remote sensing

MDPI

Article

A Few-Shot Learning Method for SAR Images Based on Weighted Distance and Feature Fusion

Fei Gao [1], Jingming Xu [1], Rongling Lang [1,*], Jun Wang [1], Amir Hussain [2] and Huiyu Zhou [3]

[1] School of Electronic and Information Engineering, Beihang University, Beijing 100191, China
[2] Cyber and Big Data Research Laboratory, Edinburgh Napier University, Edinburgh EH11 4BN, UK
[3] Department of Informatics, University of Leicester, Leicester LE1 7RH, UK
* Correspondence: ronglinglang@buaa.edu.cn

Abstract: Convolutional Neural Network (CNN) has been widely applied in the field of synthetic aperture radar (SAR) image recognition. Nevertheless, CNN-based recognition methods usually encounter the problem of poor feature representation ability due to insufficient labeled SAR images. In addition, the large inner-class variety and high cross-class similarity of SAR images pose a challenge for classification. To alleviate the problems mentioned above, we propose a novel few-shot learning (FSL) method for SAR image recognition, which is composed of the multi-feature fusion network (MFFN) and the weighted distance classifier (WDC). The MFFN is utilized to extract input images' features, and the WDC outputs the classification results based on these features. The MFFN is constructed by adding a multi-scale feature fusion module (MsFFM) and a hand-crafted feature insertion module (HcFIM) to a standard CNN. The feature extraction and representation capability can be enhanced by inserting the traditional hand-crafted features as auxiliary features. With the aid of information from different scales of features, targets of the same class can be more easily aggregated. The weight generation module in WDC is designed to generate category-specific weights for query images. The WDC distributes these weights along the corresponding Euclidean distance to tackle the high cross-class similarity problem. In addition, weight generation loss is proposed to improve recognition performance by guiding the weight generation module. Experimental results on the Moving and Stationary Target Acquisition and Recognition (MSTAR) dataset and the Vehicle and Aircraft (VA) dataset demonstrate that our proposed method surpasses several typical FSL methods.

Keywords: synthetic aperture radar (SAR); convolutional neural network (CNN); radar target recognition; few-shot learning

Citation: Gao, F.; Xu, J.; Lang, R.; Wang, J.; Hussain, A.; Zhou, H. A Few-Shot Learning Method for SAR Images Based on Weighted Distance and Feature Fusion. *Remote Sens.* **2022**, *14*, 4583. https://doi.org/10.3390/rs14184583

Academic Editors: Bo Tang, Xinghua Li, Zongxu Pan, Fan Zhang, Zhongling Huang and Wei Yao

Received: 13 August 2022
Accepted: 9 September 2022
Published: 14 September 2022

1. Introduction

Thanks to its outstanding penetrating ability, synthetic aperture radar (SAR) has all-day and all-weather ground imaging ability with high spatial resolution. For a long time, SAR image recognition has been a research hotspot for obtaining essential information from targets. The recognition pipeline usually contains two components, a feature extractor and a classifier. In traditional recognition methods, the feature extractor is used to extract hand-crafted features, including geometric features [1], transformation features [2], scattering features [3], etc. However, traditional recognition methods, i.e., the design of these hand-crafted features, usually rely excessively on expert knowledge and lack the ability to generalize to novel class targets.

With the progress of deep learning theories in recent years, the convolutional neural network (CNN) approach has received attention from researchers in the realm of SAR image recognition because of its excellent automatic feature extraction ability. CNN-based methods are developed to further improve the recognition performance of SAR images by using a dual-branch CNN [4], using DenseNet and feature reuse strategy [5], combining CNN with support vector machine (SVM) [6], or fusing multi-size convolution

kernels [7]. In addition, CNN-based methods have made further progress in several specific SAR-related tasks, such as target detection [8], image segmentation [9,10], and more. Nevertheless, CNN is a data-driven algorithm that requires numerous high-quality labeled images to train the model. Due to this limitation, CNN usually suffers from problems with overfitting due to insufficient training samples, which may degrade the recognition accuracy of the CNN model.

In practice, it is arduous and time-consuming to obtain enormous labeled SAR images due to speckle noise and background clutter. Normally, there are only a few dozen labeled images or less in each category. Under this circumstance, the recognition accuracy of CNN drops substantially [11,12]. In order to tackle this issue, researchers have proposed several methods, such as training set augmentation, transferring knowledge from other domains, designing networks with fewer parameters, and others. For instance, Ding et al. proposed augmenting the training set using three strategies: translation, adding speckle noise, and pose synthesis [13]. In [14], the authors propose transferring knowledge from the simulated data domain to alleviate the problem of insufficient training samples for SAR images. Zhao et al. propose a novel network architecture that replaces the convolution layers with highway-units to make it possible to train a deeper network using limited SAR images [15]. Zhang et al. adopt the pruning and knowledge distillation strategy to build a lightweight CNN while maintaining high recognition accuracy [16]. Even though the above-mentioned methods have achieved great success with limited training SAR images, the training process needs dozens of training samples per class to ensure that the model can achieve satisfactory performance.

Therefore, few-shot learning (FSL) has attracted interest from researchers. FSL first uses plentiful labeled images from base categories to train the model, then generalizes the prior knowledge to novel categories [17]. With very limited labeled samples (normally one or five per class) from novel categories as reference information, FSL can recognize the unlabeled samples from novel categories [18].

Common FSL frameworks can be divided into two stages: feature extraction and classification. According to the framework, FSL methods first use the feature extraction network (e.g., CNN) to map samples into the feature space with lower dimensions. In the feature space, targets from the same class tend to aggregate and stay closer, whereas samples from different categories are usually far apart. In the classification stage, FSL methods use different metric functions to measure the similarity between samples in the feature space [19]. Researchers have applied FSL to the SAR target recognition problem. For instance, Tang et al. improved the recognition accuracy and reduced the time consumption of Siamese Networks [20]. Yang et al. combined the graph neural network (GNN) with the relation network to describe the relationship between query samples and support samples, which increases the recognition accuracy for SAR few-shot recognition tasks [21].

There are two specific special characteristics of SAR images: their large inner-class variety, and their high cross-class similarity [22]. The large inner-class variety means that targets of the same class can be very different in appearance due to the high angle sensitivity of SAR images [23]. In addition, high cross-class similarity means that targets from different categories can have quite similar appearances [24]. These characteristics may result in confusion in the feature space for targets from different categories, which poses challenges for the classification stage of certain FSL methods because they make decisions based on the distance between samples. Such methods include Siamese and the prototypical networks.

Meanwhile, the number of labeled SAR images from base classes is far lower than from the optical dataset, which leads to less prior knowledge that can be generalized into novel class recognition tasks. Therefore, combining other types of features with CNN features to enrich prior knowledge can be useful. Hand-crafted features have been widely used in SAR image classification, and have good robustness and stability. Researchers have combined hand-crafted features with CNN features to improve performance, although few have been applied to FSL tasks. Both the appropriate hand-crafted features and the fusion strategy should be chosen carefully.

To tackle the aforementioned problems mentioned and improve recognition performance with limited labeled SAR images, we propose a novel few-shot SAR image recognition method in this article. First, we introduce a hand-crafted feature insertion module (HcFIM) to combine CNN features with hand-crafted features to accumulate more prior knowledge in the training stage of FSL. Here, we select the histogram of oriented gradient (HOG) feature for combination, with the HOG features representing the edge and local information of targets, which complements the CNN features. In HcFIM, we introduce the weight concatenation method to fuse two types of features. To alleviate the large inner-class variety, we design a multi-scale feature fusion module (MsFFM) for aggregating the information from features with different scales. With the aid of information from different layers, targets of the same class can be more easily aggregated and different targets with high similarity can be identified more effectively. Finally, we propose a weighted distance classifier (WDC) to tackle the problem of high cross-class similarity. The weight-generation module in WDC takes the concatenation result of query images and prototypes as input and outputs the category-specific weights for query images. By distributing these weights on the Euclidean distance, the inter-class distance and separability between different categories in the embedding space can be increased. In addition, we design the weight-generation loss to supervise the weight generation module in the training process, as these generated weights should represent the difference between query images and prototypes.

In general, the contributions of our proposed method can be summarized as follows:

(1) In order to mitigate performance deterioration problems caused by insufficient labeled data, we combine CNN features with traditional hand-crafted features in a hand-crafted feature insertion module (HcFIM). The HcFIM introduces a weighted concatenation method to combine these two types of features, enhancing the expression capability and robustness of CNN features.

(2) We design a multi-scale feature fusion module (MsFFM) to combine the information from features with different scales. MsFFM has a tree-like architecture designed for fusing the local edge information from the lower levels of CNN and the global semantic information from the higher levels of CNN in a cascade fashion. With the aid of information from different layers, targets of the same class can be more easily aggregated and different targets with high similarity can be identified more effectively. The MsFFM and the HcFIM are combined with the CNN backbone to form the multi-feature fusion network (MFFN) as the feature extractor of our method.

(3) Because the high inter-class similarity within SAR images poses a challenge for few-shot recognition, we design a weighted distance classifier (WDC) to improve the recognition process of the prototypical network. The weight generation module in WDC consists of a multi-layer neural network, which is designed to generate suitable category-specific weights for query images in a data-driven way. By distributing these weights on the Euclidean distance, the inter-class distance and separability between different categories in the embedding space can be increased. In addition, weight generation loss is proposed to improve recognition performance by effectively facilitating the weight generation module.

The structure of our paper is organized as follows. Related works on few-shot learning are briefly reviewed in Section 2. Section 3 describes our proposed method in detail. We discuss the results of our experiments on the MSTAR and VA datasets in Sections 4 and 5. Finally, we conclude this article in Section 6.

2. Related Works

2.1. Few-Shot Learning Methods

With the development of hardware and deep learning theories, CNN-based methods have become widely used in computer vision tasks represented by image classification. Starting with AlexNet in 2012 [25], many different CNN backbone structures have been proposed to improve recognition accuracy. VGG uses smaller convolution kernels with a size of 3×3, which makes it possible to construct a deep network for large-scale image

recognition tasks [26]. By adding residual learning blocks, ResNet solves the degradation problem as the network grows deeper [27]. In addition these structural modifications, other mechanisms have been developed to improve image classification performance. The attention mechanism has been adopted to help CNN focus more on specific areas to obtain more information about targets and suppress irrelevant information. For instance, SE-Net (squeeze-and-excitation network), proposed by Hu et al., improves recognition accuracy by refining the information between different channels of feature maps [28].

Due to the overfitting problem, CNN-based methods usually fail when there are no sufficient labeled images for training. Therefore, FSL has attracted growing interest from researchers in recent years. Most mainstream FSL methods use a two-stage framework involving feature extraction and classification. In the feature extraction stage, FSL methods first use the feature extraction network (e.g., CNN) to map the image sample into the embedding space with lower dimensions. In the embedding space, the closer the samples are, the more likely they are to belong to the same class. In the classification stage, prediction results can be obtained by applying different metric functions to calculate the similarity between samples, such as L1 distance, cosine similarity, Euclidean distance, and learnable metric function. Siamese networks [29] and matching networks [30] both take an image pair as input; in these networks, the similarity between the paired images is obtained by calculating the L1 distance and the cosine similarity, respectively. To avoid comparing the similarity one by one, a prototypical network [31] first calculates the prototype of each class using an inner-class averaging operation, after which unknown samples can be classified into the category corresponding to the prototype with the smallest Euclidean distance to themselves. The relation network [32] replaces the distance-based similarity measurement strategy with a learnable relation module which can automatically learn a suitable metric function in the training process. In addition to optimizing the metric function, several methods have proposed optimizing the feature extraction network for better performance. For instance, few-shot embedding adaptation transformer (FEAT) [33] modifies the feature extraction network by adding a self-attention mechanism to CNN, allowing it construct a task-specific feature space.

2.2. Few-Shot Learning in SAR Image Recognition

Most existing few-shot learning methods for SAR images can be divided into metric-based, optimized-based, and generative-based methods. Metric-based methods aim to find an appropriate embedding space and a metric function to measure the similarity between samples in the embedding space. Wang et al. replaced the typical CNN structure with a convolutional bidirectional long short-term memory (Conv-BiLSTM) network to extract features with azimuth robustness [34]. Because GNN has strong feature extraction ability and can effectively describe the relationship between samples, Yang et al. utilized graph neural networks (GNN) to obtain the relationship between query instances and each prototype [21]. Yue et al. proposed a feature-matching classifier (FMC) for SAR image classification which uses a learnable CNN to measure the similarity between query images and each prototype [11]. In order to tackle the problem caused by large depression angle variation, Yang et al. designed an inference network based on a multi-layer graph attention network to serve as the metric function and output the prediction results [35]. Optimization-based methods focus on learning a suitable parameter initialization or updating strategy to quickly generalize prior knowledge to recognize targets from unseen categories. For instance, an optimization-based meta-learning framework, MSAR [36], is proposed for learning parameter initialization for SAR FSL tasks. MSAR introduced three transfer learning strategies to exploit prior knowledge, and used a task mining strategy to emphasize the harder tasks. Generative-based methods concentrate on generating SAR-like images to enrich the training set with the aim of alleviating the overfitting problem caused by insufficient training samples.

