RESEARCH ARTICLE

Classifying Ships in SAR Images by Using Contour Bias Features and Transfer Learning

YANG LIU, TIAN LI, KUN YUAN, AND SHA LI
School of Computer and Information Technology, Liaoning Normal University, Dalian 611000, China
Corresponding author: Yang Liu (yangliu.0816@lnnu.edu.cn)

This work was supported in part by the National Nature Science Foundation of China under Grant 62002146, and in part by the Liaoning Revitalization Talents Program under Grant XLYC1807073.

ABSTRACT

With the development of synthetic-aperture radar (SAR) image interpretation technology in recent years, classifying detected ships based on SAR images has become an important trend in ocean monitoring. However, owing to the underlying imaging mechanism, the texture of SAR images typically contains noise that cannot be easily eliminated. Therefore, in this study, we consider more stable target contour features to classify SAR images of marine vessels. Based on a deep learning algorithm, we propose a method to obtain contour bias features by employing style transfer learning to classify detected ships from SAR data. We also adopt transfer learning to improve the uneven distribution of SAR datasets of representative ships. The results of experiments conducted to evaluate the proposed approach show that contour bias features improve the generalization performance and classification accuracy of the model. They also show that transfer learning effectively avoids the problem of data imbalance and, thus, improves classification accuracy on the OpenSARShip 2.0 database.

INDEX TERMS

SAR image, contour bias, transfer learning, classification.

I. INTRODUCTION

With the increasing density of marine shipping and the development of ship management automation technology, the demand for fine management of marine ships is increasing. Ship monitoring technology has been developed and widely adopted in navigational systems for narrow waters, inland rivers, and ports. However, existing monitoring networks still involve some blind spots for remote waters or non-cooperative targets. Ship classification based on synthetic-aperture radar (SAR) imaging have been proposed and applied in this context. SAR is an active microwave imaging sensing modality. Based on synthetic-aperture technology, SAR simultaneously produces two-dimensional high-resolution images by transmitting broadband signals in both the range and azimuth directions. Compared with traditional optical imaging. SAR serves as an important technique to obtain information to classify ground objects and monitor natural disaster areas owing to its capacity to penetrate physical objects and obtain images reflecting the microwave scattering features of targets. At present, SAR has been widely adopted in military and civic fields to realize target detection and classify ground objects; therefore, methods to identify, classify, and monitor ships based on SAR have attracted considerable attention as a topic of active research.

The imaging principle of SAR is complex because the results are not only related to the wavelength, angle of incidence, and polarization mode of the radar, but also to the geometric features and material of the target. Owing to the difficulty in understanding SAR images, research on SAR image target interpretation technology has largely been based on existing optical image target interpretation algorithms combined with some particularities of SAR images. These methods process SAR images as optical images. Currently, ship classification based on SAR images is mainly divided into traditional feature-based and machine-learning-based classification methods. In the former case, features are extracted manually and input to a classifier. Commonly used features include (1) geometric features such as ship length, aspect ratio, covariance coefficient, contour features, and ship
scale, and (2) scattering features such as two-dimensional comb features, local radar cross-section density, permanent symmetric scatterer, and polarization features. These methods are based on advanced and interpretable mathematical theories as an essential prerequisite, and are thus highly targeted and difficult to generalize. In contrast, classification methods based on machine learning train a classification model to recognize and classify testing samples on the basis of sufficient training samples. As an important branch of machine learning with accelerated data processing, good fault tolerance, and self-training capabilities, deep learning can process signal information produced in complex environments, with unknown background knowledge, or by unclear reasoning rules. It also allows samples to have defects or distortions, and therefore provides adaptive performance and provides high classification accuracy. However, the deep learning process requires a large number of labeled training samples to train a classification model. In addition, the training and testing data should exhibit the same data distribution.

At present, ship classification from SAR image data still involves many challenges. First, relatively few high-resolution data are available to describe ship SAR images, and samples in the existing datasets are unevenly distributed. In addition, given the particularities of the SAR imaging modality, expert interpretation is usually required to fully utilize SAR images to obtain improved classification accuracy. Considering these two factors, many methods have been proposed to apply transfer learning to SAR image classification. The amount of time and training data required to train a model can be reduced by reusing the knowledge of another model trained in another related task or domain. At present, there is a SAR database with good image quality, and considering that there will be the same imaging features among the same imaging mechanism, in this paper, we choose to learn and extract the imaging features of the source domain and apply them to the target domain.

Transfer learning can also enable deep learning models to automatically obtain edge and texture features from labeled images. Geirhos et al. [1] pointed out in 2018 that convolutional neural network (CNN) classification methods generally rely too heavily on object textures rather than global object shapes; however, SAR image textures tend to exhibit considerable noise, which cannot be perfectly removed without losing image quality. In addition, Lang et al. [2] pointed out in 2017 that the resolution of scattering features would decrease and vary with the angle of incidence in these medium-resolution images for ship SAR detection and classification, SAR sensor features, and the imaging environment due to the features of typical ships.

The contributions of this study are summarized as follows.

1) Our experimental results reveal that the obtained contour bias feature is more stable than the combination of the texture feature and contour feature, which can significantly improve accuracy and precision in SAR classification tasks.

