Journal of Imaging Science and Technology® 66(3): 030501-1–030501-10, 2022.

DCNN-based Ship Classification using Enhanced Edge Information and Inception Module

Bo Wang, Xiaoting Yu, Chengeng Huang, and Qinghong Sheng
Nanjing University of Aeronautics and Astronautics, College of Astronautics, Nanjing, China
E-mail: yuxiaoting@nuaa.edu.cn

Yuanyuan Wang
Northwest Institute of Mechanical and Electrical Engineering, South Renmin Road, Xianyang, China, 712000

Weiwei Liu
Zhejiang Institute of Surveying and Mapping Science and Technology, Hangzhou, Zhejiang Province, CN

Abstract. *The excellent feature extraction ability of deep convolutional neural networks (DCNNs) has been demonstrated in many image processing tasks, by which image classification can achieve high accuracy with only raw input images. However, the specific image features that influence the classification results are not readily determinable and what lies behind the predictions is unclear. This study proposes a method combining the Sobel and Canny operators and an Inception module for ship classification. The Sobel and Canny operators obtain enhanced edge features from the input images. A convolutional layer is replaced with the Inception module, which can automatically select the proper convolution kernel for ship objects in different image regions. The principle is that the high-level features abstracted by the DCNN, and the features obtained by multi-convolution concatenation of the Inception module must ultimately derive from the edge information of the preprocessing input images. This indicates that the classification results are based on the input edge features, which indirectly interpret the classification results to some extent. Experimental results show that the combination of the edge features and the Inception module improves DCNN ship classification performance. The original model with the raw dataset has an average accuracy of 88.72%, while when using enhanced edge features as input, it achieves the best performance of 90.54% among all models. The model that replaces the fifth convolutional layer with the Inception module has the best performance of 89.50%. It performs close to VGG-16 on the raw dataset and is significantly better than other deep neural networks. The results validate the functionality and feasibility of the idea posited.* *© 2022 Society for Imaging Science and Technology.*
[DOI: 10.2352/J.ImagingSci.Technol.2022.66.3.030501]

1. INTRODUCTION

Ship classification plays an important role in maritime safety and port security. This has been facilitated by optical remote sensing (RS) technology, which has been increasingly adopted in the recent times to identify and classify ships in RS image data in as detailed a manner as possible. However, the ship target being relatively small against the background of the vast sea surface, and significantly changing scale subjected to interference by sea clutter, clouds, and obstacles such as pirate reefs and ship wakes. In addition, deep learning-based methods must collect and label sufficient training data. Therefore, identification of surface ships faces many challenges.

Traditional ship target classification is based on classifiers and geometric, statistical, and spectral features. The quality of features and the nonlinear fitting ability of the classifier determine the final performance [1]. Using a supervised classification model, support vector machine (SVM), to classify ships is one of the most common methods. However, owing to the complex environmental conditions, traditional SVM classification has difficulty with this highly variable input [2]. Since 2012, we have witnessed the astonishing breakthrough of DCNNs in image classification. DCNNs have a powerful self-learning ability through a large amount of data, and do not require a strict selection of features, only needing to guide learning to achieve the desired purpose. Excellent models have been developed, such as AlexNet [3], VGGNet [4], GoogleNet [5], and ResNet [6]. Distinct from traditional models, DCNNs can automatically capture high-level features through multilayer propagation, so as to achieve good classification prediction without preprocessing of raw input images.

Despite their good classification performance, DCNN models have some disadvantages. Most importantly, a DCNN tends to function as a black box because the nature of the learned features is unclear, nor is it understood how the network eventually extracts them and why they provide accurate classifications from raw image data. Also, owing to the difficulty of interpretation, the targeted improvement of deep learning models is difficult [7]. Hence, it is difficult to determine the specific features that lead to the obtained classification results. This issue has been highlighted frequently. Linear classifier probes were proposed to detect features in network layers [8]. Networks were found to contain fragile co-adapted features in successive layers [9], and the concept of network dissection was proposed to quantitatively interpret convolutional neural network (CNN) structures [10]. Others have indirectly studied network

Received Mar. 11, 2021; accepted for publication Sept. 2, 2021; published online Sep. 28, 2021. Associate Editor: Yi Wang.
1062-3701/2022/66(3)/030501/10/$25.00

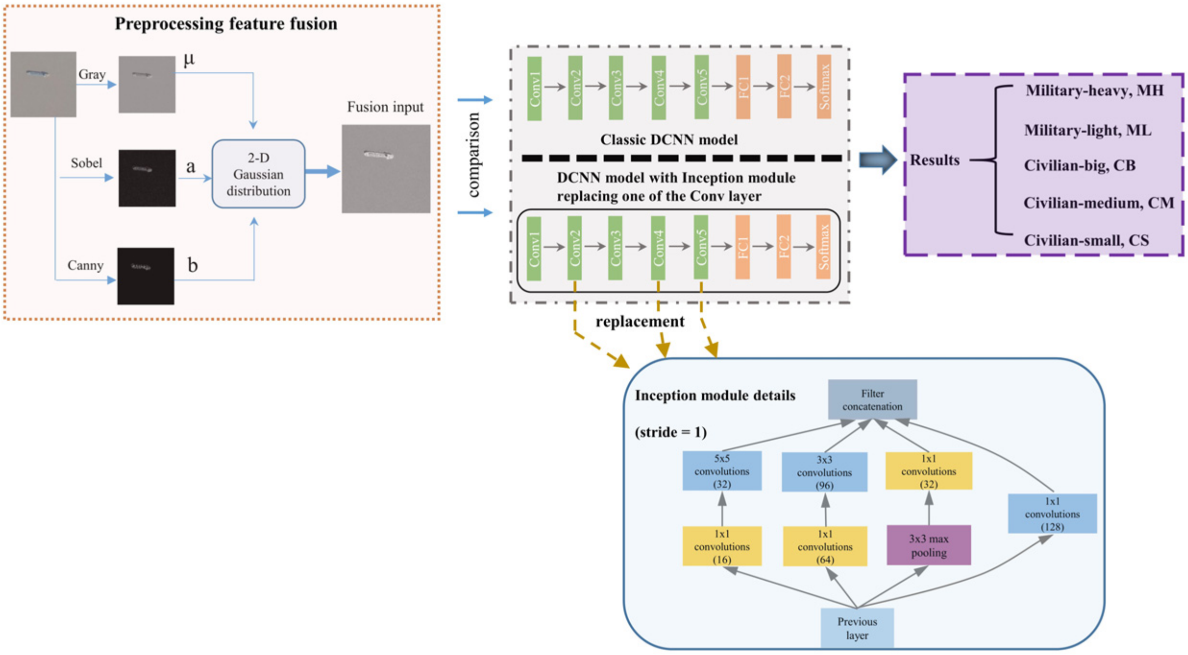

Figure 1. Schematic illustrating preprocessing of raw images and structure of DCNN classification framework. It can be seen in the middle box that both DCNN structures are typical 8-layer AlexNet models, replacing the convolutional layer in one with an Inception module, shown in lower-right box.

