# Attribute-Guided Multi-Scale Prototypical Network for Few-Shot SAR Target Classification

Siyuan Wang, Yinghua Wang, *Member, IEEE*, Hongwei Liu, *Member, IEEE*, and Yuanshuang Sun

***Abstract*—Few-shot synthetic aperture radar (SAR) target classification has received more and more attention in recent years, where most of the existing methods have applied off-the-shelf networks designed for natural images to SAR images, ignoring the special characteristics of SAR data. Therefore, in this article, we propose an attribute-guided multi-scale prototypical network (AG-MsPN) combined with subband decomposition for few-shot SAR target classification, aiming to learn more discriminative features from a few labeled data. Since the SAR images are essentially complex-valued images containing both amplitude and phase information, we implement the subband decomposition of complex-valued SAR images to explore the backscattering variations of targets, thus obtaining more complete descriptions of targets. Then, considering the complementary features extracted by different convolutional layers, based on the prototypical network, a multi-scale prototypical network (MsPN) is proposed to fuse the features of different layers to enhance the discrimination of feature representations, thus relieving the problem of high intra-class diversity and inter-class similarity for the images of SAR targets. Besides, we devise the prior binary attributes of SAR targets and add an extra attribute classification module (ACM) into the MsPN to map the images into the attribute space for classification. During the training phase, the proposed MsPN and the ACM are jointly utilized to realize the target classification in both the feature space and the attribute space, and meanwhile, the model parameters are optimized by the joint loss. Thus, the classification performance of the MsPN is further enhanced under the joint supervision of class label information from a few labeled data and the target attribute information from the prior knowledge. Therefore, we name the proposed method the AG-MsPN. We demonstrate the effectiveness of our proposed AG-MsPN on the Moving and Stationary Target Acquisition and Recognition benchmark dataset, and our method surpasses many other existing methods in the few-shot cases.**

***Index Terms*—Attribute classification, few-shot, synthetic aperture radar (SAR), target classification.**

## I. Introduction

SYNTHETIC aperture radar (SAR) has become an important ground observation device due to its advantage of working in all-day and all-weather conditions. SAR is widely used in target classification and plays an important role in military and homeland security. With the development of convolutional neural networks (CNNs), the CNN-based SAR target classification methods [1]–[6] have achieved remarkable performance. However, most of the existing methods require a substantial amount of training data, and preparing such high-quality labeled SAR data is labor-intensive and impractical. Thus, the research on SAR target classification with a few labeled data has been a challenging but active topic.

Since the labeled SAR data are hard to acquire in practice, the public Moving and Stationary Target Acquisition and Recognition (MSTAR) benchmark dataset [9] (there are ten classes of targets and no more than 300 training images per class) is widely used in SAR target classification research. Compared with the large-scale optical image datasets (e.g., ImageNet [10] with 21 841 classes and more than 14 million images in total), the number of training images in the MSTAR dataset is quite limited. Directly training the deep CNN with limited labeled SAR data may cause overfitting. To address the problem of SAR target classification with limited labeled data, data augmentation [2], [11]–[14] (e.g., translation, rotation, simulated data generation, etc.) and reduction of network parameters [2], [15], [16] are commonly considered. For example, Ding *et al.* [11] adopted various data augmentation methods, including random translation, pose synthesis, etc., to get more samples for training. In [2], a new all-convolutional network (A-ConvNet) is proposed to reduce the free parameters to avoid the overfitting. Besides, robust feature extraction and selection [17]–[20] are also beneficial. For example, an effective feature extraction approach based on a visual saliency model [18] was proposed to extract more effective discriminative features. In recent years, transfer learning methods [21]–[25] have been increasingly applied in SAR target classification with limited labeled data. The core of the transfer learning methods is to transfer the rich knowledge obtained from the source domain to the target domain. Sun *et al.* [25] designed the angular rotation generative network (ARGN) to transfer the shared angular-related information to the target domain to solve the SAR target classification with limited labeled data. Huang *et al.* [21] exploited the transferrable knowledge obtained from the pretrained convolutional layers to facilitate the SAR target classification with limited data.

Although the above methods have achieved excellent performance with limited labeled data, when the amount of labeled data is further decreased (e.g., only five labeled samples per class), the classification performance of these methods degrades severely. However, humans excel in rapid understanding of visual characteristics with few demonstrations. Inspired by the

Manuscript received September 9, 2021; revised October 22, 2021; accepted October 24, 2021. Date of publication November 9, 2021; date of current version December 10, 2021. This work was supported in part by the National Natural Science Foundation of China under Grant 61671354, in part by the National Science Fund for Distinguished Young Scholars of China under Grant 61525105, in part by the Shaanxi Innovation Team Project, and in part by the 111 Project. *(Corresponding author: Yinghua Wang.)*

The authors are with the National Laboratory of Radar Signal Processing, Xidian University, Xi'an 710071, China (e-mail: wangsiyuan@stu.xidian.edu.cn; yhwang@xidian.edu.cn; hwliu@xidian.edu.cn; sunsun5544@126.com).

Digital Object Identifier 10.1109/JSTARS.2021.3126688

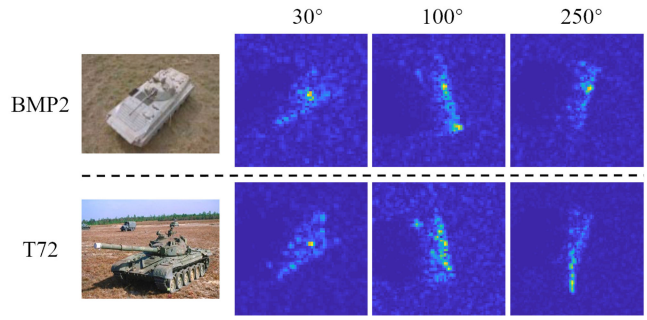


Fig. 1. SAR images of BMP2 and T72 targets with the azimuth angles of 30°, 100°, and 250°, and their corresponding optical photographs.

rapid learning ability of humans, few-shot learning (FSL) based on meta-learning [26]–[28] has attracted increasing attention, which intends to learn a classifier from a large number of training samples in the source domain and then recognizes the unseen new classes in the target domain given only a few labeled samples (usually one to ten samples per class).