2) Moreover, the results show that contour bias features can be used effectively in transfer learning, because the model can be trained to classify samples with improved accuracy by adjusting the feature learning preferences and influencing the feature weights.

This remainder of this paper is organized as follows. In Section II, we first summarize existing machine learning methods that exhibit good performance in SAR image classification and then focus on deep learning methods. In Section III, we detail the proposed transfer learning classification method for SAR images based on contour bias features. In Section IV, we present the results of experiments conducted to verify the effectiveness of the proposed approach by analyzing the features obtained from deep learning and comparing different models. Finally, in Section V, we provide conclusions and suggest some possible avenues for future research.

II. RELATED CONTENTS

At present, research on SAR image classification is primarily based on machine learning, which can be divided into traditional machine learning and deep learning methods. Classification based on traditional machine learning models usually requires researchers to be familiar with the imaging principle of SAR images and design features manually for input into predictive classifiers. In contrast, classification methods based on deep learning establish a relatively more complex model designed to learn the relationship between input data and their associated label and thus automatically extract features for classification. Compared with traditional machine learning methods, this approach reduces labor and time costs and improves generalization and classification accuracy.

In the application of traditional machine learning methods to recognize and classify ship SAR images, some methods have achieved better classification accuracy by integrating feature models in line with SAR image targets into classifiers. Lang et al. [2] proposed simple and efficient naive geometric features method for ship classification, which was designed to ascertain the essential differences between different types of ships. Multiple kernel learning was developed to learn combination weights. Lin et al. [3] proposed an improved SAR-HOG model in 2018 and applied it to ship classification using manifold learning to achieve dimensionality reduction. They also integrated a dictionary into their proposed classifier in the task-driven dictionary learning (TDDL) framework and imposed structured incoherent constraints on TDDL to achieve better classification results. In 2011, Margarit et al. developed a method to classify ship according to a fuzzy logic decision rule by combining scattering and geometric features of targets. [4]. In 2013, Xing et al. [5] combined different structural and scattering features of targets to construct an internal representation, and then used the sparse representation method to classify them. Wu et al. [6] proposed a ship classification method in 2015  which was designed to estimate feature vectors by calculating the average value of kernel density estimation, three structural features, and an average backscattering coefficient, and then to classify ships using a
support-vector machine model. In 2017, Gorovyi et al. [7] proposed the use of Haralick features for texture recognition and local binary patterns as a comparison standard and classified images by fusing azimuth and range target profiles.

Some methods have achieved improved classification performance by implementing more effective classifiers. Along these lines, Wang et al. [8] studied a hierarchical classifier based on geometric and scattering features to classify ships. In 2019, Gishkori et al. [9] proposed a method that used the invariance of pseudo-Zernike distance and introduced auxiliary atom to construct a sparse dictionary and increase the redundancy of dictionaries to perform image classification. Clustering algorithms have also been considered for this purpose. For example, in 2020, Xu et al. [10] proposed a new distance metric learning method that improved the original Laplacian regularized metric learning by adding inter-class distribution shift regularization. This method can effectively improve the separability between classes and the compression within classes, as well as the performance of fine-grained ship classification from SAR images. The author introduced a new transfer metric learning method [11], which integrated pairwise constraint, joint distribution adaptation, and geometric structure preservation to realize discriminant information preservation, geometric structure preservation, and domain displacement, and transformed manifold regularization into a unified optimization function to fully utilize their complementarity to improve the performance of SAR ship classification systems.

Most existing deep learning methods for SAR ship classification are based on CNN models. In 2017, Lin et al. [12] introduced a CNN-based architecture designed to be trained with smaller datasets, and the network was flexibly constructed by stacking a unit architecture to extract deep feature representations for classification. In 2016, Chen et al. developed a highly targeted A_ConvNets network for the MSTAR database, which achieved experimental accuracy that could not be replicated on other databases. [13]. In 2019, De Laurentiis et al. [14] proposed a CapsNets model for ship classification, which was designed to improve the use of spatial information. Dong et al. [15] developed a deep residual network for fine-grained ship classification to achieve high accuracy; however, its data are too simple to be widely used in other SAR images. He et al. [16] proposed extending a dense convolutional network to medium-resolution SAR ship classification and used a multi-task learning framework to minimize softmax log loss and triple loss to better extract deep features and improve classification accuracy. Some researchers have also considered the features of SAR images. For example, Huang et al. [17] proposed a novel deep learning framework that considers complex SAR images, spatial texture information, and backscattering patterns of ground objects. Zhang et al. [18] proposed a method that fused directional gradient histogram features with neural networks.

Most of these deep learning approaches are supervised machine learning methods that require a large, balanced dataset with high-quality annotations. Some researchers have applied transfer learning to the SAR field to better overcome the limited amount of data available and the imbalance of samples in typical datasets. For example, Wang et al. [19] proposed in 2018 to migrate a deep convolutional network pre-trained on the ImageNet dataset to small-sample SAR ship classification to extract abstract features of ships. Huang et al. [20] comprehensively studied where and how to transmit signal in SAR automatic target recognition and improved the accuracy with which a system could classify ships by fine-tuning a deep network trained on other databases. In 2020 [21], a deep migration learning method based on similarly annotated optical land-cover datasets with similar labels was proposed, and a top-2 loss smoothing loss function with cost-sensitive parameters was introduced to solve the problem of label noise and imbalance.