structures through the preprocessing of raw input image data. The Gabor LBP operator was applied to enhance rotation, edge, and other features of raw image data [11], and a dense feature pyramid network (DFPN) was developed to enhance feature propagation and encourage feature reuse [12]. These studies demonstrated that incomplete edge extraction or even edge loss, occurs after the first convolutional layer because of the inconspicuous edge information of raw images.

This approach based on preprocessing of raw input image data is applied in the present study to address the problems associated with DCNN models in ship classification based on RS image data. The proposed method preprocesses RS ship images to obtain the edge features of raw images using a combination of the Sobel [13] and Canny [14] operators. As edge information is usually unclear in raw ship images, DCNN is unlikely to sufficiently extract their edge features. The proposed preprocessing accentuates this information by increasing the ratio of gray level values between the edge and non-edge parts of ship images, which facilitates adequate extraction of edge information. In addition, to choose a proper convolution kernel for ship objects indifferent image regions, the GoogleNet Inception module [5] replaces a convolutional layer in the 8- layer AlexNet. We replace just one of the convolutional layers in order to limit modifications to the AlexNet structure, for more reasonable comparisons.

This study proposes an image preprocessing algorithm that trains the DCNN to focus on the edges of the input images. Accordingly, the factors promoting the classification results of the DCNN are identifiable and its structure is interpretable as the high-level features it abstracts must eventually evolve from the edges of input images. The present work stresses on preprocessing than on the classification accuracy of the DCNN, another reason why we employ AlexNet as the experimental model rather than a more complex model. This study also provides the option for choosing a proper convolution kernel for ship objects in different image regions, through the multi-convolution concatenation of the Inception module.

The remainder of this study is arranged as follows. Section 2 introduces the operations used by the proposed preprocessing method, and presents the DCNN classification framework used in this study. Section 3 validates the proposed approach through an analysis of experimental results. Section 4 summarizes the paper.

2. RELATED WORK

Ship classification is receiving increasing attention in the remote sensing domain, and many solutions have been proposed. Current methods are based mainly on manual features or deep learning. Low-level global features such as geometric features, i.e., scale, aspect ratio, and shape, are helpful for ship classification, but are only used in simple cases. Deep learning technology has set off a research trend in artificial intelligence, especially the application of CNNs to computer vision, which is unprecedented. Deep learning-based methods have shown impressive results in tasks such as computer vision and object classification, offering more distinguishing high-level visual features, to more effectively bridge the semantic gap between manual feature representation and remote sensing images. Features based on deep learning have strong recognition capabilities in natural image classification. These approaches are in

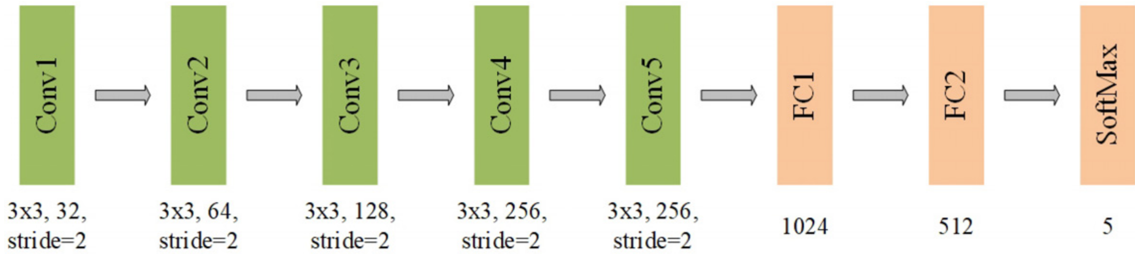

Figure 2. Structure of 8-layer AlexNet. Necessary parameters are also shown.

Figure 3. The favors a small convolution kernel, while a ship with a large area favors a larger one, suggesting that a convolutional layer with a single convolution kernel has a drawback, and that simply stacking large or small convolution kernels is not appropriate.

nascent stage and offer great potential for fine-grained ship classification.

AlexNet [3] consists of five convolutional layers and three fully connected layers. ReLU is used as the activation function to solve the problem of gradient dispersion of the sigmoid function when the network is too deep. Dropout in the training process avoids overfitting. VGGNet [4] repeatedly stacks 3×3 small convolution kernels and 2×2 maximum pooling layers, adding nonlinear operations to make the network more capable of learning features, and deepening the network structure to improve performance. Increasing the number of network layers does not introduce corresponding increase in the number of parameters. The network depth ranges from 11 to 19. VGGNet-16 and VGGNet-19 are commonly used. GoogleNet [5] is a 22-layer network using the Inception module. The convolutional layer