Even though the above-mentioned methods have obtained great results on the few-shot SAR image recognition problem, several researchers have applied methods to SAR

images after making adjustments on the basis of methods designed for optical images, ignoring the differences between SAR images and optical images, such as the large inner-class variety and high cross-class similarity of SAR images. Works cited in [35] provide solutions for the high cross-class similarity of SAR images by taking the graph attention network as the metric function. Different from the GNN-based method, we propose a novel metric function solution, i.e., using WDC to calculate the weighted distance in the feature space between query samples and prototypes, which is more intuitive and understandable. In addition, aiming to address large inner-class variety problems, we first modify the feature extraction process by adopting MsFFM to fuse fine-grained local features from lower levels and global semantic features from higher levels. By fusing features containing different information, targets from the same category can be better aggregated.

2.3. Combining of CNN Features and Traditional Hand-Crafted Features

Thanks to their ability to automatically extract features from big data, CNNs have been widely used by researchers in SAR image recognition tasks and have achieved great success. Nevertheless, the features extracted by a CNN are not very explainable, which makes the CNN model akin to an opaque "black box" model. In addition, most CNN-based recognition methods disregard traditional hand-crafted features, which have long been widely used in traditional SAR image recognition methods. These traditional hand-crafted features are designed by experienced experts and have better reliability and interpretability. Therefore, many researchers have considered combining hand-crafted features with the abstract features of CNNs for better feature representation of SAR images.

To improve the detection accuracy of traditional methods and confer better interpretability on CNN features, Zheng et al. proposed a multi-feature SAR target detection method [1]. This method first extracts geometric features of SAR images, then concatenates them with features directly captured via CNN. This multi-feature method achieves better accuracy and recall on the Sentinel-1 dataset than ones only using CNN features or geometric features. The Gabor filter has multi-orientation properties and good direction selection characteristics, and is compatible with the orientation-sensitive characteristic of SAR images. Inspired by this, Yu et al. initialized the inception module of GoogLeNet with multi-scaled and multi-orientated Gabor filters [37]. With 20% of the MSTAR training images used for training, the end-to-end model was able to achieve 90% recognition accuracy, which is significantly better than other comparable schemes. Other than geometric features, researchers have combined CNN features with attributed scattering center (ASC) features extracted from the complex data of SAR images. In order to promote better performance under the Extended Operating Conditions (EOCs) of the MSTAR dataset, a novel feature fusion framework named FEC was proposed by Zhang et al. in [38]. The complex data of SAR images is fed as the input of FEC to obtain ASCs, which are further used to construct a visual word bag. Then FEC uses the K-means algorithm to convert visual word bags into feature vectors. Finally, the discrimination correlation analysis (DCA) algorithm is adopted to fuse the CNN features and ASCs features for classification. This framework achieves over 99% recognition accuracy under SOCs and EOCs of the MSTAR dataset. Li et al. divided the SAR target by using the geometric scattering types of ASCs to learn the component information [39]. Combining the component information with the global information obtained by CNN can enhance the feature description ability and make the feature more robust. The above approaches fully illustrate that the fusion of CNN features and traditional hand-crafted features is feasible and effective. However, the ASC features are extracted from the complex data of SAR images, which means that they have certain limitations in applications, especially when complex domain data are unavailable.

3. Methods

3.1. Problem Definition

FSL is proposed to train a classifier that is capable of recognizing targets from novel categories with very few training samples. According to the requirements of the few-shot

learning recognition tasks, the dataset should comprise the training set D_{Base} and the test set D_{Novel}. The images of D_{Base} belong to the base categories, and D_{Novel} consists of images from novel categories, which are non-intersecting. We use the labeled images from D_{Base} to optimize our model. The test set D_{Novel} comprises the support set S_n and query set Q_n, which have a shared label space. Generally, images in S_n belong to C different categories, and each category has K annotated images. In this case, this few-shot recognition task is defined as a "C-way K-shot" problem and K is usually set to one or five. Q_n consists of unlabeled samples used to assess our model's performance.

Our method adopts an episodic training mechanism to match the training and testing process. To be specific, in every training episode we randomly select K images from each class of training set D_{Base} to build up the training support set S_b. Meanwhile, a certain number of images from each class are randomly selected to form the training query set Q_b.

3.2. Overall Framework

The framework of our proposed few-shot SAR image recognition method is shown in Figure 1. For better illustration, only a four-way task with one query image is shown in this figure. In the training process, the model is trained to recognize the images from Q_b based on the knowledge learned from S_b. Thus, the parameters of the few-shot recognition model can be updated with different S_b and Q_b.

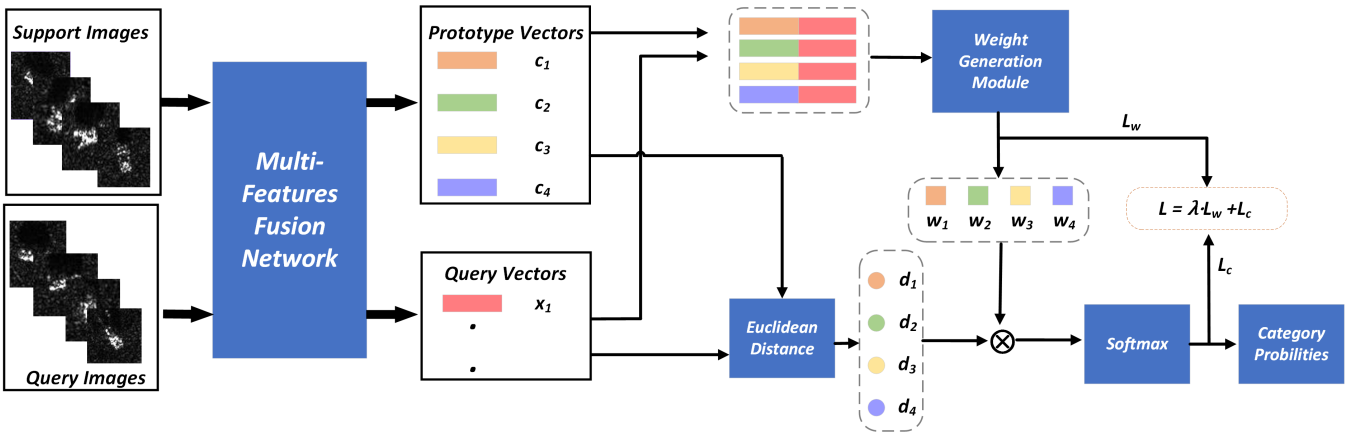


Figure 1. The framework of our proposed few-shot SAR image recognition method. The whole structure can be divided into two parts: the feature extraction network (MFFN) and the weighted Euclidean distance classifier. Support images are fed into the MFFN to extract features and then calculate the prototypes c_1, c_2, c_3, c_4 by averaging. Query feature vector x_1 is extracted by MFFN to calculate the corresponding Euclidean distance d_1, d_2, d_3, d_4. Afterwards, x_1 is concatenated with the prototypes c_i, and the concatenation vectors are sent to the weight generation module to calculate weights w_1, w_2, w_3, w_4. Finally, the weighted distances are obtained by multiplying weights with the Euclidean distance, and the softmax function is adopted to calculate the category probabilities on the weighted distances.

In each training episode, images from S_b and Q_b are first sent into the MFFN, from which the prototypes of different classes c_i can be obtained by performing the intra-class average operation on support features. The Euclidean distance d_i between query features and prototypes can be calculated as well. The MFFN contains a CNN backbone, SE-Net, MsFFM, and HcFIM, which are described in detail in Section 3.3. Afterwards, query features are concatenated with each prototype individually and the concatenated features are provided as the input of the weight generation module to generate category-specific weights w_i. Then, the weighted distance can be obtained by multiplying w_i and d_i. Finally, we adopt the softmax function on the weighted distance to calculate the category probabilities for query images. The cross-entropy loss L_c and weight-generated loss L_w are combined

to train the model jointly, and are minimized through the training process to update the parameters.

In the testing process, we use images from support set S_n and query set Q_n to assess the effectiveness of our method. S_n and Q_n have a shared label space which is non-intersecting with that of D_{Base}. First, we utilize the trained MFFN to extract the features of images from S_n and Q_n, from which the prototypes of novel categories can be obtained by class-wise averaging. Next, weights for query images from novel classes are generated by the weight generation module in WDC. Finally, the softmax function is utilized on the weighted distance to obtain the classification results of each query image.

3.3. Multi-Feature Fusion Network

Several previous research studies have indicated that features extracted by different layers contain different and complementary information. For instance, features from lower layers have more local edge details, while higher-level features include more global semantic information [40,41]. In addition, because traditional hand-crafted features have better robustness and interpretability than abstract CNN features, the feature expression capability can be enhanced by combining these two complementary features. In order to exploit the complementary information from the multi-scale features and combine CNN features with traditional hand-crafted features, MFFN is proposed in this article for feature extraction. As shown in Figure 2, MFFN mainly comprises two modules, the MsFFM and the HcFIM. The MsFFM is designed to fuse the features with different scales from each layer of CNN and the HcFIM is adopted for combining modern CNN features with the traditional hand-crafted features. In addition, the channel attention mechanism is embedded into the MFFN to optimize the feature expression.

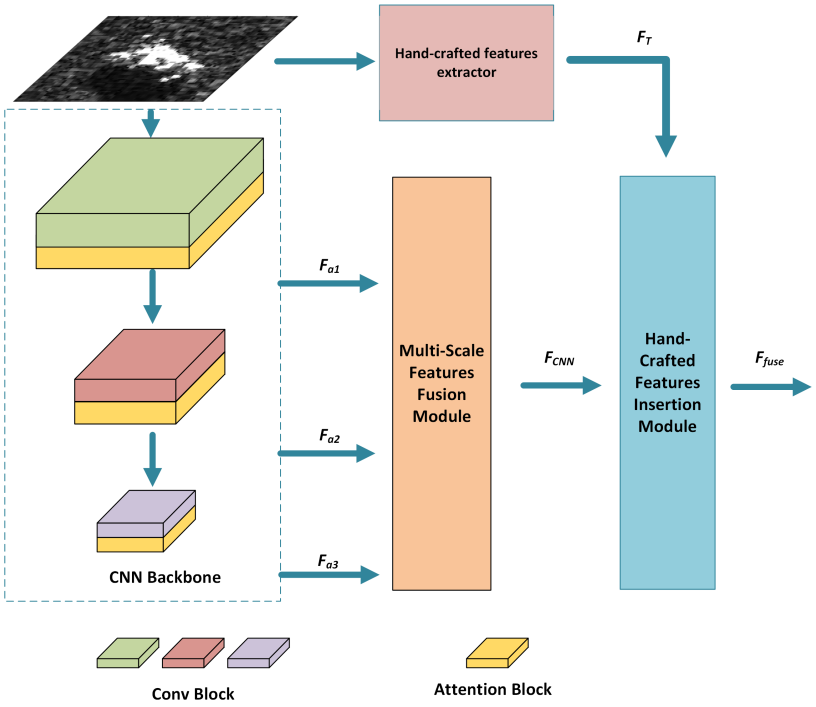


Figure 2. The overall architecture of the MFFN includes the MsFFM and the HcFIM. The channel-optimized features with different scales F_{a1}, F_{a2}, F_{a3} are fed into the MsFFM to utilize the complementary information from different layers. Afterwards, the fused feature F_{CNN} is combined with the traditional hand-crafted features F_T in the HcFIM. F_{fuse} denotes the final output feature of the MFFN.

3.3.1. Channel Attention Module

As shown in Figure 2, the attention mechanism is embedded into the backbone network, which is utilized to optimize channel-wise information. As a lightweight channel

attention module, SE-Net is capable of enhancing the feature representations without decreasing the computational efficiency [28]. Therefore, we employ SE-Net as the attention mechanism, through which the MFFN can accentuate the features with more essential information and inhibit classification-irrelevant features. Aided by the attention mechanism, our model can pay more attention to informative features.

Suppose feature maps $F_l \in \mathbb{R}^{C\times H\times W}$ are extracted by the l^{th} layer of the CNN, where C, H, and W denote the number of channels, height, and width of F_l. The operation process of SE-Net is shown in Figure 3:

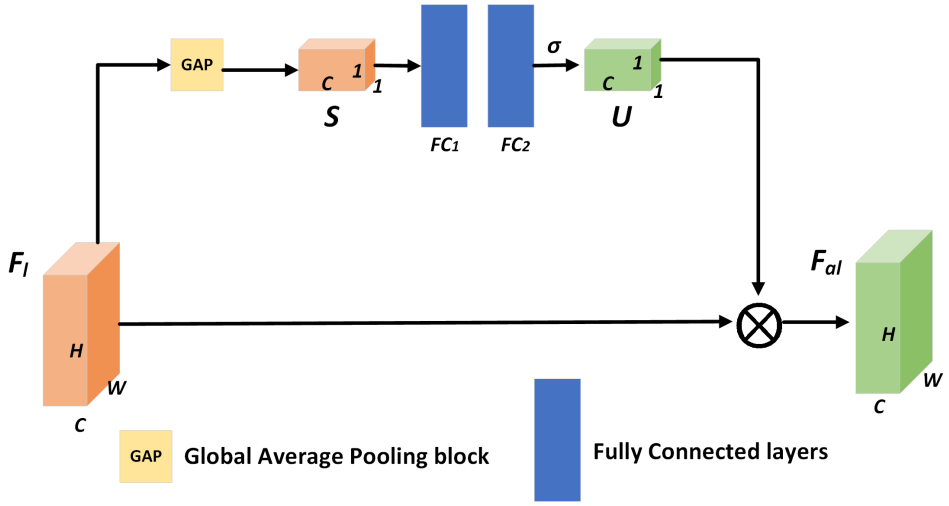


Figure 3. The operation process of the SE-Net. The channel descriptors are first generated by global average pooling, then the attention values are calculated by the fully-connected layers. Finally, the channel-refined features can be calculated by performing channel-wise multiplication. $\otimes$ represents the channel-wise multiplication operation and σ denotes the sigmoid function.

