



Article

Ship Classification in SAR Imagery by Shallow CNN Pre-Trained on Task-Specific Dataset with Feature Refinement

Haitao Lang ^{1,†} , Ruifu Wang ^{2,*,†} , Shaoying Zheng ³, Siwen Wu ^{1,†} and Jialu Li ¹¹ College of Mathematics and Physics, Beijing University of Chemical Technology, Beijing 100029, China² College of Geodesy and Geomatics, Shandong University of Science and Technology, Qingdao 266590, China³ School of Information Engineering, China University of Geosciences, Beijing 100871, China

* Correspondence: wrf@sdust.edu.cn

† These authors contributed equally to this work.

Abstract: Ship classification based on high-resolution synthetic aperture radar (SAR) imagery plays an increasingly important role in various maritime affairs, such as marine transportation management, maritime emergency rescue, marine pollution prevention and control, marine security situational awareness, and so on. The technology of deep learning, especially convolution neural network (CNN), has shown excellent performance on ship classification in SAR images. Nevertheless, it still has some limitations in real-world applications that need to be taken seriously by researchers. One is the insufficient number of SAR ship training samples, which limits the learning of satisfactory CNN, and the other is the limited information that SAR images can provide (compared with natural images), which limits the extraction of discriminative features. To alleviate the limitation caused by insufficient training datasets, one of the widely adopted strategies is to pre-train CNNs on a generic dataset with massive labeled samples (such as ImageNet) and fine-tune the pre-trained network on the target dataset (i.e., a SAR dataset) with a small number of training samples. However, recent studies have shown that due to the different imaging mechanisms between SAR and natural images, it is hard to guarantee that the pre-trained CNNs (even if they perform extremely well on ImageNet) can be finely tuned by a SAR dataset. On the other hand, to extract the most discriminative ship representation features from SAR images, the existing methods have carried out fruitful research on network architecture design, attention mechanism embedding, feature fusion, etc. Although these efforts improve the performance of SAR ship classification to some extent, they are usually based on more complex network architecture and higher dimensional features, accompanied by more time-consuming storage expenses. Through the analysis of SAR image characteristics and CNN feature extraction mechanism, this study puts forward three hypotheses: (1) Pre-training CNN on a task-specific dataset may be more effective than that on a generic dataset; (2) a shallow CNN may be more suitable for SAR image feature extraction than a deep one; and (3) the deep features extracted by CNNs can be further refined to improve the feature discrimination ability. To validate these hypotheses, we propose to learn a shallow CNN which is pre-trained on a task-specific dataset, i.e., the optical remote sensing ship dataset (ORS) instead of on the widely adopted ImageNet dataset. For comparison purposes, we designed 28 CNN architectures by changing the arrangement of the CNN components, the size of convolutional filters, and pooling formulations based on VGGNet models. To further reduce redundancy and improve the discrimination ability of the deep features, we propose to refine deep features by active convolutional filter selection based on the coefficient of variation (COV) sorting criteria. Extensive experiments not only prove that the above hypotheses are valid but also prove that the shallow network learned by the proposed pre-training strategy and the feature refining method can achieve considerable ship classification performance in SAR images like the state-of-the-art (SOTA) methods.

Keywords: convolutional neural network (CNN); deep learning; deep feature selection; ship classification; synthetic aperture radar (SAR)



Citation: Lang, H.; Wang, R.; Zheng, S.; Wu, S.; Li, J. Ship Classification in SAR Imagery by Shallow CNN Pre-Trained on Task-Specific Dataset with Feature Refinement. *Remote Sens.* **2022**, *14*, 5986. <https://doi.org/10.3390/rs14235986>

Academic Editors: Emanuele Salerno, Claudio Di Paola, Angelica Lo Duca and Piotr Samczynski

Received: 5 October 2022

Accepted: 22 November 2022

Published: 25 November 2022

Publisher's Note: MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.



Copyright: © 2022 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

Ship classification based on high-resolution synthetic aperture radar (SAR) imagery plays an increasingly important role in various maritime affairs, such as marine transportation management, maritime emergency rescue, marine pollution prevention and control, marine security situational awareness, and so on [1]. To classify a ship accurately in SAR images, the key point is to excavate the discriminative features of SAR images so that they can reflect the essential difference between different types of ships and highlight the similarity in ships of the same classes and the dissimilarity in ships of different classes. In recent years, the supervised learning methods, including traditional machine learning methods [2–13] and newly emerging deep learning methods [14–33], have shown promising results on the tasks of ship classification in SAR images. The traditional machine learning methods are characterized by using handcrafted features, such as scattering features [2,3,13], superstructure scattering features [4,12], geometric features [5,7], histogram of oriented gradient (HOG) features [8], fusion features [6,11], and so on. In contrast, deep learning methods have the ability to automatically learn discriminative features and have achieved enormous success and state-of-the-art performance.

Nevertheless, they still have some limitations in real-world applications that need to be taken seriously by researchers. One is the insufficient number of SAR ship training samples, which limits the training of a satisfied CNN with megabytes of parameters to be learned [34–36]. The other is the limited information that SAR imagery can provide (compared with natural images), which limits the extraction of discriminative ship representation features.

To alleviate the bottleneck caused by insufficient training data that hinder the further improvement of ship classification accuracy, existing approaches primarily leverage two kinds of strategies. A widely adopted strategy is to pre-train CNNs on a generic dataset with massive samples (such as ImageNet [37]) and fine-tune the pre-trained networks on the target dataset (i.e., a SAR dataset) with only a small number of training samples [38–40] (as shown in Figure 1a). However, recent studies have shown that due to the different imaging mechanisms of SAR imagery and natural images, it is hard to guarantee that the pre-trained CNNs (even if they perform extremely well on ImageNet) can be finely tuned by a small SAR dataset just right enough to extract the discriminative features of the ships [39]. Wang et al. claim that considering the difference between SAR imagery and natural images (e.g., imaging mechanisms and target information), features extracted from natural images via the ImageNet pre-trained model are not suitable for SAR imagery. The other strategy, which is termed transfer learning (TL) or domain adaptation (DA), improves classifier training on the target dataset (i.e., the SAR dataset) by transferring the knowledge from a different but related source dataset (usually another ship dataset) [41–48]. This strategy requires that the source dataset (the source domain) and the target dataset (the target domain) have an intrinsic connection [41]. Therefore, the source dataset often deliberately chooses another ship dataset (which usually has a different modality from the target dataset, such as optical ship images) to ensure that there is transferable knowledge between the two domains [45,46,48]. A typical scheme is to extract features in each of the two domains, then map them to a common feature space, and make the two consistent by adjusting the marginal or/and conditional distribution(s), so as to solve the problem of shortfall of labeled data in the target domain by using a large amount of labeled data from the source domain [45,46,48]. Existing methods focus on aligning the feature-level distribution of the source and target domains to minimize the domain gap. In the case of SAR imagery, the network finds it challenging to extract features from the grayscale images which can be used for a classification task.

On the other hand, in order to extract the most discriminative ship representation features from SAR imagery, the existing methods have carried out fruitful research on network architecture design [17,19,21,22,29–31], attention mechanism embedding [23,25], feature fusion [23–26,29,33], etc. Ref. [17] verifies the superior performance of the ResNet architecture on the SAR ship classification task. Ref. [21] proposes the Siamese network equipped with a pre-processing module and a feature fusion module. Ref. [30] proposes

a searched binary neural network (SBNN) with neural architecture search technique to obtain the optimal CNN architecture for SAR ship classification. Ref. [29] proposes a group bilinear convolutional neural network (GBCNN) model to deeply extract discriminative representations of ship targets from the pairwise vertical–horizontal polarization (VH) and vertical–vertical polarization (VV) SAR imagery. To fully explore the polarization information, the multi-polarization fusion loss (MPFL) is constructed to train the proposed model for superior SAR ship representation learning. Ref. [25] proposes HOG-ShipCLSNet to integrate four mechanisms: multi-scale classification mechanism, global self-attention mechanism, fully connected balance mechanism, and HOG feature fusion mechanism. The work in [24] discusses how to effectively fuse traditional handcrafted features and deep convolutional features in CNN and recommends to perform feature fusion in the last fully connected layer. Ref. [26] proposes a ship classification network (PFGFE-Net) with polarization coherence feature fusion and geometric feature embedding to utilize polarization information and traditional handcrafted features. Although these efforts improve the performance of SAR ship classification to some extent, they are usually based on more complex network architecture and higher dimensional features, accompanied by more time-consuming storage expenses. Not only that, a small SAR dataset is not enough to effectively learn so many parameters of complex CNN, resulting in overfitting, and the ship features extracted by CNN are highly redundant, which directly impairs the discrimination ability of the features. This situation becomes more serious as the depth of the network deepens.