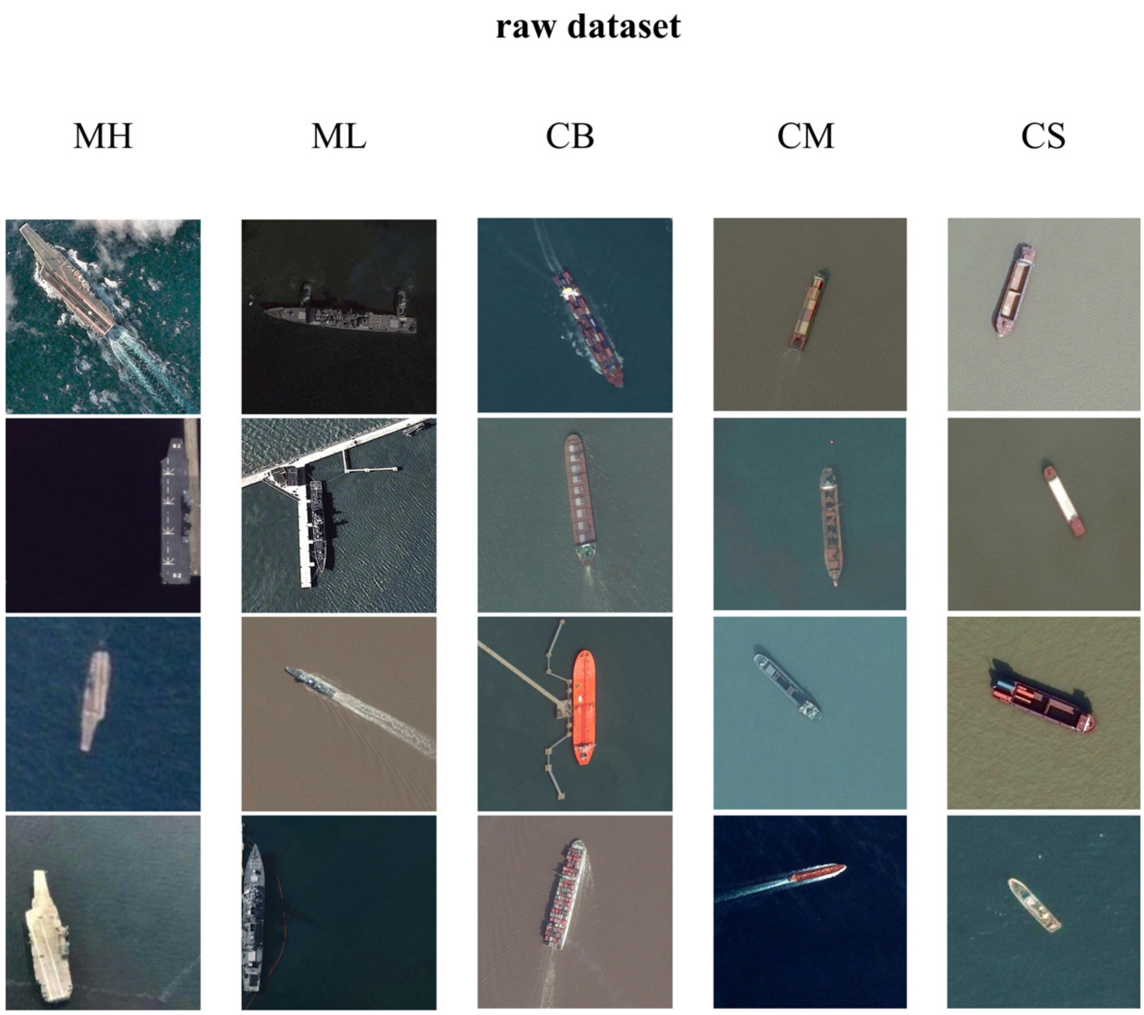

Figure 4. Representative images of raw dataset. Each column represents a single class with classes shown in the order of MH, ML, CB, CM and CS from left to right.

is parallel to the pooling layer instead of being stacked, which avoids overfitting due to an excessive number of parameters, and minimizes calculation. ResNet [6] uses residual learning, which solves the problem of information loss during transmission in a traditional CNN. The direct transmission of input information to the output solves the problem of gradient disappearance or explosion, leading to a network to be too deep and untrainable. DenseNet [15] connects each layer to all other layers in a feedforward manner. The input of a layer includes the feature maps of all previous layers, and its feature map is used as the input of all subsequent layers, which reduces the phenomenon of gradient disappearance, and efficiently reuses features while reducing the number of parameters. Overfitting is a challenge in DCNN because the neural network performs well on training set but cannot replicate the same on unseen test data [16]. Zheng [17] used a two-stage training method (pretraining and implicit regularization training) to optimize the feature boundary of DCNN to reduce overfitting. It has been proved that the quality of training and test datasets as well as redundancy of the parameter space have an impact on the generalization ability of deep neural networks [18].

The classic CNN network has achieved breakthrough results in target recognition applications, prompting researchers to apply deep learning techniques to ship classification. Fouad Bousetouane et al. [19] used four types of CNNs, OverFeat, AlexNet, VGG, and GoogLeNet, to extract features used for one-to-many training of support vector machines for ship target classification. Sergey Voinov et al. [20] proposed a CNN-based high-resolution image ship target recognition method, using MobileNet to quickly preselect the ship target in an image, and Faster R-CNN Inception-ResNet for ship target detection and classification. Maxime Leclerc et al. [21] combined a pretrained Inception network and ResNet network to classify ship targets on the MARVEL, ship dataset, mainly using transfer learning to fine-tune the deep neural network to find the appropriate structure of the network, number of layers, learning rate, and regularization parameters. These methods do not take advantage of a ship's edge characteristics.