SE-Net first shrinks the space feature to obtain a feature descriptor by performing global average pooling. The global pooling feature map can be calculated using (1)

$$s_c = GAP(f_c) = \frac{1}{H\times W}\sum_{i=1}^{H}\sum_{j=1}^{W} f_c(i,j) \tag{1}$$

where f_c denotes the feature map from the c^{th} channel. Next, global pooling features $S = [s_1, s_2, \ldots, s_C] \in R^{C\times 1\times 1}$ are delivered to the fully-connected layers to calculate the attention values which reflect the importance of different channels:

$$U = Sigmoid(FC_2(ReLU(FC_1(S)))) \tag{2}$$

where $FC_1(\cdot)$ and $FC_2(\cdot)$ are two fully-connected layers and $ReLU(\cdot)$ and $Sigmoid(\cdot)$ are ReLU and sigmoid functions, respectively. Finally, the channel-refined feature can be calculated by performing channel-wise multiplication on U and F_l:

$$F_{al} = f_{Att}(F) = U \otimes F_l \tag{3}$$

where $\otimes$ represents the channel-wise multiplication operation. By assigning weights to different channels, CNN can place more attention on the essential channels and inhibit the redundant channels.

3.3.2. Multi-Scale Feature Fusion Module

SAR images from different categories are quite similar, such as T62 and T72. Furthermore, the same SAR targets with different azimuth angles have diverse backscattering

behaviors, which leads to various patterns in the same category. Therefore, SAR images have large inner-class variety and high cross-class similarity.

Most typical CNN models only utilize the features extracted by the last layer for classification, i.e., only single-scale features are adopted as final features. Although the features can be passed through the whole network layer by layer, shallow ones are likely to be diluted, resulting in feature loss and decreasing the feature representation capability. In this case, the local fine-grained features from the lower level of CNN are ignored. As the network grows deeper, the feature maps' spatial resolution continuously shrinks and the feature maps contain more semantic information rather than local edge information. The lower-level features contain more fine-grained local information, which complements the higher-level features containing global semantic information. In order to decrease the feature loss risk and exploit complementary multi-scale information, a cascade tree-like multi-scale feature fusion module (MsFFM) is designed to fuse the features with different scales. The overall structure of the MsFFM is shown in Figure 4.

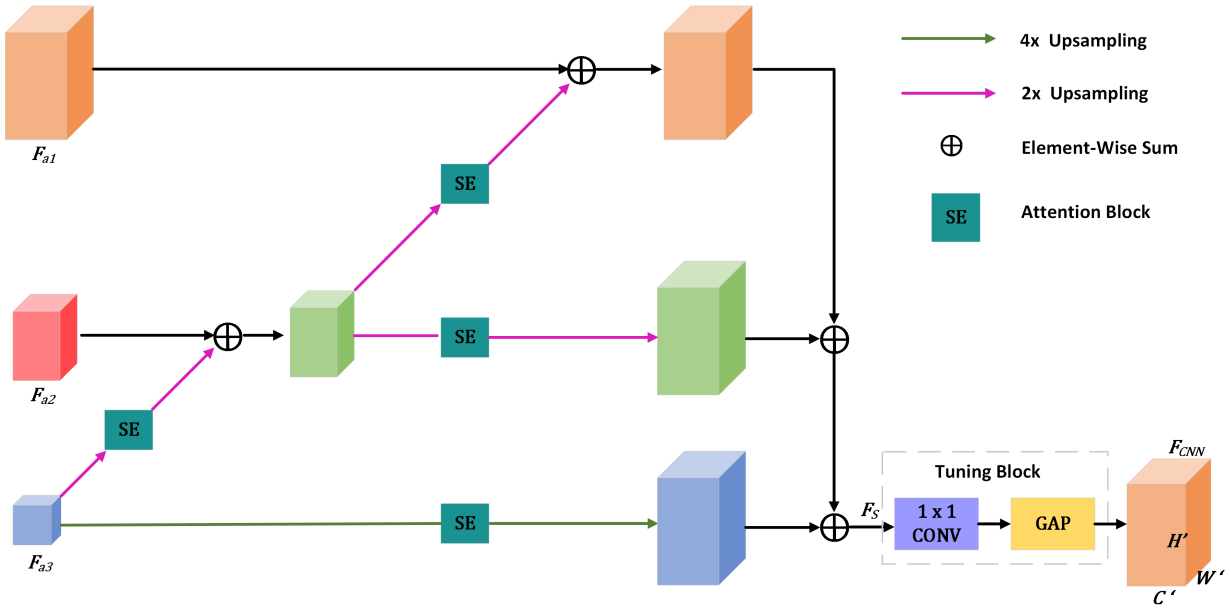


Figure 4. Framework of MsFFM. Features with smaller resolution are upsampled and then provided as the input to the channel attention block. The channel-optimized upsampled features are added element-wise to the adjacent features with higher resolution. Finally, we merge the all-upsampled feature map to further aggregate information, and the tuning block is adopted to refine the fused multi-scale feature. In this figure, the magenta arrow and green arrow represent the two-times upsampling operation and four-times upsampling operation, respectively, $\oplus$ denotes the elementwise sum operation, and F_{CNN} is the feature map after multi-scale fusion.

We use a three-layer CNN as the backbone network of our proposed method; the channel-refined SAR feature maps can be represented as $F_{al} = [F_{a1}, F_{a2}, F_{a3}]$. First, feature maps with smaller resolutions are upsampled by the deconvolution method and then provided as the input to the attention block to refine channel information. The channel-optimized upsampled features are added elementwise to the adjacent features with higher resolution. Different scales of features from adjacent layers can be fused successively through the process. On the right-hand side of Figure 4, we merge all upsampled feature maps to further aggregate information from different layers in the fusing process. Specifically, we first perform quadruple upsampling on F_{a3}, then perform double upsampling on the fusion maps of F_{a2} and F_{a3}. Finally, all the upsampled feature maps are aggregated via element-wise sum operation. After each upsampling, we utilize SE-Net to optimize the channel information in order to enhance the feature representations.

Even though the fused features reserve the essential information from features of different scales, the MsFFM unavoidably conserves a few irrelevant features in the fusing process. Therefore, a tuning block is used to refine the fused features. The tuning block consists of a convolutional layer with 1×1 kernels and a global average pooling layer. Convolutional kernels with a size of 1×1 are capable of mixing the information from different channels, through which certain category-related features can be further extracted for classification; in this way, the recognition performance can be increased. The tuned features can be produced as

$$F_{CNN} = GAP(ReLU(Conv(F_s))) \tag{4}$$

3.3.3. Hand-Crafted Feature Insertion Module

Currently, methods based on CNN are widely applied in SAR target recognition tasks. One significant advantage of these modern CNN-based methods is the capability of automatic feature extraction without expert experience. However, features extracted by CNN are abstract and less explainable, which limits their further application in the present context due to the high reliability requirements of SAR target recognition tasks. In addition, the sparseness of labeled SAR images on account of the difficulty of sample annotations lead to overfitting problems for CNN-based methods, in turn degrading the recognition performance.

Prior to the wide application of CNN-based methods, many different features were developed to describe the information in images, including the hu moment feature [42], gabor feature [43], local binary patterns (LBP) [44], histogram of oriented gradient (HOG), principal component analysis (PCA) [45], etc. These traditional hand-crafted features are designed by experienced experts based on mature theories, contributing to their good interpretability and high robustness. Methods based on these features have achieved satisfactory accuracy in SAR target recognition tasks. For example, Huang et al. utilized Hu moment features, scattering center features, and local radar cross section (LRCS) features to identify the attributes of different types of SAR targets [46]. They adopted the KNN classifier to recognize different ship targets obtained by Sentinel-1 SAR radar based on these hand-crafted features. The recognition accuracy of their method is comparable with the CNN-based method proposed by Wang et al. [47], demonstrating the effectiveness of hand-crafted features in SAR target recognition tasks.

Generally, the advantage of CNN-based methods is their automatic feature extraction capability without need for expert experience. However, the recognition performance of CNN-based methods can become degraded with limited training data due to overfitting problems. Considering that traditional hand-crafted features have good interpretability and high robustness, we seek to fuse CNN features and traditional hand-crafted features with the aim of enhancing the feature extraction capability of CNN-based methods with limited training data. Because these two types of features have complementary benefits, CNN features can be optimized and corrected by hand-crafted features under conditions of limited training sample data, allowing the robustness and representation of CNN features to be improved.

In order to combine these two types of features, we added a hand-crafted feature insertion module (HcFIM) to our feature extraction network. As shown in Figure 5, we adopt the weighted concatenation method proposed by Zhang et al. in [48] to construct the HcFIM. Considering the disparity in feature dimensions between the two types of features, the model is mainly optimized for features with higher dimensions. In comparison, the impact of features with lower dimensions is weakened, which leads to the occurrence of overfitting. Therefore, the balanced feature dimension is needed to avoid overfitting occurrences. Specifically, fully-connected layers are adopted to balance the dimension between the two types of features:

$$F_{To} = FC(F_T) \tag{5}$$

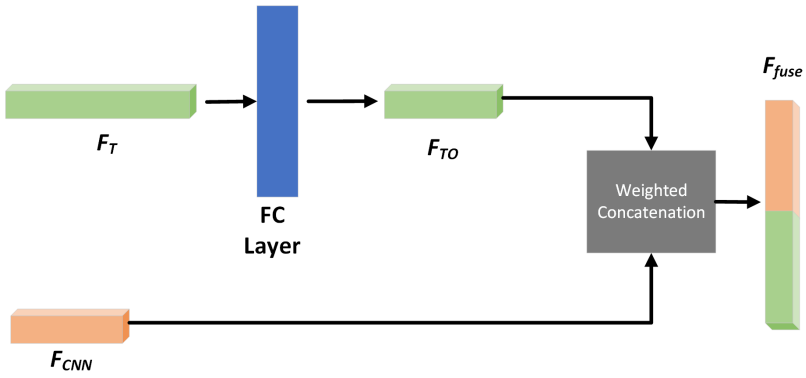


Figure 5. The weighted concatenation method for fusing CNN features F_{CNN} and traditional hand-crafted features F_T. First, a fully-connected layer is adopted to ensure that the dimension of F_T is unified with that of F_{CNN}. Then, we set two learnable weight parameters α and β to reveal the importance of different features. Finally, the two features are fused by the concatenation operation.

In order to reveal the importance of different features, two learnable parameters α and β are designed to be the weight coefficient for each feature. We use two extra neurons in our network to adaptively learn alpha and beta, respectively. In the training process, a softmax function is adopted to ensure that their sum equals one. These learnable parameters can reflect the importance of different types of features. During the training process, these two parameters can be updated through backward propagation together with the whole model. The weighted concatenation can be represented as Equation (6):

$$F_{fuse} = Cat(\alpha \cdot F_{CNN}, \beta \cdot F_{To}) \tag{6}$$

where F_{CNN} denotes the multi-scale fused CNN feature and F_{To} denotes the hand-crafted feature with the same balanced dimension. After fusing the CNN and hand-crafted features, F_{fuse} is the output feature of the MFFN, which is utilized for few-shot recognition.

3.4. Weighted Distance Classifier

We adopt the episodic training strategy to correspond the training process with the testing process. More specifically, we randomly select K images from each class of the training set D_{Base} to build up the training support set S_b in each training episode. A certain number of images from each class are randomly chosen to form the training query set Q_b. Suppose x_{ij} denotes the ith sample of the jth class from S_b; then, the prototype of the jth class can be calculated as follows:

$$c_j = \frac{1}{M}\sum_{i=1}^{M} f_\varphi(x_{ij}) \tag{7}$$

where M represents the number of images from the jth class in S_b, $f_\varphi(\cdot)$ represents the functional expression of MFFN that can extract input images into a feature vector, and φ represents parameters in the network.

After obtaining the prototype for each category, the Euclidean distance between query image features and each prototype can be calculated. The lower Euclidean distance represents the higher similarity with the prototype in the embedding space, which means this query image has a higher probability of belonging to the class corresponding to the prototype.