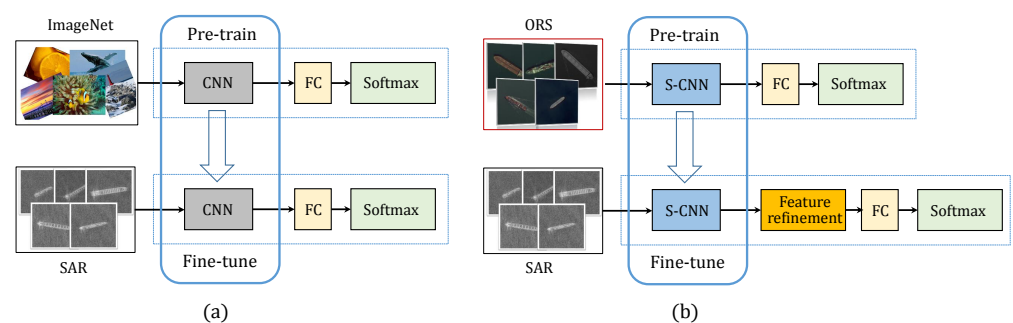


Figure 1. Comparison between a widely adopted flowchart and the proposed flowchart. (a) A widely adopted pre-trained and fine-tuned flowchart. (b) Flowchart of the proposed method. Generally, a CNN is used to extract the deep features from the input images. Then, the deep features are refined further by fully connected layers (FC). Finally, the refined features are input into the softmax classifier for the final ship classification. Compared with the widely adopted flowchart, the proposed one explores the performance of the shallow CNN (S-CNN) on SAR ship classification. In addition, we add a feature refinement operation between CNN and FC aiming to reduce the redundancy and increase the discriminative ability of the deep features further.

1.1. Motivation

Through the analysis of SAR image characteristics and the CNN feature extraction mechanism, this study puts forward three hypotheses: (1) pre-training a CNN on a task-specific dataset (specifically, on an optical remote sensing ship dataset (ORS)) may be more effective than that on a generic dataset; (2) a shallow CNN may be more suitable for SAR image feature extraction than a deep one; and (3) the deep features extracted by a CNN can be further refined to improve the feature discrimination ability.

The first hypothesis is motivated by the assumption that compared with the ImageNet, which is a generic dataset with tens of thousands of categories, the vast majority of which have completely different attributes from ships, the optical remote sensing ship dataset has the same target attributes as the SAR ship dataset, making it easier to train a transferable network to improve the successive fine-tuning performance. The second hypothesis is motivated by the characteristics of SAR imagery and the feature extraction mechanism of CNNs. The mature and widely used CNN classifiers, such as AlexNet [49], VGGNet [50],

ResNet [51], and DenseNet [52], are initially designed for natural image classification tasks. Natural images have abundant color information, while SAR image pixels represent the reflected electromagnetic wave intensity of the object. Thus, there exist enormous domain differences between natural and SAR imagery. It is a well-known fact that SAR imagery contains far less information than natural images. While the CNN classifier extracts the feature maps of input images and uses the last layer of feature vectors for object classification, different feature layers of a CNN have different spatial resolutions and semantic information. For example, the lower layers utilize lower-level semantic features in comparison with those utilized by the final layers. Due to the lack of information in the SAR image itself, deeper layers cannot extract more effective discriminative features. As a result, a shallow CNN may be more suitable for SAR image feature extraction. The third hypothesis is based on the consensus that the deep features extracted by CNN (especially the fused features) are highly redundant, and it is reasonable to believe that these features can be further compressed into lower dimensions without losing their discriminative ability, or even further improving their discriminative ability.

To validate these hypotheses, we propose to learn a shallow CNN which is pre-trained on a task-specific dataset, i.e., optical remote sensing ship dataset (ORS) instead of on the widely adopted ImageNet dataset. The proposed flowchart is illustrated in Figure 1b. Compared with the widely adopted method (Figure 1a), our study conducts three improvements. Firstly, we pre-train the CNN model on a task-specific dataset, i.e., an optical remote sensing ship dataset (ORS) instead of the generic dataset ImageNet. Secondly, we explore the performance of shallow CNNs in SAR ship classification tasks. For comparison purposes, we designed 28 CNN architectures by changing the arrangement of the CNN components, size of convolutional filters, and pooling formulations on the basis of VGGNet models [50] and present a thorough evaluation. Thirdly, in order to avoid overfitting, extract more discriminative deep features, and reduce feature redundancy, we propose to refine deep features by active convolutional filters selection based on the coefficient of variation (CoV) sorting criteria and performing the feature refinement operation by selecting and reserving the active convolutional kernels on the last convolutional layer of a CNN.

Extensive experiments not only prove that the above hypotheses are valid but also prove that the shallow network learned by the proposed pre-training strategy and the feature refining method can achieve considerable ship classification performance in SAR imagery like the state-of-the-art (SOTA) methods.

1.2. Contribution

The contribution of this study is threefold:

(1) Through the analysis of SAR image characteristics and the CNN feature extraction mechanism, we put forward three hypotheses and designed an experimental flowchart to prove their validity.

(2) We introduce the ORS dataset as a pre-training dataset serving the SAR ship classification task. Compared with the widely used generic ImageNet dataset, ORS is smaller but more suitable for SAR ship classification scenarios. To the best of our knowledge, no existing studies have attempted this.

(3) We propose a novel feature refinement method by extracting active convolutional filters which have a high response for the purpose of reducing feature dimensions, avoiding over-fitting, and extracting more discriminative deep features in SAR ship imagery.

1.3. Organization

The remainder of this paper is organized as follows. Section 2 introduces the CNN architectures we designed for research purposes (Section 2.1) and the details of the proposed feature refinement method (Section 2.2). In Section 3, we introduce the dataset (Section 3.1) used to conduct our experiments (Section 3.2) and the common experimental protocol (Section 3.3). Then, we analyze and discuss the experimental results in Section 4. The

effectiveness of using a task-specific dataset to pre-train CNNs (Section 4.1), the feasibility of feature refinement (Section 4.2), the overall performance of the three proposed improved schemes (Section 4.3), and the comparison with the state-of-the-art methods (Section 4.4) are introduced sequentially. Finally, we conclude this study in Section 5.

2. Methodology

2.1. CNN Architecture Design

A CNN can learn multi-level representations of ships from SAR imagery. A suitable CNN is crucial in the ship classification task, which can improve prediction accuracy and reduce prediction error. So far, many CNN architectures, such as AlexNet [49], VGGNet [50], ResNet [51], and DenseNet [52], have been successfully applied to SAR ship classification tasks. A typical CNN architecture generally is composed of alternate layers of convolution and pooling followed by one or more fully connected layers at the end. Furthermore, different regulatory units such as batch normalization and dropout are also incorporated to optimize CNN performance [53]. The mechanism and role of these components are briefly described as follows:

- Convolutional layer (C). The convolutional layer is composed of a set of convolutional kernels where each neuron acts as a kernel. A convolutional kernel works by dividing the image into small slices, commonly known as receptive fields. The kernel convolves with the images using a specific set of weights by multiplying its elements with the corresponding elements of the receptive field. The convolution operation can be expressed as:

$$\mathbf{f}_l^k(p, q) = \sum_c \sum_{x, y} I_c(x, y) \cdot f_l^k(p, q) \quad (1)$$

where $I_c(x, y)$ represents an element of the input image located at the position (x, y) . $I_c(x, y)$ is element-wise multiplied by $f_l^k(p, q)$ index of the k -th convolutional kernel of the l -th layer. The output feature-map of the k -th convolutional operation can be expressed as

$$\mathbb{F}_l^k(p, q) = [\mathbf{f}_l^k(1, 1), \dots, \mathbf{f}_l^k(p, q), \dots, \mathbf{f}_l^k(P, Q)] \quad (2)$$

where p, q denote the row and column position in a feature matrix, and P, Q denote the total number of rows and columns of feature matrix, respectively.

- Pooling layer (P). Pooling or down-sampling is a local operation, which sums up similar information in the neighborhood of the receptive field and outputs the dominant response within this local region.

$$\mathbb{Z}_l^k(p, q) = g_p(\mathbb{F}_l^k(p, q)) \quad (3)$$

Equation (3) shows the pooling operation in which \mathbb{Z}_l^k represents the pooled feature map of input feature map \mathbb{F}_l^k , whereas $g_p(\cdot)$ defines the type of pooling operation. Different types of pooling formulations such as max, average, overlapping, spatial pyramid pooling, etc., are used in CNNs. Boureau et al. [54] performed both a theoretical comparison and experimental validation of max and average pooling and proved that when the clutter is homogeneous and has low variance across images, average pooling has good performance and is robust to intrinsic variability.

- Activation function. Activation function $g_a(\cdot)$ serves as a decision function and helps in learning of intricate patterns. For a convolved feature map, the activation function is defined as

$$\mathbb{T}_l^k(p, q) = g_a(\mathbb{F}_l^k(p, q)) \quad (4)$$

Different activation functions such as sigmoid, tanh, maxout, ReLU, and its variants (leaky ReLU) are used to inculcate non-linear combination of features. In real applications, ReLU and its variants are preferred, as they help in overcoming the vanishing gradient problem [49].