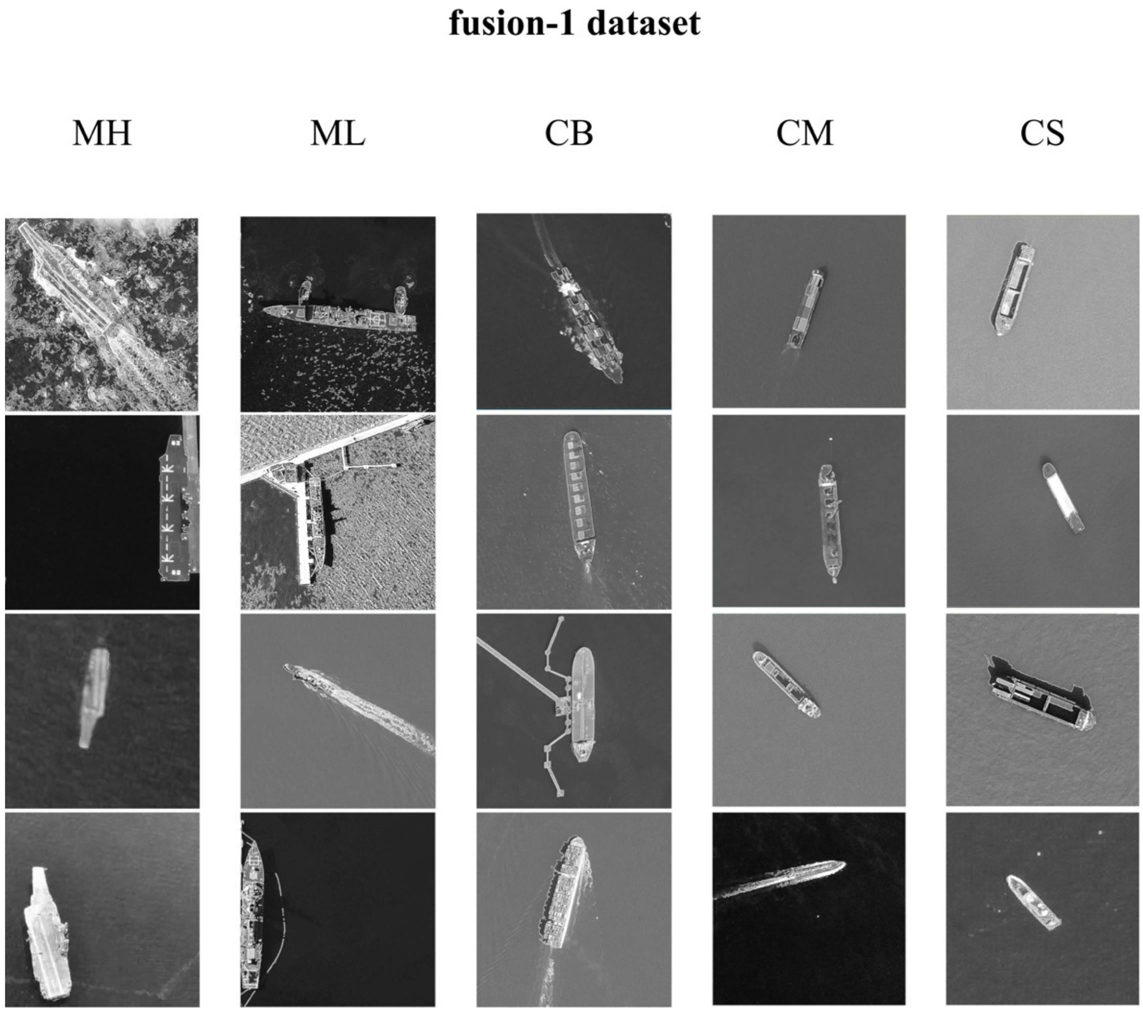

Figure 5. Representative images of fusion-1 dataset obtained with weighting coefficients (0.3, 0.8). Each column represents a single class, shown in the order of MH, ML, CB, CM and CS from left to right.

3. PROPOSED METHOD AND CLASSIFICATION FRAMEWORK

We describe the preprocessing fusion method and classification framework, as shown in Figure 1. We introduce the feature fusion operations, followed by the DCNN classification framework.

3.1 *Selection of Edge Operators*

The edge feature is a basic feature of the image that contains a considerable part of its information, especially when features needed for modeling in image tasks are based on edge features and are continuously abstracted into deeper feature expressions. The importance of edge information in image processing and computer vision is evident. The edge is essentially a boundary line, and the edge of the image is a collection of pixels that are distinguished from neighbouring pixels. The image can be regarded as a two-dimensional function whose derivative can quantify the change of pixels in the image. Well-known edge detection operators include the Roberts, Sobel, Laplacian, and Canny operators.

For optical remote sensing ship detection tasks, because noise and edges belong to the high-frequency part of the image, when the image quality is high and the image noise is insignificant, even the simple Roberts operator can obtain good results. However, with declining image quality, noise interference increases, which covers a part of the ship, and the use of complex edge operators will cause false detection of the noise. While Sobel operator is not good at edge extraction [22], it has a strong ability to smooth random noise [23]. The Canny operator is more sensitive to weakened edge information and can extract rich edge detail information [24], but it will cause some noise misdetection. These two operators have obvious complementary characteristics, combining them through weighted fusion to enhance the edge features is as follows:

$$P(x, y) = \mu * G(x, y) + a * S(x, y) + b * C(x, y), \qquad (1)$$

where $P(x, y)$ is the new input image, with size $M \times N$; $G(x, y)$ is the raw image after grayscale transformation; $S(x, y)$ and $C(x, y)$, are the images obtained after applying