For a given query sample $(x_i, y_i) \in Q_b$, the feature extracted by MFFN can be represented by $f_\varphi(x_i)$ and the Euclidean distance between $f_\varphi(x_i)$ and one of the prototypes c_j can be calculated as Equation (8):

$$d\left(f_\varphi(x_i), c_j\right) = \left\| f_\varphi(x_i) - c_j \right\|^2 \tag{8}$$

Due to the high cross-class similarity, SAR targets of different categories can be easily confused in the embedding space, which presents a challenge for classification. To alleviate this problem, we propose a weighted distance classifier (WDC) to maximize the Euclidean distance between different classes in the embedding space, which makes instances from different categories more distinguishable.

As shown on the upper right-hand side of Figure 1, the query feature maps and prototypes are first concatenated separately, then the concatenation vectors are fed into the weight generation module to produce the class-specific weights of query images. More specifically, the weight generation module consists of a convolution layer and two fully-connected layers, through which we can obtain the weights corresponding to different categories for query images. The parameters of the weight generation module can be updated during the training process in a data-driven way. Therefore, it can generate suitable weights for each query image, which can contribute to increasing the inter-class separability. Suppose $g_\psi(\cdot)$ denotes the weight generation module, where ψ represents the learnable parameters; then, the weight can be calculated as follows:

$$w_{ij} = g_\psi\big(cat\big(f_\varphi(x_i), c_j\big)\big) \tag{9}$$

where w_{ij} is the weight of the input sample x_i for class j, and $cat(\cdot)$ denotes the vector concatenation. Ideally, the generated weights measure the difference between the query image and each prototype. Specifically, a greater the difference between a query image and a prototype leads to a higher weight being generated. On the contrary, as the query image becomes more similar to the corresponding prototype, lower weights are generated. Then, the weighted Euclidean distance can be calculated by multiplying the Euclidean distance by the corresponding weights. The weighting operation can increase the inter-class distance by "pushing" the query feature maps away from the prototypes of other categories. Thus, the few-shot recognition performance can be improved effectively. Finally, the softmax classifier is adopted to obtain the probability of x_i belonging to class j:

$$p(y_i = j|x_i) = \frac{\exp\big(-w_{ij} \cdot d\big(f_\varphi(x_i), c_j\big)\big)}{\sum_k \exp\big(-w_{ik} \cdot d\big(f_\varphi(x_i), c_k\big)\big)} \tag{10}$$

3.5. Loss Function

The classification loss and the weight-generation loss are combined to guide the operation of our proposed method jointly. The classification loss uses the cross-entropy loss, which is defined as Equation (11):

$$\mathcal{L}_c = -\frac{1}{N}\sum_{i=1}^{N} logP(y_i = j|x_i) \tag{11}$$

where (x_i, y_i) denote the input sample and the corresponding ground-truth label, respectively, and N represents the number of samples.

In our proposed method, the WDC is designed to generate weights and calculate the weighted Euclidean distance. In particular, the weight generation module of the WDC is composed of a learnable multi-layer network, through which we can obtain the class-specific weights. As we mentioned above, the generated weights should represent the difference between the query image and each prototype. To be more specific, if a query image i is quite different from class j, the corresponding weight w_{ij} should be larger in order to make the weighted distance larger, which can increase the separability from other categories. However, this module cannot produce pertinent weights without a specific loss function guiding the module. Thereby, the weight-generation loss is designed to guide this module during the training process, ensuring that the WDC can increase the separability between different categories. The weight-generation loss can be calculated as follows:

$$L_w = -\frac{1}{Q}\sum_{i=1}^{Q}\sum_{j=1}^{C}(-w_{ij})\cdot \mathbf{y}_{i,j} \tag{12}$$

where w_{ij} denotes the weight of the ith sample for the jth class. In the training process, the WDC can generate suitable weights by optimizing $\mathcal{L}_w$, meaning that the inter-class distance can be increased, which contributes to the improvemed performance of our method. The joint loss of our proposed method is defined by Equation (13):

$$\mathcal{L} = \mathcal{L}_c + \lambda \cdot \mathcal{L}_w \tag{13}$$

where λ is the balance hyperparameter.

4. Results

4.1. Datasets

Here, we first evaluate our method on the Moving and Stationary Target Acquisition and Recognition (MSTAR) benchmark dataset. Then, we perform several auxiliary experiments on the Vehicle and Aircraft (VA) dataset to compare the generalization capability of different methods.

4.1.1. MSTAR Dataset

The MSTAR dataset is a benchmark dataset for SAR image recognition. The images in the MSTAR dataset were obtained by an imaging radar with a resolution of 0.3 m $\times$ 0.3 m. This dataset includes ten different types of targets, namely, T62, T72, BMP2, BRDM2, BTR60, BTR70, D7, ZIL131, 2S1, and ZSU234. Figure 6 shows a comparison of SAR and optical images.

(**a**) T62 (**b**) T72 (**c**) BMP2 (**d**) BRDM2 (**e**) BTR60

(**f**) BTR70 (**g**) D7 (**h**) ZIL131 (**i**) 2S1 (**j**) ZSU234

Figure 6. The optical and SAR images in the MSTAR dataset.

The size of the images is 64 × 64, and each type of target contains two different depression angles of 15° and 17°. Table 1 displays the number of targets from each class in the MSTAR dataset.

Table 1. The number of images from each class in the MSTAR dataset.

Class	Depression Angle	No.	Depression Angle	No.
T62	15°	273	17°	299
T72	15°	196	17°	232
BMP2	15°	195	17°	233
BRDM2	15°	274	17°	298
BTR60	15°	195	17°	256
BTR70	15°	196	17°	233
D7	15°	274	17°	299
ZIL131	15°	274	17°	299
2S1	15°	274	17°	299
ZSU234	15°	274	17°	299
Total		2425		2747

According to the few-shot problem definition mentioned in Section 3.1, we split the MSTAR dataset into a training set D_{Base} and test set D_{Novel}. The test set is composed of a support set S_n and a query set Q_n. In this paper, images corresponding to the image types T62, T72, BRDM2, BTR60, BTR70, and ZIL131 with depression angles of both 15° and 17° are used to construct the training set. The test set contains the target images of four types (BMP2, D7, 2S1, and ZSU234). From Table 2, we can see that K labeled images of these four categories with a depression angle of 17° are randomly selected to construct the support set. Following the common few-shot settings, K is set to one and five. The query set Q_n contains 1017 target images from these four categories with a depression angle of 15° used to evaluate the recognition performance of each method.

Table 2. The number of images in the training, support, and query sets of the MSTAR dataset.

Training Set D_{Base}			Test Set D_{Novel}					
			Support Set S_n			Query Set Q_n		
Class	Depression Angle	No.	Class	Depression Angle	No.	Class	Depression Angle	No.
T62	15°, 17°	572	BMP2	17°	K	BMP2	15°	195
T72	15°, 17°	428	D7	17°	K	D7	15°	274
BRDM2	15°, 17°	572	2S1	17°	K	2S1	15°	274
BTR60	15°, 17°	451	ZSU234	17°	K	ZSU234	15°	274
BTR70	15°, 17°	429						
ZIL131	15°, 17°	573						
Total		3025	Total	17°	4 × K	Total	15°	1017

4.1.2. VA Dataset

The images in the VA dataset were obtained using a C-band SAR sensor with a resolution of 0.5 m × 0.5 m. This dataset includes five types of vehicle targets and two types of aircraft targets: car, truck, bus, MPV, fire truck, airliner, and helicopter. In this dataset, the image size of aircraft targets is 128 × 128 and that of vehicle targets is 64 × 64. We cropped the aircraft images from 128 × 128 to 64 × 64 for a consistent input size. Table 3 shows the number of different targets in the VA dataset.

Following similar few-shot settings as those for the MSTAR dataset, the VA dataset is partitioned into a training set D_{Base} and test set D_{Novel}. The test set is composed of a support set S_n and a query set Q_n. Table 4 displays the details of the three sets. The training set contains car, truck, and helicopter images. From the remaining four categories, one labeled image from each category is randomly chosen to form the support set. Finally,

the remaining 139 images are used to construct the query set Q_n used to evaluate the recognition performance of different methods on the VA dataset.

Table 3. The number of images from each class in the VA dataset.

Class	Car	Truck	Bus	MPV	Fire Truck	Airliner	Helicopter	Total
No.	35	35	35	35	35	38	70	283

Table 4. The number of images in the training, support and query sets of the VA dataset.

Training Set D_{Base}		Test Set D_{Novel}			
		Support Set S_n		Query Set Q_n	
Class	No.	Class	No.	Class	No.
Car	35	Bus	1	Bus	34
Truck	35	MPV	1	MPV	34
Helicopter	70	Fire Truck	1	Fire Truck	34
		Airliner	1	Airliner	37
Total	140	Total	4	Total	139

4.2. Implementation Details

The backbone CNN of MFFN includes three convolution modules, each consisting of a convolution layer, a batch normalization layer, and a max pooling layer. Each convolution layer contains 64 kernels and the kernel size is 3×3. The SE-Net comprises a global average pooling layer and two fully-connected layers, which contain 8 and 64 neurons, respectively.

We resize the input images to 64×64 uniformly without any other preprocessing, and our model is trained with no data augmentation. The ADAM optimizer is adopted to optimize our model in the training process. In order to avoid the local optimum, we adopt the learning rate decay strategy through the training process. The learning rate is initialized to 0.001 and attenuated according to Equation (14) every iteration:

$$new\_lr = lr \times \left(1 - \frac{current\_iter}{max\_iter}\right)^{0.8} \tag{14}$$

where *current_iter* and *max_iter* represent the current and total number of iterations, respectively, and *lr* and *new_lr* denote the current learning rate and the updated learning rate, respectively.

Because the gradient mainly exists on the edge of the target, the local information and edge information of targets can be well described by the gradient. The histogram of oriented gradient (HOG) feature was first designed for human detection [49,50]; later, scholars applied the HOG feature to the SAR ship classification tasks [51,52]. As a hand-crafted local feature, the HOG feature contains more edge and shape information, which can complement the global features extracted by CNN. Therefore, we fuse the HOG and CNN features in HcFIM to improve recognition accuracy. We discuss the effectiveness of inserting different hand-crafted features in the following section.

The weight generation module in the WDC comprises two fully-connected layers. The neuron numbers of these two fully-connected layers are 256 and 1, respectively. In addition, we use the deconvolution layers as the up-sampling method. We set the kernel size at 4×4 and the step at two. The channel number is not changed during the up-sampling process.

4.3. Evaluation Metrics

Accuracy is the main criterion for evaluating the recognition performance of all methods, and is defined as follows:

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \tag{15}$$

where TP denotes the number of true positives, TN denotes the number of true negatives, FP denotes the number of false positives, FN denotes the number of false negatives. The number of correctly classified targets is $TP + TN$ and the total number of targets is $TP + TN + FP + FN$.

In order to avoid randomness in our experimental results, all the experiments consist of 20 repetitions and a support set is randomly selected each time.

4.4. Recognition Performance Comparison

We first evaluate the performance of our method on the MSTAR benchmark dataset. The support set of the MSTAR dataset contains four types of targets, each composed of K labeled images. Following the typical settings, K is set to one and five in our experiments, e.g., the experimental setting of the MSTAR dataset is a "4-way 5-shot" and "4-way 1-shot". To further verify the effectiveness of our method, we construct comparative experiments on the VA dataset, although we only perform the "4-way 1-shot" experiment on the VA dataset due to the limited number of images. The comparison methods are as follows:

(1) Siamese Network. A Siamese network takes an image pair as input and then uses two share-weighted CNNs to extract features, from which the Siamese network can measure the similarity between the feature pairs to judge whether they belong to the same category.
(2) Matching Network. Feature extraction and classifier modules are combined to form the matching network. The classifier module adopts the cosine metric function to obtain the similarity score of two feature maps extracted by the feature extraction module. The matching network uses attention and external memory mechanisms to accelerate the training process.
(3) Prototypical Network. A prototypical network first uses a standard CNN to extract image features, then calculates the prototypes for each category using a feature averaging operation. The softmax function is introduced to classify query images based on the Euclidean distance between their features and each prototype.
(4) Relation Network. A relation network creates a relation module which can automatically learn a metric function in the training process to measure the similarity between features and prototypes.
(5) Few-Shot Embedding Adaptation with Transformer (FEAT) utilizes a self-attention mechanism to modify the feature extraction process, which makes the extracted features more discriminative and task-specific.

