrical and Electronic Engineering, Nanyang Technological University, Singapore 639798.

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TGRS.2014.2335751

first one, deep neural network (DNN) is applied to obtain high-level features and overcome the limitations of existing ship detection methods. According to the recent works in [12]–[18], deep architecture has multiple levels of feature representation, and the higher levels represent more abstract information. From a concept point of view, DNN trains multiple hidden layers with unsupervised initialization, and after such initialization, the entire network will be fine-tuned by a supervised back-propagation algorithm [12], [14]. Practically, DNN can be implemented with autoencoder, which reconstructs the input by using the corresponding output of the network [12]. In addition, learning sparse representations from original image pixels leads to better classification performance than using raw pixels or nonsparse representations obtained by principal component analysis. Moreover, by exploiting the sparsity of wavelet features, feature detectors localized in both time and frequency domains can be obtained [19], [20].

As for the second issue, compressed domain is adopted to improve detection efficiency and utilize sparse features from both space and frequency domains. Instead of reducing processing time by passively cutting an image into tiles or scaling to a low-resolution version, extracting wavelet features from JPEG 2000 codec can impressively increase the detection efficiency. In addition, discrete wavelet transform (DWT) has its unique advantages: multiresolution analysis and singularity detection. The human visual system detects objects over two fundamental properties: 1) observation is performed simultaneously in space and frequency and 2) images are perceived in a multiscale manner so that different elements can be focused with different scales [21]. After 2-D DWT, the original image is decomposed into a low-frequency subband (denoted as LL) and several horizontal/vertical/diagonal high-frequency subbands (denoted as LH, HL, and HH). Different subbands describe different sparse features of input images. In our work, wavelet domain sparsity is further exploited for DNN-based feature extraction.

Until recently, few works have been done on object detection in JPEG 2000 compressed domain. Xiong and Huang [22] extracted wavelet-based texture features from compressed domain. Delac *et al.* [23] analyzed image compression effects in face recognition systems and proposed a face recognition method in JPEG 2000 compressed domain with independent component analysis and principal component analysis [24]. Ni [25] developed the concept of information tree and proposed a novel tree-distance measurement for JPEG 2000 compressed-domain image retrieval. Teynor *et al.* [26] presented a nonuniform image retrieval method using color/texture features and modeling wavelet coefficient distribution with Gaussian mixture densities. Zargari *et al.* [27] exploited packet header information for JPEG 2000 image retrieval, including the number of nonzero bit planes, the number of coding passes, and the code block length.

However, the aforementioned approaches face difficulty in handling ship detection under various conditions, and particularly high-frequency subbands are not effectively utilized. In the proposed work, low-frequency subband and high-frequency subbands are exploited for feature extraction using two DNNs. The singularities of LL are extracted to train the first DNN; as LH, HL, and HH describe the image details in different orientations (horizontal, vertical, and diagonal), these subbands are combined before training the second DNN. Then, the outputs of the two DNNs are pooled for final decision making by extreme learning machine (ELM) [28], [29]. ELM is a novel and efficient training algorithm for the single-hidden-layer neural network. Compared with traditional neural network or SVM methods, ELM can be trained hundred times faster since its input weights and hidden node biases are randomly generated and the output weights are analytically computed.

The contributions of the proposed work are threefold: 1) a new compressed-domain framework is developed for fast ship detection; 2) DNN is employed for hierarchical ship feature extraction in wavelet domain; compared with existing feature descriptors, the proposed learning-based features are more robust under variant conditions; and 3) a new training model, the ELM, is adopted for feature fusion and classification, and thus, faster and better ship detection is achieved.

Using these novel techniques, the proposed framework is more suitable for ship detection than the aforementioned approaches with the following advantages.

1) **Faster detection**. Compressed domain achieves much faster detection than pixel domain.
2) **More reliable results**. High-level feature representations are extracted by hierarchical deep architecture to ensure more accurate classification.
3) **Better utilization of information**. Two DNNs are trained with multisubbands coefficients to make full use of the wavelet information.

The remainder of this paper is organized as follows. Section II overviews the proposed framework; Section III describes the preprocessing method for coarse ship locating; Section IV explains the ship feature representation in detail, including the selection of the inputs of autoencoders, fine tuning, and ELM-based decision making for deep networks; Section V demonstrates the simulation results comparing the proposed method and other relevant state-of-the-art ship detection methods; and Section VI concludes this paper.

II. Proposed Method

The typical JPEG 2000 compression is shown in Fig. 1. To clearly illustrate the proposed approach, it is necessary to define compressed domain in advance. According to the work in [23], the *compressed domain* is anywhere in the compression or decompression procedure, after transform or before inverse transform. Therefore, object detection can be conducted in compressed domain from points 1 to 6 in Fig. 1.

Unlike the other points, entropy coding (points 3 and 4 in Fig. 1) will obviously change the spatial distribution of the object features and destroy the structure information. Hence, points 1, 2 and 5, 6 are more suitable for ship detection. Furthermore, as points 5, 6 are symmetry to points 1, 2 in codec implementation, only points 1, 2 are discussed hereinafter.

At the encoder side, DWT is first performed (point 1 in Fig. 1). Then, the resulting coefficients are mapped to different bit planes by quantization (point 2 in Fig. 1). The bit-plane encoding will not obviously change the properties of wavelet

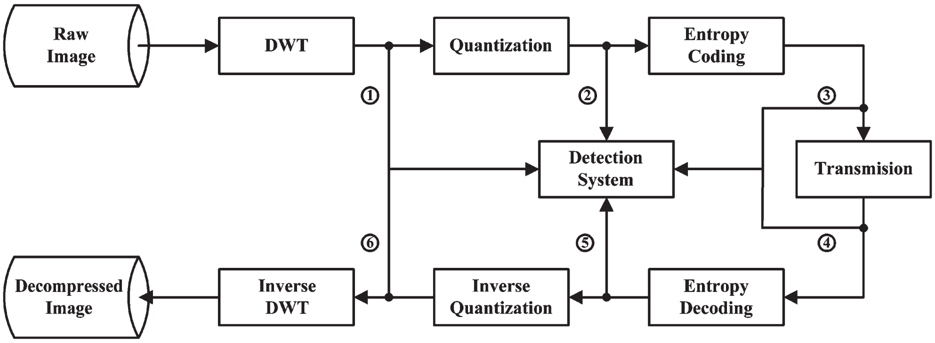

Fig. 1. JPEG2000 image compression.

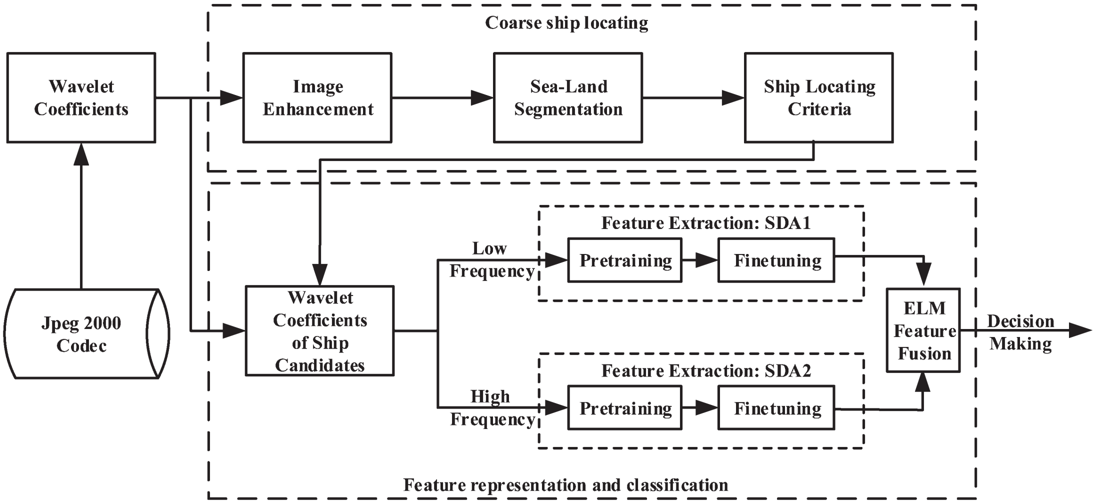

Fig. 2. Proposed ship detection framework.

features [30], and thus, the detection accuracy will not be severely affected. Based on this analysis, point 1 is viewed as the ideal place for ship detection.