- Batch normalization. Batch normalization is used to address the issues related to the internal covariance shift within feature maps, which slows down the convergence. Furthermore, it smoothens the flow of gradient and acts as a regulating factor, which thus helps in improving the generalization of the network.
- Dropout. Dropout introduces regularization within the network, which ultimately improves generalization by randomly skipping some units or connections with a certain probability. This random dropping of some connections or units produces several thinned network architectures, and finally, one representative network is selected with small weights.
- Fully connected layer (FC). The fully connected layer is mostly used at the end of the network for classification. It makes a non-linear combination of selected features, which are used for the classification of data.
- Softmax layer. In probability theory, the output of the softmax function can be used to represent a categorical distribution. The softmax function is used as output layer in the CNN networks, and it can be expressed as

$$p(x_c) = \frac{\exp(x_c)}{\sum_{t=1}^C \exp(x_t)} \quad (5)$$

where x_c is the output for c -th class, C is the number of classes, and $p(x_c)$ is probability of the c -th class.

A recent survey has found that among the various improvements that can improve the performance of CNNs, such as the use of different activation and loss functions, parameter optimization, regularization, and architectural innovations, the significant improvement in the representational capacity of the deep CNNs is achieved through architectural innovations [53]. The arrangement of CNN components plays a fundamental role in designing new architectures and thus achieving enhanced performance.

This study investigates the impact of different CNN architectures, especially shallow networks, on SAR ship classification. To this end, we built 28 CNNs of different architectures based on VGGNet [50], which is widely regarded as the milestone design template for the follow-up networks, by changing the network depth (i.e., the connection configuration of the convolutional layers and pooling layers), the convolutional kernels, and the pooling formulations. The detailed architecture of the designed CNNs and the corresponding parameter size is listed in Table 1. This table illustrates 28 stacks of convolutional layers (C) and pooling layers (P) with different depths in different arrangements. The fully connected layer (FC) and softmax layer following these stacks are not illustrated in this table. Specifically, in this study, we utilize three fully connected layers: the first two have 4096 channels each, and the third performs SAR ship classification and thus contains C channels (one for each class), which equals the number of classes of ship to be classified. The final layer is the softmax layer. As for other basic components, i.e., activation function (ReLU), batch normalization, and dropout, we utilize the same functions for all 28 CNNs.

Among these 28 CNNs, we are able to find the standard VGGNets and their counterparts, i.e., CNN-XIV (VGG-8) vs. CNN-XIII, CNN-XIX (VGG-11) vs. CNN-XVIII, CNN-XXI (VGG-13) vs. CNN-XX, CNN-XXIII (VGG-16) vs. CNN-XXI, and CNN-XXV (VGG-19) vs. CNN-XXIV. The difference between the standard VGGNets and their counterparts is that we replace the maximum pooling (P_m) with average pooling (P_a) motivated by the study of [54]. The size of the convolutional kernel used in CNN-VIII and CNN-XVII is 7×7 and 5×5 for CNN-XV and CNN-XXVIII. These models are sorted according to the number of parameters they need to learn. The minimum is 0.37M of CNN-I, and the maximum is 25.61M of CNN-XXVIII.

Table 1. The 28 designed CNNs with various architectures.

Name	Architecture(CNN)	Parameters(M)
CNN-I	C11 – P_a – C12 – P_a – C14 – P_a	0.37
CNN-II	C11 – P_a – C12 – P_a – C12 – P_a – C12 – P_a – C12 – P_a	0.52
CNN-III	C11 – P_a – C12 – P_a – C12 – P_a – C12 – P_a – C13 – P_a	0.67
CNN-IV	C11 – P_a – C12 – P_a – C13 – P_a – C13 – P_a	0.96
CNN-V	C11 – P_a – C12 – P_a – C12 – P_a – C13 – P_a – C13 – P_a	1.11
CNN-VI	C11 – C11 – P_a – C12 – P_a – C13 – P_a – C13 – P_a – C13 – P_a	1.15
CNN-VII	C11 – P_a – C12 – P_a – C13 – P_a – C13 – P_a – C13 – P_a	1.55
CNN-VIII	C31 – P_a – C31 – P_a – C32 – P_a – C32 – P_a – C32 – P_a	2.11
CNN-IX	C11 – P_a – C12 – P_a – C13 – P_a – C13 – P_a – C14 – P_a	2.14
CNN-X	C11 – P_a – C12 – P_a – C14 – P_a – C14 – P_a	3.03
CNN-XI	C11 – P_a – C11 – P_a – C12 – P_a – C12 – P_a – C14 – C14 – P_a	3.21
CNN-XII	C11 – P_a – C12 – P_a – C13 – C13 – P_a – C13 – C13 – P_a – C14 – P_a	3.32
CNN-XIII	C11 – P_a – C12 – P_a – C13 – P_a – C14 – P_a – C14 – P_a	3.91
CNN-XIV	C11 – P_m – C12 – P_m – C13 – P_m – C14 – P_m – C14 – P_m	3.91
CNN-XV	C21 – P_a – C22 – P_a – C23 – P_a – C23 – P_a – C23 – P_a	4.31
CNN-XVI	C11 – P_a – C12 – P_a – C13 – C13 – P_a – C14 – C14 – P_a – C14 – P_a	6.86
CNN-XVII	C31 – P_a – C32 – P_a – C33 – P_a – C33 – P_a – C33 – P_a	8.44
CNN-XVIII	C11 – P_a – C12 – P_a – C13 – C13 – P_a – C14 – C14 – P_a – C14 – C14 – P_a	9.22
CNN-XIX	C11 – P_m – C12 – P_m – C13 – C13 – P_m – C14 – C14 – P_m – C14 – C14 – P_m	9.22
CNN-XX	C11 – C11 – P_a – C12 – C12 – P_a – C13 – C13 – P_a – C14 – C14 – P_a – C14 – C14 – P_a	9.38
CNN-XXI	C11 – C11 – P_m – C12 – C12 – P_m – C13 – C13 – P_m – C14 – C14 – P_m – C14 – C14 – P_m	9.38
CNN-XXII	C11 – C11 – P_a – C12 – C12 – P_a – C13 – C13 – C13 – P_a – C14 – C14 – C14 – P_a – C14 – C14 – C14 – P_a	14.70
CNN-XXIII	C11 – C11 – P_m – C12 – C12 – P_m – C13 – C13 – C13 – P_m – C14 – C14 – C14 – P_m – C14 – C14 – C14 – P_m	14.70
CNN-XXIV	C11 – C11 – P_a – C12 – C12 – P_a – C13 – C13 – C13 – C13 – P_a – C14 – C14 – C14 – C14 – P_a – C14 – C14 – C14 – P_a	20.02
CNN-XXV	C11 – C11 – P_m – C12 – C12 – P_m – C13 – C13 – C13 – C13 – P_m – C14 – C14 – C14 – C14 – P_m – C14 – C14 – C14 – P_m	20.02
CNN-XXVI	C11 – C11 – C11 – C11 – P_a – C12 – C12 – C12 – C12 – P_a – C13 – C13 – C13 – C13 – P_a – C14 – C14 – C14 – P_a – C14 – C14 – C14 – P_a	20.39
CNN-XXVII	C11 – C11 – C11 – C11 – P_m – C12 – C12 – C12 – C12 – P_m – C13 – C13 – C13 – C13 – P_m – C14 – C14 – C14 – C14 – P_m – C14 – C14 – C14 – P_m	20.39
CNN-XXVIII	C21 – P_a – C22 – P_a – C23 – C23 – P_a – C24 – C24 – P_a – C24 – C24 – P_a	25.61

Note: c_n^m denotes the corresponding convolutional layer, which is composed of m convolutional kernels with the size of $n \times n$. p_a and p_m represents average and max pooling, respectively. Simplified representation: C11 \rightarrow c_3^{64} , C12 \rightarrow c_3^{128} , C13 \rightarrow c_3^{256} , C14 \rightarrow c_3^{512} , C21 \rightarrow c_5^{64} , C22 \rightarrow c_5^{128} , C23 \rightarrow c_5^{256} , C24 \rightarrow c_5^{512} , C31 \rightarrow c_7^{64} , C32 \rightarrow c_7^{128} , C33 \rightarrow c_7^{256} .

2.2. Feature Refinement

The existing research shows that CNNs prefer a high response from a filter when it fires on an input [55], and there are active convolutional kernels and attribute-centric nodes in the CNN network [55,56]. It motivates us to perform feature refinement by extracting active convolutional filters which have a high response to reduce feature dimensions, avoid overfitting, and extract more discriminative deep features in SAR ship images. In our implementation, we perform active convolutional kernel selection on the last convolutional layer of the CNN.

To obtain the active convolutional filters from the last convolutional layer, in our implementation, we compute the mean (\mathcal{E}) and standard deviation (\mathcal{S}) of each convolutional filter's output on a 2×2 window, which are defined as:

$$\mathcal{E}_c^k = \frac{\sum_{t=1}^{n_c} \sum_{p=1, q=1}^2 \mathbf{f}_{last}^k(p, q)}{2 \times 2 \times n_c} \quad (6)$$

$$\mathcal{S}_c^k = \frac{1}{n_c} \sqrt{\sum_{t=1}^{n_c} \frac{\sum_{p=1, q=1}^2 \mathbf{f}_{last}^k(p, q)}{2 \times 2} - \mathcal{E}_c^k} \quad (7)$$

where c indicates ship class ($c = 0, 1, 2, \dots, C$), and n_c denotes all the input data which belong to class c . $\mathbf{f}_{last}^k(p, q)$ denotes the response value of the k -th filter in the last convolutional layer, and \mathcal{E}_c^k and \mathcal{S}_c^k are the mean value and standard deviation of the k -th filter in the last convolution layer for ship class c .