4.4.1. "4-Way 5-Shot" Experiment on the MSTAR Dataset

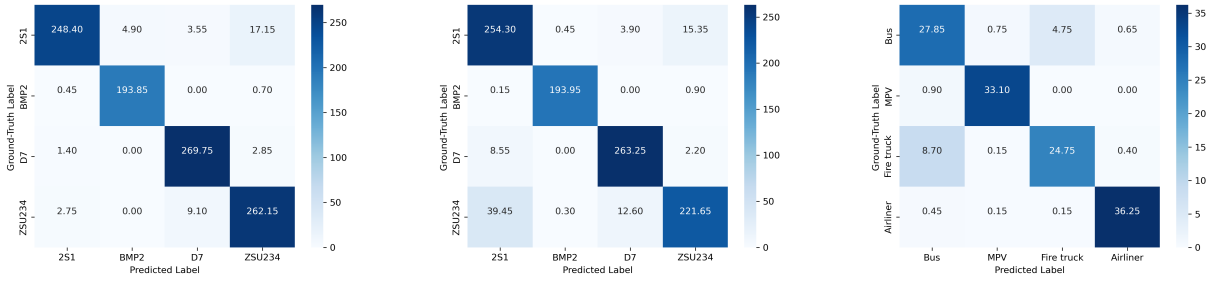
For the MSTAR dataset, Table 5 displays the "4-way 5-shot" recognition performance comparison between different methods. Our proposed method achieves 95.91% accuracy, which outperforms other comparison methods. The average accuracy of the matching network, Siamese network, prototypical network, and FEAT is 91.58%, 72.49%, 87.80%, and 81.23%, respectively, all of which obviously underperform our proposed method. This is because the similarity metric functions of these four comparison methods are distance-based with fixed parameters, which directly calculate the Euclidean or Cosine distance between query features and prototypes. In our method, we propose a WDC which contains a multi-layer neural network to calculate the weights for every query sample. Then, the weights are distributed to the Euclidean distance to the prototypes. The WDC is capable of updating the category-specific weights for query samples in the training process, meaning that the learned weights can compensate for defects of the parameter-fixed Euclidean

metric. In conjunction with our newly designed weight-generation loss, the weighted Euclidean distance can effectively decrease the inter-class similarity of SAR images, thereby improving the recognition accuracy.

Similar to the relation network's relation module, the weight generation module in the WDC is a learnable similarity metric function; however, the average accuracy of our method is superior to the relation network by 15.06%. The first reason for this improvement is that our method has better feature extraction capability through the help of SE-Net, MsFFM, and HcFIM. Another reason is that our method combines the outputs of the weight-generation module with the Euclidean distance. In other words, the self-learning metric function and the distance-based metric function are combined in the WDC, from which we can make full use of the advantages of these two metric functions, leading to further improved recognition performance.

Compared to other methods that only use CNN to extract features for SAR images, we modify the feature extraction process by adopting SE-Net, MsFFM, and HcFIM. Embedding SE-Net into CNN can make CNN focus more on essential channels and suppress unnecessary ones. The MsFFM fuses the features of different scales in a tree-like cascade fashion. Unlike single-scale CNN models, the MsFFM is capable of fusing the features of different scales from different levels of the CNN. As several traditional hand-crafted features have achieved good accuracy in SAR target recognition tasks, we introduce the HcFIM to fuse CNN and traditional hand-crafted features in order to improve the robustness and representation of features.

Figure 7a displays the average confusion matrix of our method on the MSTAR dataset under the "4-way 5-shot" condition. It can be seen that targets of BMP2 have the highest average recognition accuracy. The reason for this is that the BMP2's distinctive features as an infantry fighting vehicle make it more easy to recognize correctly.



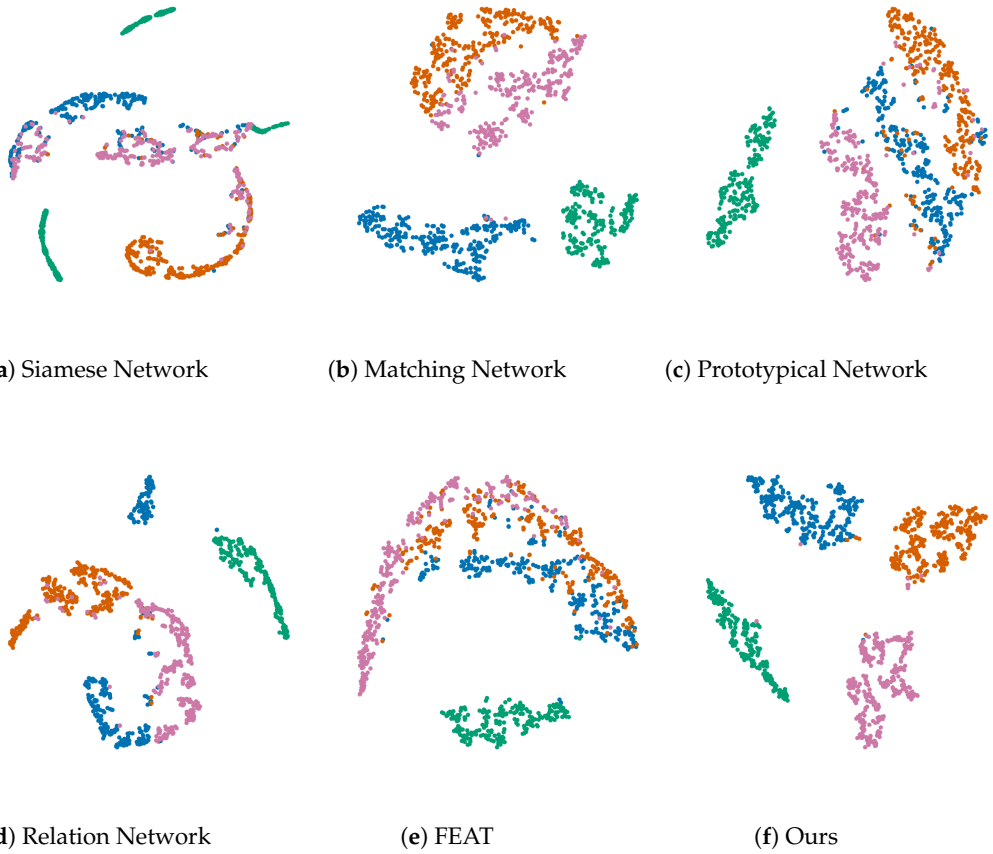
(**a**) (**b**) (**c**)

Figure 7. The average confusion matrix of our proposed method: (**a**) MSTAR "4-way 5-shot" condition, (**b**) MSTAR "4-way 1-shot" condition, (**c**) VA "4-way 1-shot" condition.

In Figure 8, the t-SNE algorithm is adopted to illustrate the output features extracted by different methods. Figure 8a–e shows that samples from different categories have quite similar distances, and the samples from the same category are not aggregated very well, which indicates that the SAR images have large inner-class variety and high cross-class similarity. In Figure 8a,e, many samples from different classes are seriously confused, leading to poor recognition accuracy. In Figure 8f, the distribution of targets from the same class is tighter and the cross-class distance is larger, which demonstrates that our method can enhance the aggregation of the same categories and the separability of different categories.

To provide a more intuitive comparison between our method and the baseline method, Figure 9 displays the feature maps of 2S1 from different layers. The first row of Figure 9 represents the visualization of features extracted by our method and the second row of Figure 9 represents the visualization of features extracted by the baseline method. From Figure 9b–d, we can see that our method is able to obtain more information from the input image with the help of SE-Net. In addition, Figure 9e represents the output of the MsFFM. After fusing multi-scale features, the fused feature map contains more local information

than the feature shown in Figure 9c. The first row of Figure 10 displays all channels in the first and second convolution layers. Compared with Figure 10c,e, our model focuses more on the key parts of the target area and places less attention on the background area.



(**a**) Siamese Network (**b**) Matching Network (**c**) Prototypical Network

(**d**) Relation Network (**e**) FEAT (**f**) Ours

Figure 8. The visualization results of different methods on the MSTAR dataset under the "4-way 5-shot" condition, where (**a**–**e**) denote the different comparison methods and (**f**) denotes our proposed method.

Table 5. "4-way 5-shot" recognition performance of different methods on the MSTAR dataset.

Methods	Min Accuracy (%)	Max Accuracy (%)	Average Accuracy (%)
Siamese	63.42	83.48	72.49 ± 5.46
FEAT	70.50	89.97	81.23 ± 4.08
Prototypical	75.42	92.14	87.80 ± 4.35
Matching	80.53	94.11	91.58 ± 3.01
Relation	68.93	94.30	80.85 ± 5.84
Ours	**93.31**	**99.21**	**95.79 ± 1.27**

The bold row represents the method with the highest average recognition accuracy.

4.4.2. "4-Way 1-Shot" Experiment on the MSTAR Dataset

From Table 6, it can be seen that the accuracy of our method under the "4-way 1-shot" condition achieves 92.67%, which is inferior to the "4-way 5-shot" condition by only 3.24%. Our proposed method achieves the best performance in terms of accuracy compared with the other methods, which demonstrates that our method can effectively recognize targets from novel classes with very limited training samples. Distance-based similarity metric functions with fixed parameters are adopted by the Matching Network, Siamese Network, Prototypical Network, and FEAT for classification. Therefore, when the query samples have high cross-class similarity, the recognition accuracy of these methods is inferior to our method. Compared with other methods, we designed the HcFIM to utilize

the complementary advantages of CNN features and traditional hand-crafted features, allowing the feature representations can be enhanced.

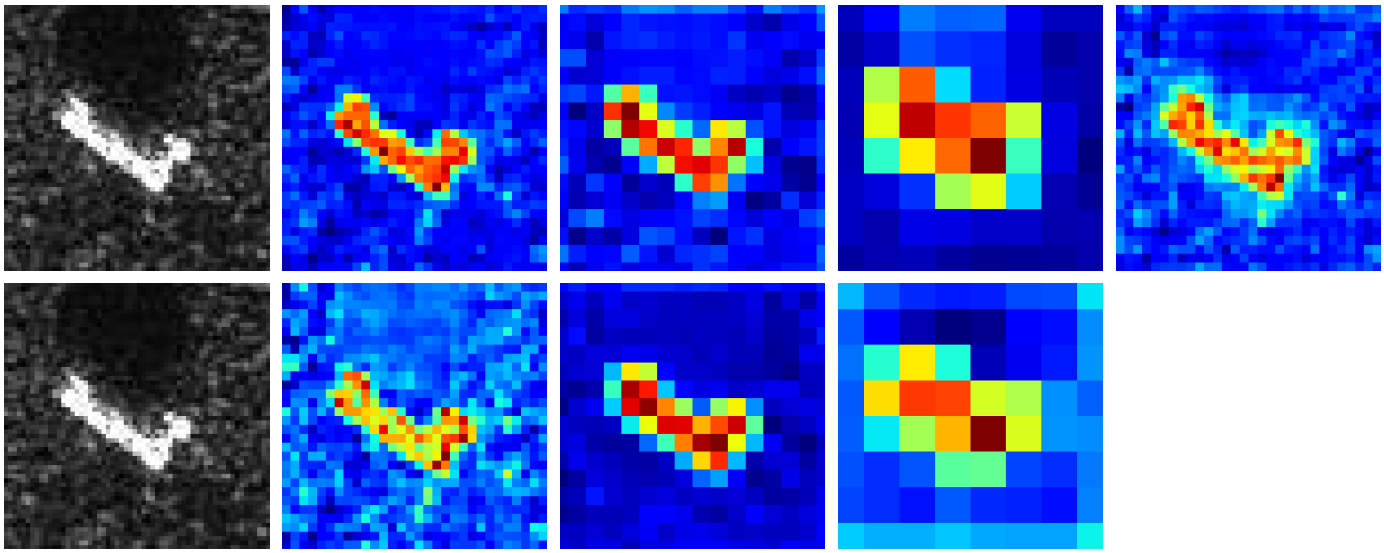


Figure 9. The comparison of feature maps between our model and the baseline model. The first row shows the feature maps of our method and the second row the feature maps of the baseline method: (**a**) the input image of 2S1, (**b**) the feature maps on the first convolution layer, (**c**) the feature maps on the second convolution layer, (**d**) the feature maps on the third convolution layer, (**e**) the feature map after multi-scale feature fusion.

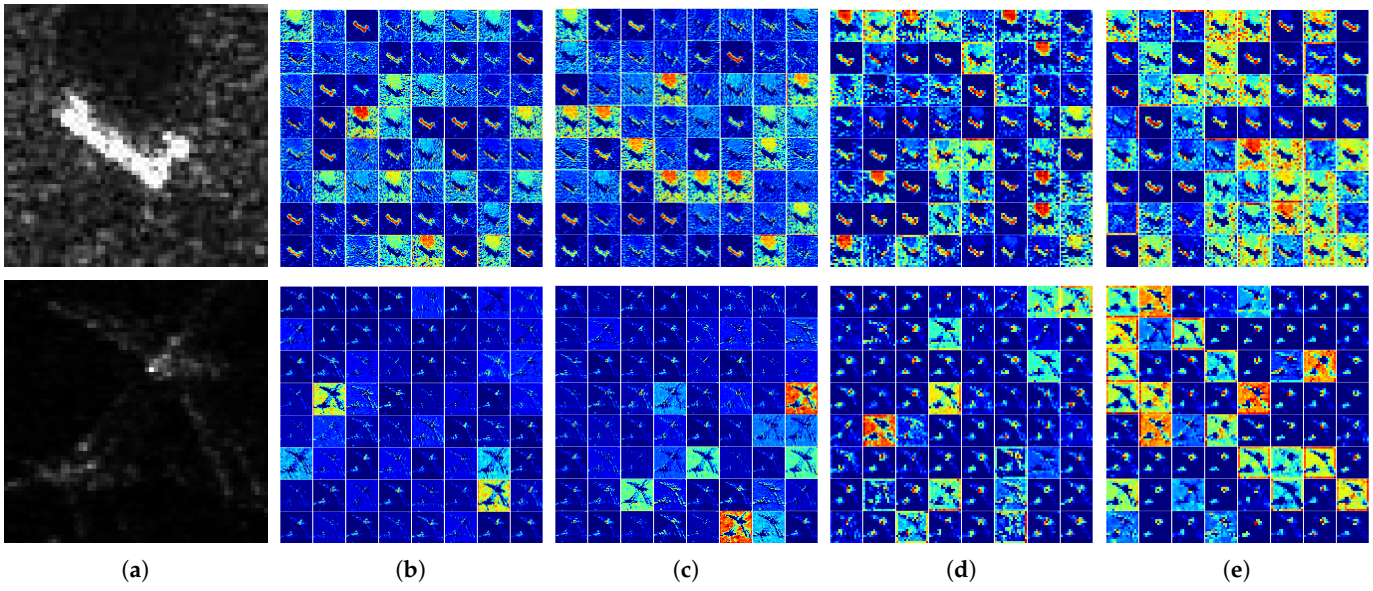


Figure 10. Feature map comparison between baseline and our proposed model, illustrating: (**a**) the input image, (**b**) the feature maps from the first convolutional layer of our proposed model, (**c**) the feature maps from the first convolutional layer of the baseline model, (**d**) the feature maps from the second convolutional layer of our proposed model, (**e**) the feature maps from the first convolutional layer of the baseline model.