The block diagram of the proposed framework is depicted in Fig. 2. It can be decomposed into two main steps: image preprocessing (for coarse ship locating), and feature representation and classification (for ship object detecting).

In the preprocessing, CDF 9/7 wavelet coefficients are extracted from JPEG 2000 codec. The wavelet coefficients in different subbands tend to reflect different properties of the original image [31]. Generally speaking, the low frequency contains most of the global information, while the high frequency represents local or detail information. In the proposed model, the low-frequency subband LL is exploited for the extraction of the regions of ship candidates.

On the other hand, the low-frequency coefficients and high-frequency coefficients (HFCs) are individually processed for feature extraction by two DNNs, which are to be discussed in Section IV. Moreover, to fully exploit the information of the original image in wavelet domain, the resulting features from low and high subbands are further fused by ELM, for more accurate feature classification (i.e., higher ship detection accuracy). The detailed implementations are listed as follows.

1) Wavelet singularities of LL are detected to train a stacked denoising autoencoder (SDA) 1. Note that SDA is one of the implementation strategies of DNN and will be introduced in Section IV-A.
2) The combination of the wavelet coefficients in high-frequency subbands (i.e., LH, HL, and HH) are used to train an SDA2.
3) The weight matrices of the trained SDAs are considered as feature extractors for low- and high-frequency subbands, respectively. The obtained features are then combined to train an online sequential ELM (OS-ELM) [28], [29], [32], [33].

It should be mentioned that the third step can be regarded as decision pooling of SDA1 and SDA2, or training a high-performance classifier of ship features. Since we are to make our algorithm more robust to various environmental conditions, online training is adopted to further improve the network's performance. The experiments in [29], [34], and [35] showed that ELM is fast and more accurate in large class training and the generalization performance of ELM turns out to be very stable.

III. Coarse Ship Locating

As shown in Fig. 2, fast ship locating (i.e., ship candidate extraction) is performed in LL subband, which includes image enhancement, sea–land segmentation, and ship locating based on shape criteria.

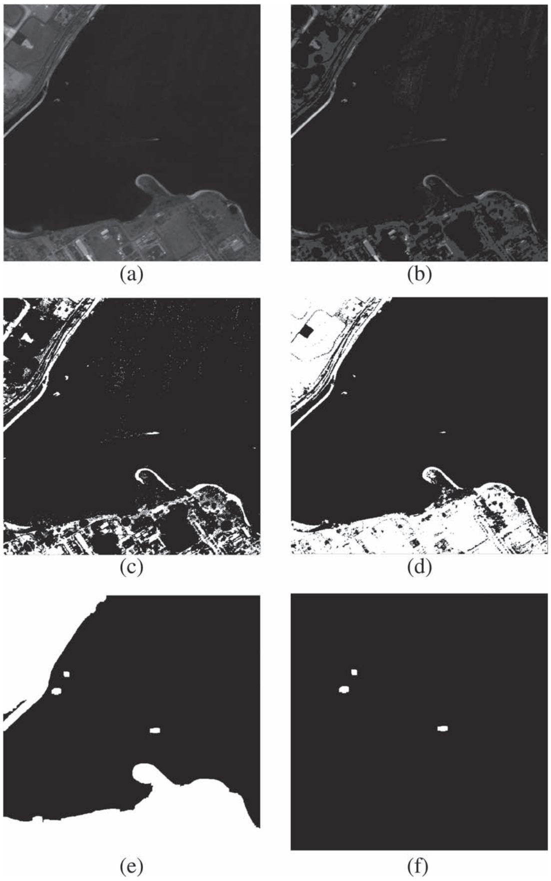

Fig. 3. Coarse ship segmentation: (a) Input image, (b) top-hat-transformed image, (c) binarized with the first threshold T, (d) corrected with the second threshold T', (e) refined by morphology dilation and erosion, and (f) coarse ship location.

A. Image Enhancement

In order to remove uneven illumination, a morphological operator, i.e., top-hat transform (THT), is used for ship extraction and background suppression. As ships are usually brighter than their surroundings, the white THT is employed in the proposed work [shown in Fig. 3(b)]. The mathematical definition of white THT is as follows:

$$\mathrm{Tw}(f) = f - f \circ b \tag{1}$$

where f is the input LL coefficients of the original image, $\circ$ denotes opening operation, and Tw is the enhanced image. In the simulations, b is set as a circular structuring element with a radius of 12.

B. Sea–Land Segmentation

Different from the traditional intensity histogram and maximum variance segmentation, here, a statistical Gaussian model is adopted to adaptively estimate the probabilistic distribution of the sea regions [36], and the algorithm is as follows.

1) Binarize the input image by the Otsu algorithm [37], and then label the connected regions.
2) Find the geometrical center P of the largest connected region R.
3) Use point P as the starting point; traverse R to obtain another set of points P' satisfying that the $A \times A$ (empirically set as 60 in the experiments) neighboring regions of P' are inside the region R. Label the points P' as all-sea region S.
4) Compute the mean μ and variance σ of S, and use them as the statistical parameters of the Gaussian model.

The resulting μ and σ are used to compute a threshold (T) for image binarization, as follows:

$$T = \mu + \lambda\sigma \tag{2}$$

where λ is the weight of variation (σ) and set as three according to the Gaussian distribution.

The binarized image obtained by T [shown in Fig. 3(c)] usually remains *holes* in large lands or clouds. In this case, a new threshold (T') for the elimination of *hole regions* and incorrectly marked lands is chosen as

$$T' = \lambda'\sigma \tag{3}$$

where λ' is a parameter to control the similarity of land and sea, empirically set as four. After thresholding with T' [shown in Fig. 3(d)], the median filtering (with size of [3 3]), morphology dilation, and erosion (circular structuring element with a radius of three) are applied to fill the isolated holes. Then, the masks of land, cloud, and ship candidates are segmented [shown in Fig. 3(e)]. In the following, ship candidates will be further extracted by using the unique shape properties of ships. Note that some of the pseudotargets may be included in the extracted regions; however, they can be removed in the process of feature fusion and classification in Section IV.

C. Ship Locating Criteria

In the previous section, several connected regions are extracted from the resultant masks by labeling the eight-connected neighbors. Geometric properties of the connected regions are then used for the locating of ship candidates, which are listed as follows [37], [38].

1) **Area**: It equals the number of pixels in the corresponding connected region. Area is used to cut off the lands, clouds, and other obviously large/small false targets.
2) **Major minor axis ratio**: It is defined as

$$R_{ls} = \frac{L_{\mathrm{axis}L}}{L_{\mathrm{axis}S}} \tag{4}$$

where $L_{\mathrm{axis}L}$ and $L_{\mathrm{axis}S}$ are the length of long and short axes of the bounding rectangle, respectively.
3) **Compactness**: Compactness measures the degree of circular similarity, and it is defined as

$$\text{Compactness} = \frac{\text{Perimeter}^2}{\text{Area}}. \tag{5}$$

By using these shape criteria, we can obtain the coarse locations of ship candidates [shown in Fig. 3(f)]. In the experiments, the size of testing images is 2000 × 2000 (in pixels) with a resolution of 5 m. The size of ship candidates is supposed to be smaller than 100 × 100 (or larger than 10 × 10). In this case, the regions with area larger than 10 000 (or smaller than 100) would be removed. Moreover, as the long axis of ship should be longer than the minor one, the major minor axis ratio is selected as 1.5. Compactness is set as 40 to exclude the regions which are obviously irregular.

It is also worth to note that, compared with original images, using a low-frequency subband (LL) for coarse ship locating would decrease the detection accuracy by 0.32% (in statistical average), but the detection speed is improved by more than 60%.