Usually, ships are very similar within a class, and the convolutional filter always has a high mean value and low standard deviation when it "fires" on a certain ship class. Hence, we employ the coefficient of variation (COV) to determine active convolutional kernels and select more discriminative features. The COV of the k -th filter in the last convolutional layer for class c can be computed as:

$$\text{COV}_c^k = \frac{\mathcal{E}_c^k}{\mathcal{S}_c^k} \quad (8)$$

Based on experience from pre-experiments, we select the first 50 convolutional kernels with the largest COV value for each class and extract the refined deep features from these selected convolutional kernels. The pseudo-code is described in Algorithm 1.

Algorithm 1 Deep feature refinement.

Require: Deep features of SAR ship images;

Ensure: Deep features refinement based on COV criterion;

- 1: **for** $ii \in \{1, \dots, k, \dots, K\}$ **do**
 - 2: **for** $jj \in \{1, \dots, c, \dots, C\}$ **do**
 - 3: Compute \mathcal{E}_{jj}^{ii} based on Equation (6);
 - 4: Compute \mathcal{S}_{jj}^{ii} based on Equation (7);
 - 5: Compute COV_{jj}^{ii} based on Equation (8);
 - 6: **return** COV_{jj}^{ii} ;
 - 7: **end for**
 - 8: **end for**
 - 9: **for** $jj \in \{1, \dots, c, \dots, C\}$ **do**
 - 10: Sort($\text{COV}_{jj}^1, \dots, \text{COV}_{jj}^k, \dots, \text{COV}_{jj}^K$)
 - 11: Select the first 50 convolutional kernels with the largest COV value for each class.
 - 12: **return** $\{\text{filters}_{jj} | \text{Sort}(\text{COV}_{jj}^1, \dots, \text{COV}_{jj}^k, \dots, \text{COV}_{jj}^K) | \max 50\}$;
 - 13: **end for**
 - 14: Extracting new deep features from the selected filters;
 - 15: **return** Refined deep features of ship SAR images.
-

3. Experiments

3.1. Dataset and Data Pre-Processing

In our experiments, we utilized the generic dataset ImageNet [37] and task-specific dataset ORS [45] to pre-train the CNNs, respectively. ImageNet is one of the most famous and widely used datasets for pre-training CNNs. It contains more than 20,000 categories with roughly 500–1000 images per category [37]. In this study, we used ImageNet as the benchmark to validate the performance when using the ORS dataset [45] to pre-train the CNN. ORS dataset was released by [45] to provide a reliable source domain with a large number of labeled ship samples for transfer learning study. The dataset has eight ship classes, including cargo, container, oil tanker, bulk carrier, car carrier, chemical tanker, dredge, and tug. All the ship slices were segmented from Google Earth with sub-meter resolution, as shown in Figure 2.

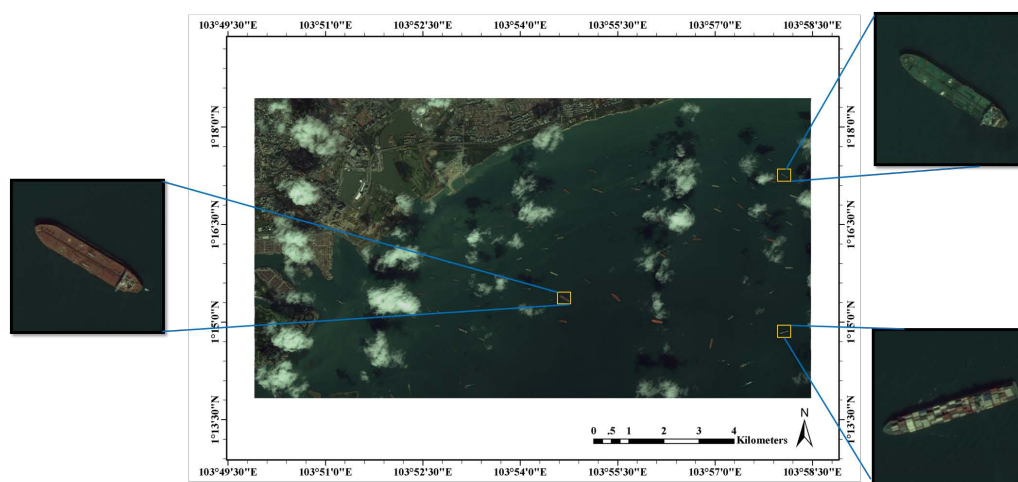


Figure 2. Ship slices in ORS dataset were segmented from Google Earth imagery.

On the other hand, we utilized two SAR ship datasets to fine-tune the pre-trained CNNs and test the final performance of the proposed method. The first SAR dataset (SD1) was collected by [3] from six strip map-mode VV-polarization TerraSAR-X images with $2.0 \text{ m} \times 1.5 \text{ m}$ resolution in azimuth and range directions, respectively. The dataset contains three ship categories, i.e., carrier, container, and oil tanker. Each category (class) has 50 ship samples. The second SAR dataset (SD2) is the FUSAR dataset, which was collected by [35] from high-resolution Gaofen-3 imagery. In this study, we chose a subset that contains four common ship classes, i.e., cargo, bulk carrier, container, and oil tanker, from the original dataset. Moreover, we kept 50 ship samples per class to conduct our experiments. Several typical samples from ORS, SD1, and SD2 datasets are shown in Figure 3.

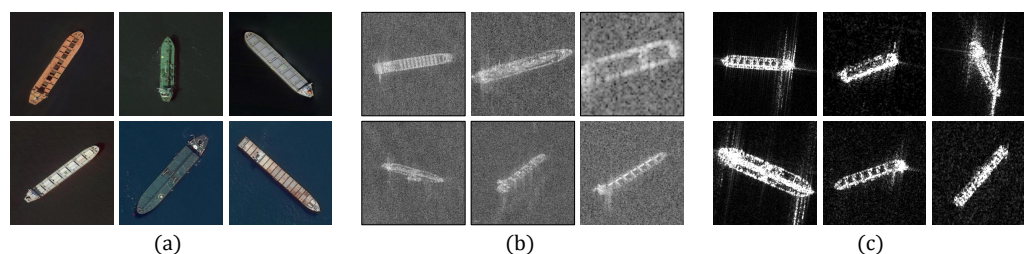


Figure 3. Typical samples from (a) ORS, (b) SD1, and (c) SD2.

Considering that pre-training a CNN requires a large number of samples, in our experiments, we used a data augmentation scheme involving scaling, flipping, and rotating to enrich the samples in the ORS dataset to 5000 per class. For two SAR datasets, we enriched them to 100 samples per class. A total of 80 samples were randomly selected from

each class to build a training set for fine-tuning the CNNs pre-trained on ImageNet/ORS, and the remaining 20 samples per class were used to validate and compare the performance of various CNNs and methods. To obtain reliable evaluation results, we ran it ten times and reported the averaged classification accuracy.

3.2. Experimental Content

As shown in Figure 1, this paper proposes three changes compared with the traditional method, i.e., (1) pre-training CNNs with task-specific ORS rather than ImageNet, (2) replacing deep CNNs with shallow CNNs to perform SAR ship classification, and (3) adding a feature refinement operation between the last convolutional layer of CNNs and FC layer to improve the discrimination ability of the feature. To validate the feasibility of our ideas and corresponding changes, we conducted four experiments for comparison.

- E1: CNNs were pre-trained on ImageNet and then fine-tuned on the SAR dataset without feature refinement.
- E2: CNNs were pre-trained on ImageNet and then fine-tuned on the SAR dataset with feature refinement.
- E3: CNNs were pre-trained on ORS and then fine-tuned on the SAR dataset without feature refinement.
- E4: CNNs were pre-trained on ORS and then fine-tuned on the SAR dataset with feature refinement.

Furthermore, to obtain an objective evaluation of the performance of the proposed method, we compare it with several SOTA methods. These methods include: [17] conducting ship classification based on deep residual network, [18] utilizing a CNN embedding and metric learning, [24] injecting traditional handcrafted features into the deep CNN, and [48] proposing a dual-branch network embedding attention mechanism to conduct deep sub-domain adaptation. In our implementation, we reproduced these algorithms and conducted experiments on SD1 and SD2 datasets. It should be noted that, for [17], we utilized ResNet50 as the backbone network to experiment. For [24], we injected naive geometric features (NGFs) [7] into the VGG16 network and performed feature fusion in the last FC layer by the method of concatenation with feature normalization, as recommended by the authors. While for [48], since it is a domain adaptation method, following the original literature, we utilized the ORS dataset as the source domain and SD1/SD2 as the target domain.