In this article, we tackle the SAR target classification with extremely few labeled samples via meta-learning [27]. The FSL methods based on meta-learning have attained remarkable performance in optical image classification [26]–[32], but there are only a handful of such kind of methods in the field of SAR target classification [33]–[39]. For instance, Wang *et al.* [38] introduced the simulated SAR data into the training procedure of meta-learning, aiming to employ the transferrable knowledge learned from the simulated data to assist the classification of SAR targets in the target domain. To extract more discriminative features, Li *et al.* [36] improved the structure of relation network (RN) [31] and adopted the pretraining strategy. In the aforementioned approaches in SAR field, the researchers only slightly improve the off-the-shelf networks designed for optical images and then apply them to SAR images. The special characteristics of SAR images are not fully taken into account, which include the following three aspects.

1) *Bidimensional complex-valued images:* Due to the special active microwave imaging mechanism, the SAR images are essentially bidimensional complex-valued images that contain the amplitude information and phase information. Previous works [40]–[43] have verified that the phase information of complex-valued SAR images can provide effective target information. For instance, some methods based on time–frequency analysis (TFA) [40], [41], [44] and subband decomposition (SD) [45]–[47] are employed to analyze the physical scattering properties of SAR targets. Therefore, it is necessary to jointly use the amplitude information and phase information of the few complex-valued SAR images to fully exploit the rich target information.
2) *High intra-class diversity and large inter-class similarity:* Differing from the optical images, SAR images reveal the radar backscattering of the ground objects. The slight variation in imaging condition, as well as the configuration of targets, may result in large differences in SAR images. Fig. 1 shows the SAR images of BMP2 and T72 with different azimuth angles from the MSTAR dataset, and their corresponding optical photographs. It can be seen that the same target has various backscattering behaviors at different azimuth angles, and the different targets may also look quite similar. The high intra-class diversity and the large inter-class similarity of SAR images make the few-shot SAR target classification more challenging.
3) *Limited transferrable knowledge from the source domain to the target domain for SAR image datasets:* In few-shot SAR target classification, on the one hand, there are extremely few labeled samples in the target domain; on the other hand, compared to the optical image datasets, the amount of labeled data in the source domain of SAR image datasets is also much less, which leads to the limited transferrable knowledge from the source domain to the target domain. Due to the limited transferrable knowledge, the performance of the few-shot SAR target classification methods [33]–[39] using only the high-level visual features is still limited. For instance, with only five labeled samples per class, the prototypical network (PN) [30] achieves the classification accuracy of 99.7% on the optical Omniglot dataset [48], but only 75.12% on the MSTAR dataset, as shown in the experimental result in Section V. Using prior knowledge obtained from other sources, such as the prior attributes of targets obtained from the optical photographs and text descriptions from the Internet, to assist the meta-learning may improve the performance of few-shot classification. Therefore, we hope to combine the visual feature classification with the attribute classification during the meta-learning, to further improve the performance of few-shot SAR target classification.

Considering the three issues raised above, based on the PN [30], an attribute-guided multi-scale prototypical network (AG-MsPN) is proposed, as shown in Fig. 2. The proposed AG-MsPN consists of a multi-scale prototypical network (MsPN) and an attribute classification module (ACM). Meanwhile, the complete amplitude information and phase information contained in the complex-valued SAR images are taken into account. First of all, the azimuth and range SD [45]–[47] of the complex-valued SAR images is performed to reveal the backscattering variations of targets under different azimuth look angles and range chirp bands. To tackle the problems of high intra-class diversity and inter-class similarity within SAR images, we improve the feature extractor of the PN and propose the MsPN to fuse the output multi-scale features from different layers. Therefore, both the fine-grained local discriminative information in the low-level layers and the global semantic information in the high-level layers are considered to strengthen the feature representations of targets. Thereby, the targets of different categories with high similarity become easier to be distinguished. Finally, to compensate for the performance degradation of meta-learning under limited data of source domain, the attribute classification of targets is merged into the MsPN. Specifically, we devise the prior binary attributes for each class of targets and present an ACM to map the SAR target images into the attribute space for attribute classification. The class labels of the SAR targets are