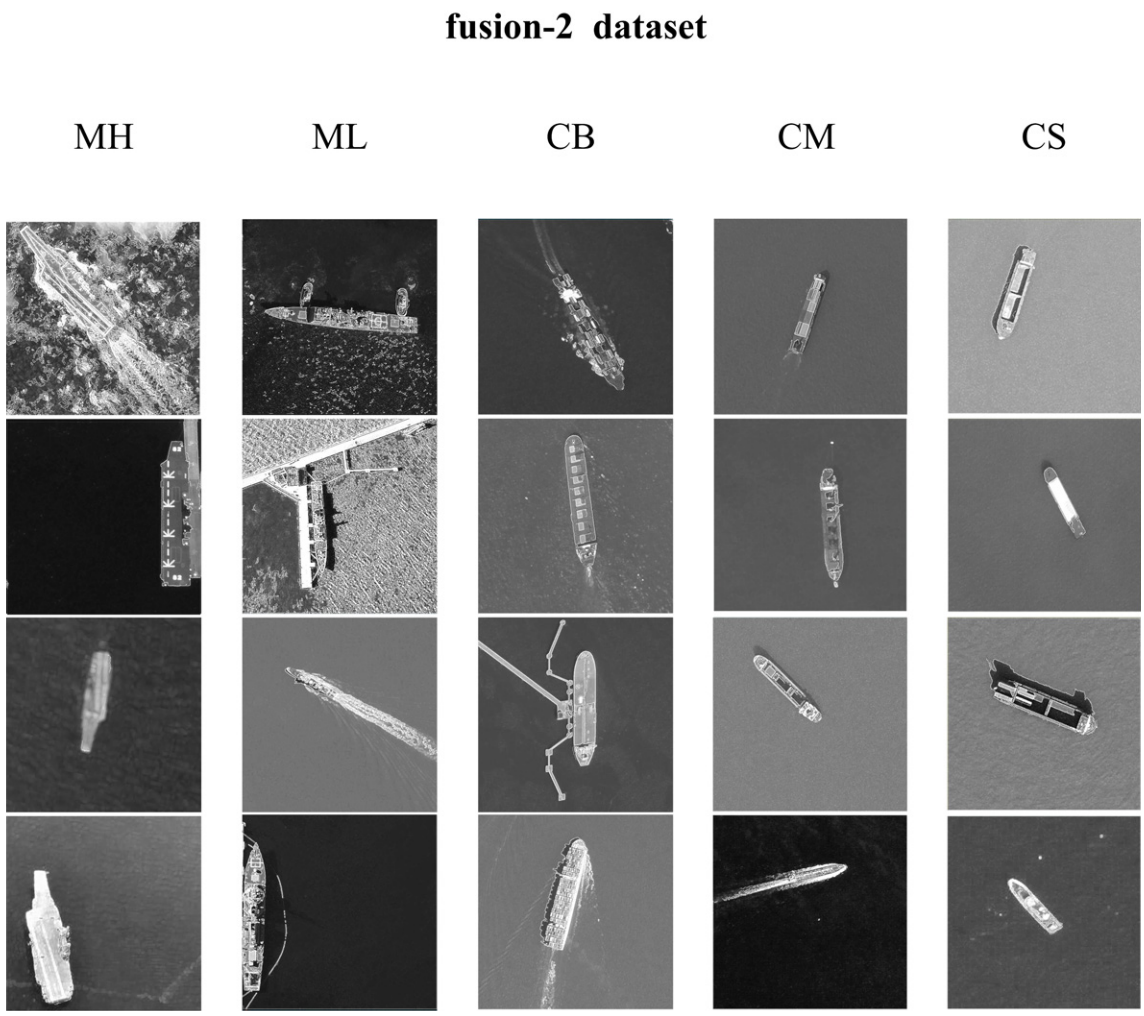

Figure 6. Representative images of fusion-2 dataset obtained with weighting coefficients (0.5, 0.5). Each column represents a single class, shown in the order of MH, ML, CB, CM and CS from left to right.

the Sobel and Canny transformations, respectively, to the raw image; and μ, a, and b are weight coefficients that determine the combinations of these three transformations. In this work, $G(x, y)$ serves as the basis of $P(x, y)$. Therefore, $\mu = 1$.

3.2 *Determination of Weight Coefficients*

In edge extraction of the original input image, the Sobel and Canny operators are independent, with separate processes. The adopted fusion algorithm must not be biased toward a certain edge operator in the initial fusion; The two selected edge operators will eventually be connected in the fusion process due to weight coefficient. We subsequently select the two-dimensional Gaussian distribution to constrain the weight coefficients. Its rotational symmetry ensures consistent smoothness [25] in two directions, representing the degree of demand for the Sobel and Canny operators. Therefore, it ensures that feature-weighted fusion will not be initially biased toward any operator to ensure neutrality. The two-dimensional Gaussian distribution function is

$$f(a, b, \mu) = \frac{1}{2\pi\sigma_1\sigma_2\sqrt{1-\rho^2}} \exp\left[\frac{-1}{2(1-\rho^2)}\right] \left[\frac{\|a-\mu_1\|^2}{\sigma_1^2} - 2\rho\frac{(a-\mu_1)(b-\mu_2)}{2\sigma_1\sigma_2} + \frac{\|b-\mu_2\|^2}{\sigma_2^2}\right], \tag{2}$$

where ρ is a coefficient reflecting the degree of independence of a and b, and μ_1, μ_2, and σ_1, σ_2, are the expectations and variances, respectively, in the related marginal distributions. Because the Sobel and Canny operators function independently, a and b are independent. Therefore, we set $\rho = 0$. In addition, μ_1 and μ_2 reflect the central tendency of these distributions as the mean centers of a and b. Moreover, because $G(x, y)$ serves as the basis during the fusion process, we set $\mu = \mu_1 = \mu_2 = 1$. We set $\sigma_1 = \sigma_2 = 1$ for convenience. We update '(2)' as

$$f(a, b) = \frac{1}{2\pi} \exp\left(\frac{-1}{2}\right) \left[\|a-1\|^2 + \|b-1\|^2\right], \tag{3}$$

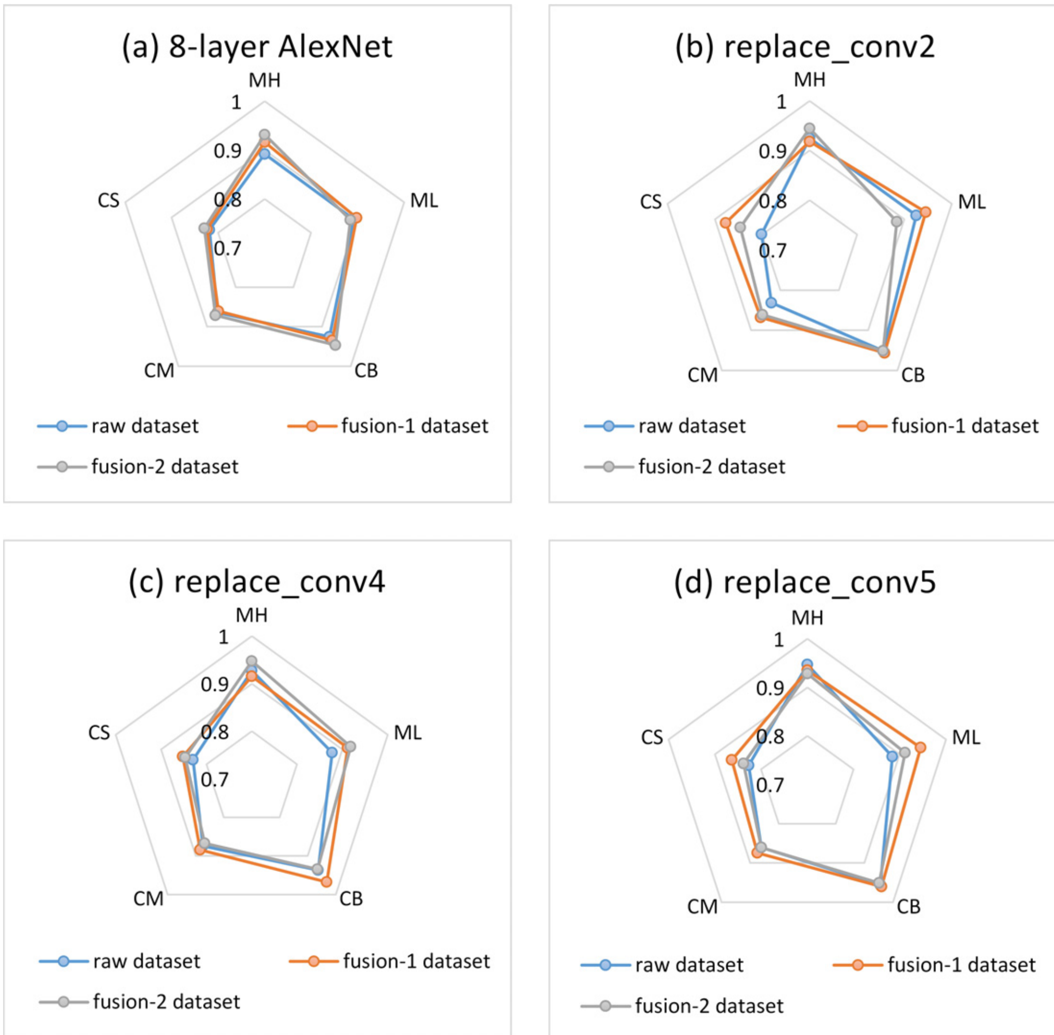

Figure 7. Radar charts showing the classification accuracies of the four models. (a) 8-layer AlexNet; (b) replace_conv2; (c) replace_conv4; (d) replace_conv5. It can be seen intuitively that the network obtains better classification accuracy on the fused dataset. We note that accuracies of the CS class are almost lowest, even if the proposed method is applied.