III. METHOD

Inspired by transfer learning, in this study, we propose a transfer learning method that adopts a model trained on a balanced SAR image set to an unbalanced SAR image set. We consider two SAR image datasets, including MSTAR [22] and OpenSARShip 2.0 [23]. MSTAR is a SAR image dataset comprising military vehicles, which is used as the source domain for transfer learning in our proposed approach. It contains ten kinds of target images acquired of military vehicles at various azimuth angles. The data distribution is balanced, the recognized targets are located in the center of the image. The OpenSARShip 2.0 dataset comprises SAR images of ships acquired by the Sentinel-1 satellite, including 16 types of ship images acquired from different backgrounds. We adopted this dataset as the target domain for the transfer learning process in the proposed method. However, the data distribution of OpenSARShip 2.0 is very uneven, the images contain a considerable variety of content other than the target, and the ship targets account for a relatively small proportion of the image, which is not conducive to training deep learning models. Therefore, the proposed approach transfers the parameters trained on the MSTAR database to the OpenSARShip 2.0 database.

Multiplicative noise is difficult to remove in SAR imaging owing to the special imaging mechanism. Therefore, the proposed method extracts contour bias features by using the style transfer method on SAR images. One SAR image is used as the content image and ten different style images are used for style transfer. The SAR image after style transfer retains the original contour but exhibits different textures. When the deep model learns these images with different textures but is marked with the same label, the contour bias features are obtained, and SAR images of different styles are generated. Our experimental results show that this can improve classification accuracy. In addition, we also propose the implementation of transfer learning from tank SAR images to ship SAR images, and our experimental results demonstrate prove that this approach is an effective method to improve the accuracy of ship classification methods based on SAR images.
The key contributions of the present work can be divided into two parts. 1) Because image noise in the MSTAR dataset cannot be removed without degrading the quality of the image, stylized images are used for training, and the network is forced to learn more stable contour bias features for classification. 2) Owing to the poor image quality, relatively few samples, and unbalanced data in the OpenSARShip 2.0 dataset, deep learning models typically overfit the data. To avoid this phenomenon, our proposed approach implements transfer learning from the tank dataset to the ship dataset. The model is trained with the MSTAR dataset, transfer learning is performed to the OpenSARShip 2.0 dataset, and then the last several layers of the network are fine-tuned. The process flow of the proposed approach is shown in Fig. 1.

A. CLASSIFICATION MODEL

In the classification model, this study chose the CNN, which is widely used and has many derivative models. A CNN is composed of an activation function, convolutional layers, pooling layers, and fully connected layers. For any given input image, the neural network cannot know exactly which parts of comprise the salient features of the image are, so every pixel $X$ in the original image is considered in the convolutional process. A convolution operation is performed with a square matrix of a given $N \times N$ size, called a convolution kernel. The convolution kernel $K$ is traversed from beginning to end through a picture of size $W \times W \times D$, as shown in Equation 3.1, where $D$ is the depth of the picture consistent with the depth of the convolution kernel.

$$z_{i,j} = \sum_{i=1}^{N} \sum_{j=1}^{N} K_{ij}D_{i,j}$$  \hspace{1cm} (3.1)

In addition, there are some parameters to determine how to convolve, for example, padding, which determines whether to fill zeros around the input picture, and strides, which determine the distance that the convolution kernel moves. Finally, the size of the output matrix is $M \times M$ (see Equation 3.2).

$$M = \frac{W - N + 2P}{S} + 1.$$  \hspace{1cm} (3.2)

Convolution is essentially an input-output mapping that can learn a large number of mapping relationships between the input and output without any accurate mathematical expressions between the input and output. A convolutional neural network trained using known patterns learns a mapping between the input and output. The activation function is a nonlinear function that maps the output results of the convolutional layer; commonly used activation functions are Sigmoid, Tanh, ReLu.

The pooling layer reduces the computational complexity from the upper layer, and simultaneously exhibits translation invariance. The extracted features remain unchanged even with a small displacement, which can eliminate the influence of image distortion. According to specific needs, we can choose max pooling, that is, keeping the maximum value in the receptive field, or mean pooling, which retains the average value in the receptive field.

The fully connected layer plays a classification role in the entire neural network; after deep layers such as convolutional, activation, or pooling, features are identified and classified through the fully connected layer. Finally, the backpropagation algorithm is used to optimize the network structure, and the weights are adjusted by calculating the difference between the actual and ideal outputs.

B. IMAGE STYLE TRANSFER

SAR images are processed by using image-style transfer. First, the SAR imaging system is based on the coherence principle; that is, in the radar echo signal, the gray value of adjacent pixels exhibits some random changes owing to coherence, and this random change is varies around a certain mean value. Thus, speckle noise is produced in the image, and at present no method exists to remove such noise without losing the image information. Image style transfer is adopted to weaken the influence of image texture on image classification, preserve image contour information, and to guide the network learn more stable contour bias information for classification. Secondly, compared with the general datasets used in optical image classification, the MSTAR database has less data and is prone to overfitting. Therefore, we adopt image style transfer processing, and one SAR image is processed with ten different style images to generate ten stylized images with different textures and consistent contours (Fig. 4) to augment the data. Moreover, we verified that image-style transfer processing can improve the accuracy of image classification.