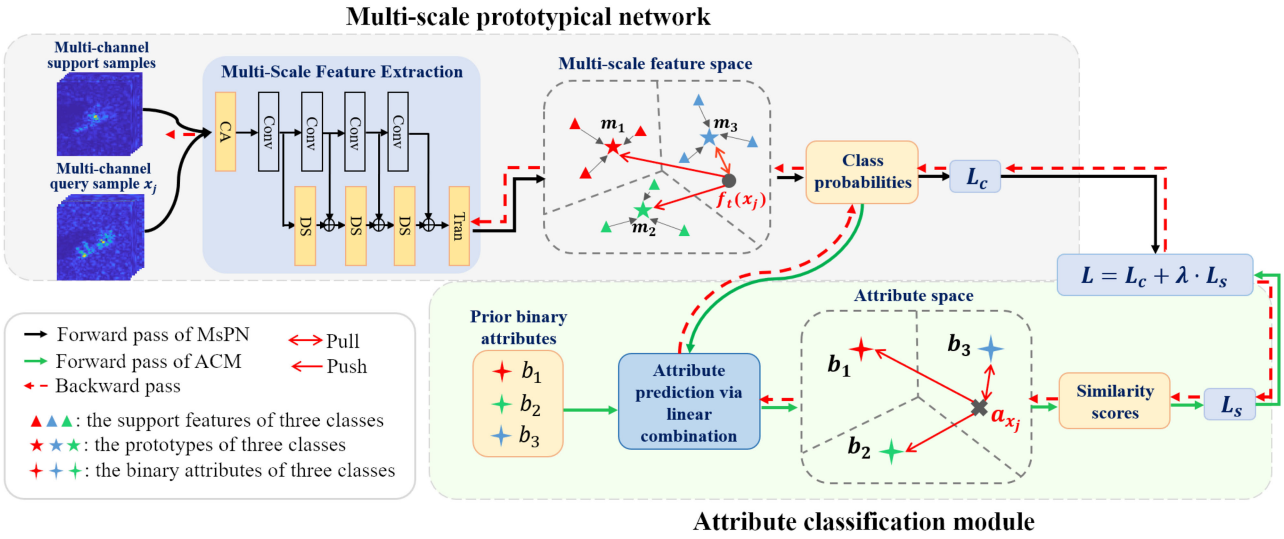


Fig. 2. Overall framework of attribute-guided multi-scale prototypical network in the case of three-class classification. MsPN denotes the multi-scale prototypical network and ACM is the attribute classification module. CA denotes the channel attention [7] module, and Conv represents the convolutional block. DS is the downsampling block, which is realized by using the same architecture as Conv. Tran is the proposed transformation block and is composed of a batch normalization [8] layer, a channel attention module, and a convolutional block. $f_t(x_j)$ denotes the multi-scale feature of query sample $x_j$ extracted by the MsPN, and $a_{x_j}$ is the attribute of $x_j$ predicted by the ACM. $L_c$ and $L_s$ are the classification loss of the MsPN and the similarity loss of the ACM, respectively. $L$ is the total loss, in which $\lambda$ is a balance parameter.

predicted by comprehensively considering the classification results of the MsPN and the ACM. Through an end-to-end training of the MsPN and the ACM, the classification performance of the MsPN can be improved under the supervision of both the class label information from a few labeled data and the prior attribute information of targets. Therefore, we name the proposed method as the AG-MsPN.

In general, the contributions of this article are as follows.

1) To address the problems of high intra-class diversity and inter-class similarity within SAR images and enhance the feature representations of targets, we improve the feature extractor of the PN and propose the MsPN to fuse the output multi-scale features from different layers. Hence, aided with the fused fine-grained local discriminative features, the targets of different classes with high similarity become easier to be distinguished.
2) To compensate for the performance degradation of meta-learning with limited data in the source domain, we devise the prior binary attributes of SAR targets in the MSTAR dataset and propose an ACM to predict the class labels of SAR targets together with the MsPN, yielding the proposed AG-MsPN. During the end-to-end optimization process of the MsPN and the ACM, the performance of the MsPN is enhanced under the supervision of both the label information and the prior attribute information of SAR targets.
3) A novel few-shot SAR target classification framework, AG-MsPN, is proposed, in which the complete amplitude information and phase information contained in the complex-valued SAR images are taken into account. We implement the range and azimuth SD of the complex-valued SAR images to explore the backscattering variations of SAR targets hidden during the imaging phase, aiming to better extract the discriminative features of SAR targets. Extensive experiments on the MSTAR dataset demonstrate that the proposed few-shot SAR target classification framework AG-MsPN outperforms many state-of-the-art methods under various settings of the target domain and the source domain.

The rest of this article is organized as follows. Section II introduces the related work. Preliminary is briefly reviewed in Section III. A detailed description of the proposed AG-MsPN is presented in Section IV, and the implementation details and experiments are provided in Section V. Finally, Section VI concludes this article.

## II. Related Work

### A. Few-Shot SAR Target Classification

In recent years, FSL has received increasing attention in the field of SAR target classification [33]–[39], which aims to learn a classifier for unseen classes with few labeled samples. Meta-Learning [27] is commonly used in FSL, which extracts the transferrable knowledge from the relevant source domain to assist the few-shot classification task in the target domain. In contrast to transfer learning, the special episodic training strategy [27] is applied in meta-learning to acquire the transferrable knowledge, allowing for more rapid adaptation of the classification model to the unseen classes.

Meta-Learning can be mainly divided into three streams: metric-based [30], [31], optimization-based [29], and memory-module-based [49], where metric-based [33], [34], [36], [37] and optimization-based [35], [38] methods are mainly considered in few-shot SAR target classification. For instance, in [35], an optimization-based meta-learning model is designed to obtain good initialization parameters that can be applied to new classes

with only a few steps of gradient descent. However, training the optimization-based meta-learning models is time-consuming. The metric-based meta-learning dedicates to learning a good metric space, in which sample features of the same category are clustered, and the class labels of the samples are determined via finding the nearest class prototype. For example, in [37], the triplet loss is applied to constrain the sample distance within the same class and among different classes to facilitate the learning of metric space. Wang *et al.* [33] merged a bidirectional recurrent neural network [50] into the PN [30] to extract target features, which are robust to the variation of poses. Tang *et al.* [34] replaced the feature extractor of the Siamese network [51] with the one in the A-ConvNet [2], in order to obtain better feature representation and improve the performance of metric-based meta-learning.

Although the aforementioned methods have improved the performance of few-shot SAR target classification, the special characteristics of SAR images are not fully taken into account and the prior auxiliary information is ignored. Thus, different from the previous works [33]–[38], a novel few-shot SAR target classification model AG-MsPN is proposed, in which the complete target information contained in the complex-valued SAR images and the prior attribute information of targets are taken into account.

### B. Applications of Complex-Valued SAR Images

Differing from optical images, SAR images are complex valued, which contain the amplitude information and phase information. Nevertheless, only a few studies [40]–[43], [52], [53] have jointly utilized the amplitude information and phase information of complex-valued SAR images to analyze SAR targets. For example, the complex-valued convolutional neural networks (CV-CNNs) [43], [52], [53] are designed to extend all the elements of the CNN to the complex domain to fully explore the complete target information contained in the complex-valued SAR images. However, training the CV-CNN still requires a large number of labeled SAR images. Besides, some methods based on TFA [40]–[42] are also presented to reveal the backscattering variations along the azimuth and range directions during the imaging phase. In [40], Spigai *et al.* proposed the TFA-based method to quickly calculate the radar spectrogram to characterize the backscattering variations of point targets during the acquisition of SAR images. Singh and Datcu [41] leveraged the TFA algorithm to analyze the backscattering properties of targets to better distinguish the targets that look quite similar. However, the TFA-based methods only analyze the physical properties of target at the pixel level, and the backscattering behaviors of the entire target are not directly depicted. Moreover, the TFA-based methods are mainly used in scene classification or the physical property analysis of targets and could not be directly applied to target recognition.