IV. Ship Feature Representation and Classification

The state-of-the-art ship detection approaches extract complicated features and combine them with learning-based classification. These feature operators or descriptors, e.g., scale-invariant feature transform [39], speeded up robust features [40], histogram of gradient [41], local multiple pattern (LMP) in [1], and shape/texture features in [11], are engineered to be invariant under certain rotations or scale variations and chosen for some specific vision tasks.

Features extracted by these methods generally have some fundamental limitations in practical applications. For example, they may have poor performances when the images are corrupted by blur, distortion, or illumination, which commonly exist in the remote sensing images. Relatively, learning features from image would help to tackle these issues. Recent works in [42] and [43] have shown that the features extracted by the unsupervised learning outperform those manually designed ones on object detection or recognition.

However, ship detection is usually under complicated environmental conditions, and the processed images may contain various pseudotargets, e.g., islands, clouds, coastlines, etc. Bengio *et al.* [44]–[46] indicated that traditional machine learning algorithms, e.g., SVM, may have difficulties in efficiently handling such highly varying inputs. These learning schemes usually use a few layers of computational units to establish the training model. When dealing with highly variant conditions, the computation is exponentially increased.

DNN, decomposing inputs into multiple nonlinear processing layers, achieves better performance with much less parameters in each layer. For example, the d inputs require $O(2^d)$ parameters to be represented by SVM with a Gaussian kernel, $O(d^2)$ parameters for a single-layer neural network, $O(d)$ parameters for a multilayer network with $O(\log_2 d)$ layers, and $O(1)$ parameters for a recurrent neural network. In other words, DNN would obtain much better feature representation than traditional learning methods using the same amount of parameters.

With the aids of pyramid-like hierarchical architecture, sparse features can be extracted by DNN. Since the higher hidden layer is more sparse than the lower one, high-level representations are obtained by this layer-by-layer extraction. The works in [12]–[18] have shown that DNN achieves excellent performance in vision and other applications.

Hence, our ship detection is based on deep architecture. Since, in wavelet domain, different subbands contain different information of the original image, the features of low and high frequencies are learned using two DNNs, respectively. The resultant features are then fused by ELM [28] at the top level, and the reason for selecting ELM as the final pooling layer is to be discussed in Section IV-D.

A. Introduction of SDA

As shown in Fig. 2, the SDAs [14] are used for training ship feature extractors in wavelet domain. In this section, we first briefly overview the concepts and theories of SDA.

SDA used in our work is one of the implementation strategies of DNN, and related research studies have gained great traction. It is based on the traditional autoencoder but uses the corrupted inputs rather than the original ones. The work in [14] has proved that, by using corrupted inputs, SDA can achieve better learning accuracy than that of the original autoencoder. Practically, denoising autoencoder is used as the building block of SDA, by feeding the latent representation of the layer below. Each level has a representation of the input pattern that is more abstract than the previous one [47].

The denoising autoencoder maps the input vector $x \in [0,1]^d$ to a higher level representation and then uses latent representation $y \in [0,1]^{d'}$ through a deterministic mapping $y = f_\theta(x) = s(Wx + b)$, parameterized by $\theta = \{W, b\}$, where $s(\cdot)$ is the activation function, W is a $d' \times d$ weight matrix, and b is a bias vector. The resulting latent representation y is then mapped back to a reconstructed vector $z \in [0,1]^d$ in the input space $z = g_{\theta'}(y) = s(W'y + b)$ with $\theta' = \{W', b'\}$. The reverse weight matrix W' mapping can optionally be constrained by $W' = W^{\mathrm{T}}$, and in this case, the autoencoder is said to have tied weights.

Each training sample $x(i)$ is mapped to a corresponding $y(i)$ and a reconstruction $z(i)$. The objective function of autoencoder is as follows:

$$\begin{aligned}\theta^*, \theta^{*'} &= \arg\min_{\theta,\theta'} \frac{1}{n} \sum_{i=1}^{n} L\left(x^{(i)}, z^{(i)}\right) \\ &= \arg\min_{\theta,\theta'} \frac{1}{n} \sum_{i=1}^{n} L\left(x^{(i)}, g_{\theta'}\left(f_\theta\left(x^{(i)}\right)\right)\right) \qquad (6)\end{aligned}$$

where L is a loss function, and it can be the traditional squared error $L(x,z) = \|x - z\|^2$.

An alternative loss, i.e., average reconstruction cross-entropy, is as follows:

$$\begin{aligned}L_H(x,z) &= H(B_x \| B_z) \\ &= -\sum_{k=1}^{d} \left[x_k \log z_k + (1-x)\log(1-z_k)\right] \qquad (7)\end{aligned}$$

where H denotes the cross-entropy and B represents the probabilities of training samples [14].

Combining (6) with $L = L_H$, the final objective function that we optimized can be written as

$$\theta^*, \theta^{*'} = \arg\min_{\theta,\theta'} E_{q^0(X)} L\left(x^{(i)}, g_{\theta'}\left(f_\theta\left(X^{(i)}\right)\right)\right) \qquad (8)$$

where $q^0(X)$ denotes the distribution associated with n training samples and E denotes the expectation operator. This optimization can be solved by an algorithm called *stochastic*

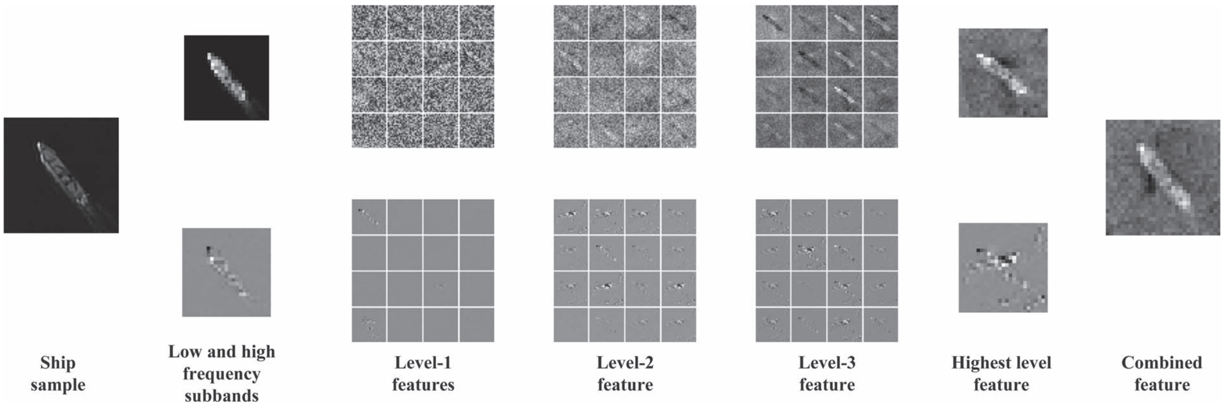

Fig. 4. Visualization of feature representation in different levels.

gradient descent. Stochastic gradient descent is a widely used optimization method for minimizing an objective function [like the one in (8)] that could be written as a sum of differentiable functions. Generally, the stochastic gradient descent can avoid local minima of the training error and thus achieves good classification performance.

B. Inputs of SDAs

As mentioned in Section II and shown in Fig. 2, the low-frequency (LL) and high-frequency (LL, LH, and HH) subbands are trained by two SDAs, respectively.

Singularities represent the sparse structures of LL, and therefore, they are extracted to train the SDA1. As the LH, HL, and HH already reflect the sparseness of the image, they are combined and used as the inputs of SDA2.

Before training, the input data need to be initialized by a zero-mean or z-score normalization [48]

$$Z = \frac{M - \text{mean}(M)}{\text{std}(M)} \tag{9}$$

where M denotes the input feature matrix and mean and std denote the 2-D mean and standard deviation, respectively. The z-score obeys the standard normal distribution that each of the dimensional data has zero mean and standard deviation, and it helps to correct outlier data points and remove light effect, which may significantly improve the training performance.