In the experimental evaluation, images from the MSTAR dataset were used as content images, images from Kaggle’s Painter by Numbers dataset were used as style images, ten different style images and content images were randomly selected, and the AdaIN style transfer algorithm [24] was used for style transfer. First, the content image $c$ and style image $s$ are input into the VGG network, and the feature images obtained from them are saved as $c'$ and $s'$, respectively. According to Equation 3.3, we first standardize the content image, then multiply it by the standard deviation of the style image and add the mean value of the style image to keep the mean value and standard deviation of the content image consistent with the mean value and standard deviation of the style image as much as possible, where $\sigma$ and $\mu$ are the mean value and standard deviation, respectively. The results are then input to a VGG decoder model to obtain the initial stylized image, as shown
in Equation 3.4, where \( g \) represents the VGG decoder.

\[
\begin{align*}
t &= \text{AdaIN} \left( c', s \right) = c' + \left( c' - \mu(c') \right) \left( \frac{s}{\sigma(s)} \right) + \mu(s), \\
T(c, s) &= g(t)
\end{align*}
\]

(3.3) (3.4)

To make the stylized image closer to the style image, the initial stylized image is input into the VGG network to obtain its feature map \( t' \), and then compared with the original stylized and content images, and the weighted loss of the initial stylized image and the sum of the two are optimized. The total loss is shown in Equation 3.5, which consists of content loss \( L_c \) and style loss \( L_s \), where \( \lambda \) is the weight hyperparameter. The content loss, \( L_c \), is the two-norm of the contraposition difference obtained by \( t \) and \( t' \) feature graphs, as shown in Equation 3.6. The style loss, \( L_s \), is the two-norm of the mean and standard deviation of each channel of each layer feature map between \( t' \) and \( s \), where \( l \) is the subscript of the convolutional layer in VGG (see Equation 3.7).

\[
\begin{align*}
L &= L_c + \lambda L_s, \\
L_c &= \|t' - t\|_2^2, \\
L_s &= \sum_{l=1}^{L} \| \mu(\phi_l(t')) - \mu(\phi_l(s)) \|_2 + \sum_{l=1}^{L} \| \sigma(\phi_l(t')) - \sigma(\phi_l(s)) \|_2.
\end{align*}
\]

(3.5) (3.6) (3.7)

Two experiments were conducted to compare and analyze the effects of extracting more stable contour bias features from the stylized training images, as described below.

Experiment 1 (Expt-1): In the first experiment, we used the original MSTAR images as the training and verification datasets.

Experiment 2 (Expt-2): The stylized MSTAR image was used as the training set, and the original MSTAR image was used as the verification set.

The models for the two experiments were trained separately using the same deep learning network, and the extracted feature maps were retained for visual analysis.
using the TensorFlow open-source machine learning library. CUDA 11.4 was used to call the GPU for accelerated training.

A. SAR IMAGE STYLE TRANSFER

In this study, the Painter by Numbers dataset from Kaggle was used as the style source, and the texture of images was transferred to the MSTAR database to construct a new dataset, referred to as Stylized-MSTAR Data. The results of image style transfer are shown in Fig. 4. It may be observed from the image that after the style transfer, the SAR image in MSTAR changes the original image texture by retaining the original target contour and adding texture features and color information consistent with the style image.

B. APPLICATION ANALYSIS OF STYLE TRANSFER

To compare the MSTAR database with the Stylized-MSTAR database, we evaluated them with the same general classification network. Under the same network structure and using the same batch size, loss function, and optimization function, the learning rate and its decay speed were adjusted for different databases for the best comparison. See Table 1 for the database category distributions for Expt-1 and Expt-2.

To compare Expt-1 with Expt-2, we used two kinds of datasets with the AlexNet network, set the same parameters, such as batch size, epochs, and loss function and optimization functions, and set learning rate.

Accuracy of the verification set converged at approximately 15 epochs without obvious fluctuation. In terms of classification accuracy, in Table 2, Expt-2 exhibited better performance than Expt-1, which shows that after image style transfer, neural networks show an obvious bias to contour feature response. By analyzing the results of the comparison between Expt-1 and Expt-2, we found that when using SAR images after style transfer as contour features extracted from training sets for classification, the accuracy of classification fluctuated less and steadily improved. According to Expt-3, we found that style transfer is an effective data enhancement method that can improve the performance of learning models without any architectural changes. Compared with Expt-4, the results of Expt-2 show that the network only exhibits a strong generalization ability for stylized images. Although such images can be used to transfer the features learned from stylized MSTAR images to the MSTAR database, transferring the features learned from the MSTAR database to stylized MSTAR images was difficult.

For a better comparison between Expt-1 and Expt-2, the accuracy, recall, and other evaluation indices were recorded and compared to obtain the highest accuracy, as shown in Table 3. The precision rate of Expt-2 shown in the table is much higher than that of Expt-1, indicating that the contour bias features extracted by the model are more stable, significantly reducing the overfitting phenomenon, and the experimental method was more effective.