SD [45]–[47] is also utilized to fully leverage the rich target information contained in the complex-valued SAR images. For instance, in [45], the maximum-likelihood statistics based on the Wishart probability distribution is used to detect the nonstationary pixels of the subapertures, and then, the influence of azimuth backscattering variations on conventional polarimetric SAR data analysis is eliminated by using the coherent restoration. In [47], a series of subband images are generated in the azimuth and range directions, and then, the two-looks internal Hermitian product based on the nonnormalized interferometric coherence of the two sublooks is proposed to distinguish the point targets from the nearby speckle. To the best of our knowledge, the SD of complex-valued SAR data has still not been applied in the field of SAR target classification. Therefore, we implement the SD in the azimuth and range directions to explore the rich target information contained in the complex-valued SAR images, aiming to enhance the performance of few-shot SAR target classification.

Different from the previous works [45]–[47], after generating the subband images, we do not use these methods to process the subband images, like the nonnormalized interferometric coherence and the coherent restoration, but merge the amplitude images of the original SAR image with the decomposed subband images via the channel-wise concatenation. Then, considering the remarkable performance of CNN, we send the obtained multi-channel SAR images into the CNN, aiming to adaptively explore the valuable information contained in the original SAR image and the subband images.

### C. Prior Attribute Information

Attributes are high-level descriptions of objects [54], such as shape, color, or geographic information. The same attribute can be shared by multiple categories of objects. In general, the attributes of objects can be obtained from the prior knowledge, such as searching for the optical photographs and text descriptions from the website. Given a set of attributes, humans can quickly recognize the targets with the prior knowledge. For instance, the elephant can be reliably recognized when looking for the attributes of *gray animals*, *four legs*, and *long trunks*.

Benefiting from the additional target information provided by the attributes, attribute-based zero-shot learning (ZSL) [54]–[58] has attracted more and more attention, which aims to recognize the novel class never seen before without any labeled samples. In the attribute-based ZSL methods [54]–[58], an embedding function from the images to the high-level attributes is commonly learned, to facilitate the recognition of the unseen classes. For instance, Lampert *et al.* [54] first designed the direct attribute prediction model and the indirect attribute prediction model to recognize the images of unseen classes by predicting the posterior probability of each attribute. However, these two attribute prediction models are trained based on each individual attribute; thus, the correlations between attributes are ignored. To construct the relationships between different attributes, some ZSL methods [57], [58] work on learning a transformation model to map the images into a shared attribute space. For instance, in [57], the images are mapped into the shared semantic space, and then, the classification is performed by finding the nearest class semantic vector.

Inspired by the method of [57], we design the prior binary attributes of MSTAR SAR targets and present an ACM based on the MsPN, aiming to enhance the performance of few-shot SAR target classification. Compared with [57], only the way

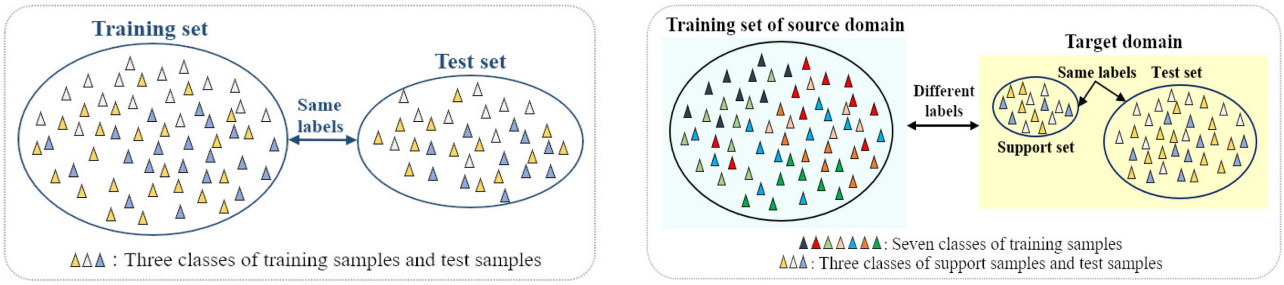


Fig. 3. Differences of datasets in traditional classification and few-shot classification. (a) Datasets in the traditional classification. (b) Datasets in the few-shot classification. The samples with the same category are represented by triangles of the same color.

of attribute prediction in our method is similar to it, while the attribute classification and the training strategy of our method are different from it, which will be depicted in Section IV-C.

## III. Preliminary

In this section, the definitions of FSL, PN, and SD are briefly introduced.

### A. Brief Introduction of FSL

FSL aims to recognize the unseen classes with only few labeled samples. Fig. 3(a) and (b) shows the datasets of traditional classification and few-shot classification, respectively. As shown in Fig. 3(a), in the traditional classification, the training set and the test set share the same label space. However, as illustrated in Fig. 3(b), there are three datasets in the few-shot classification: the training set in the source domain, the support set, and the test set in the target domain. The support set and the test set in the target domain share the same label space, which are similar to the training set and the test set in the traditional classification. However, the class labels of the training set in the source domain and data in the target domain are disjoint. Suppose that there are $N$ classes and $K$ labeled samples per class in the support set (generally $K \leq 10$); then, this few-shot classification problem is defined as an $N$-way $K$-shot task, which aims to classify the test samples given only $N \times K$ support samples.

From Fig. 3(b), we can see that, in the target domain, the labeled samples in the support set are insufficient to train a classifier. Hence, the training set of the relevant source domain is introduced to assist the classification of test samples. Since the class labels of the training set and the test set are disjoint, the classifier trained by the traditional training method on the training set cannot generalize well to the unseen test samples. Therefore, the episodic training strategy [27] is introduced into the meta-training procedure, in order to enable the classifier to learn the transferrable knowledge and obtain good generalization ability.