where (a, b) is artificially determined, $a \in (0, 1)$, $b \in (0, 1)$, and the intervals between successive values of a and b (i.e., the stride) are uniformly 0.1. We adhere to the idea of the arithmetic average in the selection of weight coefficients.

According to '(3)', we total the function values obtained when the weighting coefficients (a, b) take different values, and take the arithmetic average. The final value is closest to (0.3, 0.7), so we choose (0.3, 0.7) as the first set of experimental weight coefficients. To further verify the effectiveness of the preprocessing method, we randomly select (0.5, 0.5) as the second set of experimental weight coefficients.

3.3 *DCNN Classification Framework*

We employ the 8-layer AlexNet model, consisting of five convolutional layers, two fully connected layers, and a softmax layer, whose structure is shown in Figure 2. Owing to its single convolution kernel size, this network suffers when ship objects are in different regions of the input image, as shown in Figure 3. Therefore, we apply an Inception module (blue box at bottom right, Fig. 1) to replace one of the convolutional layers, so that, with its multi-convolution concatenation, the robustness of the convolution operation against different ship positions can be improved without extensive modification to the network structure.

4. EXPERIMENTS AND ANALYSIS

4.1 *Experimental Datasets*

Nearly 25000 optical ship images were manually captured from Google Earth as the raw dataset. These were divided into two major categories: military ship, with military-heavy (MH) and military-light (ML) subclasses, and civilian ship, with civilian-large (CL), civilian-medium (CM), and civilian-small (CS) subclasses, as shown in Figure 4. Each category had 5000 images with a 4:1 ratio of training to test sets. The first preprocessed dataset used in the experiments, fusion-1, was obtained according to values of the weighting coefficients (a, b) that came closest to providing the mean value of the

Table I. Classification results of the three datasets with four models.

Dataset	Model	Category				
		MH	ML	CB	CM	CS
Raw dataset	8-layer AlexNet	0.8920	0.8912	0.9254	0.8642	0.8184
	Replace_conv2	0.9250	0.9252	0.9510	0.8311	0.8017
	Replace_conv4	0.9272	0.8776	0.9371	0.8742	0.8296
	Replace_conv5	0.9471	0.8844	0.9557	0.8609	0.8268
Fusion-1	8-layer AlexNet	0.9162	0.8980	0.9347	0.8609	0.8240
	Replace_conv2	0.9184	0.9456	0.9557	0.8675	0.8771
	Replace_conv4	0.9151	0.9116	0.9674	0.8841	0.8520
	Replace_conv5	0.9350	0.9456	0.9604	0.8742	0.8631
Fusion-2	8-layer AlexNet	0.9316	0.8844	0.9464	0.8709	0.8296
	Replace_conv2	0.9449	0.8844	0.9510	0.8609	0.8464
	Replace_conv4	0.9471	0.9184	0.9347	0.8675	0.8464
	Replace_conv5	0.9283	0.9116	0.9510	0.8609	0.8380

two-dimensional distribution in '(3)', i.e., $(a, b) = (0.3, 0.7)$, as shown in Figure 5. To further verify the validity of the preprocessing method, we randomly selected the set (a, b), (0.5, 0.5), fusion-2, as shown in Figure 6. We replaced the second, fourth, and fifth convolutional layers with an Inception module, denoted as replace_conv2, replace_conv4 and replace_conv5, respectively. All experimental models were optimized by the Adam optimizer, and had the same initial learning rate of 0.01 with a decay rate of 0.8 per 500 iterations in a total of 50000 iterations.

4.2 *Classification Performance and Analysis*

Table I present the accuracies obtained from the raw dataset, fusion-1, and fusion-2 with the 8-layer AlexNet, replace_conv2, replace_conv4, and replace_conv5 models. For a more intuitive display, the results shown in Table I are also presented in Figure 7. It can be found that the worst performance was almost always achieved in the CS class, because ships in this class are quite similar to ships in other classes, with only small tonnage differences and similar shapes, and are divided into different categories based on a rough division strategy. Some confusion can be attributed to the vague images manually captured from Google Earth.

We averaged the classification results of these models, as shown in Figure 8. We note that the performance of the green and brown bars is obviously better than that of the blue bar, and the green one performed best. It is worth noting that the average accuracies of the four models with the raw dataset (blue bar) are less than 0.9000, while the four models with the fusion-2 dataset (brown bar) have average accuracies very close to 0.9000, and the average accuracies of the four models with the fusion-1 dataset (green bar) are mostly greater than 0.9000. In the process of abstracting high-level features, DCNN tends to lose shallow features, i.e., edge information. Our preprocessing method enhances the edge information of ships in the image, somewhat compensates for the loss of shallow features, and improves

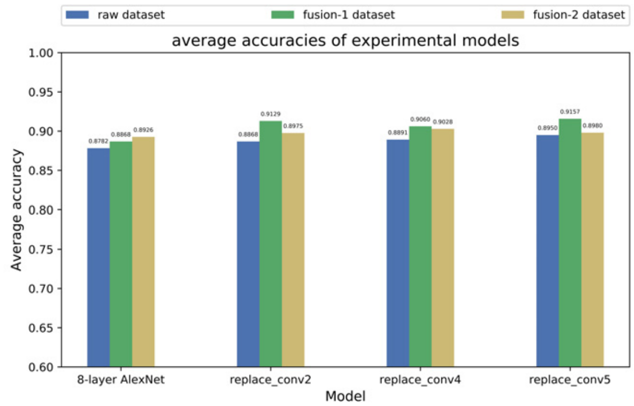

Figure 8. Bar chart of average accuracies of experimental models. It can be seen that the green bar, which represents the fusion-1 dataset, has the highest average accuracies overall. The replace_conv5 model with the fusion-1 dataset has the best average accuracy of 91.57%, while the AlexNet model with the raw dataset has the lowest accuracy, 87.82%.