In the comparison between Expt-1 and Expt-2, by analyzing the training chart (Figure 5), we used the same network, batch size, loss function, and optimization function for both experiments (VGG11) and observed the trend of the resultant curve when adjusting parameters to achieve their respective optimal results. Under the same epochs, the network convergence in Expt-2 was obviously faster than that in Expt-1, the fluctuation of the verification set in Expt-2 was smaller, and the network was more stable.

C. APPLICATION ANALYSIS OF PARAMETER TRANSFER

After obtaining the weight of MSTAR training, transfer learning was adopted for training, and the parameters trained on the MSTAR database were transferred to the OpenSARShip 2.0 database considering the small number of samples in OpenSARShip 2.0 databases and uneven distribution among classes. During transfer learning, the top-level parameters can be fine-tuned, or all transfer weights can be taken as initialization parameters before training the entire network. In this study, we only performed fine-tuning for the parameters of the top layer, because the MSTAR and OpenSARShip 2.0 databases comprise SAR images with similar bottom-level features, and the abstract features of the top layer are different. Fine-tuning of the convolutional kernel of the top layer can be performed. If only weights are used as initialization parameters, the efficiency of the retraining parameters is relatively low.

| Type | No. of | No. of | No. of | No. of |
|------|--------|--------|--------|--------|
|      | train  | valid  | train  | valid  |
| 2S1  | 931    | 233    | 1165   | 582    |
| BMP2 | 558    | 140    | 1165   | 349    |
| BRDM | 1132   | 283    | 1165   | 707    |
| BTR  | 361    | 90     | 804    | 226    |
| BTR70| 186    | 47     | 1000   | 117    |
| D7   | 458    | 115    | 1129   | 286    |
| T62  | 458    | 114    | 1142   | 286    |
| T72  | 553    | 138    | 1165   | 346    |
| ZIL131| 458    | 115    | 1165   | 286    |
| ZSU  | 1121   | 280    | 1165   | 701    |

TABLE 1. Data distribution for Expt-1 and Expt-2.
TABLE 2. Test accuracy of MSTAR database under different classical networks for each experiment.

| Method      | MSTAR→MSTAR (Expt-1) | Stylized-MSTAR→MSTAR (Expt-2) | Stylized-MSTAR→Stylized-MSTAR (Expt-3) | MSTAR→Stylized-MSTAR (Expt-4) |
|-------------|-----------------------|-------------------------------|----------------------------------------|-------------------------------|
| LeNet[25]   | 97.15±0.31            | 98.38±0.24                    | 99.91±0.47                             | 48.86±1.77                    |
| AlexNet[26] | 98.24±0.39            | 99.76±0.184                   | 99.36±0.37                             | 40.62±2.45                    |
| VGG11[27]   | 98.92±0.49            | 99.67±0.27                    | 99.08±0.34                             | 95.52±0.26                    |
| VGG16[27]   | 97.97±1.25            | 99.05±0.74                    | 98.09±0.94                             | 95.68±0.99                    |
| VGG19[27]   | 94.14±2.28            | 98.06±1.27                    | 98.24±1.09                             | 92.83±2.59                    |
| GoogleNet[28]| 95.15±0.16            | 98.28±0.13                    | 97.56±0.19                             | 15.28±1.07                    |
| ResNet18[29] | 98.65±6.68            | 99.44±0.46                    | 98.73±0.84                             | 47.10±14.11                   |
| ResNet50[29] | 95.18±8.12            | 96.71±0.37                    | 98.62±0.43                             | 62.66±18.64                   |
| DenseNet121[30] | 98.79±0.745        | 98.71±0.825                   | 98.8±0.15                              | 27.29±1.25                    |
| DenseNet169[30] | 99.82±0.79         | 98.57±0.81                    | 98.53±0.98                             | 28.71±1.08                    |
| MobileNet v1[31] | 95.70±0.72        | 96.08±0.38                    | 99.15±0.43                             | 22.91±1.62                    |
| Inception v4[32] | 96.87±0.53         | 97.73±0.21                    | 99.16±0.37                             | 40.22±12.49                   |
| A_ConvNets[13] | 98.96±0.39          | 98.49±0.31                    | 98.89±0.61                             | 93.34±0.93                    |

TABLE 3. Quantitative evaluation indices of the MSTAR database classification results under different classical networks in Expt-1 and Expt-2.