Fig. 4 shows the episodic training strategy in the 3-way 1-shot case. Specifically, a set of auxiliary few-shot classification tasks $\{T_1, T_2, \ldots, T_n\}$ is sampled from the training set of the source domain to imitate the 3-way 1-shot classification task in the target domain. As demonstrated in Fig. 4, in the $i$th iteration,

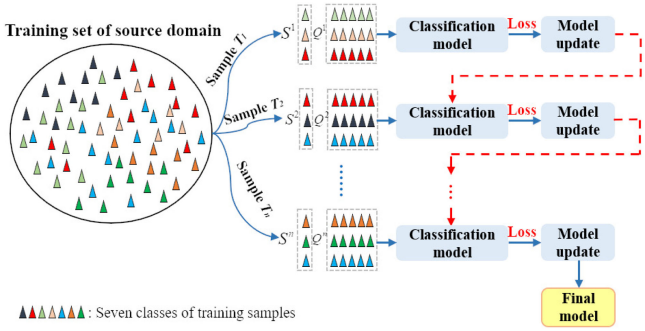


Fig. 4. Episodic training strategy of the few-shot classification. $T_i$ is the sampled few-shot classification task in the $i$th iteration and $S^i$ and $Q^i$ denote the sampled support set and query set, respectively.

the support set $S^i$ with three classes and one sample per class is first randomly selected from the training set. Then, a batch of samples (e.g., five samples per class as shown in Fig. 4) from the remaining of the selected three classes are randomly sampled to serve as the query set $Q^i$. The few-shot classification task $T_i$ is formed of the selected support set $S^i$ and query set $Q^i$, and then, the samples in $S^i$ and $Q^i$ are fed into the classification model with the aim of recognizing the query samples given the $3 \times 1$ support samples. The iterative update of the model parameters is based on each few-shot classification task.

After the episodic training, the classification model has adapted to a lot of few-shot classification tasks. Therefore, the classification model can obtain the transferrable knowledge from the training set, so that it quickly adapts to the unseen classification task in the target domain.

### B. Prototypical Network

The PN [30] is a typical metric-based meta-learning network for few-shot object classification. The feature extractor of the PN consists of four convolutional blocks, and each convolutional block is composed of a convolutional layer, a batch normalization (BN) [8] layer, a ReLU nonlinear activation function [59], and a max-pooling layer, as shown in Fig. 5. The key idea of the PN is to learn an embedding function, with which the embedded features of the samples within the same category are grouped together. With the embedded features, each class is represented by its prototype, which is defined as the average feature of this

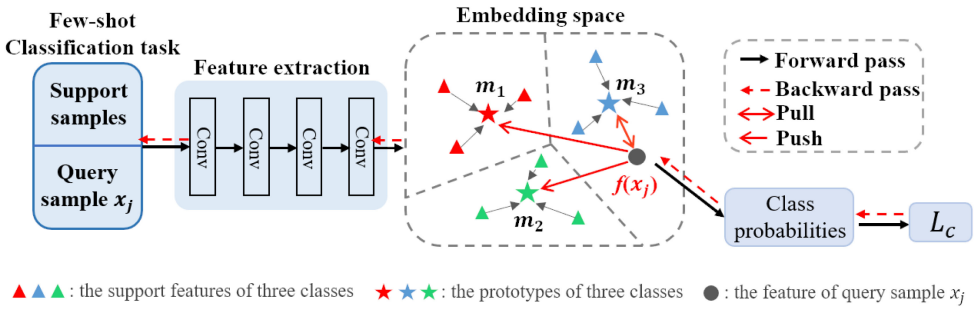


Fig. 5. Classification process of the PN. Conv denotes the convolutional block. $m_1$, $m_2$, and $m_3$ are the prototypes of three classes. $f(x_j)$ is the extracted feature of query sample $x_j$. $L_c$ is the classification loss of the PN.

class. Then, the distances from the embedded features to each prototype are calculated to determine which class the samples belong to. In view of the small number of parameters and the excellent achievements of the PN in the few-shot optical image classification, we choose the PN as the baseline to tackle the problem of few-shot SAR target classification. We simply introduce it by referring to [30] as follows.

Given a few-shot classification task sampled in the $i$th iteration, the support set and the query set are denoted as $S^i = \{(x_i, y_i), i = 1, \ldots, m\}$ and $Q^i = \{(x_j, y_j), j = 1, \ldots, n\}$, respectively, where $y_i$ and $y_j$ are the corresponding labels of samples $x_i$ and $x_j$.

As described in Fig. 5, the support samples are mapped into the embedding space. Then, the prototype of class $k$ is generated by averaging the embedded features of support samples within class $k$, which is shown as follows:

$$m_k = \frac{1}{|S_k|} \sum_{(x_i, y_i) \in S_k} f(x_i) \tag{1}$$

where $S_k$ denotes the set of support samples labeled with class $k$ and $f$ denotes the embedding function.

After calculating the prototype of each class, the query samples are classified by finding the nearest prototype in the embedding space. Given a query sample $x_j \in Q^i$, we first calculate the Euclidean distance between its embedded feature $f(x_j)$ and each prototype, to evaluate their similarities. Then, based on the softmax function, the probability that the query sample $x_j$ belongs to class $k$ is predicted as follows:

$$p(y_j = k \mid x_j) = \frac{\exp\left(-d_{x_j}^k\right)}{\sum_{k'} \exp\left(-d_{x_j}^{k'}\right)} \tag{2}$$

where $d_{x_j}^k$ denotes the Euclidean distance between the feature $f(x_j)$ and the prototype of class $k$, i.e., $m_k$.

As can be seen from (2), the larger the Euclidean distance, the smaller the probability that the query sample belongs to the class. Thus, the classification loss $L_c$ for query sample $x_j$ with true class $k$ is defined as

$$L_c = -\log p(y_j = k \mid x_j). \tag{3}$$

By minimizing the classification loss of query sample $x_j$, the embedded feature $f(x_j)$ of $x_j$ is pulled closer to the prototype of true class $k$ in the embedding space, so that the query sample $x_j$ can be correctly classified.

In the meta-training procedure of the PN, a large number of training classes and training samples from the relevant source domain are required to learn the transferrable knowledge and improve the generalization ability on the unseen test data. However, compared to the optical image datasets, such as miniImageNet [60] (64 training classes and 600 samples per class), the number of training classes and training samples in the source domain of the SAR image dataset is much fewer, leading to the poor few-shot classification performance via meta-learning. Therefore, we improve the feature extractor and the label prediction of the PN to tackle the few-shot SAR target classification under not only the extremely few labeled samples in the target domain, but also the limited training samples of the source domain.

### C. Subband Decomposition

In this article, we follow the SD method for azimuth and range directions proposed in [45]–[47], aiming to reveal the rich target information from both the amplitude and the phase of complex-valued SAR data. The SD of azimuth and range directions is introduced, respectively, as follows.