Table II. Parameter settings of different models.

	VGG-16	ResNet-50	DenseNet	Replace_conv5
Initial learning rate	0.00001	0.00001	0.0006	0.0009
Epoch	100	300	500	100

the accuracy of network classification. Using the rotational invariance of the two-dimensional Gaussian distribution, the most suitable weight coefficients for the combination of the two operators are found, so that the fused image has richer edge information and less noise misdetection, thereby improving the accuracy of network classification. It is proved that our method had a good effect in the experiments, and the fusion-1 dataset obtained the best average accuracies.

We also analyzed the performance of each category in different models, as shown in Figure 9. We notice that the classification performance is generally better after replacing the convolutional layer. In these three subgraphs, we also found that the improvement of replace_conv5 was more obvious than for the other models. This is because the output of the fifth layer is the input of the fully connected layer, and after replacing the fifth convolutional layer with the Inception module, the multi-convolution operation can concatenate more feature information, which increases of the feature information entering the fully connected layer, thus affecting the last softmax layer.

To confirm the effectiveness of the Inception module, we compared the replace_conv5 model with other representative deep neural networks running on the raw dataset. Table II shows the parameter settings of different models.

The results of four models in each category are shown in Table III. It can be seen intuitively from Figure 10 that VGG-16 and replace_conv5 get better effects in five categories. VGG-16 has the highest accuracy rate among the three categories of CB, CM, and CS. DenseNet and replace_conv5 have the highest accuracy rates in the ML

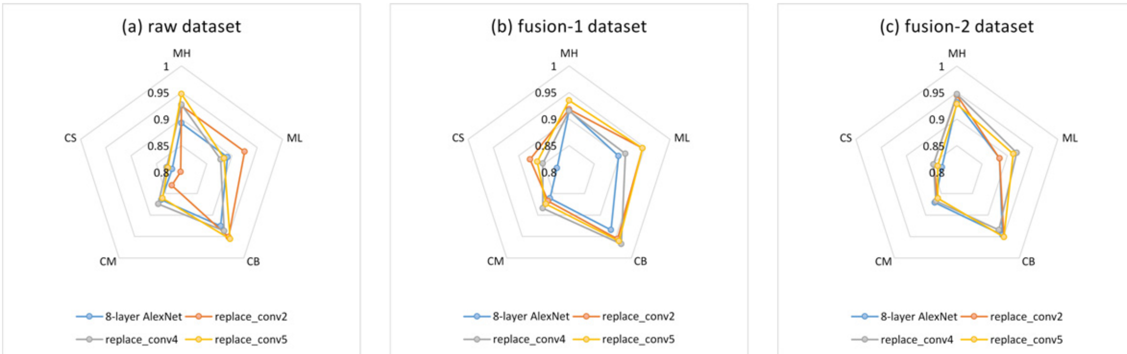

Figure 9. Radar charts representing trends of classification results of the five categories among the three experimental datasets. (a) raw dataset; (b) fusion-1 dataset; (c) fusion-2 dataset. In general, it can be seen that replacing the convolutional layer with the Inception module improves classification results.

Table III. Classification results of the raw dataset with five different models.

	MH	ML	CB	CM	CB	Average accuracy
VGG-16	0.7780	0.9624	0.9611	0.9525	0.8452	0.8998
ResNet-50	0.6534	0.8055	0.9165	0.4676	0.5743	0.6835
DenseNet	0.8366	0.9749	0.8370	0.7858	0.6987	0.7866
Replace_conv5	0.9471	0.8844	0.9557	0.8609	0.8268	0.8950

and MH categories respectively while ResNet-50 performs poorly.

VGG-16 obtained the best average accuracy, reaching 89.98%, while replace_conv5 achieved an average accuracy of 89.50% similar to that of VGG-16. It can be seen that the proposed model had better average accuracy than the other networks. The idea of VGG is to increase the depth of the network and reduce the size of the convolution kernel, which is the predecessor of the Inception module, which concatenates the extracted feature information through its multi-convolution operation, making the output feature information more sufficient, compared with a single convolution kernel. When the target is located at different positions in the image and its size is different, the Inception module can improve the robustness of convolution and improve the classification accuracy.

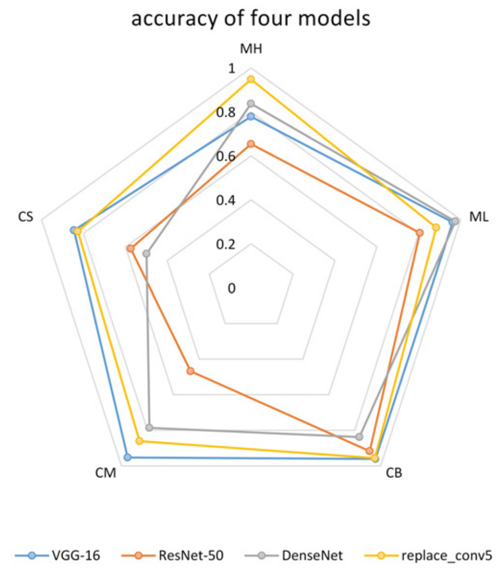

Figure 10. Radar chart showing the classification accuracies of VGG-16, ResNet-50, DenseNet, and replace_conv5 models.

5. CONCLUSIONS

Our proposed method is to preprocess RS ship images to obtain edge features using a combination of the Sobel and Canny operators, and replace the convolutional layer with the Inception module in experiments. The approach seeks to clearly identify the factors promoting the classification results of the DCNN and facilitate the indirect interpretation of DCNN functionality, because the high-level features abstracted by the DCNN, and the features obtained by multi-convolution concatenation of the Inception module, must ultimately derive from the edge information of the input images. This indicates that the predicted classification results are based on the input edge features, which indirectly interpret the classification results to some extent. We selected two weight coefficients to make the datasets, and replaced the second, fourth and fifth convolutional layers with the Inception module. Our experimental results showed that the proposed preprocessing method is effective, the fusion-1 dataset has the highest average accuracy of the models, at 90.54%, and the raw dataset has only 88.72% accuracy. The replacement of the convolutional layer with the Inception module was also validated. The performance of the replace_conv5 model on the raw dataset was 89.50%, which was close to VGG-16 and significantly higher than other networks. The present study has room for improvement. For example, the experimental ship dataset can be expanded, and a more representative feature can be extracted. Finally, while the present work helps to address the interpretation of the classification results, future work should concentrate more on deep learning frameworks.