The average confusion matrix of the MSTAR dataset "4-way 1-shot" condition is shown in Figure 7b. The main difference between Figure 7a,b is the recognition performance for ZSU234 targets. It can be seen that under the "5-shot" condition, the recognition accuracy of

our method for ZSU234 is 95.66%, while under the "1-shot" condition this indicator drops to 80.89%. Under the "1-shot" condition, there are more ZSU234 targets wrongly classified as 2S1 targets. The reason is that both targets are self-propelled guns with relatively similar features. As there was only one image of each class in the support set under the "1-shot" condition, the features extracted were not sufficient, leading to confusion between these two targets.

4.4.3. "4-Way 1-Shot" Experiment on the VA Dataset

Table 7 displays the results of the "4-way 1-shot" experiments on the VA dataset. Our proposed method obtains the best performance in terms of average accuracy. Our method outperforms other comparison methods on both SAR datasets, demonstrating that our method has better generalization ability. In addition, the training set of the VA dataset contains fewer images than that of the MSTAR dataset, which means that there is less prior knowledge that can be transferred from the base categories to the novel categories. Under this circumstance, our method shows better feature extraction ability with the help of HcFIM. By introducing hand-crafted features, our method can obtain more vital information from the training process. From Figure 7c, the bus and the fire truck are prone to being confused, mainly because both are large vehicles with similar features. The airliner is the only aircraft target in the test set, and can be easily distinguished from other ground targetsl thus, the airliner has the highest recognition accuracy among all types of targets in the test set of the VA dataset.

Table 6. "4-way 1-shot" recognition performance of different methods on the MSTAR dataset.

Methods	Min Accuracy (%)	Max Accuracy (%)	Average Accuracy (%)
Siamese	57.72	77.09	68.07 ± 5.12
FEAT	70.50	89.97	71.04 ± 8.23
Prototypical	56.44	91.35	74.65 ± 8.43
Matching	60.87	86.43	79.59 ± 7.25
Relation	55.95	83.38	69.19 ± 8.84
Ours	**85.94**	**96.85**	**91.76 ± 3.40**

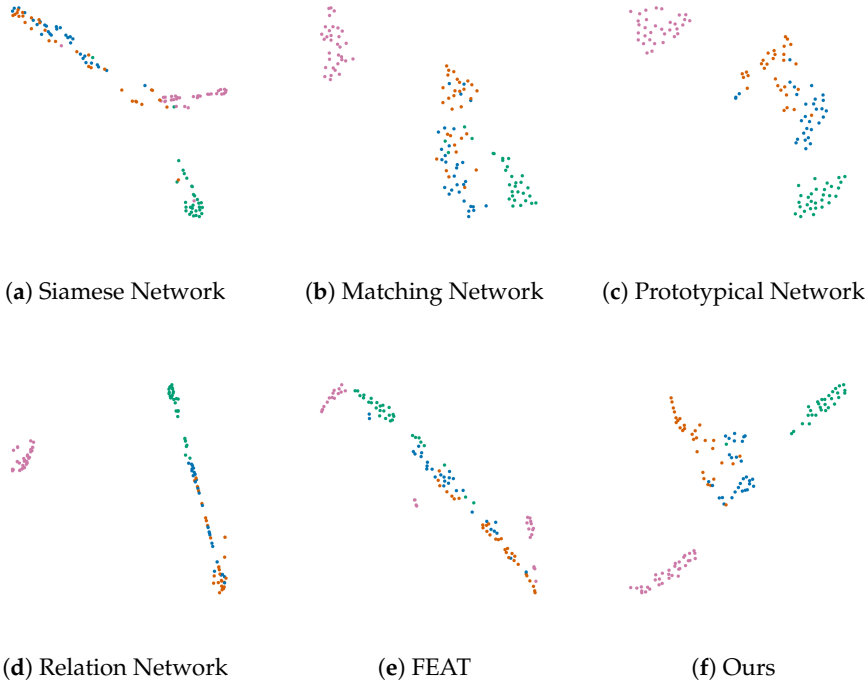
The bold row represents the method with the highest average recognition accuracy.

Table 7. "4-way 1-shot" recognition performance of different methods on the VA dataset.

Methods	Min Accuracy (%)	Max Accuracy (%)	Average Accuracy (%)
FEAT	58.74	82.93	73.03 ± 6.36
Siamese	53.15	76.42	69.94 ± 5.86
Prototypical	68.53	82.12	78.07 ± 4.14
Matching	60.84	84.56	73.46 ± 6.24
Relation	58.04	87.80	77.33 ± 6.77
Ours	**76.92**	**90.21**	**85.28 ± 3.82**

The bold row represents the method with the highest average recognition accuracy.

Likewise, the t-SNE algorithm is applied to illustrate the features extracted by different methods. As shown in Figure 11f, our method has larger inter-class distances and smaller intra-class distances than the comparison methods shown in Figure 11a–e, meaning that our method can properly identify more query images than the comparison methods. In addition, the second row of Figure 10 shows all the channel information of a plane target from the VA dataset. By comparing the second row of Figure 10b,d with that of Figure 10a,c, we can see the attention on the background noise degrade. With the addition of SE-Net, our method can focus more on the vital areas of the plane, e.g., the head of the plane, the empennage, and the wings.



(**a**) Siamese Network (**b**) Matching Network (**c**) Prototypical Network

(**d**) Relation Network (**e**) FEAT (**f**) Ours

Figure 11. The visualization result of different methods on the VA dataset under the "4-way 1-shot" condition, where (**a**–**e**) denote different comparison methods and (**f**) denotes our proposed method.

5. Discussion

5.1. Ablation Study

On the basis of the prototypical network, our proposed method employs four modules to modify the network structure and classification strategy. First, we embed the SE-Net attention mechanism between the layers of the CNN to refine the channel-wise information. Then, the MsFFM is used fuse the multi-scale features extracted by the CNN's different layers. We introduce the HcFIM to combine the traditional hand-crafted features and CNN features to enhance the feature representations for classification. In addition, we design the WDC strategy to generate appropriate weights, which represent the correlation of query images and each class. The weighted Euclidean distance is obtained by multiplying the weights with the corresponding Euclidean distance. Finally, the joint loss function of our method comprises the classification and the weight-generation losses. In order to analyze the validity of each module, ablation experiments on the MSTAR dataset wewre performed 20 times under the "4-way 5-shot" setting.

In Figure 12, the prototypical network from [31] is the "baseline" method. It can be seen that all four modules have better recognition performance than the "baseline" method in terms of the average accuracy, which demonstrates the effectiveness of our modules. The WDC with the weight-generation loss function is the most effective module of our method on the MSTAR dataset. The WDC generates weights for query images and distributes weights to the Euclidean distance, allowing the separability between different classes in the embedding space to be increased. In the training process, the weight-generation loss of our new loss function can effectively guide the WDC module to increase the inter-class Euclidean distance. With the help of WDC and weight-generation loss, the recognition accuracy outperforms the baseline model by 8.11%. Because the SE-Net can distribute weights for different channels of feature maps, the CNN can place more focus on the informative channels and suppress the unnecessary ones. The MsFFM successively fuses the features from different layers of the CNN, which can aggregate information from different scales of features and decrease the feature loss risk. By fusing modern CNN features and traditional hand-crafted features, the HcFIM can improve the

feature extraction capability for the CNN under limited training samples, consequently enhancing the robustness and representation of features. By combining all modules, the feature extraction ability can be improved and the separability between different categories can be increased, leading to better performance.

To show the effectiveness of each module on different datasets, we performed an ablation study on the VA dataset. The orange columns in Figure 12 show the average results of adding different modules to the baseline method. Similar to the results on the MSTAR dataset, MsFFM, SE-net, and 'WDC + new loss' are the most effective modules. Among these three modules, the most useful is the MsFFM, which can improve the accuracy of the baseline model by 4.87%. The remaining two modules have a similar effect on the increase in recognition accuracy at approximately 4.2%. Even though the HcFIM is not as valid as other modules, the average accuracy of the HcFIM is nonetheless 2.24% better than the baseline methods. The average accuracy can be further improved by 84.09% when combining all modules together.

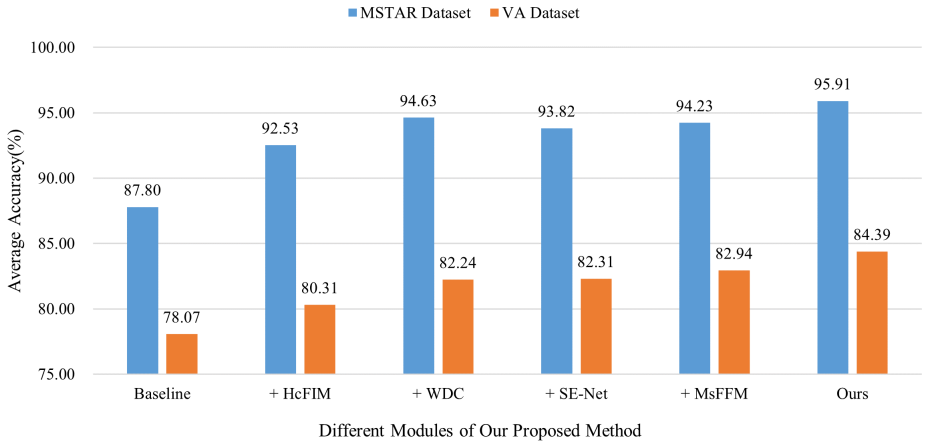


Figure 12. Results of the ablation study on both datasets.

With the aim of further analyzing the effect of each module, we further refined the ablation experiments by overlaying different modules onto the baseline method one by one. The experimental results are shown in Table 8. It can be seen that by adding each module to the baseline method in turn, the accuracy of the recognition gradually increases, and the proposed method achieves the best performance when all of the modules are added to the baseline method.

Table 8. The results of further ablation study on both datasets.

Baseline	MsFFM	HCFIM	WDC + New Loss	SE-Net	Average Accuracy (%)	
					MSTAR Dataset	VA Dataset
✓					87.80	78.07
	✓				94.23	82.94
	✓	✓			94.35	83.00
	✓	✓	✓		95.18	84.23
	✓	✓	✓	✓	**95.79**	**84.39**

The bold accuracy denotes the method with the best recognition performance.

5.2. Influence of Different Concatenation Methods

We adopt the weighted concatenation method proposed by Zhang et al. in [48] to fuse CNN features and traditional hand-crafted features. As shown in Figure 5, the traditional hand-crafted features are embedded by fully-connected layers, then two learnable parameters are used to reveal the weight of different features.

We compare different feature fusion methods for two types of features in Table 9. The prototypical network is the baseline method that only utilizes CNN features to recognize the SAR target. The 'Concatenation' method denotes that two types of features are fused by concatenation, the 'Learnable Coefficients' method stands for the learnable weight coefficients during training process, and the 'Feature Embedding' method represents the embedding process of traditional hand-crafted features implemented by fully-connected layers.

Table 9. The results of different concatenating schemes for hand-crafted features and CNN features.

Baseline	Concatenation	Learnable Coefficients	Feature Embedding	Average Accuracy (%)
✓				87.80
	✓			91.48
	✓	✓		92.06
	✓		✓	91.86
	✓	✓	✓	**92.53**

The bold accuracy denots the method with the best recognition performance.

As shown by the average accuracy in Table 9, the hand-crafted features can improve the recognition accuracy from 87.80% at baseline to 91.48% with the 'Concatenation' method, indicating that the insertion of hand-crafted features can enhance feature representation. In addition, using the learnable weight coefficients during the concatenation process can reveal the importance of different features, bringing the recognition accuracy up to 92.06%. Aligning the dimensions of the hand-crafted features and CNN features through the 'Feature Embedding' method can avoid overfitting caused by dimension imbalance between the two types of features, increasing the recognition accuracy by 0.38%. Finally, the recognition accuracy of our proposed method reaches 92.53% after merging all three approaches.