| Method      | MSTAR→MSTAR (Expt-1) | Stylized-MSTAR→MSTAR (Expt-2) | Accuracy |
|-------------|-----------------------|-------------------------------|----------|
| LeNet[25]   | 96.53%                | 96.76%                        | 96.39%   | 97.46%   | 98.33%  | 98.34%  | 98.33%  | 98.62%  |
| AlexNet[26] | 97.36%                | 97.65%                        | 97.40%   | 98.63%   | 99.85%  | 99.85%  | 99.85%  | 99.84%  |
| VGG11[27]   | 98.91%                | 98.93%                        | 98.91%   | 99.41%   | 99.64%  | 99.64%  | 99.64%  | 99.84%  |
| VGG16[27]   | 99.04%                | 99.05%                        | 99.04%   | 99.22%   | 99.77%  | 99.77%  | 99.77%  | 99.79%  |
| VGG19[27]   | 95.69%                | 95.99%                        | 95.75%   | 96.42%   | 99.82%  | 99.82%  | 99.82%  | 99.84%  |
| GoogleNet[28] | 44.24%                | 46.21%                        | 38.18%   | 95.31%   | 93.93%  | 94.30%  | 93.92%  | 98.41%  |
| ResNet18[29] | 99.23%                | 99.24%                        | 99.23%   | 99.35%   | 99.87%  | 99.87%  | 99.87%  | 99.90%  |
| ResNet50[29] | 90.55%                | 90.63%                        | 90.63%   | 96.00%   | 96.68%  | 96.79%  | 96.69%  | 97.08%  |
| DenseNet121[30] | 86.04%                | 93.87%                        | 87.63%   | 99.50%   | 98.04%  | 98.19%  | 98.07%  | 98.31%  |
| DenseNet169[30] | 95.62%                | 96.23%                        | 95.46%   | 99.61%   | 99.02%  | 99.04%  | 99.02%  | 99.38%  |
| MobileNet v1[31] | 90.61%                | 91.76%                        | 90.09%   | 96.42%   | 96.32%  | 96.53%  | 96.35%  | 96.46%  |
| Inception v4[32] | 97.55%                | 97.55%                        | 97.52%   | 97.40%   | 97.91%  | 97.95%  | 97.91%  | 97.94%  |
| A_ConvNets[13] | 98.90%                | 98.92%                        | 98.90%   | 99.35%   | 99.87%  | 99.89%  | 99.89%  | 99.95%  |

FIGURE 5. Training result diagram. (a) Result of the training of Expt-1 on VGG11. (b) Result of the training of Expt-2 on VGG1.

The OpenSARShip 2.0 source database include many images with very small sizes. We input these samples into the network and expanded them to an input image size of 224. The interpolation pixel content affects the extraction of image features. Therefore, before the experiment, we performed data cleaning and eliminated the images with length or height less than 50 pixels. The final data distribution is shown in Table 4.

The previously trained parameters were transferred to OpenSARShip 2.0, and the results were fine-tuned, as shown in Table 5. The fine-tuning areas selected for different networks differed. For example, the LeNet network includes relatively few network layers, and only the parameters of the fully connected layer are fine-tuned, whereas Inception v4 network does not include a fully connected layer, and the parameters of the last inception module are adjusted in the
TABLE 4. OpenSARShip 2.0 database.

| Type                 | No. of train | No. of valid |
|----------------------|--------------|--------------|
| Cargo                | 3502         | 3503         |
| Dredging or underwater ops | 57           | 57           |
| Fishing              | 14           | 14           |
| High speed craft     | 1            | 1            |
| Passenger            | 40           | 41           |
| Port Tender          | 2            | 2            |
| Search and Rescue vessel | 3            | 3            |
| Tanker               | 892          | 892          |
| Tag                  | 14           | 14           |
| Wing in ground       | 3            | 3            |

TABLE 5. Comparison of the results of transfer parameters obtained by two methods to OpenSARShip 2.0 database.

| Method                        | MSAR→OpenSARShip 2.0 | Stylized-MSAR→OpenSARShip 2.0 | FGSCR→OpenSARShip 2.0 |
|-------------------------------|-----------------------|-------------------------------|------------------------|
| LeNet[25]                     | 77.66±0.23            | 77.68±0.86                    | 77.67±0.92            |
| AlexNet[26]                   | 78.33±0.525           | 79.39±0.425                   | 76.86±0.25            |
| VGG11[27]                     | 77.69±0.19            | 77.81±0.17                    | 76.26±0.7             |
| VGG16[27]                     | 77.99±0.17            | 78.51±0.17                    | 77.55±0.7             |
| VGG19[27]                     | 78.59±0.204           | 78.71±0.192                   | 77.71±0.24            |
| GoogleNet[28]                 | 77.58±0.01            | 77.58±0.01                    | 76.24±0.01            |
| ResNet18[29]                  | 78.08±0.5             | 78.25±0.43                    | 77.65±0.7             |
| ResNet50[29]                  | 77.49±0.01            | 77.53±0.01                    | 76.08±0.01            |
| DenseNet121[30]               | 77.71±0.325           | 77.99±0.245                   | 76.52±0.57            |
| DenseNet169[30]               | 77.64±0.00            | 78.02±0.035                   | 77.32±0.7             |
| MobileNet_v1[31]              | 77.25±0.01            | 77.51±0.03                    | 77.14±0.7             |
| Inception_v4[32]              | 77.45±0.1             | 77.53±0.17                    | 77.48±0.17            |
| A_ConvNets[33]                | 77.83±0.06            | 78.51±0.185                   | 75.54±0.7             |