ACKNOWLEDGMENT

The authors would like to thank the anonymous reviewers who patiently reviewed this letter and gave valuable feedback.

Disclosure Statement

The authors have no relevant financial interests in the manuscript and no other potential conflicts of interest to disclose.

Funding

This work was supported in part by the National Natural Science Foundation of China (Project No. 41701531) and National Key Laboratory Foundation (Contract No. 6142411193209).

REFERENCES

1 Q. Zheng, P. Zhao, Y. Li, H. Wang, and Y. Yang, "Spectrum interference-based two-level data augmentation method in deep learning for automatic modulation classification," Neural Comput. Appl. **33**, 7723–7745 (2020).

2 J. Xiao, Y. Wang, and H. Su, "Combining support vector machines with distancebased relative competence weighting for remote sensing image classification: A case study," J. Imaging Sci. Technol. **64**, 01 (2019).

3 A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," *Advances in Neural Information Processing Systems* (Morgan Kaufmann Publishers, USA, 2012), pp. 1097–1105.

4 K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," *Computer Science* (2014).

5 C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich, "Going deeper with convolutions," *Proc. IEEE Conf. on Computer Vision and Pattern Recognition* (IEEE, Piscataway, NJ, 2015), pp. 1–9.

6 K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," *Proc. IEEE Conf. on Computer Vision and Pattern Recognition* (IEEE, Piscataway, NJ, 2016), pp. 770–778.

7 Q. Zheng, M. Yang, X. Tian, N. Jiang, and D. Wang, "A full stage data augmentation method in deep convolutional neural network for natural image classification," Discrete Dyn. Nature Soc. **2020**, 1–11 (2020).

8 G. Alain and Y. Bengio, Understanding intermediate layers using linear classifier probes. 2016.

9 J. Yosinski, J. Clune, Y. Bengio, and H. Lipson, "How transferable are features in deep neural networks?," *NIPS'14: Proc. 27th Int'l. Conf. on Neural Information Processing Systems* (ACM, New York, NY, 2014), Vol. 27, pp. 3320–3328.

10 D. Bau, B. Zhou, A. Khosla, A. Oliva, and A. Torralba, "Network dissection: Quantifying interpretability of deep visual representations," *Proc. IEEE Conf. on Computer Vision and Pattern Recognition* (IEEE, Piscataway, NJ, 2017), pp. 6541–6549.

11 Q. Wang, Y. Zheng, G. Yang, W. Jin, X. Chen, and Y. Yin, "Multiscale rotation-invariant convolutional neural networks for lung texture classification," IEEE J. Biomed. Health Inform. **22**, 184–195 (2017).

12 X. Yang, H. Sun, K. Fu, J. Yang, X. Sun, M. Yan, and Z. Guo, "Automatic ship detection in remote sensing images from google earth of complex scenes based on multiscale rotation dense feature pyramid networks," Remote Sens. **10**, 132 (2018).

13 W. Gao, X. Zhang, L. Yang, and H. Liu, "An improved sobel edge detection," *2010 3rd Int'l. Conf. on Computer Science and Information Technology* (IEEE, Piscataway, NJ, 2010), Vol. 5, pp. 67–71.

14 L. Ding and A. Goshtasby, "On the canny edge detector," Pattern Recognit. **34**, 721–725 (2001).

15 G. Huang, Z. Liu, L. Van Der Maaten, and K. Q. Weinberger, "Densely connected convolutional networks," *Proc. IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)* (IEEE, Piscataway, NJ, 2017), pp. 4700–4708.

16 Q. Zheng, X. Tian, N. Jiang, and M. Yang, "Layer-wise learning based stochastic gradient descent method for the optimization of deep convolutional neural network," J. Intell. Fuzzy Syst. **37**, 1–14 (2019).

17 Q. Zheng, M. Yang, J. Yang, Q. Zhang, and X. Zhang, "Improvement of generalization ability of deep cnn via implicit regularization in two-stage training process," IEEE Access **6**, 15844–15869 (2018).

18 Q. Zheng, X. Tian, M. Yang, Y. Wu, and H. Su, "Pac-bayesian framework based drop-path method for 2d discriminative convolutional network pruning," Multidimens. Syst. Signal Process. **31** (2020).

19 F. Bousetouane and B. Morris, "Off-the-shelf cnn features for fine-grained classification of vessels in a maritime environment," *Conf. Int'l. Symposium on Visual Computing* (Springer, Cham, 2015), pp. 379–388.

20 S. Voinov, D. Krause, and E. Schwarz, "Towards automated vessel detection and type recognition from vhr optical satellite images," *IGARSS2018* (IEEE, Piscataway, NJ, 2018), pp. 4823–4826.

21 M. Leclerc, R. Tharmarasa, M. C. Florea, A. C. Boury-Brisset, T. Kirubarajan, and N. Duclos-Hindié, "Ship classification using deep learning techniques for maritime target tracking," *2018 21st Int'l. Conf. on Information Fusion (FUSION)* (IEEE, Piscataway, NJ, 2018), pp. 737–744.

22 A. Dixit and H. Wagatsuma, "Investigating the effectiveness of the sobel operator in the mca-based automatic crack detection," *2018 4th Int'l. Conf. on Optimization and Applications (ICOA)* (IEEE, Piscataway, NJ, 2018), pp. 1–6.

23 A. Haiqing Niu, B. Shaofeng Guo, C. Tao Xu, D. Tinghan Song, and E. Leping XU, "Infrared image edge extraction of cable terminal based on improved eight direction sobel operator," *2018 Int'l. Conf. on Power System Technology (POWERCON)* (IEEE, Piscataway, NJ, 2018), pp. 3295–3300.

24 Md. Khurram Monir Rabby, Brinta Chowdhury, and Jung H. Kim, "A modified canny edge detection algorithm for fruit detection & classification," *2018 10th Int'l. Conf. on Electrical and Computer Engineering (ICECE)* (IEEE, Piscataway, NJ, 2018), pp. 237–240.

25 E. Reinhard, W. Heidrich, P. Debevec, S. Pattanaik, G. Ward, and K. Myszkowski, *High dynamic range imaging: acquisition, display, and image-based lighting* (Morgan Kaufmann, USA, 2010).