5.3. Influence of Different Hand-Crafted Features

Before CNN entered wide used, many hand-crafted features were utilized in SAR image recognition tasks, such as the hu moment feature [42], Gabor feature [43], local binary patterns (LBP) [44], histogram of oriented gradients (HOG), principal component analysis (PCA) [45], etc.

To verify whether inserting any one type of hand-crafted feature can be helpful, we performed "4-way 5-shot" experiments on the MSTAR dataset by fusing the above features with CNN features. From the results shown in Figure 13, it can be seen that the average recognition accuracy is increased when inserting different hand-crafted features, with the LBP and HOG features being the most effective. The reason for this is that both the LBP and HOG features are local features that highlight the edge and shape information of targets, while the features extracted via CNN represent the global information. Combining these two complementary features can enhance the discriminative ability of feature representation, which leads to better recognition accuracy.

5.4. The Hyperparameter in the Loss Function

We combined the classification loss and the weight-generation loss to form the joint loss function of our method. To be specific, the classification loss $\mathcal{L}_c$ uses the cross-entropy loss function, from which the gap between the prediction results and the truth value can be measured. During the process of minimizing the classification loss, the feature extraction ability of the MFFN can be enhanced. The weight-generation loss is designed to supervise the training process of the weight generation module in the WDC, from which the category-specific weights for query images can be obtained. According to the form of Equation (13), we combine these two losses by adding them and use the hyperparameter λ to adjust the proportion of the two parts in the loss function.

In order to obtain an appropriate λ on the MSTAR and VA datasets, we set λ as 0, 0.1, 1, 10, 100, and 1000, respectively. Then, "4-way 5-shot" experiments were performed

on the MSTAR dataset and "4-way 1-shot" experiments on the VA dataset. With each λ, we separately conducted 20 experiments on both datasets. We determined the recognition performance of our proposed method on both datasets with different λ, as shown in Figure 14.

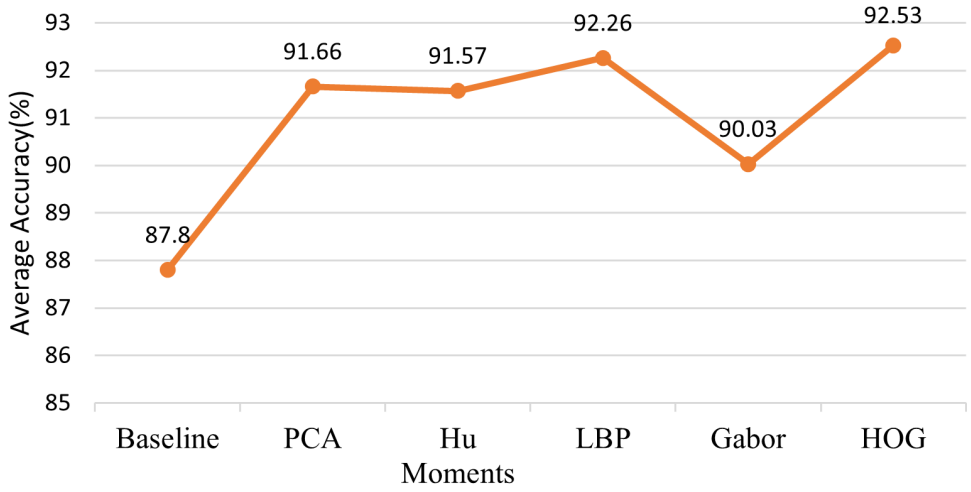


Figure 13. The results of inserting different hand-crafted features.

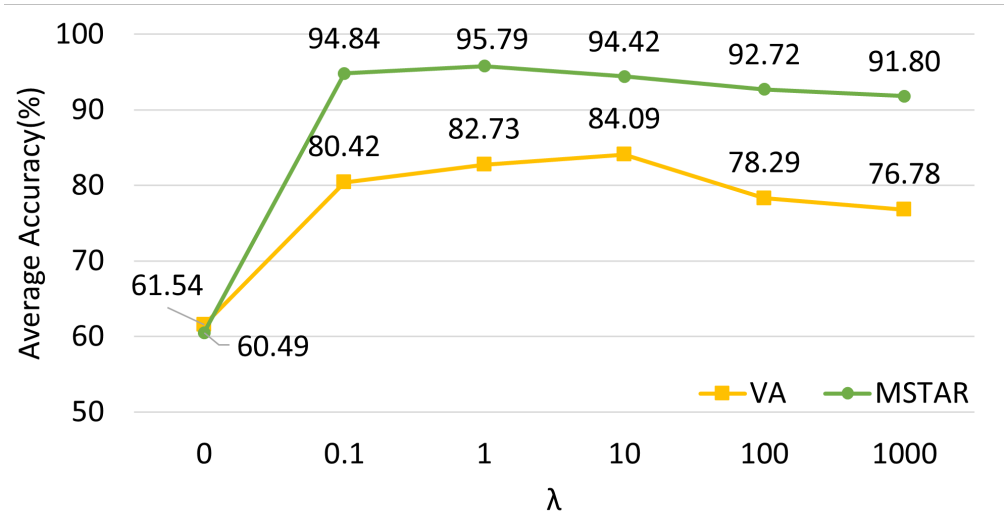


Figure 14. The accuracy of our method under different λ on VA and the MSTAR datasets.

With λ set as 0, the accuracy of our method is 61.54% and 60.49%, respectively, on the two datasets. Without the weight-generation loss, the weight generation module cannot be effectively supervised, and thus the weights could not represent the similarity between query samples and prototypes. In this case, multiplying weights and Euclidean distances cannot improve class separation, resulting in lower accuracy. In contrast, the recognition accuracy increases by approximately 20% and 30% on both datasets when λ is set as 0.1, which demonstrates the effectiveness of the weight-generation loss.

The optimal values of λ on the MSTAR and VA datasets is 1 and 10, respectively. This is because the MSTAR dataset and the VA dataset are captured by imaging radar with different work bands and polarization. The image data in these two datasets have significant differences, which leads to different optimal values of λ. As the VA dataset is much smaller than the MSTAR dataset, the number of images used for training on the VA dataset is only about 5% of the MSTAR dataset. Coupled with the fact that the support set of the VA dataset only contains one image for reference, the recognition performance

improvement generated by the WDC and the weight-generation loss on the VA dataset is slightly lower than on the MSTAR dataset. Similarly, for both datasets, the recognition accuracy first increases and then gradually decreases as the value of λ increases.

The reasons for this trend are as follows: The role of weight-generation loss $\mathcal{L}_w$ is to guide the weight generation module, meaning that the weights can increase the separability of different classes by "pushing" query samples from the prototypes of other classes. With higher λ, $\mathcal{L}_w$ occupies a higher proportion of the loss function, and the model becomes biased towards optimizing the weight generation module rather than optimizing the MFFN and weight generation module together, which decreases the model's feature extraction capability and results in a decrease in recognition accuracy.

5.5. Computational Efficiency

In order to discuss the computational efficiency of different few-shot learning methods, we calculated the FLOPs, parameters, and running time during the testing process. All algorithms were implemented by PyTorch 1.8.0 and run on an Nvidia GeForce RTX 2080Ti GPU with 11GB of memory.

Figure 15 compares the number of parameters and FLOPs of all methods. As shown in the figure, the Siamese network has the largest FLOPs and number of parameters due to the dual share-weighted CNN structure, which displays the complexity of this method. The FLOPs of our method, the prototypical network, and FEAT are less than those of the matching network and relation network. The reason for this is that the former methods calculate the prototype of each class in order to avoid comparing the similarity of samples one by one. Our methods have more parameters than any of the other methods except for the Siamese network. This is because the weight generation module in the WDC is a neural network, which comprises many parameters. In addition, the MsFFM and HcFIM contain layers with a large number of parameters, including deconvolution layers and fully-connected layers.

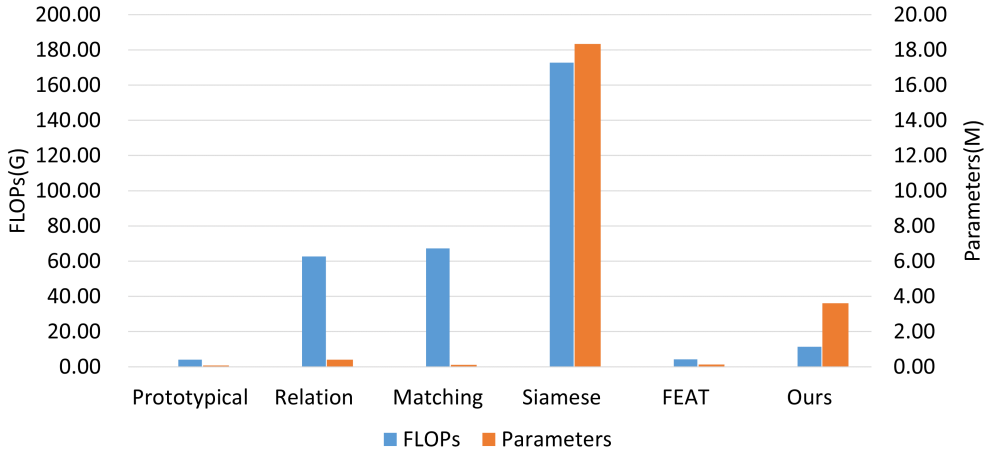


Figure 15. Comparison of the number of parameters and FLOPs between different methods.

The test times of the different methods with the "4-way 1-shot" settings on the MSTAR dataset are shown in Figure 16. It can be seen from Figure 16 that the test time of the Siamese network (2.53 s) and matching network (2.33 s) are obviously longer than that of our proposed method. This is because they need to measure the similarity between the input image pairs one by one in order to judge whether or not they belong to the same class, which makes these methods less efficient. Our method performs the inner-class average operation on support feature maps to obtain the corresponding prototypes of each class, thereby expediting this process.

Figure 16 shows that the prototypical network (1.09 s) and FEAT (1.33 s) operate faster than our proposed method. The reason for this is that we use the WDC to recognize query

samples, which makes the complexity of the model slightly higher than these two methods. Despite the fact that it has higher computational complexity, our proposed method achieves higher recognition accuracy than the other methods on both the MSTAR and VA datasets.

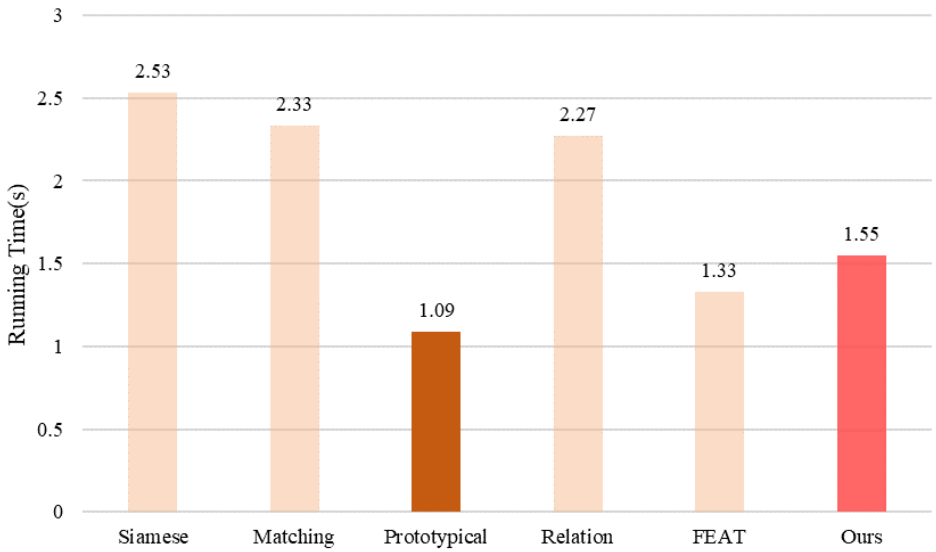


Figure 16. The running time of different methods on the MSTAR dataset.

6. Conclusions

In this article, we have proposed a novel few-shot recognition method for SAR images based on weighted distance and feature fusion. In order to aggregate targets from the same category with high diversity, we designed the MsFFM to fuse different scale features in a cascade fashion. With the help of edge and fine-grained information on targets from the lower levels of the CNN, the influence of the inner-class variety of SAR images can be decreased. To alleviate the problem of overfitting induced by the sparseness of labeled SAR images, we introduced HcFIM, which adopts a weighted concatenation approach to fuse hand-crafted features and CNN features. The HcFIM fusion process can enhance the reliability and robustness of the features used for classification. The MsFFM and HcFIM, together with a three-layer CNN backbone, form the MFFN. The MFFN is used as the feature extractor to map query and support images into feature vectors, from which prototypes can be obtained for each class. A WDC is used to avoid inter-class confusion caused by large cross-class similarities in the classification stage. The WDC first uses the concatenation features of query vectors and each prototype to generate appropriate category-specific weights, then multiplies the weights by the Euclidean distance from query vectors to the prototype of the corresponding class. In addition to the cross-entropy loss, we use the weight-generation loss in the joint loss function for our proposed method, with the aim of maximizing the gap between different classes by guiding the weight generation module. Comparative experiments on the MSTAR and VA datasets demonstrate that the proposed method is better able to deal with the high inter-class similarity problem in SAR images than other approaches. In addition, the experimental results highlight that our proposed method is superior to other comparative few-shot learning methods in terms of multiple metrics.