3.3. Experimental Protocol

In our experiments, for a fair comparison, for all CNNs, the batch size was set to 64 and momentum to 0.9. The training was regularized by weight decay (the L2 penalty multiplier set to 5×10^{-3}) and dropout regularization for the first two fully connected layers (dropout ratio set to 0.5). The learning rate was initially set to 10^{-3} and then decreased by a factor of 10 when the validation set error stopped decreasing within 10 epochs. The training was stopped when the validation set error stopped decreasing within 30 epochs.

All the experiments are performed on a 64-bit Core i7-6800k (3.40 GHz) computer with 64 GB of RAM and one Nvidia GTX1080ti GPU (11 GB RAM), which is supported by compute unified device architecture (CUDA) 8.0, and the proposed method is implemented by KERAS [57] with tensorflow [58] backend.

4. Results and Discussion

We conducted all four experiments on two datasets and listed the experimental results in Table 2. It should be noted that CNN-I has a serious overfit due to its too shallow network that prevents convergence during training, so there are no test results reported. Other CNN architectures have high robustness and can achieve stable results. Analyzing the experimental results, we draw the following basic understanding.

Table 2. Performance evaluation of various CNNs on four experiments for two datasets. The classification accuracy (%) averaged on ten runs are reported.

Name	SD1				SD2			
	E1	E2	E3	E4	E1	E2	E3	E4
CNN-I	–	–	–	–	–	–	–	–
CNN-II	75.33	76.67	75.33	76.33	72.00	73.25	76.50	76.75
CNN-III	73.33	75.00	73.33	77.00	73.50	75.00	74.33	76.75
CNN-IV	75.67	77.00	79.00	80.00	72.67	73.75	74.25	74.50
CNN-V	78.00	78.67	80.33	81.33	75.00	76.67	76.50	76.67
CNN-VI	79.33	81.33	75.33	76.67	73.67	74.00	70.67	72.00
CNN-VII	81.33	82.00	83.33	87.67	77.67	79.33	78.67	82.33
CNN-VIII	75.67	76.33	73.67	74.33	71.33	73.75	66.00	66.50
CNN-IX	78.00	79.33	75.67	76.00	72.50	74.67	77.33	78.67
CNN-X	79.00	79.33	82.33	82.67	74.75	75.33	78.00	80.50
CNN-XI	79.67	81.33	73.00	73.33	72.75	73.50	76.00	76.67
CNN-XII	80.67	82.00	76.00	78.33	72.67	74.00	72.00	72.67
CNN-XIII	81.67	83.33	72.67	76.67	75.00	75.00	76.00	77.33
CNN-XIV	81.67	82.00	79.33	82.33	78.75	78.75	79.75	80.50
CNN-XV	78.67	78.67	65.00	68.33	74.00	75.00	76.67	78.67
CNN-XVI	79.00	79.33	82.33	84.00	77.50	78.33	79.33	80.75
CNN-XVII	76.00	77.33	74.00	75.33	75.00	76.67	74.50	75.00
CNN-XVIII	82.33	83.67	74.56	78.67	78.00	78.75	70.75	70.67
CNN-XIX	82.33	84.67	75.67	75.67	79.33	80.67	72.67	72.67
CNN-XX	81.67	83.33	71.33	76.00	75.00	78.50	72.67	72.75
CNN-XXI	78.67	79.67	66.67	66.33	78.67	79.33	72.75	73.50
CNN-XXII	82.33	84.00	65.33	68.33	77.25	77.67	65.75	67.33
CNN-XXIII	84.67	86.33	65.67	66.67	80.50	81.75	64.00	70.00
CNN-XXIV	82.67	83.67	63.33	66.67	76.67	77.33	60.67	62.25
CNN-XXV	83.33	84.33	54.00	54.67	77.00	77.00	52.75	56.00
CNN-XXVI	82.00	83.67	59.00	62.33	78.75	79.33	57.25	58.75
CNN-XXVII	81.67	83.33	54.33	54.67	75.67	75.00	58.50	60.67
CNN-XXVIII	78.00	79.33	59.67	60.00	76.75	78.50	62.00	67.50

4.1. About the Pre-Trained Dataset

Observing the results of E1 and E3 on both SD1 and SD2, it can be seen that deep CNN architectures perform better than shallow networks when using ImageNet as a pre-trained dataset. Overall, as the depth of the network progressively increases, the classification accuracy gradually increases from about 75.00% of the shallow network (CNN-II) to about 85.00% of the deeper network (CNN-XXIII) on SD1 (whereas for SD2, it is from about 72.00% (CNN-II) to about 80.00% (CNN-XXIII)). The extreme performance (84.67% for SD1 and 80.50% for SD2) is obtained by CNN-XXIII (i.e., VGG-16 [50]). Thereafter, the classification performance has not risen but has fallen as the depth increased. The opposite situation happens when using ORS as a pre-trained dataset. It is very interesting that, this time, the shallow networks perform better than their deeper counterparts. The leading performances are obtained by CNN-VII (83.33% on SD1) and CNN-XIV (i.e., VGG-8 [50], 79.75% on SD2), which are all shallow CNNs. This experimental rule is reflected in Figure 4a,b. From a macro-trend point of view, CNN-XVIII, which is an 11-layer network, seems to be a watershed. CNNs that are shallower than CNN-XVIII perform better when pre-training on ORS, while networks that are deeper than it are pre-trained better with ImageNet.

A direct comparison of results from E1 and E3 on two datasets shows that for SD1, the best performance is 84.67%, obtained by CNN-XXIII (E1), followed by 83.33% obtained by CNN-XXV (E1) and CNN-VII (E3); the next are 82.67%, obtained by CNN-XXIV (E1), and 82.33%, obtained by CNN-X and CNN-XVI from E3, and CNN-XVIII, CNN-XIX, and CNN-XXII from E1. The overall performance of SD2 is slightly lower than that of SD1,

mainly since SD2 is a 4-classification task, which is slightly more difficult than SD1's 3-classification task. The best performance is 80.50%, obtained by CNN-XXIII (E1), the second is 79.75%, obtained by CNN-XIV (E3), and the third is 79.33%, obtained by CNN-XIX (E1) and CNN-XVI (E3).

In our view, the reason behind this experimental discovery may be that ImageNet contains a large number of class-rich samples, so it is enough to pre-train a relatively deep network and then fine-tune on the target dataset (even if it has nothing to do with ImageNet) to obtain satisfactory performance. In contrast, because ORS has far fewer training samples than ImageNet, it is not enough to pre-train a relatively deep network with good performance. However, because the task-specific ORS dataset is intrinsically linked to the target dataset (more essential common attributes), the shallow network it trains can achieve performance comparable to the deeper network trained by ImageNet. In this sense, it proves that it is perfectly feasible to train a shallow network with a smaller task-oriented pre-training dataset that does not lose performance to a deeper network trained with a larger dataset such as ImageNet.

4.2. About Feature Refinement

We explore the effectiveness of the proposed feature refinement by comparing E1 with E2 (E3 vs. E4). The gap between E1 (E3) and E2 (E4) (after feature refinement) on two datasets for all CNN architectures is illustrated in Figure 5. There are two main findings. The first is that the proposed feature refinement consistently helps CNN improve classification performance, except for the only three exceptions with negative gain. The results of this experiment prove that the proposed feature refinement is feasible and effective. Secondly, take a closer look at Figure 5, it shows that feature refinement is more effective at boosting the CNN pre-trained using the ORS dataset than those with the ImageNet dataset. For example, on SD1, four networks receive a gain of more than 4.00% (the maximum is 4.67% by CNN-XX), and six networks received a boost of more than 3.00%, which can be seen by looking at the green bars in the figure. This outcome does not occur on CNNs pre-trained with ImageNet, none of which achieves a gain of more than 3.00% (the maximum is 2.34% by CNN-XIX), which can be seen by checking the blue bars. A similar situation occurs on SD2, which can be seen by comparing the yellow with the red bars. On average, the gain of E4-E3 on SD1 and SD2 are 1.86% and 1.56%, which outperforms E2-E1 (1.22% and 1.05%) by 0.64% and 0.51%, respectively.

Feature refinement plays a role similar to principal component analysis (PCA) operation; selecting the active convolutional kernels is equivalent to selecting the principal component of the feature, so it significantly reduces the feature dimension, while further improving the feature discriminative ability. The effect of this process can be illustrated by Figure 6. As shown in Figure 6a, the feature map of the last layer of CNN (CNN-VII in this demo) before feature refinement has a lot of redundancy (refer to the all-black sub-plots in the map). After feature refinement, as shown in Figure 6b, the redundant features are greatly eliminated, preserving only dimensions with useful information.

4.3. About the Shallow Network Pre-Trained on Task-Specific ORS Dataset with Feature Refinement

Looking back at Table 2, it can be found that the best classification performance on the two datasets is 87.67% (E4) and 82.33% (E4), respectively, and they are obtained by CNN-VII, which is a shallow network pre-trained by the task-specific ORS dataset with feature refinement. In contrast, it is very interesting and very coincidental that the second best performance on both datasets (86.33% (E2) and 81.75% (E2)) is obtained by the deep network CNN-XXIII pre-trained by the ImageNet dataset with feature refinement. This result proves that for the specific task of ship classification in SAR imagery, the proposed scheme of "shallow network + task-specific ORS pre-training + feature refinement" is completely feasible and effective, and it is fully capable of performing this task and obtaining the best results.