1) *Azimuth SD:* In order to get the full-resolution SAR images, many low-resolution echoes of targets received at different azimuth angles are integrated in the imaging phase. Thus, the azimuth SD [45] of complex-valued SAR images is performed to generate several subband images, which are observed under different azimuth look angles. The generation steps are as follows.
   a) *1-D Fourier transform:* First, we perform the 1-D Fourier transform in the azimuth direction on complex-valued SAR images to get the azimuth spectrum.
   b) *Azimuth weighting function estimation:* The amplitude of the azimuth spectrum is averaged along the range direction, and then, we can obtain the estimated azimuth weighting function [46].
   c) *Azimuth spectrum correction:* In order to correct the spectrum, we multiply the inverse of the estimated azimuth weighting function with the azimuth spectrum.
   d) *Azimuth spectrum division:* According to the number we need, the corrected spectrum is divided into several

equal nonoverlapping subspectrums along the azimuth direction, and the center frequency of each subspectrum is moved to zero frequency.

e) *Subspectrum weighting:* After obtaining the subspectrums, we use the estimated azimuth weighting function in step (b) to weight each subspectrum and implement the zero-padding operation around the subspectrums to restore the original size of the complex-valued SAR image.

f) *1-D inverse Fourier transform:* Finally, the subband images decomposed in the azimuth direction are produced by performing the 1-D inverse Fourier transform on each subspectrum.

2) *Range SD:* Similar to the azimuth SD, the range SD [47] produces a series of subband images under different range chirp bands. The process is as follows.

a) *1-D Fourier transform:* First, we perform the 1-D Fourier transform in the range direction on complex-valued SAR images to get the range spectrum.

b) *Range weighting function estimation:* The amplitude of the range spectrum is averaged along the azimuth direction, and then, we can obtain the estimated range weighting function.

c) *Range spectrum correction:* We multiply the inverse of the estimated range weighting function with the range spectrum, thereby getting the corrected spectrum.

d) *Range spectrum division:* According to the number we need, the corrected range spectrum is divided into several equal nonoverlapping subspectrums along the range direction, and the center frequency of each subspectrum is moved to zero frequency.

e) *Subspectrum weighting:* After obtaining the subspectrums, we use the estimated range weighting function in step (b) to weight each subspectrum and implement the zero-padding operation around the subspectrums.

f) *1-D inverse Fourier transform:* Finally, the subband images decomposed in range direction are produced by performing the 1-D inverse Fourier transform on each subspectrum.

## IV. Methodology

In this article, a novel few-shot SAR target classification network AG-MsPN is proposed, as illustrated in Fig. 2, in which the complete target information revealed via the SD is taken into account. Compared with the PN shown in Fig. 5, we first improve the feature extractor to fuse the output multi-scale features from different layers and propose the MsPN. Then, to utilize the attribute information to assist the meta-learning, the binary attribute of each type of target is designed, and an ACM is proposed and merged into the MsPN to predict the class labels of samples together with the MsPN. Next, we will give a detailed introduction about the SD, the MsPN, and the ACM of the AG-MsPN.

### A. SD of Complex-Valued SAR Data

As illustrated in Fig. 1, due to the special active microwave imaging mechanism, the high inter-class similarity and intra-class diversity in SAR amplitude images provide misleading information to the models, thus increasing the challenge of few-shot SAR target classification. Considering the fact that SAR images are essentially the bidimensional complex-valued signals, we follow the idea of [45]–[47] and use the range and azimuth SD for complex-valued SAR images, to fully exploit the target amplitude information and phase information. Thus, we can obtain richer descriptions of SAR targets so that they become easier to be discriminated.

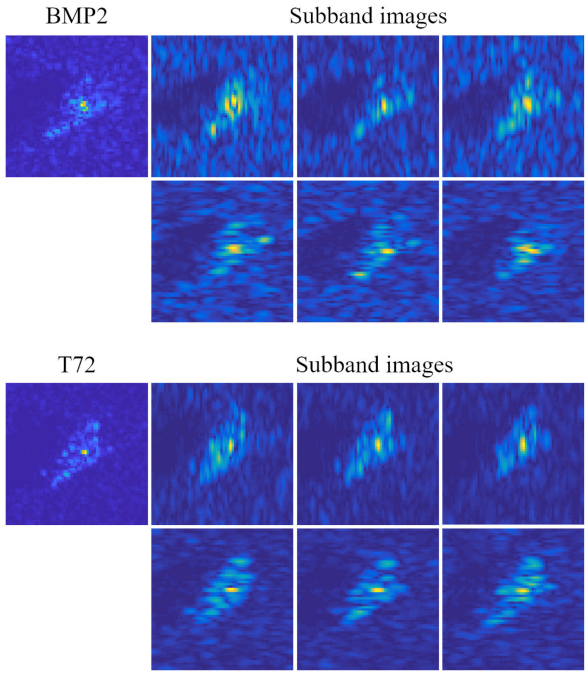


Fig. 6. SAR images and their corresponding subband images of BMP2 and T72 targets. The subband images are obtained by splitting the azimuth spectrum and range spectrum into three equal nonoverlapping parts, respectively.

Specifically, we perform the azimuth and range SD of complex-valued SAR images according to the methods described in Section III-C. First, we conduct the azimuth SD to decompose the SAR images into three subband images, which characterize the backscattering of targets under different azimuth look angles during the imaging phase. Then, considering the target responses at different range chirp bands, the range subband decomposition is also implemented to gather three subband images. Therefore, six complex-valued subband images are obtained for each complex-valued SAR image. With these subband images, the various scattering behaviors of SAR targets during the imaging process, which are hidden in the processed full-band SAR images, are discovered to express the targets more comprehensively.

Fig. 6 shows the SAR images and their corresponding subband images of BMP2 and T72 targets under different azimuth look angles and range chirp bands. It can be seen that the targets of BMP2 and T72 that look similar in original SAR images can be easily distinguished, aided with the various backscattering behaviors revealed in subband images. Thus, we can better explore the discriminative features of targets by comprehensively considering the SAR images and their subband images.