1) SDA1—Low-Frequency Module: DWT provides localized information about the variation of the image around a certain point or its local regularity. Therefore, irregularities will be sharpened after DWT. More exactly, the existence of discontinuities in the original image will result in local maxima in the wavelet domain [21]. It has also been demonstrated that finding wavelet modulus local maxima is an effective way to detect the singularities [31].

The singularities and irregular structures often contain the most important information, and thus, they are particularly meaningful for object recognition. Moreover, the characteristics of singularities are suitable for ship detection with its unique intensity distribution compared with that of the background.

Based on the aforementioned analysis, we use the local maxima of LL as the inputs of the SDA1.

2) SDA2—High-Frequency Module: The HFCs contain various object details, e.g., edgelike lines and distinct points, in different orientations. They reflect the sparse features of the original image. Hence, the coefficients of high-frequency subbands are utilized for training the SDA2.

Practically, the high-frequency subbands are combined as follows:

$$\begin{aligned} \text{IH} &= \alpha \cdot \text{LH} + \beta \cdot \text{HL} + \gamma \cdot \text{HH} \\ \text{s.t.} \qquad & \alpha + \beta + \gamma = 1 \end{aligned} \tag{10}$$

where α, β, and γ denote the weight parameters of different subbands. These parameters are set equally to avoid artificially intervening the weights of each subband. Please note that, here, symmetric even-order wavelet decomposition filters are used, where CDF 9/7 wavelets are applied.

C. Pretraining and Fine Tuning

In this section, we introduce the details of SDA training for ship feature extraction in low and high frequencies. Generally speaking, SDA-based feature extractor involves two main steps: pretraining and fine tuning.

The unsupervised layer-by-layer pretraining can help to achieve good generalization and low variance of testing error. Each layer is trained as a denoising autoencoder by minimizing the reconstruction of its input (which is the output code of the previous layer).

Based on the recent works in [49]–[51], some additional parameters are set to further improve the performance of the SDA. Before training, the coefficients are scaled to $[0, 1]^d$, and the learning rate is set as 0.1. The number of training batches depends on the size of data set, usually between [10, 100]. Different training batches should contain different classes of training samples to achieve better performance. Compared with 5% noise that is typically used in SDA [14], the simulations in [51] indicated that it is better drop out 20% inputs combined with 50% hidden units.

Once all of the layers are pretrained, the network needs a second stage of supervised training called fine tuning. The supervised fine tuning is used to minimize the prediction error. Practically, a logistic regression layer is added on top of the pretrained network.

Fig. 4 shows the different feature representations in different levels. It can be seen that each layer's mapping feature is more meaningful than that of the previous one. The resultant feature in the low frequency contains more implicit information of the ship, while the high-frequency one contains more edges and structures. Their combination reflects the full representation of a ship, and it has certain semantic meaning.

D. Feature Fusion With ELM

After the two SDAs have been trained, ELM is used for the fusion of ship features obtained from low and high frequencies, and the fused 100-dimensional feature vector is then utilized for classification and final decision making (i.e., ship detection).

ELM is a training algorithm for single-hidden-layer feedforward neural networks, and the input weights and hidden-layer bias are randomly set and need not to be tuned. ELM not only learns extremely fast but also achieves good generalization performance [28], [29]. In addition, when the training data are one by one or chunk by chunk, OS-ELM [32], [33] is preferred since retraining is not required whenever a new chunk of data is received.

In the following, we will introduce the ELM and OS-ELM algorithms in detail.

1) ELM Training: Given a training set $N = \{(x_i, t_i)|x_i \in R^n, t_i \in R^m,\ i = 1, \ldots, L\}$, where x_i is the feature vector extracted above and t_i represents the class label of each sample. $G(\cdot)$ denotes the activation function, and L is the number of hidden nodes. The ELM training algorithm can be summarized as follows [28].

1) Randomly assign hidden node parameters, e.g., input weight w_i and bias b_i, $i = 1, \ldots, L$.
2) Calculate the hidden-layer output matrix H.
3) Calculate the output weight β

$$\beta = H^{\dagger}T \tag{11}$$

where $T = [t_1, \ldots, t_N]^{\mathrm{T}}$, $H^{\dagger}$ is the Moore–Penrose generalized inverse of matrix H [29].

The orthogonal projection method can be efficiently used in ELM: $H^{\dagger} = (H^{\mathrm{T}}H)^{-1}H^{\mathrm{T}}$, if H^TH is nonsingular, or $H^{\dagger} = H^{\mathrm{T}}(H^{\mathrm{T}}H)^{-1}$, if HH^{T} is nonsingular. According to the ridge regression theory [29], it was suggested that a positive value $1/\lambda$ is added to the diagonal of H^TH or HH^{T} in the calculation of the output weights β. The resultant solution is more stable and achieves better generalization performance. That is, in order to improve the stability of ELM, we can have

$$\beta = H^{\mathrm{T}}\left(\frac{1}{\lambda} + HH^{\mathrm{T}}\right)^{-1}T \tag{12}$$

and the corresponding output function of ELM is

$$f(x) = h(x)\beta = h(x)H^{\mathrm{T}}\left(\frac{1}{\lambda} + HH^{\mathrm{T}}\right)^{-1}T \tag{13}$$

or we can have

$$\beta = \left(\frac{1}{\lambda} + HH^{\mathrm{T}}\right)^{-1}H^{\mathrm{T}}T \tag{14}$$

and the corresponding output function of ELM is

$$f(x) = h(x)\beta = h(x)\left(\frac{1}{\lambda} + HH^{\mathrm{T}}\right)^{-1}H^{\mathrm{T}}T. \tag{15}$$

2) OS-ELM: The sequential implementations of the least squares solution of OS-ELM [32], [33] are as follows.

Step 1) **Initialization:**

Initialize the learning using a small chunk of initial training data $N_0 = \{(x_i, t_i)\}_{i=1}^{N_0}$ from the given training set $N = \{(x_i, t_i)|x_i \in R^n, t_i \in R^m, i = 1, \ldots, L\}$.

a) Randomly generate the hidden node parameters $(a_i, b_i),\ i = 1, \ldots, L$.

b) Calculate the initial hidden-layer output matrix H_0

$$H_0 = \begin{pmatrix} G(a_1, b_1, x_1) & \ldots & G(a_L, b_L, x_1) \\ \vdots & \ddots & \vdots \\ G(a_1, b_1, x_{N_0}) & \cdots & G(a_L, b_L, x_{N_0}) \end{pmatrix}.$$

c) Estimate the initial output weight $\beta^{(0)} = P_0H_0^{\mathrm{T}}T_0$, where $P_0 = (H_0^{\mathrm{T}}H_0)^{-1}$ and $T_0 = [t_1, \ldots, t_{N_0}]^{\mathrm{T}}$.

d) Set $k = 0$.

Step 2) **Sequential Learning:**

a) Present the $(k+1)$th chunk of new observations: $N_{k+1} = \{(x_i, t_i)\}_{i=\sum_{j=0}k+1+1}^{\sum_{j=0}k+1N_j}$, where N_{k+1} denotes the number of observations in the $(k+1)$th chunk.

b) Calculate the partial hidden-layer output matrix H_{k+1} for the $(k+1)$th chunk of data N_{k+1}.

c) Calculate the output weight $\beta^{(k+1)}$

$$P_{k+1} = P_k - P_kH_{k+1}^{\mathrm{T}}\left(I + H_{k+1}P_kH_{k+1}^{\mathrm{T}}\right)^{-1}H_{k+1}P_k \tag{16}$$

$$\beta^{(k+1)} = \beta^{(k)} + P_{k+1}H_{k+1}^{\mathrm{T}}\left(T_{k+1} - H_{k+1}\beta^{(k)}\right). \tag{17}$$

d) Set $k = k + 1$; go to step 2)-a).

V. Experiments and Analysis

Extensive experiments are conducted in this section. Since SDA-based feature extraction, ELM-based feature fusion, and classification are adopted in this work, we term the proposed method as SDA-ELM, which is compared with the relevant state-of-the-art methods in [1] and [11]. In [1], multiple features are fused by SVM (denoted as MF-SVM), while in [11], salient regions are detected before SVM-based classification (denoted as SA-SVM). In addition, another method (SDA-based feature combined with SVM-based classification) is also tested (denoted as SDA-SVM).