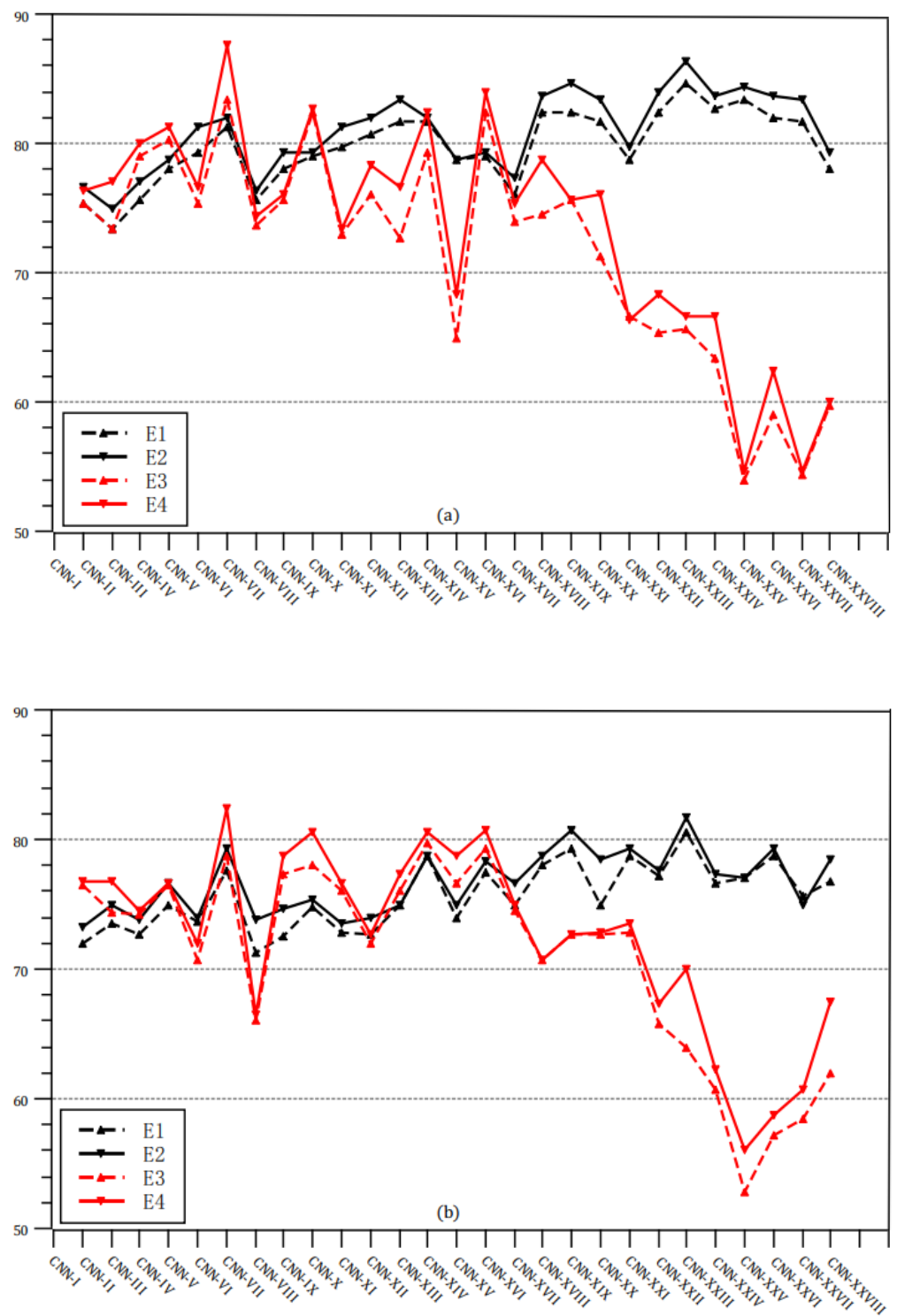


Figure 4. The curves of four experimental results on (a) SD1 and (b) SD2. The ordinate represents the classification accuracy (%), and the abscissa represents various CNN architectures.

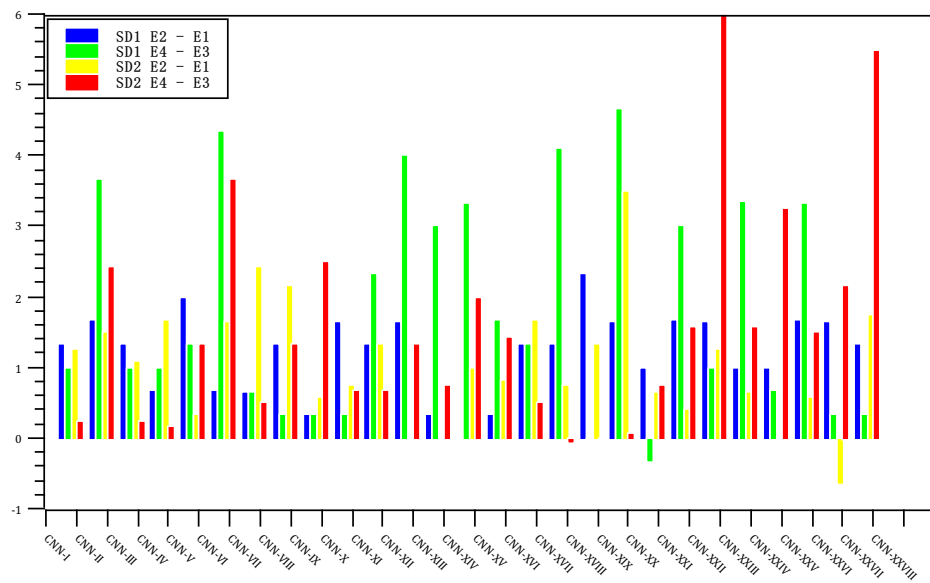
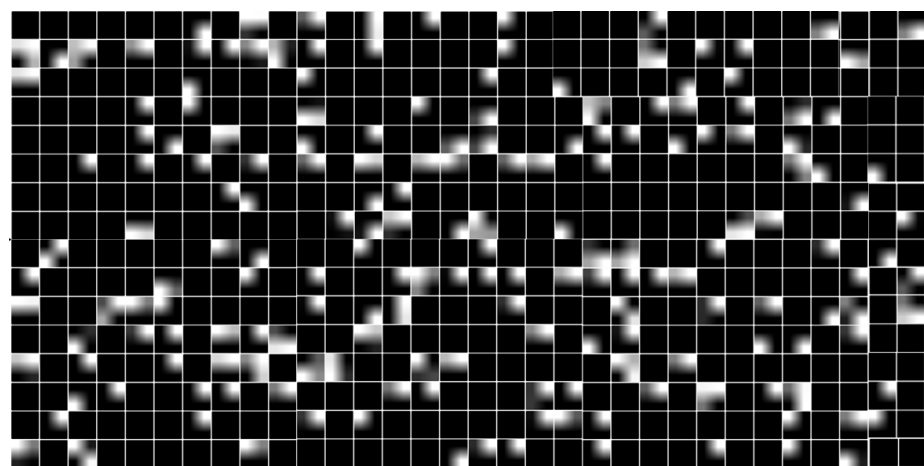
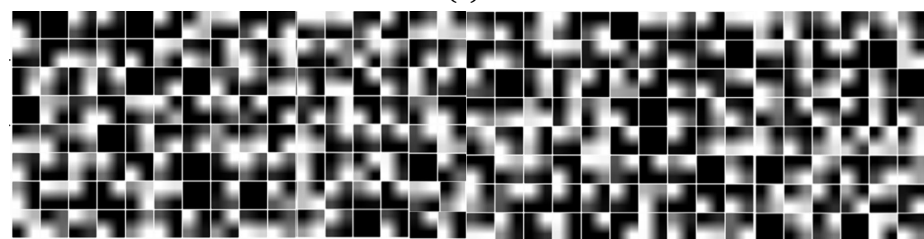


Figure 5. Gain obtained by feature refinement. The ordinate represents the classification accuracy (%), and the abscissa represents various CNN architectures.



(a)



(b)

Figure 6. Feature map (partial) of the last layer of CNN (CNN-VII in this demo) (a) before and (b) after the feature refinement.

In addition to the performance advantages, the shallow network (CNN-VII) needs to learn fewer parameters (1.55M vs. 14.7M refer to Table 1, the former is only one-tenth of the latter) compared with deep networks (CNN-XXIII); feature refinement improves the discriminative ability while also reducing the dimension of deep features, coupled with the use of task-oriented smaller ORS training dataset, training expenses have been greatly reduced, and overfitting has been almost eliminated.

4.4. Comparison with State-of-the-Art Methods

We compared the performance of the shallow network CNN-VII learned by the proposed method (E4) to several state-of-the-art (SOTA) methods. The results are listed in Table 3. It shows that the shallow network learned by the proposed pre-training strategy and the feature refining method can achieve considerable ship classification performance in SAR imagery like the state-of-the-art (SOTA) methods. The performance of CNN-VII on both SD1 and SD2 datasets are all in second place, only slightly behind method [48] and ahead of the other methods [17,18,24].