V. CONCLUSION

In this study, we have proposed a deep learning method to improve the accuracy of SAR image classification. Considering that noise in SAR images is difficult to divide without loss of image quality, we have first proposed a method to obtain stylized SAR images by stylize transfer to construct a SAR dataset to be used as a training set as input to a deep learning network to learn the contour bias feature of SAR images. The results of an experiment using the original SAR image as the verification set showed that the contour bias feature can significantly improve the accuracy with which SAR images can be classified and improve the generalization performance and stability of the model. To address the problem of limited amounts of images and uneven data distribution in available SAR image dataset, we used the proposed approach to transfers the parameters obtained from learning stylized MSTAR SAR image models to the task of classifying the OpenSARShip 2.0 database and fine-tune the top-level parameters of the network for different tasks to improve the classification performance of the model. The experimental results show that the accuracy with which SAR images can be classified can be improved by using image style transfer as a pretreatment of the training set and extracting contour bias features from SAR datasets. The training parameters on SAR datasets with larger data volumes and more uniform distributions are transferred to learn to classify SAR data from small and uneven datasets. In contrast, the contour bias features obtained after style transfer and parameter transfer are more generalized and can improve classification accuracy.

REFERENCES

[1] R. Geirhos, P. Rubisch, C. Michaelis, M. Bethge, F. A. Wichmann, and W. Brendel, “ImageNet-trained CNNs are biased towards texture; increasing shape bias improves accuracy and robustness,” 2018, arXiv:1811.12231.
[2] H. Lang and S. Wu, “Ship classification in moderate-resolution SAR image by naive geometric features-combined multiple kernel learning,” IEEE Geosci. Remote Sens. Lett., vol. 14, no. 10, pp. 1765–1769, Sep. 2017.
[3] H. Lin, S. Song, and J. Yang, “Ship classification based on MSHOG feature and task-driven dictionary learning with structured incoherent constraints in SAR images,” Remote Sens., vol. 10, no. 2, p. 190, Jan. 2018.
[4] G. Margarit and A. Tabasco, “Ship classification in single-Pol SAR images based on fuzzy logic,” IEEE Trans. Geosci. Remote Sens., vol. 49, no. 8, pp. 3129–3138, Aug. 2011.
[5] X. Xing, K. Ji, H. Zou, W. Chen, and J. Sun, “Ship classification in TerraSAR-X images with feature space based sparse representation,” IEEE Geosci. Remote Sens. Lett., vol. 10, no. 6, pp. 1562–1566, Nov. 2013.
[6] F. Wu, C. Wang, S. Jiang, H. Zhang, and J. Sun, “Classification of vessels in single-pol COSMO-SkyMed images based on statistical and structural features,” Remote Sens., vol. 7, no. 5, pp. 5511–5533, May 2015.
[7] I. M. Gorovyi and D. S. Sharapov, “Efficient object classification and recognition in SAR imagery,” in Proc. 18th Int. Radar Symp. (IRS), Jun. 2017, pp. 1–7.
Y. Liu et al.: Classifying Ships in SAR Images by Using Contour Bias Features and Transfer Learning

IEEE Access

VOLUME 10, 2022

129957

[8] C. Wang, H. Zhang, F. Wu, S. Jiang, B. Zhang, and Y. Tang, “A novel hierarchical ship classifier for COSMO-SkyMed SAR data,” IEEE Geosci. Remote Sens. Lett., vol. 11, no. 2, pp. 484–488, Feb. 2014.

[9] S. Gishikari and B. Mulgrew, “Pseudo-Zernike moments based sparse representations for SAR image classification,” IEEE Trans. Aerosp. Electron. Syst., vol. 55, no. 2, pp. 1037–1044, Apr. 2019.

[10] Y. Xu and H. Lang, “Distribution shift metric learning for fine-grained ship classification in SAR images,” IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens., vol. 13, pp. 2276–2285, 2020.

[11] Y. Xu and H. Lang, “Ship classification in SAR images with geometric transfer metric learning,” IEEE Trans. Geosci. Remote Sens., vol. 59, no. 8, pp. 6799–6813, Aug. 2021.

[12] Z. Lin, K. Ji, M. Kang, X. Leng, and H. Zou, “Deep convolutional highway unit network for SAR target classification with limited labeled training data,” IEEE Trans. Geosci. Remote Sens., vol. 14, no. 7, pp. 1091–1095, Jul. 2017.

[13] S. Chen, H. Wang, F. Xu, and Y.-Q. Jin, “Target classification using the deep convolutional networks for SAR images,” IEEE Trans. Geosci. Remote Sens., vol. 54, no. 8, pp. 4806–4817, Apr. 2016.

[14] L. De Laurentiis, A. Pomenti, F. Del Frate, and G. Schiavon, “Capsule and convolutional neural network-based SAR ship classification in Sentinel-1 data,” in Active and Passive Microwave Remote Sensing for Environmental Monitoring III, vol. 11154, SPIE, Oct. 2019, p. 1115405.

[15] Y. Dong, H. Zhang, C. Wang, and Y. Wang, “Fine-grained ship classification based on deep residual learning for high-resolution SAR images,” Remote Sens. Lett., vol. 10, no. 11, pp. 1095–1104, Nov. 2019.