In the following sections, to verify the effectiveness of each component of the proposed method (i.e., ship locating, feature extraction, feature fusion, and classification), the performance of ship candidate segmentation is first tested; then, the proposed SDA-based feature extraction is compared with other feature representation methods; classification performance of ELM is further evaluated against SVM, by using different combinations of extracted ship features; and finally, the overall ship detection accuracy is compared to demonstrate the advantages of the proposed scheme under practical testing conditions.

For fair comparison, 4000 5-m-resolution SPOT 5 panchromatic images with the size of 2000 × 2000 (in pixels) are

TABLE I
CLASSIFICATION SAMPLES

Class	*Number of Samples*
Ships	400
Oceanwaves	400
Clouds	400
Coastlines	200
Islands	200

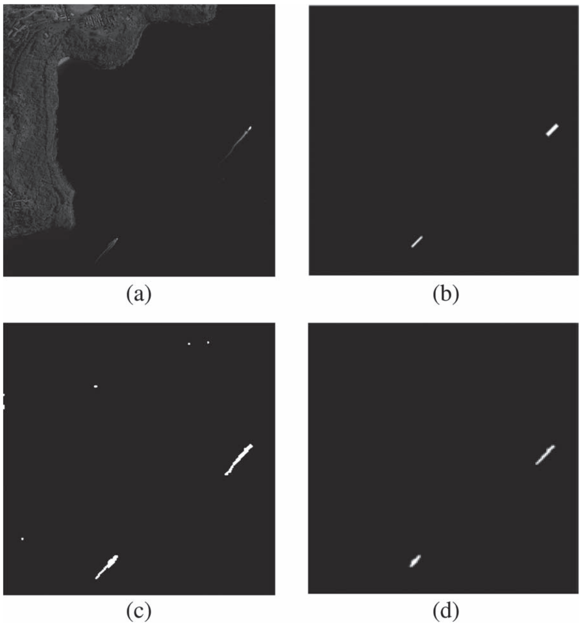

Fig. 5. Performance comparison of ship candidate segmentation. (a) Original image. (b) Manually labeled ground truth (the white pixels indicate ship candidates, while the black pixels represent land/sea regions). (c) Results by Chan–Vese model in [1]. (d) Results by the proposed method.

prepared to build an image data set to compare the performances of MF-SVM [1], SA-SVM [11], SDA-SVM, and SDA-ELM, and 1600 training samples are extracted for feature learning, shown in Table I. It should be emphasized that the images for the extraction of 1600 training samples have not been included in the testing images.

The testing hardware and software conditions are listed as follows: Intel-i7 2.4 G CPU, 8 G DDR3 RAM, Windows 7, Matlab R2012b, and Microsoft Visual Studio 2010.

A. Comparison of Coarse Ship Locating

The coarse ship locating is performed in the low-frequency subband LL. Fig. 5 shows the comparison of segmentation results of ship candidates, and one can see that the proposed method achieves more accurate segmentation than the Chan–Vese model in [1]. In addition, we also conducted objective comparisons to evaluate the performances of different methods. Three commonly used criteria were computed: false positive rate (FPR), false negative rate (FNR), and false error (FE). They are defined as follows:

$$\mathrm{FPR}(\mathrm{SR},\mathrm{GT}) = \frac{\#(\mathrm{SR}\cap\overline{\mathrm{GT}})}{\#(\mathrm{GT})} * 100\%$$
$$\mathrm{FNR}(\mathrm{SR},\mathrm{GT}) = \frac{\#(\overline{\mathrm{SR}}\cap\mathrm{GT})}{\#(\mathrm{GT})} * 100\%$$
$$\mathrm{FE}\,(\mathrm{SR},\mathrm{GT}) = (\mathrm{FPR}+\mathrm{FNR}) * 100\%$$

TABLE II
AVERAGE PERFORMANCE OF DIFFERENT SEGMENTATION METHODS

Method	*FPR* (%)	*FNR* (%)	*FE* (%)
Proposed model	1.5	3.9	5.4
Chan-Vese model in [1]	6.2	8.6	14.8

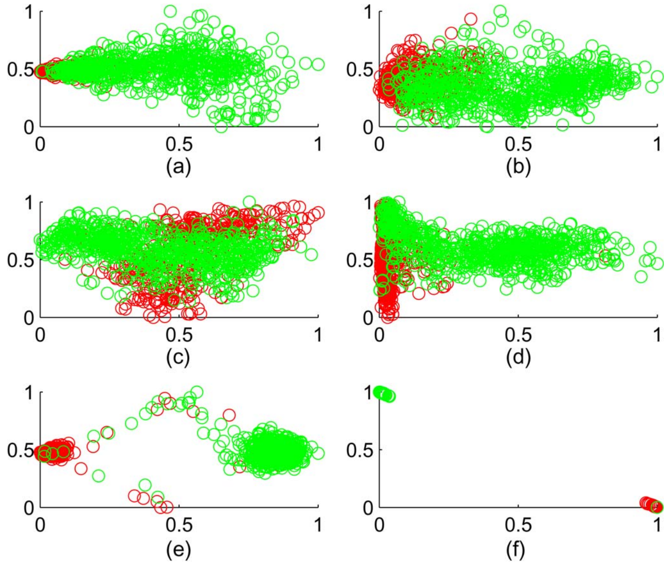

Fig. 6. Visualized 2-D space distributions of the first two principal components of different features. (a) Singularities of LL. (b) HFCs. (c) Shape/texture features of SA-SVM [11]. (d) LMPs of MF-SVM [1]. (e) Proposed SDA-based features. (f) Two-dimensional outputs of ELM pooling.

where SR denotes the segmentation results (ship candidates), GT denotes the manually labeled ground truth, $\#(\cdot)$ is the number of pixels in the corresponding region, and $\overline{\mathrm{GT}}$ and $\overline{\mathrm{ER}}$ denote the regions which are not included in GT and ER, respectively.

The averaging comparison results of the proposed method and Chan–Vese model in [1] are demonstrated in Table II. It is shown that the proposed model has lower FPR, FNR, and FE (better performance).

B. Comparison of Feature Representations

In this experiment, representation performances of different features are compared, including the LL singularities (LLSs), the HFCs, the LMPs in [1], the shape/texture features in [11], and the proposed SDA-based features.

Principal component analysis [52] is used for visualizing different features in 2-D space. Fig. 6 shows the first two principal components of each feature, where the red points represent ships and the green ones represent other subclasses shown in Table I.

As can be seen, the distributions of LLS and HFC are completely blended together. Relatively, the distances of the feature points in [11] expand little in the Cartesian coordinates, and still, a large amount of feature points are overlapped. LMP outperforms the aforementioned features; nearly half of the red points are separated from the green ones. The proposed

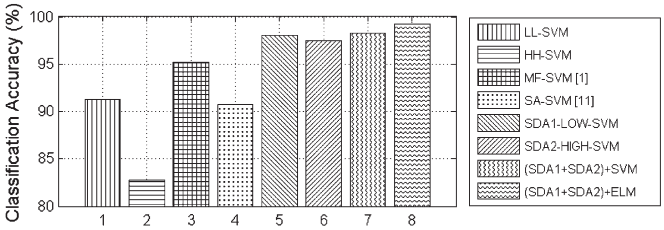

Fig. 7. Classification comparison of different features and learning algorithms.

SDA-based feature performs the best; almost all points are well separated. Noted that the outputs are nearly converged after the feature vector is clustered by ELM [in Fig. 6(f)].

The reason for the poor performances of LLS and HFC is obvious; they are simply the raw DWT coefficients exacted from the JPEG 2000 codec. As for the features used in SA-SVM [11], the feature vector includes some simple and straightforward shape or texture information of the objects, such as size, contrast, and energy. It is obvious that clouds and ships may share similarities in shape (e.g., object size, major minor axis ratio, etc.) and texture (e.g., contrast, energy, etc.). Unlike the others, LMP obtains more precise information of the samples by enlarging the dimension of its feature vector. The proposed SDA-based features capture the unique characteristics of ship by multilayer feature learning.