Considering that [17,18] and [24] follow the same “pre-train + fine tune” scheme as the proposed method, it indicates that the proposed method is very effective in this scheme. It is worth mentioning that the proposed method is far lower than the above methods in terms of both network complexity and training cost, which makes it more valuable to popularize in practical applications. Ref. [48] is essentially a transfer learning method, it utilizes the ORS dataset as the source domain for knowledge transfer instead of using it as pre-training data.

Table 3. Comparison with the SOTA methods on two datasets. The classification accuracy (%) averaged on ten runs are reported.

	[17]	[18]	[24]	[48]	Proposed.(CNN-VII)
SD1	85.33	80.67	84.33	89.50	87.67
SD2	81.67	78.67	80.50	83.67	82.33

5. Conclusions

To alleviate the bottleneck caused by the insufficient SAR training dataset that hinders the further performance improvement of ship classification in SAR imagers, this study makes three improvements to the existing widely adopted “pre-train + finetune” scheme and proposes a novel scheme of a shallow network with feature refinement pre-trained on a task-specific dataset. Extensive experiments have demonstrated the effectiveness of this scheme, and in-depth analysis has revealed the reasons behind the experimental outcomes.

This paper utilizes CNNs in the form of VGGNet as the backbone network for research, but it does not prevent the proposed scheme from being generalized to other types of network architectures, thanks to the fact that task-specific pre-training and feature refinement have no relevance to the network types. Some of the newly emerging methods aimed at improving feature discriminative ability or network performance, such as attention mechanisms, network trimming, etc., can also be used in conjunction with the proposed scheme.

Author Contributions: Conceptualization, H.L.; methodology, H.L. and S.W.; software, S.W.; validation, S.Z. and S.W.; formal analysis, H.L.; investigation, S.Z.; resources, R.W.; data curation, S.Z. and R.W.; writing—original draft preparation, S.W. and S.Z.; writing—review and editing, H.L. and R.W.; visualization, J.L.; supervision, H.L.; project administration, H.L.; funding acquisition, H.L. All authors have read and agreed to the published version of the manuscript.

Funding: This work was supported by the National Natural Science Foundation of China under Grant 62071030, Natural Science Foundation of Shandong Province under Grant ZR2022MD002, and Marine Project under Grant 2205cxzx040431.

Data Availability Statement: Not applicable.

Acknowledgments: The authors would like to thank the contributors to the FUSAR-Ship dataset and HR-SAR dataset for providing data support for this research. Meanwhile, the authors would like to thank all peer reviewers and associate editors for their constructive comments that significantly improved this paper.

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

References

1. National Research Council. Critical infrastructure for ocean research and societal needs in 2030. In *Technical Report*; National Academy of Sciences: Washington, DC, USA, 2011.
2. Margarit, G.; Tabasco, A. Ship classification in single-pol SAR images based on fuzzy logic. *IEEE Trans. Geosci. Remote Sens.* **2011**, *49*, 3129–3138. [[CrossRef](#)]
3. Xing, X.; Ji, K.; Zou, H.; Chen, W.; Sun, J. Ship classification in TerraSAR-X images with feature space based sparse representation. *IEEE Geosci. Remote Sens. Lett.* **2013**, *10*, 1562–1566. [[CrossRef](#)]
4. Jiang, M.; Yang, X.; Dong, Z.; Fang, S.; Meng, J. Ship classification based on superstructure scattering features in SAR images. *IEEE Geosci. Remote Sens. Lett.* **2016**, *13*, 616–620. [[CrossRef](#)]
5. Lang, H.; Wu, S.; Lai, Q.; Ma, L. Capability of geometric features to classify ships in SAR imagery. In Proceedings of the Image and Signal Processing for Remote Sensing XXII. International Society for Optics and Photonics, Edinburgh, UK, 26–28 September, 2016; Volume 10004, p. 1000415.
6. Lang, H.; Zhang, J.; Zhang, X.; Meng, J. Ship classification in SAR image by joint feature and classifier selection. *IEEE Geosci. Remote Sens. Lett.* **2016**, *13*, 212–216. [[CrossRef](#)]
7. Lang, H.; Wu, S. Ship Classification in Moderate-Resolution SAR Image by Naive Geometric Features-Combined Multiple Kernel Learning. *IEEE Geosci. Remote Sens. Lett.* **2017**, *14*, 1765–1769. [[CrossRef](#)]
8. Lin, H.; Song, S.; Yang, J. Ship classification based on MSHOG feature and task-driven dictionary learning with structured incoherent constraints in SAR images. *Remote Sens.* **2018**, *10*, 190. [[CrossRef](#)]
9. Xu, Y.; Lang, H.; Chai, X.; Ma, L. Distance metric learning for ship classification in SAR images. In Proceedings of the Image and Signal Processing for Remote Sensing XXIV. SPIE, Berlin, Germany, 10–12 September, 2018; Volume 10789, pp. 441–451.
10. Xu, Y.; Lang, H. Distribution shift metric learning for fine-grained ship classification in SAR images. *IEEE J. Sel. Top. Appl. Earth Obs. Remote Sens.* **2020**, *13*, 2276–2285. [[CrossRef](#)]
11. Zhou, G.; Zhang, G.; Xue, B. A maximum-information-minimum-redundancy-based feature fusion framework for ship classification in moderate-resolution SAR image. *Sensors* **2021**, *21*, 519. [[CrossRef](#)] [[PubMed](#)]
12. Wang, X.; Liu, C.; Li, Z.; Ji, X.; Zhang, X. Superstructure scattering features and their application in high-resolution SAR ship classification. *J. Appl. Remote Sens.* **2022**, *16*, 036507. [[CrossRef](#)]
13. Salerno, E. Using Low-Resolution SAR Scattering Features for Ship Classification. *IEEE Geosci. Remote Sens. Lett.* **2022**, *19*, 4509504. [[CrossRef](#)]
14. Bentes, C.; Velotto, D.; Tings, B. Ship classification in TerraSAR-X images with convolutional neural networks. *IEEE J. Ocean. Eng.* **2017**, *43*, 258–266. [[CrossRef](#)]
15. Li, J.; Qu, C.; Peng, S. Ship classification for unbalanced SAR dataset based on convolutional neural network. *J. Appl. Remote Sens.* **2018**, *12*, 035010. [[CrossRef](#)]
16. Xi, Y.; Xiong, G.; Yu, W. Feature-loss double fusion Siamese network for dual-polarized SAR ship classification. In Proceedings of the 2019 IEEE International Conference on Signal, Information and Data Processing (ICSIDP), Chongqing, China, 11–13 December 2019; pp. 1–5.
17. Dong, Y.; Zhang, H.; Wang, C.; Wang, Y. Fine-grained ship classification based on deep residual learning for high-resolution SAR images. *Remote Sens. Lett.* **2019**, *10*, 1095–1104. [[CrossRef](#)]
18. Li, Y.; Li, X.; Sun, Q.; Dong, Q. SAR image classification using CNN embeddings and metric learning. *IEEE Geosci. Remote Sens. Lett.* **2020**, *19*, 4002305. [[CrossRef](#)]
19. He, J.; Wang, Y.; Liu, H. Ship classification in medium-resolution SAR images via densely connected triplet CNNs integrating Fisher discrimination regularized metric learning. *IEEE Trans. Geosci. Remote Sens.* **2020**, *59*, 3022–3039. [[CrossRef](#)]
20. Zeng, L.; Zhu, Q.; Lu, D.; Zhang, T.; Wang, H.; Yin, J.; Yang, J. Dual-polarized SAR ship grained classification based on CNN with hybrid channel feature loss. *IEEE Geosci. Remote Sens. Lett.* **2021**, *19*, 4011905. [[CrossRef](#)]
21. Raj, J.A.; Idicula, S.M.; Paul, B. One-shot learning-based SAR ship classification using new hybrid Siamese network. *IEEE Geosci. Remote Sens. Lett.* **2021**, *19*, 4017205. [[CrossRef](#)]
22. Xu, X.; Zhang, X.; Zhang, T. Multi-Scale SAR Ship Classification with Convolutional Neural Network. In Proceedings of the 2021 IEEE International Geoscience and Remote Sensing Symposium IGARSS, Brussels, Belgium, 11–16 July 2021; pp. 4284–4287.
23. Zhang, T.; Zhang, X. Squeeze-and-excitation Laplacian pyramid network with dual-polarization feature fusion for ship classification in sar images. *IEEE Geosci. Remote Sens. Lett.* **2022**, *19*, 4019905. [[CrossRef](#)]
24. Zhang, T.; Zhang, X. Injection of traditional hand-crafted features into modern CNN-based models for SAR ship classification: What, why, where, and how. *Remote Sens.* **2021**, *13*, 2091. [[CrossRef](#)]
25. Zhang, T.; Zhang, X.; Ke, X.; Liu, C.; Xu, X.; Zhan, X.; Wang, C.; Ahmad, I.; Zhou, Y.; Pan, D.; et al. HOG-ShipCLSNet: A novel deep learning network with hog feature fusion for SAR ship classification. *IEEE Trans. Geosci. Remote Sens.* **2021**, *60*, 5210322. [[CrossRef](#)]
26. Zhang, T.; Zhang, X. A polarization fusion network with geometric feature embedding for SAR ship classification. *Pattern Recognition* **2022**, *123*, 108365. [[CrossRef](#)]
27. Li, Y.; Lai, X.; Wang, M.; Zhang, X. C-SASO: A Clustering-Based Size-Adaptive Safer Oversampling Technique for Imbalanced SAR Ship Classification. *IEEE Trans. Geosci. Remote Sens.* **2022**, *60*, 5231112. [[CrossRef](#)]