Fig. 7. Optical photographs of ten classes of SAR targets in the MSTAR dataset.

Differing from the previous works [45]–[47], in our method, the amplitude images of the original SAR images and their subband images are merged via the channel-wise concatenation to obtain the multi-channel SAR images. Then, we employ the proposed AG-MsPN to adaptively explore the valuable target information contained in the multi-channel SAR images, instead of using some methods like the interferometric coherence or coherent restoration to process the obtained subband images as in the existing works [45]–[47].

### *B. Multi-Scale Prototypical Network*

Fig. 7 shows the optical photographs of the vehicle targets in the MSTAR dataset that we are interested in. From Fig. 7, we can see that some vehicle targets have shared structures and subtle differences, e.g., the targets "T62" and"T72," or the targets "BTR60" and "BTR70." Due to the appearance similarity and the pose variations of the vehicle targets, the acquired SAR images have higher inter-class similarity and intra-class diversity than optical images. To tackle the problems of high inter-class similarity and intra-class diversity within SAR images, except for the global information, the local details of targets (e.g., the cannon of the target) should also be considered to facilitate the discrimination of different classes of targets.

Previous studies [61]–[63] have demonstrated the abundant complementary texture information between different layers of the CNN. The high-level layers of the CNN extract global information and rich semantic information about the targets, but the descriptions of local details are scarce, while the low-level ones focus more on the discriminative local details of the targets. Hence, as depicted in Fig. 2, by improving the fusion method of [64], we design an MsPN to fuse the low-level local detailed information and the high-level global information. Thus, the feature representations of SAR targets are strengthened aided with the fused local detailed features. Next, before describing the classification of the MsPN, we first provide the explanation of the multi-scale feature extraction.

*1) Multi-Scale Feature Extraction:* Fig. 8(b) describes the multi-scale feature extraction network of MsPN. Compared with the feature extraction network of the PN shown in Fig. 8(a), the channel attention (CA) [7] module, the downsampling block for multi-scale feature fusion (MsFF), and the transformation block are newly added.

*a) CA module:* As shown in Fig. 6, there are differences in the target information contained in each subband image

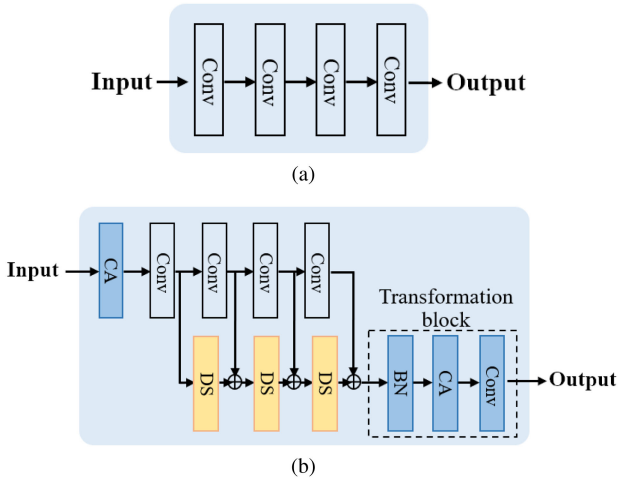


Fig. 8. Feature extractor of the PN and the proposed MsPN. (a) Original feature extractor of the PN. (b) Proposed multi-scale feature extractor of the MsPN. Conv denotes the convolutional block, which consists of a convolutional layer, a BN layer, a ReLU nonlinear activation function, and a max-pooling layer. CA represents the channel attention module, and DS is the downsampling block that is realized by using the same architecture as Conv.

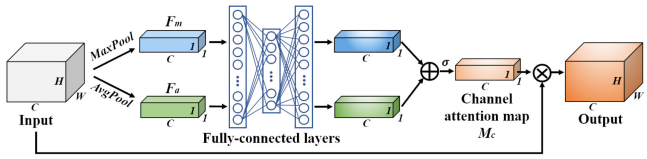


Fig. 9. Process of CA by referring to [7]. $AvgPool$ and $MaxPool$ are the average-pooling and max-pooling operations, respectively. $\sigma$ denotes the sigmoid activation function, and $\otimes$ is the element-wise multiplication.

and the original SAR image, resulting in different degrees of contribution to the classification by them. Since the inputs of the AG-MsPN are the multi-channel SAR images, the CA is first applied to selectively emphasize the informative features and suppress the redundant ones.

The process of CA is shown in Fig. 9. Given an input SAR image $x \in R^{C\times H\times W}$ with a size of $H \times W$ and $C$ channels, we conduct the average-pooling and max-pooling operations for images in each channel to generate the channel descriptors $F_a \in R^{C\times 1\times 1}$ and $F_m \in R^{C\times 1\times 1}$, respectively, to denote the whole information of each channel. Then, the channel descriptors are sent to the shared fully connected layers to learn the inter-channel relationship and generate the CA map $M_c \in R^{C\times 1\times 1}$, thereby representing "what" to emphasize in channels. In summarize, the CA map of $x$ is computed as

$$\begin{aligned} M_c(x) &= \sigma(FC(AvgPool(x)) + FC(MaxPool(x))) \\ &= \sigma\left(FC\left(F_a\right) + FC\left(F_m\right)\right) \end{aligned} \tag{4}$$

where $\sigma$ denotes the sigmoid activation function [65], $FC$ represents the fully connected layers, and $AvgPool$ and $MaxPool$ are the average-pooling and max-pooling operations, respectively.

Then, the channel-refined SAR image $x^c \in R^{C\times H\times W}$ is produced by implementing the element-wise multiplication of

sample $x$ and the CA map $M_c(x)$, as shown in the following equation:

$$x^c = M_c(x) \otimes x \tag{5}$$

where $\otimes$ represents the element-wise multiplication.

*b) Multi-scale feature fusion:* After obtaining the channel-refined SAR images, we gradually fuse the low-level local detailed features and the high-level global features of the channel-refined SAR images by using feature downsampling and concatenation as that in [64].