C. Comparison of Feature Fusion and Classification

In this section, the classification performance of ELM is compared against that of SVM, by using different combinations of extracted features. The classification accuracy of each method is computed as

$$T = \frac{\text{Number of correctly classified samples}}{\text{Number of tested samples}} * 100\%.$$

The experiments are based on k-fold cross-validation [53], which provides a better Monte Carlo estimate than simply randomly divided data set. First, the data set is randomly split into k mutually exclusive subsets $S_1, S_2, \ldots, S_k$ of approximately equal size. Then, the classifier is trained and tested k times; for each testing $t \in \{1, 2, \ldots, k\}$, it is trained on S without S_t and tested on S_t. As our training data set has 1600 samples, we select $k = 4$. The classification accuracy of each feature is shown in Fig. 7.

Due to the high-dimensional (679) feature vector, MF-SVM achieves better performance than SA-SVM by 4.5%. It is not surprising that SDA-SVM and SDA-ELM perform better than the other ones, since the features visualized in Fig. 6 are well separated. The results further demonstrate the effectiveness of the proposed SDA-based features. Moreover, ELM outperforms SVM in terms of learning accuracy, and this is also consistent to the conclusions obtained in [28] and [29].

To make full use of the information in low- and high-frequency subbands, in the proposed framework, ELM is adopted to achieve the fusion of features extracted by the corresponding two SDAs. An additional experiment has been conducted to verify the effectiveness of the fused feature, and the results are shown in Fig. 8. It can be seen that the performance (in terms of classification error) of the proposed method using both low-frequency coefficient and HFC (denoted as (SDA1+SDA2)-ELM) outperform other methods using low-frequency coefficient (denoted as SDA1-LOW-ELM) or HFC (denoted as SDA2-HIGH-ELM).

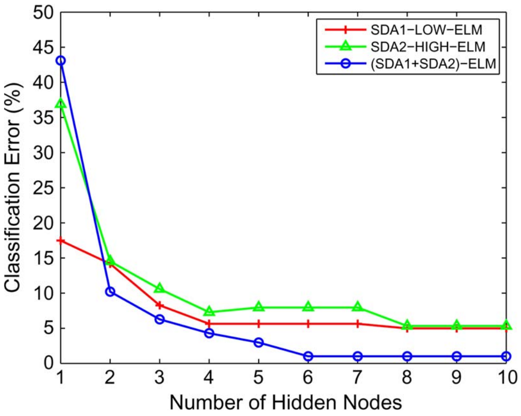

Fig. 8. Classification error of different features from low and high frequencies.

TABLE III
Classification Time Cost Using Different Methods

Method	MF-SVM [1]	SA-SVM [11]	SDA-SVM	SDA-ELM
Extracting Time (s)	6.8562	4.6893	2.4267	2.4267
Training Time (s)	0.6330	0.3606	0.4866	0.1986
Testing Time (s)	0.3402	0.1334	0.1448	0.0591
Overall Time (s)	7.8294	5.1833	3.0581	**2.6844**

TABLE IV
Detection Performances of Different Methods

Method	MF-SVM [1]	SA-SVM [11]	SDA-SVM	SDA-ELM
Accuracy (%)	94.57	91.63	96.45	**97.58**
Missing ratio (%)	5.43	8.37	3.55	2.42
Faslse ratio (%)	3.32	5.64	2.03	1.44
Error ratio (%)	8.75	14.01	5.58	3.86

Apart from classification accuracy, the feature extraction, training, and testing time of SA-SVM [11], MF-SVM [1], SDA-SVM, and SDA-ELM are compared, and the results are listed in Table III. Due to the relatively tight feature vector extracted in the wavelet domain, the extraction times of SDA-SVM and SDA-ELM are much less than those of SA-SVM and MF-SVM. The training time of SDA-ELM is less than those of SA-SVM, MF-SVM, and SDA-SVM by 50%, and the testing time is less than those by 30%. This advantage will be increasing when larger data set is used. That is, our approach performs better in extracting, training, and testing time, and these results benefit from all aforementioned advantages: faster feature extraction and higher learning efficiency of ELM.

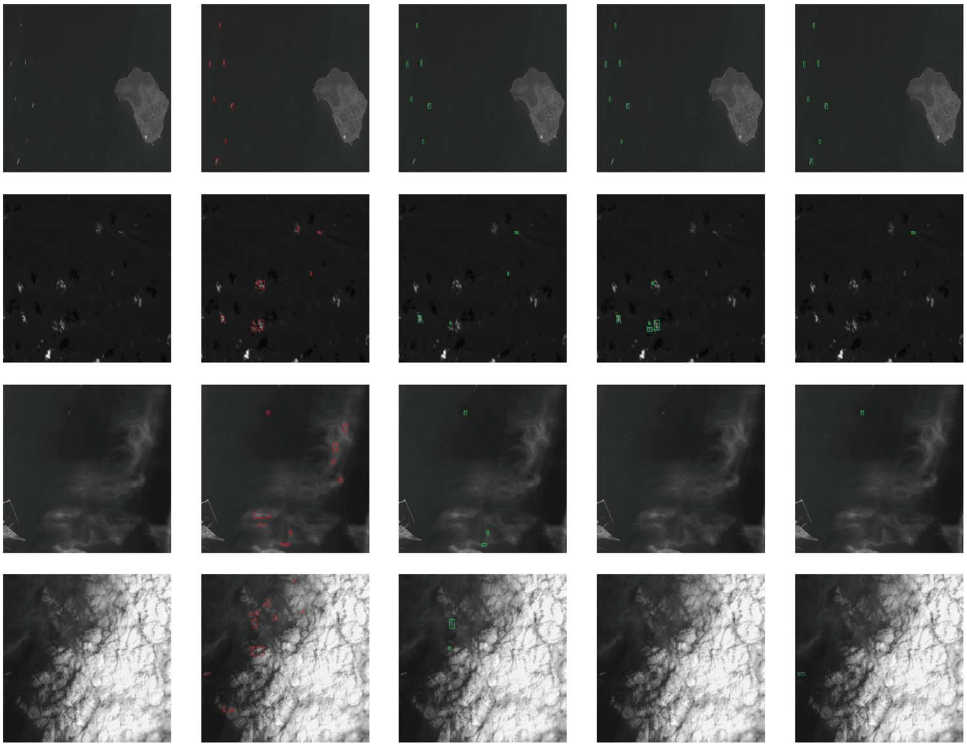

Fig. 9. Detection results of different methods under different experimental conditions. (Rows 1–4) Land, little clouds, cotton shaped cloud with mist, and large area of floccus. (Column 1) Input images. (Column 2) Coarse location of ship candidates. (Column 3) Classification results of MF-SVM [1]. (Column 4) Classification results of SA-SVM [11]. (Column 5) Classification results of our method.

D. Comparison of Overall Detection Performances

Finally, the detection performances of MF-SVM [1], SA-SVM [11], SDA-SVM, and SDA-ELM are compared. For fair comparison, the coarse ship locating described in Section III is applied with MF-SVM, SA-SVM, and SDA-SVM algorithms. The evaluation criteria are defined as [1]

$$\text{Accuracy} = \frac{\text{Number of correctly detected ships}}{\text{Number of real ships}} * 100\%$$

$$\text{Missing ratio} = 100\% - \text{Accuracy}$$

$$\text{False ratio} = \frac{\text{Number of falsely detected candidates}}{\text{Number of detected ships}} * 100\%$$

$$\text{Error ratio} = \text{Missing ratio} + \text{False ratio}.$$

The results are shown in Table IV, and one can see that SDA-ELM achieves the best performance. In addition, Fig. 9 shows several detection results (SDA-ELM against the other two existing methods) of optical spaceborne images with various false targets: island, small clouds, mist, large floccus, etc. One can see that SA-SVM and MF-SVM cannot well adapt to the large complicated area. SA-SVM eliminates part of pseudotargets but also some ships in the salient region. MF-SVM detects some of the ships with several false targets still remained. Compared with SA-SVM and MF-SVM, the proposed approach achieves the best results in various weather conditions.