28. Ma, F.; Zhang, F.; Xiang, D.; Yin, Q.; Zhou, Y. Fast task-specific region merging for SAR image segmentation. *IEEE Trans. Geosci. Remote Sens.* **2022**, *60*, 5222316. [[CrossRef](#)]
29. He, J.; Chang, W.; Wang, F.; Liu, Y.; Wang, Y.; Liu, H.; Li, Y.; Liu, L. Group Bilinear CNNs for Dual-Polarized SAR Ship Classification. *IEEE Geosci. Remote Sens. Lett.* **2022**, *19*, 4508405. [[CrossRef](#)]
30. Zhu, H.; Guo, S.; Sheng, W.; Xiao, L. SBNN: A Searched Binary Neural Network for SAR Ship Classification. *Appl. Sci.* **2022**, *12*, 6866. [[CrossRef](#)]
31. Zheng, H.; Hu, Z.; Liu, J.; Huang, Y.; Zheng, M. MetaBoost: A Novel Heterogeneous DCNNs Ensemble Network With Two-Stage Filtration for SAR Ship Classification. *IEEE Geosci. Remote Sens. Lett.* **2022**, *19*, 4509005. [[CrossRef](#)]
32. Sun, Y.; Wang, Z.; Sun, X.; Fu, K. SPAN: Strong Scattering Point Aware Network for Ship Detection and Classification in Large-Scale SAR Imagery. *IEEE J. Sel. Top. Appl. Earth Obs. Remote Sens.* **2022**, *15*, 1188–1204. [[CrossRef](#)]
33. Ai, J.; Mao, Y.; Luo, Q.; Jia, L.; dao Xing, M. SAR Target Classification Using the Multikernel-Size Feature Fusion-Based Convolutional Neural Network. *IEEE Trans. Geosci. Remote Sens.* **2022**, *60*, 5214313. [[CrossRef](#)]
34. Meng, B.; Min, M.J.; Xi, Z.; Wang, L.G. A High Resolution SAR Ship Sample Database and Ship Type Classification. In Proceedings of the IEEE IGARSS 2020–2020 IEEE International Geoscience and Remote Sensing Symposium, Waikoloa, HI, USA, 26 September–2 October 2020; pp. 1774–1777.
35. Hou, X.; Ao, W.; Song, Q.; Lai, J.; Wang, H.; Xu, F. FUSAR-Ship: Building a high-resolution SAR-AIS matchup dataset of Gaofen-3 for ship detection and recognition. *Sci. China Inf. Sci.* **2020**, *63*, 140303. [[CrossRef](#)]
36. Rizaev, I.G.; Achim, A. SynthWakeSAR: A Synthetic SAR Dataset for Deep Learning Classification of Ships at Sea. *Remote Sens.* **2022**, *14*, 3999. [[CrossRef](#)]
37. Deng, J.; Dong, W.; Socher, R.; Li, L.J.; Li, K.; Li, F.-F. Imagenet: A large-scale hierarchical image database. In Proceedings of the Computer Vision and Pattern Recognition, Miami, FL, USA, 20–25 June 2009; pp. 248–255.
38. Oquab, M.; Bottou, L.; Laptev, I.; Sivic, J. Learning and Transferring Mid-level Image Representations Using Convolutional Neural Networks. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Columbus, OH, USA, 23–28 June, 2014; pp. 1717–1724.
39. Wang, Y.; Wang, C.; Zhang, H. Ship classification in high-resolution SAR images using deep learning of small datasets. *Sensors* **2018**, *18*, 2929. [[CrossRef](#)] [[PubMed](#)]
40. Lu, C.; Li, W. Ship classification in high-resolution SAR images via transfer learning with small training dataset. *Sensors* **2018**, *19*, 63. [[CrossRef](#)] [[PubMed](#)]
41. Lang, H.; Wu, S.; Xu, Y. Ship Classification in SAR Images Improved by AIS Knowledge Transfer. *IEEE Geosci. Remote Sens. Lett.* **2018**, *15*, 439–443. [[CrossRef](#)]
42. Xu, Y.; Lang, H.; Chai, X. Distribution discrepancy maximization metric learning for ship classification in synthetic aperture radar images. In Proceedings of the IEEE IGARSS 2019–2019 IEEE International Geoscience and Remote Sensing Symposium, Yokohama, Japan, 28 July–2 August 2019; pp. 1208–1211.
43. Xu, Y.; Lang, H.; Niu, L.; Ge, C. Discriminative adaptation regularization framework-based transfer learning for ship classification in SAR images. *IEEE Geosci. Remote Sens. Lett.* **2019**, *16*, 1786–1790. [[CrossRef](#)]
44. Xu, Y.; Lang, H. Ship classification in SAR images with geometric transfer metric learning. *IEEE Trans. Geosci. Remote Sens.* **2020**, *59*, 6799–6813. [[CrossRef](#)]
45. Lang, H.; Li, C.; Xu, J. Multi-Source Heterogeneous Transfer Learning via Feature Augmentation for Ship Classification in SAR Imagery. *IEEE Trans. Geosci. Remote Sens.* **2022**, *60*, 5228814. [[CrossRef](#)]
46. Lang, H. Semi-supervised Heterogeneous Domain Adaptation via Dynamic Joint Correlation Alignment Network for Ship Classification in SAR Imagery. *IEEE Geosci. Remote Sens. Lett.* **2022**, *19*, 4508105.
47. Song, Y.; Li, J.; Gao, P.; Li, L.; Tian, T.; Tian, J. Two-Stage Cross-Modality Transfer Learning Method for Military-Civilian SAR Ship Recognition. *IEEE Geosci. Remote Sens. Lett.* **2022**, *19*, 4506405. [[CrossRef](#)]
48. Zhao, S.; Lang, H. Improving Deep Subdomain Adaptation by Dual-Branch Network Embedding Attention Module for SAR Ship Classification. *IEEE J. Sel. Top. Appl. Earth Obs. Remote Sens.* **2022**, *15*, 8038–8048. . [[CrossRef](#)]
49. Krizhevsky, A.; Sutskever, I.; Hinton, G.E. Imagenet classification with deep convolutional neural networks. In *Advances in Neural Information Processing Systems*; Harrahs and Harveys: Stateline, CA, USA, 2012; pp. 1097–1105.
50. Simonyan, K.; Zisserman, A. Very Deep Convolutional Networks for Large-Scale Image Recognition. In Proceedings of the International Conference on Learning Representations, San Diego, CA, USA, 7–9 May, 2015; pp. 1–14.
51. He, K.; Zhang, X.; Ren, S.; Sun, J. Deep Residual Learning for Image Recognition. In Proceedings of the Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 27–30 June, 2016; pp. 770–778.
52. Huang, G.; Liu, Z.; Weinberger, K.Q. Densely Connected Convolutional Networks. In Proceedings of the 2017 IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July, 2017; pp. 2261–2269.
53. Khan, A.; Sohail, A.; Zahoora, U.; Qureshi, A.S. A survey of the recent architectures of deep convolutional neural networks. *Artif. Intell. Rev.* **2020**, *53*, 5455–5516. [[CrossRef](#)]
54. Boureau, Y.L.; Bach, F.; LeCun, Y.; Ponce, J. Learning mid-level features for recognition. In Proceedings of the 2010 IEEE Computer Vision and Pattern Recognition, San Francisco, CA, USA, 13–18 June 2010; pp. 2559–2566.

55. Xu, H.; Chen, Y.; Lin, R.; Kuo, C.C.J. Understanding CNN via deep features analysis. In Proceedings of the Asia-Pacific Signal and Information Processing Association Annual Summit and Conference (APSIPA ASC), Kuala Lumpur, Malaysia, 12–15 December 2017; pp. 1052–1060.
56. Escorcia, V.; Carlos Niebles, J.; Ghanem, B. On the relationship between visual attributes and convolutional networks. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Boston, MA, USA, 7–12 June, 2015; pp. 1256–1264.
57. Chollet, F. Keras. 2016. Available online: <https://github.com/fchollet/keras> (accessed on 1 September 2022) .
58. Abadi, M.; Agarwal, A.; Barham, P.; Brevdo, E.; Chen, Z.; Citro, C.; Corrado, G.S.; Davis, A.; Dean, J.; Devin, M.; et al. TensorFlow: Large-Scale Machine Learning on Heterogeneous Systems . *arXiv* **2016**, arXiv:abs/1603.04467