[16] J. He, Y. Wang, and H. Liu, “Ship classification in medium-resolution SAR images via densely connected triplet CNNs integrating Fisher discrimination regularized metric learning,” IEEE Trans. Geosci. Remote Sens., vol. 59, no. 4, pp. 3022–3039, Apr. 2021.

[17] Z. Huang, M. Datcu, Z. Pan, and B. Lei, “Deep SAR-Net: Learning objects from signals,” ISPRS J. Photogramm. Remote Sens., vol. 161, pp. 179–193, Mar. 2020.

[18] T. Zhang, X. Zhang, X. Ke, C. Liu, X. Xu, X. Zhan, C. Wang, I. Ahmad, Y. Zhou, D. Pan, J. Li, H. Su, J. Shi, and S. Wei, “HOG-ShipCLSNet: A novel deep learning network with HOG feature fusion for SAR ship classification,” IEEE Trans. Geosci. Remote Sens., vol. 60, pp. 1–22, 2022.

[19] Y. Wang, C. Wang, and H. Zhang, “Ship classification in high-resolution SAR images using deep learning of small datasets,” Sensors, vol. 18, no. 9, p. 2929, Sep. 2018.

[20] Z. Huang, Z. Pan, and B. Lei, “What, where, and how to transfer in SAR target recognition based on deep CNNs,” IEEE Trans. Geosci. Remote Sens., vol. 58, no. 4, pp. 2324–2336, Apr. 2020.

[21] Z. Huang, C. O. Dumitru, Z. Pan, B. Lei, and M. Datcu, “Classification of large-scale high-resolution SAR images with deep transfer learning,” IEEE Geosci. Remote Sens. Lett., vol. 18, no. 1, pp. 107–111, Jan. 2021.

[22] The Air Force Moving and Stationary Target Recognition Database. [Online]. Available: https://www.sdms.afrl.af.mil/datasets/mtsra/

[23] B. Li, B. Liu, L. Huang, W. Guo, Z. Zhang, and W. Yu, “OpenSARShip 2.0: A large-volume dataset for deeper interpretation of ship targets in Sentinel-1 imagery,” in Proc. SAR Big Data Era: Models, Methods Appl. (BIGSARDATA), Nov. 2017, pp. 1–5.

[24] X. Huang and S. Belongie, “Arbitrary style transfer in real-time with adaptive instance normalization,” in Proc. IEEE Int. Conf. Comput. Vis. (ICCV), Oct. 2017, pp. 1501–1510.

[25] Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner, “Gradient-based learning applied to document recognition,” Proc. IEEE, vol. 86, no. 11, pp. 2278–2324, Nov. 1998.

[26] A. Krizhevsky, I. Sutskever, and E. Hinton Geoffrey, “ImageNet classification with deep convolutional neural networks,” in Proc. Adv. Neural Inf. Process. Syst., vol. 25, 2012, pp. 1097–1105.

[27] K. Simonyan and A. Zisserman, “Very deep convolutional networks for large-scale image recognition,” 2014, arXiv:1409.1556.

[28] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich, “Going deeper with convolutions,” in Proc. IEEE Conf. Comput. Vis. Pattern Recognit., Jun. 2015, pp. 1–9.

[29] K. He, X. Zhang, S. Ren, and J. Sun, “Deep residual learning for image recognition,” in Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR), Jun. 2016, pp. 770–778.

[30] G. Huang, Z. Liu, L. Van Der Maaten, and K. Q. Weinberger, “Densely connected convolutional networks,” in Proc. IEEE CVPR, Jun. 2017, pp. 4700–4708.

[31] A. G. Howard, M. Zhu, B. Chen, D. Kalenichenko, W. Wang, T. Weyand, M. Andreetto, and H. Adam, “MobileNets: Efficient convolutional neural networks for mobile vision applications,” 2017, arXiv:1704.04861.

[32] C. Szegedy, S. Ioffe, V. Vanhoucke, and A. A. Alemi, “Inception-v4, inception-ResNet and the impact of residual connections on learning,” in Proc. 31st AAAI Conf. Artif. Intell., Feb. 2017, pp. 1–15.

[33] Y. Di, Z. Jiang, and H. Zhang, “A public dataset for fine-grained ship classification in optical remote sensing images,” Remote Sens., vol. 13, no. 4, p. 747, Feb. 2021.

YANG LIU received the B.S. degree in mechanical engineering from Dalian Jiaotong University, in 2010, and the M.S. and Ph.D. degrees in computer science from Pukyong National University, in 2012 and 2016, respectively. Since 2016, he has been a Lecturer with Liaoning Normal University. His research interests include computer vision, machine learning, and SAR image applications.

TIAN LI received the B.S. degree (Engineering) in computer science and technology (normal) from Liaoning Normal University, Dalian, Liaoning, China, in 2020, where she is currently pursuing the M.S. degree in computer science.

KUN YUAN received the B.S. degree (Engineer- ing) in computer science and technology from the University of Jinan, Shandong, China, in 2020. He is currently pursuing the M.S. degree in computer science with Liaoning Normal University, Dalian, Liaoning, China.

SHA LI received the B.S. degree (Engineering) in computer science and technology (normal) from Liaoning Normal University, Dalian, Liaoning, China, in 2021, where she is currently pursuing the M.S. degree in computer science.

* * *