VI. Conclusion

In this paper, we have proposed a compressed-domain ship detection framework using DNN and ELM for optical spaceborne images. Compared with the previous works, the proposed approach achieves better classification by deep-learning-based feature representation model with faster detection in compressed domain. After ship candidates are extracted, the singularities in LL are detected to train the SDA1. Then, the combination of high-frequency components (i.e., LH, HL, and HH) is used to train the SDA2. The two SDAs are viewed as feature extractors to obtain high-level features, and the resultant features are fused by ELM to further improve the classification results. ELM learns extremely faster and has better generalization than other traditional learning algorithms. Extensive experiments demonstrate that our proposed scheme outperforms the state-of-the-art methods in terms of detection time and accuracy.

As for the possible shortcomings of the proposed work, the parameters in coarse ship locating should be more adaptive to the image contents. In addition, due to the availability of image data sets, the simulations in the proposed work are conducted using panchromatic images, and other remote sensing image could be further tested or verified in a future work.

Moreover, in the experiments, the images of resolution of 5 m are used. In this case, the ships that we succeed to detect may be larger than 50 m (10 × 10 pixels). However, the limitation on the size of the detected ship is not induced by the proposed framework; it is mainly due to the resolution of the original images. In other words, when 5-m-resolution images are used for ship detection, it is not reasonable to detect smaller ships (for example, 20 m), as very limited object details/features can be extracted from such a small region (4 × 4 pixels).

Finally, since the object features are extracted and fused by DNN and ELM, the extracted features are with high robustness. Thus, the proposed framework is expected to work well for multispectral or synthetic aperture radar images. Our future work may focus on the use of the proposed work for ship detection from multiple sensors.

References

[1] C. Zhu, H. Zhou, R. Wang, and J. Guo, "A novel hierarchical method of ship detection from spaceborne optical image based on shape and texture features," *IEEE Trans. Geosci. Remote Sens.*, vol. 48, no. 9, pp. 3446–3456, Sep. 2010.

[2] C. Corbane, L. Najman, E. Pecoul, L. Demagistri, and M. Petit, "A complete processing chain for ship detection using optical satellite imagery," *Int. J. Remote Sens.*, vol. 31, no. 22, pp. 5837–5854, Jul. 2010.

[3] F. Y. Lure and Y.-C. Rau, "Detection of ship tracks in AVHRR cloud imagery with neural networks," in *Proc. IEEE IGARSS*, 1994, vol. 3, pp. 1401–1403.

[4] J. M. Weiss, R. Luo, and R. M. Welch, "Automatic detection of ship tracks in satellite imagery," in *Proc. IEEE IGARSS*, 1997, vol. 1, pp. 160–162.

[5] A. N. de Jong, "Ship infrared detection or vulnerability," in *Proc. SPIE*, 1993, pp. 216–224, International Society for Optics and Photonics.

[6] J.-W. Lu, Y.-J. He, H.-Y. Li, and F.-L. Lu, "Detecting small target of ship at sea by infrared image," in *Proc. IEEE Int. CASE*, 2006, pp. 165–169.

[7] C. DeSilva, G. Lee, and R. Johnson, "All-aspect ship recognition in infrared images," in *Proc. Electron. Technol. Directions Year 2000*, 1995, pp. 194–198.

[8] D. W. Burgess, "Automatic ship detection in satellite multispectral imagery," *Photogramm. Eng. Remote Sens.*, vol. 59, no. 2, pp. 229–237, 1993.

[9] G. Wu, J. de Leeuw, A. K. Skidmore, Y. Liu, and H. H. Prins, "Performance of Landsat TM in ship detection in turbid waters," *Int. J. Appl. Earth Observ. Geoinf.*, vol. 11, no. 1, pp. 54–61, Feb. 2009.

[10] C. Corbane, F. Marre, and M. Petit, "Using SPOT-5 HRG data in panchromatic mode for operational detection of small ships in tropical area," *Sensors*, vol. 8, no. 5, pp. 2959–2973, 2008.

[11] F. Bi, F. Liu, and L. Gao, "A hierarchical salient-region based algorithm for ship detection in remote sensing images," in *Advances in Neural Network Research and Applications*. New York, NY, USA: Springer-Verlag, 2010, pp. 729–738.

[12] G. E. Hinton, S. Osindero, and Y.-W. Teh, "A fast learning algorithm for deep belief nets," *Neural Comput.*, vol. 18, no. 7, pp. 1527–1554, Jul. 2006.

[13] G. E. Hinton and R. R. Salakhutdinov, "Reducing the dimensionality of data with neural networks," *Science*, vol. 313, no. 5786, pp. 504–507, Jul. 2006.

[14] P. Vincent, H. Larochelle, Y. Bengio, and P.-A. Manzagol, "Extracting and composing robust features with denoising autoencoders," in *Proc. ICML*, 2008, pp. 1096–1103, ACM.

[15] D. Erhan, P.-A. Manzagol, Y. Bengio, S. Bengio, and P. Vincent, "The difficulty of training deep architectures and the effect of unsupervised pre-training," in *Proc. ICAIS*, 2009, pp. 153–160.

[16] Y. Bengio, "Learning deep architectures for AI," *Found. Trends Mach. Learn.*, vol. 2, no. 1, pp. 1–127, 2009.

[17] P. Vincent, H. Larochelle, I. Lajoie, Y. Bengio, and P.-A. Manzagol, "Stacked denoising autoencoders: Learning useful representations in a deep network with a local denoising criterion," *J. Mach. Learn. Res.*, vol. 9999, pp. 3371–3408, 2010.

[18] A. Coates and A. Y. Ng, "Learning feature representations with k-means," in *Neural Networks: Tricks of the Trade*. New York, NY, USA: Springer-Verlag, 2012, pp. 561–580.

[19] Y. Bengio and A. Courville, "Deep learning of representations," in *Handbook on Neural Information Processing*. New York, NY, USA: Springer-Verlag, 2013, pp. 1–28.

[20] E. C. Smith and M. S. Lewicki, "Efficient auditory coding," *Nature*, vol. 439, no. 7079, pp. 978–982, 2006.

[21] M. Tello, C. López-Martínez, and J. J. Mallorqui, "A novel algorithm for ship detection in SAR imagery based on the wavelet transform," *IEEE Geosci. Remote Sens. Lett.*, vol. 2, no. 2, pp. 201–205, Apr. 2005.

[22] Z. Xiong and T. S. Huang, "Wavelet-based texture features can be extracted efficiently from compressed-domain for JPEG 2000 coded images," in *Proc. ICIP*, 2002, vol. 1, p. I-481, IEEE.

[23] K. Delac, M. Grgic, and S. Grgic, "Effects of JPEG and JPEG 2000 compression on face recognition," in *Pattern Recognition and Image Analysis*. New York, NY, USA: Springer-Verlag, 2005, pp. 136–145.

[24] "Face recognition in JPEG and JPEG 2000 compressed domain," *Image Vis. Comput.*, vol. 27, no. 8, pp. 1108–1120, Jul. 2009.

[25] L. Ni, "A novel image retrieval scheme in JPEG 2000 compressed domain based on tree distance," in *Proc. ICICS*, 2003, vol. 3, pp. 1591–1594, IEEE.

[26] A. Teynor, W. Müller, and W. Kowarschick, "Compressed Domain Image Retrieval Using JPEG2000 and Gaussian Mixture Models," in *Visual Information and Information Systems*. New York, NY, USA: Springer-Verlag, 2006, pp. 132–142.

[27] F. Zargari, A. Mosleh, and M. Ghanbari, "A fast and efficient compressed domain JPEG 2000 image retrieval method," *IEEE Trans. Consum Electron.*, vol. 54, no. 4, pp. 1886–1893, Nov. 2008.