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

Funding: This research was funded in part by the National Natural Science Foundation of China under Grant 61771027. The work of Amir Hussain was supported by the UK Engineering and Physical Sciences Research Council (EPSRC) under Grant EP/M026981/1, Grant EP/T021063/1, and Grant EP/T024917/1. The work of Huiyu Zhou was supported by the Royal Society's Newton Advanced Fellowship under Grant NA160342 and by the European Union's Horizon 2020 research and innovation program under the Marie Sklodowska Curie Grant 720325.

Acknowledgments: The Aerospace Information Research Institute, Chinese Academic of Science, provided the Vehicle and Aircraft (VA) dataset for our experiments. We would like to express our appreciation for their assistance.

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

References

1. Zheng, T.; Wang, J.; Lei, P. Deep learning based target detection method with multi-features in SAR imagery. In Proceedings of the 2019 6th Asia-Pacific Conference on Synthetic Aperture Radar (APSAR), Xiamen, China, 26–29 November 2019; pp. 1–4.
2. Wang, S.; He, Z. The fast target recognition approach based on PCA features for SAR images. *J. Natl. Univ. Def. Technol.* **2008**, *30*, 136–140.
3. Ding, B.; Wen, G. Target reconstruction based on 3-D scattering center model for robust SAR ATR. *IEEE Trans. Geosci. Remote Sens.* **2018**, *56*, 3772–3785. [CrossRef]
4. Gao, F.; Huang, T.; Wang, J.; Sun, J.; Hussain, A.; Yang, E. Dual-branch deep convolution neural network for polarimetric SAR image classification. *Appl. Sci.* **2017**, *7*, 447. [CrossRef]
5. Dong, H.; Zhang, L.; Zou, B. Densely connected convolutional neural network based polarimetric SAR image classification. In Proceedings of the IGARSS 2019–2019 IEEE International Geoscience and Remote Sensing Symposium, Yokohama, Japan, 28 July–2 August 2019; pp. 3764–3767.
6. Gao, F.; Huang, T.; Sun, J.; Wang, J.; Hussain, A.; Yang, E. A new algorithm for SAR image target recognition based on an improved deep convolutional neural network. *Cogn. Comput.* **2019**, *11*, 809–824. [CrossRef]
7. Ai, J.; Mao, Y.; Luo, Q.; Jia, L.; Xing, M. SAR target classification using the multikernel-size feature fusion-based convolutional neural network. *IEEE Trans. Geosci. Remote Sens.* **2021**, *6*, 1–13. [CrossRef]
8. An, Q.; Pan, Z.; Liu, L.; You, H. DRBox-v2: An improved detector with rotatable boxes for target detection in SAR images. *IEEE Trans. Geosci. Remote Sens.* **2019**, *57*, 8333–8349. [CrossRef]
9. 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*, 1–16. [CrossRef]
10. Ma, F.; Zhang, F.; Yin, Q.; Xiang, D.; Zhou, Y. Fast SAR image segmentation with deep task-specific superpixel sampling and soft graph convolution. *IEEE Trans. Geosci. Remote Sens.* **2021**, *60*, 1–16. [CrossRef]
11. Yue, Z.; Gao, F.; Xiong, Q.; Wang, J.; Huang, T.; Yang, E.; Zhou, H. A novel semi-supervised convolutional neural network method for synthetic aperture radar image recognition. *Cogn. Comput.* **2021**, *13*, 795–806. [CrossRef]
12. Gao, F.; Yang, Y.; Wang, J.; Sun, J.; Yang, E.; Zhou, H. A deep convolutional generative adversarial networks (DCGANs)-based semi-supervised method for object recognition in synthetic aperture radar (SAR) images. *Remote Sens.* **2018**, *10*, 846. [CrossRef]
13. Ding, J.; Chen, B.; Liu, H.; Huang, M. Convolutional neural network with data augmentation for SAR target recognition. *IEEE Geosci. Remote Sens. Lett.* **2016**, *13*, 364–368. [CrossRef]
14. Malmgren-Hansen, D.; Kusk, A.; Dall, J.; Nielsen, A.A.; Engholm, R.; Skriver, H. Improving SAR automatic target recognition models with transfer learning from simulated data. *IEEE Geosci. Remote Sens. Lett.* **2017**, *14*, 1484–1488. [CrossRef]
15. Lin, Z.; Ji, K.; Kang, M.; Leng, X.; Zou, H. Deep convolutional highway unit network for SAR target classification with limited labeled training data. *IEEE Geosci. Remote Sens. Lett.* **2017**, *14*, 1091–1095. [CrossRef]
16. Zhang, F.; Liu, Y.; Zhou, Y.; Yin, Q.; Li, H.C. A lossless lightweight CNN design for SAR target recognition. *Remote Sens. Lett.* **2020**, *11*, 485–494. [CrossRef]
17. Che, J.; Wang, L.; Bai, X.; Liu, C.; Zhou, F. Spatial-Temporal Hybrid Feature Extraction Network for Few-shot Automatic Modulation Classification. *IEEE Trans. Veh. Technol.* **2022**, 1–6. [CrossRef]
18. Wang, Y.; Yao, Q.; Kwok, J.T.; Ni, L.M. Generalizing from a few examples: A survey on few-shot learning. *ACM Comput. Surv.* **2020**, *53*, 1–34. [CrossRef]
19. Chen, W.Y.; Liu, Y.C.; Kira, Z.; Wang, Y.C.F.; Huang, J.B. A closer look at few-shot classification. *arXiv* **2019**, arXiv:1904.04232.
20. Tang, J.; Zhang, F.; Zhou, Y.; Yin, Q.; Hu, W. A Fast Inference Networks for SAR Target Few-Shot Learning Based on Improved Siamese Networks. In Proceedings of the IGARSS 2019–2019 IEEE International Geoscience and Remote Sensing Symposium, Yokohama, Japan, 28 July–2 August 2019; pp. 1212–1215. [CrossRef]
21. Yang, R.; Xu, X.; Li, X.; Wang, L.; Pu, F. Learning relation by graph neural network for SAR image few-shot learning. In Proceedings of the IGARSS 2020–2020 IEEE International Geoscience and Remote Sensing Symposium, Waikoloa, HI, USA, 26 September–2 October 2020; pp. 1743–1746.
22. Wang, S.; Wang, Y.; Liu, H.; Sun, Y. Attribute-Guided Multi-Scale Prototypical Network for Few-Shot SAR Target Classification. *IEEE J. Sel. Top. Appl. Earth Obs. Remote Sens.* **2021**, *14*, 12224–12245. [CrossRef]

23. Luo, D.; Li, L.; Mu, F.; Gao, L. Fusion of high spatial resolution optical and polarimetric SAR images for urban land cover classification. In Proceedings of the 2014 Third International Workshop on Earth Observation and Remote Sensing Applications (EORSA), Changsha, China, 11–14 June 2014; pp. 362–365. [CrossRef]
24. Hou, Y.; Xu, T.; Hu, H.; Wang, P.; Xue, H.; Bai, Y. MdpCaps-Csl for SAR Image Target Recognition with Limited Labeled Training Data. *IEEE Access* **2020**, *8*, 176217–176231. [CrossRef]
25. Krizhevsky, A.; Sutskever, I.; Hinton, G.E. Imagenet classification with deep convolutional neural networks. *Commun. ACM* **2017**, *60*, 84–90. [CrossRef]
26. Simonyan, K.; Zisserman, A. Very deep convolutional networks for large-scale image recognition. *arXiv* **2014**, arXiv:1409.1556.
27. He, K.; Zhang, X.; Ren, S.; Sun, J. Deep residual learning for image recognition. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 21–30 June 2016; pp. 770–778.
28. Hu, J.; Shen, L.; Sun, G. Squeeze-and-excitation networks. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–23 June 2018; pp. 7132–7141.
29. Koch, G.; Zemel, R.; Salakhutdinov, R. Siamese neural networks for one-shot image recognition. In Proceedings of the ICML Deep Learning Workshop, Lille, France, 6–11 July 2015; Volume 2.
30. Vinyals, O.; Blundell, C.; Lillicrap, T.; Wierstra, D. Matching networks for one shot learning. *Adv. Neural Inf. Process. Syst.* **2016**, 3637–3645.
31. Snell, J.; Swersky, K.; Zemel, R. Prototypical networks for few-shot learning. *Adv. Neural Inf. Process. Syst.* **2017**, 4078–4088.
32. Sung, F.; Yang, Y.; Zhang, L.; Xiang, T.; Torr, P.H.; Hospedales, T.M. Learning to compare: Relation network for few-shot learning. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–23 June 2018; pp. 1199–1208.
33. Ye, H.J.; Hu, H.; Zhan, D.C.; Sha, F. Few-shot learning via embedding adaptation with set-to-set functions. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Seattle, WA, USA, 13–19 June 2020; pp. 8808–8817.
34. Wang, L.; Bai, X.; Xue, R.; Zhou, F. Few-shot SAR automatic target recognition based on Conv-BiLSTM prototypical network. *Neurocomputing* **2021**, *443*, 235–246. [CrossRef]
35. Yang, M.; Bai, X.; Wang, L.; Zhou, F. Mixed loss graph attention network for few-shot SAR target classification. *IEEE Trans. Geosci. Remote Sens.* **2021**, *60*, 1–13. [CrossRef]
36. Fu, K.; Zhang, T.; Zhang, Y.; Wang, Z.; Sun, X. Few-shot SAR target classification via metalearning. *IEEE Trans. Geosci. Remote Sens.* **2021**, *60*, 1–14. [CrossRef]
37. Yu, Q.; Hu, H.; Geng, X.; Jiang, Y.; An, J. High-performance SAR automatic target recognition under limited data condition based on a deep feature fusion network. *IEEE Access* **2019**, *7*, 165646–165658. [CrossRef]
38. Zhang, J.; Xing, M.; Xie, Y. FEC: A feature fusion framework for SAR target recognition based on electromagnetic scattering features and deep CNN features. *IEEE Trans. Geosci. Remote Sens.* **2020**, *59*, 2174–2187. [CrossRef]
39. Li, Y.; Du, L.; Wei, D. Multiscale CNN based on component analysis for SAR ATR. *IEEE Trans. Geosci. Remote Sens.* **2021**, *60*, 1–12. [CrossRef]
40. Lin, T.Y.; Dollár, P.; Girshick, R.; He, K.; Hariharan, B.; Belongie, S. Feature pyramid networks for object detection. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July 2017; pp. 2117–2125.
41. He, K.; Zhang, X.; Ren, S.; Sun, J. Spatial pyramid pooling in deep convolutional networks for visual recognition. *IEEE Trans. Pattern Anal. Mach. Intell.* **2015**, *37*, 1904–1916. [CrossRef]
42. Hu, M.K. Visual pattern recognition by moment invariants. *IRE Trans. Inf. Theory* **1962**, *8*, 179–187.
43. Lee, T.S. Image representation using 2D Gabor wavelets. *IEEE Trans. Pattern Anal. Mach. Intell.* **1996**, *18*, 959–971.
44. Ahonen, T.; Hadid, A.; Pietikainen, M. Face description with local binary patterns: Application to face recognition. *IEEE Trans. Pattern Anal. Mach. Intell.* **2006**, *28*, 2037–2041. [CrossRef] [PubMed]
45. Mishra, A.K. Validation of pca and lda for sar atr. In Proceedings of the TENCON 2008–2008 IEEE Region 10 Conference, Hyderabad, India, 19–21 November 2008; pp. 1–6.
46. Huang, L.; Liu, B.; Li, B.; Guo, W.; Yu, W.; Zhang, Z.; Yu, W. OpenSARShip: A dataset dedicated to Sentinel-1 ship interpretation. *IEEE J. Sel. Top. Appl. Earth Obs. Remote Sens.* **2017**, *11*, 195–208. [CrossRef]
47. Wang, C.; Shi, J.; Zhou, Y.; Yang, X.; Zhou, Z.; Wei, S.; Zhang, X. Semisupervised learning-based SAR ATR via self-consistent augmentation. *IEEE Trans. Geosci. Remote Sens.* **2020**, *59*, 4862–4873. [CrossRef]
48. 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]
49. Dalal, N.; Triggs, B. Histograms of oriented gradients for human detection. In Proceedings of the 2005 IEEE Computer Society Conference On Computer Vision and Pattern Recognition (CVPR'05), San Diego, CA, USA, 20–25 June 2005; Volume 1, pp. 886–893.
50. Chen, P.Y.; Huang, C.C.; Lien, C.Y.; Tsai, Y.H. An efficient hardware implementation of HOG feature extraction for human detection. *IEEE Trans. Intell. Transp. Syst.* **2013**, *15*, 656–662. [CrossRef]
51. Song, S.; Xu, B.; Yang, J. SAR target recognition via supervised discriminative dictionary learning and sparse representation of the SAR-HOG feature. *Remote Sens.* **2016**, *8*, 683. [CrossRef]
52. 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]