[28] G.-B. Huang, Q.-Y. Zhu, and C.-K. Siew, "Extreme learning machine: Theory and applications," *Neurocomputing*, vol. 70, no. 1–3, pp. 489–501, Dec. 2006.

[29] G.-B. Huang, H. Zhou, X. Ding, and R. Zhang, "Extreme learning machine for regression and multiclass classification," *IEEE Trans. Syst., Man, Cybern. B*, vol. 42, no. 2, pp. 513–529, Apr. 2012.

[30] M. D. Adams, The JPEG 2000 Still Image Compression Standard, 2001.

[31] S. Mallat and W. L. Hwang, "Singularity detection and processing with wavelets," *IEEE Trans. Inf. Theory*, vol. 38, no. 2, pp. 617–643, Mar. 1992.

[32] G.-B. Huang, N.-Y. Liang, H.-J. Rong, P. Saratchandran, and N. Sundararajan, "On-line sequential extreme learning machine," in *Proc. Comput. Intell.*, 2005, pp. 232–237.

[33] N.-Y. Liang, G.-B. Huang, P. Saratchandran, and N. Sundararajan, "A fast and accurate on-line sequential learning algorithm for feedforward networks," *IEEE Trans. Neural Netw.*, vol. 17, no. 6, pp. 1411–1423, Nov. 2006.

[34] M. Şahin, "Comparison of modelling ANN and ELM to estimate solar radiation over Turkey using NOAA satellite data," *Int. J. Remote Sens.*, vol. 34, no. 21, pp. 7508–7533, 2013.

[35] M. Pal, A. E. Maxwell, and T. A. Warner, "Kernel-based extreme learning machine for remote-sensing image classification," *IEEE Geosci. Remote Sens. Lett.*, vol. 4, no. 9, pp. 853–862, 2013.

[36] X. You and W. Li, "A sea–land segmentation scheme based on statistical model of sea," in *Proc. CISP*, 2011, vol. 3, pp. 1155–1159, IEEE.

[37] R. C. Gonzalez and R. E. Woods, *Digital Image Processing*, 3rd ed. Knoxville, TN, USA: Gatesmark Publishing, 2007, pp. 742–745.

[38] M. Sonka, V. Hlavac, and R. Boyle, *Image Processing, Analysis, Machine Vision*. Stamford, CT, USA: Cengage Learning, 1999.

[39] D. G. Lowe, "Distinctive image features from scale-invariant keypoints," *Int. J. Comput. Vis.*, vol. 60, no. 2, pp. 91–110, Nov. 2004.

[40] H. Bay, T. Tuytelaars, and L. Van Gool, "SURF: Speeded up robust features," in *Proc. ECCV*, 2006, pp. 404–417.

[41] N. Dalal and B. Triggs, "Histograms of oriented gradients for human detection," in *Proc. CVPR*, 2005, vol. 1, pp. 886–893.

[42] M. Baccouche, F. Mamalet, C. Wolf, C. Garcia, and A. Baskurt, "Sequential deep learning for human action recognition," in *Human Behavior Understanding*. New York, NY, USA: Springer-Verlag, 2011, pp. 29–39.

[43] Y. Netzer *et al.*, "Reading digits in natural images with unsupervised feature learning," in *Proc. NIPS Workshop Deep Learn. Unsupervised Feature Learn.*, 2011, vol. 2011, pp. 1–9.

[44] Y. Bengio, P. Lamblin, D. Popovici, and H. Larochelle, "Greedy layer-wise training of deep networks," *Adv. Neural Inf. Process. Syst.*, vol. 19, p. 153, 2007.

[45] Y. Bengio, O. Delalleau, and N. L. Roux, "The curse of highly variable functions for local kernel machines," in *Proc. Adv. Neural Inf. Process. Syst.*, 2005, pp. 107–114.

[46] Y. Bengio and Y. LeCun, "Scaling learning algorithms towards AI," in *Large-Scale Kernel Machines*, vol. 31. Cambridge, MA, USA: MIT Press, 2007, pp. 1–41.

[47] Y. Bengio, P. Lamblin, D. Popovici, and H. Larochelle, "Greedy layer-wise training of deep networks," *Proc. Adv. Neural Inf. Process. Syst.*, vol. 19, pp. 1–8, 2007.

[48] L. A. Shalabi, Z. Shaaban, and B. Kasasbeh, "Data mining: A preprocessing engine," *J. Comput. Sci.*, vol. 2, no. 9, p. 735, 2006.

[49] G. Hinton, "A practical guide to training restricted Boltzmann machines," *Momentum*, vol. 9, no. 1, 2010.

[50] D. Erhan *et al.*, "Why does unsupervised pre-training help deep learning?" *J. Mach. Learn. Res.*, vol. 11, pp. 625–660, Feb. 2010.

[51] N. Srivastava, "Improving neural networks with dropout," Ph.D. dissertation, Univ. Toronto, Toronto, ON, Canada, 2013.

[52] I. Jolliffe, *Principal Component Analysis*. Hoboken, NJ, USA: Wiley, 2005.

[53] R. Kohavi, "A study of cross-validation and bootstrap for accuracy estimation and model selection," in *Proc. IJCAI*, 1995, vol. 14, no. 2, pp. 1137–1145.

Jiexiong Tang (S'13) received the B.Eng. degree from the Beijing Institute of Technology, Beijing, China, in 2013, where he is currently working toward the M.S. degree in the School of Electrical and Information Engineering.

His research interests include remote sensing image processing, feature learning and extraction, extreme learning machines, deep neural networks, and applications in computer vision.

Chenwei Deng (M'09) received the Ph.D. degree in signal and information processing from the Beijing Institute of Technology, Beijing, China, in 2009.

He is currently an Associate Professor with the School of Information and Electronics, Beijing Institute of Technology. He has authored or coauthored over 20 technical papers in refereed international journals and conference proceedings, and coedited one book. His current research interests include image/video coding, quality assessment, perceptual modeling, feature representation/fusion, and object detection/recognition/tracking.

Prof. Deng was awarded the titles of Beijing Excellent Talent and Excellent Young Scholar of Beijing Institute of Technology in 2013.

Guang-Bin Huang (M'98–SM'04) received the B.Sc. degree in applied mathematics and the M.Eng. degree in computer engineering from Northeastern University, Shenyang, China, in 1991 and 1994, respectively, and the Ph.D. degree in electrical engineering from Nanyang Technological University, Singapore, in 1999. During undergraduate period, he also concurrently studied in the Applied Mathematics Department and the Wireless Communication Department, Northeastern University.

From June 1998 to May 2001, he was a Research Fellow with Singapore Institute of Manufacturing Technology (formerly known as Gintic Institute of Manufacturing Technology), where he led/implemented several key industrial projects (e.g., Chief Designer and Technical Leader of Singapore Changi Airport Cargo Terminal Upgrading Project, etc.). Since May 2001, he has been an Assistant Professor and an Associate Professor with the School of Electrical and Electronic Engineering, Nanyang Technological University. He serves as an Associate Editor of *Neurocomputing*, *Neural Networks*, and *Cognitive Computation*. His current research interests include machine learning, computational intelligence, and extreme learning machines.

Dr. Huang serves as an Associate Editor of the IEEE TRANSACTIONS ON CYBERNETICS.

Baojun Zhao received the Ph.D. degree in electromagnetic measurement technology and equipment from the Harbin Institute of Technology, Harbin, China, in 1996.

From 1996 to 1998, he was a Postdoctoral Fellow with the Beijing Institute of Technology, Beijing, China, where he has been engaged in teaching and research work in the Radar Research Laboratory since 1998. He has been the Project Leader of more than 30 national projects. He is currently a Professor, a Ph.D. Supervisor, and the National Professional Laboratory Director of Signal Acquisition and Processing. He has authored or coauthored over 100 publications. His main research interests include image/video coding, image recognition, infrared/laser signal processing, and parallel signal processing. He has received four ministerial-level scientific and technological progress awards in these fields.