As shown in Fig. 8(b), the convolutional blocks hierarchically extract the features $\{F_l, l = 1, \ldots, L\}$ of the channel-refined SAR image $x^c$, where $F_l$ denotes the feature of the $l$th layer. The structure of each convolutional block [30] is the same, consisting of a convolutional layer composed of 64 $3 \times 3$ filters, a BN layer, a ReLU nonlinear activation function, and a $2 \times 2$ max-pooling layer with a stride of 2. For the fusion of the two adjacent features, we follow the same way of [64]. Specifically, we first downsample the features with larger spatial resolution and then merge the downsampled features with the features of smaller spatial resolution by channel-wise concatenation, as shown in Fig. 8(b). Based on the above feature fusion strategy, we can obtain the multi-scale feature $f(x)$ by aggregating the output features from layer 1 to layer $L$:

$$f(x) = \mathrm{DS}\left(\ldots \mathrm{DS}\left(\mathrm{DS}\left(F_1\right) \oplus F_2\right) \oplus F_3 \ldots\right) \oplus F_L \tag{6}$$

where DS represents the downsampling block and $\oplus$ is the operation of concatenating features along the channel dimension.

*c) Transformation block:* For the fused multi-scale feature $f(x)$ obtained above, the features from each layer (each $F_l$ has 64 feature maps) are preserved; meanwhile, some redundant information might be also preserved and is not beneficial for classification. In order to refine the fused multi-scale feature $f(x)$, we design another transformation block, as shown in Fig. 8(b). The transformation block is composed of a BN layer, a CA module, and a convolutional block, where BN is used to normalize the distribution of the input features to accelerate the convergence of the CNN. The CA is adopted to learn the inter-channel relationship, to enable the CNN to focus on channels with critical discriminative information and suppress channels with irrelevant redundant information. The convolutional block is used to reduce the dimension of the multi-scale feature $f(x)$ and further extract the abstract category-related features. Therefore, the transformed multi-scale feature of sample $x$ is produced as

$$\begin{aligned} f_t(x) &= \mathrm{Tran}(f(x)) \\ &= \mathrm{Conv}(\mathrm{CA}(\mathrm{BN}(f(x)))) \end{aligned} \tag{7}$$

where BN and CA are the batch normalization and channel attention, respectively, and Conv denotes the convolutional block.

Compared to the multi-scale feature fusion strategy proposed in [64], the proposed MsFF model in the AG-MsPN has two newly added elements, i.e., the CA module after the input and the transformation block before the output, as shown in Fig. 8(b). As demonstrated in the experimental results in Table IX, the newly added elements indeed improve the feature extraction ability of the network, which results in the higher classification accuracy of the proposed MsFF model than only using the feature downsampling and concatenation as that in [64].

*2) Classification of the MsPN:* In our method, each sample is, in fact, a multi-channel SAR image. According to (4)–(7), we can calculate the transformed multi-scale features of samples in the support set $S^i$ and the query set $Q^i$ selected within a few-shot classification task in the $i$th iteration, respectively. Then, the class labels of query samples $\{x_j, x_j \in Q^i\}$ are predicted via finding the nearest prototypes, as depicted in Fig. 2. The classification of the MsPN is explained in detail as follows.

First, the transformed multi-scale features of the support samples within the same class are averaged to obtain the prototype of this class. The prototype of class $k$ is calculated as

$$m_k = \frac{1}{|S_k|} \sum_{(x_i, y_i) \in S_k} f_t\left(x_i\right) \tag{8}$$

where $x_i$ is the support sample in the support set, $y_i$ is the corresponding class label of $x_i$, $f_t(x_i)$ denotes the transformed multi-scale feature of $x_i$, and $S_k \subset S^i$ denotes the set of support samples labeled with class $k$.

After generating the prototype of each class, the Euclidean distances between the transformed multi-scale features of query samples and the prototypes are calculated to evaluate their similarities. The smaller the distance, the higher the similarity between the transformed multi-scale feature of one query sample and the prototype.

Specifically, given a query sample $x_j \in Q^i$, the Euclidean distance between its transformed multi-scale feature $f_t(x_j)$ and the prototype of class $k$ is calculated as

$$d_{x_j}^k = \left\| f_t\left(x_j\right) - m_k \right\|_2 . \tag{9}$$

According to (9), we can calculate the Euclidean distances between $f_t(x_j)$ and the prototypes of all the classes. Then, based on the softmax function, the Euclidean distances are normalized into the interval (0,1) to get the probabilities that $x_j$ belongs to each class. The probability that $x_j$ belongs to class $k$ is defined as

$$p\left(y_j = k \mid x_j\right) = \frac{\exp\left(-d_{x_j}^k\right)}{\sum_{k'} \exp\left(-d_{x_j}^{k'}\right)}. \tag{10}$$

It is noticed from (10) that when the query sample $x_j$ has smaller distance with the prototype $m_k$, it will be predicted to belong to class $k$ with a higher probability.

The classification loss $L_c$ for query sample $x_j$ with true class $k$ is defined as

$$L_c = -\log p\left(y_j = k \mid x_j\right). \tag{11}$$

### C. Attribute Classification Module

Since the numbers of training classes and the total training samples in the source domain of SAR image datasets are far fewer than those of optical image datasets, the MsPN learned with limited training samples in the source domain still cannot generalize well enough to the unseen test data. However, with the prior attribute information of the object, even without the

TABLE I
DESIGNED PRIOR BINARY ATTRIBUTES OF TEN CLASSES OF TARGETS IN THE MSTAR DATASET

| Type | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| BMP2 | 1 | 0 | 1 | 1 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 |
| BTR70 | 0 | 1 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 1 |
| T72 | 1 | 0 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 1 |
| BTR60 | 0 | 1 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 |
| 2S1 | 1 | 0 | 1 | 1 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 |
| BRDM2 | 0 | 1 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 |
| D7 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 |
| T62 | 1 | 0 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| ZIL131 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 |
| ZSU23/4 | 1 | 0 | 1 | 1 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 |

The numbers 1–12 represent 12 different attributes, which are *pedrail*, *machine gun*, *cannon*, *battery*, *tank*, *armored personnel carrier*, *armored reconnaissance vehicle*, *rocket launcher*, *air defense unit*, *truck*, *bulldozer*, and *derived type*, respectively.

training samples of it, humans can recognize the unseen object by its attributes. Therefore, except for the visual feature classification of the MsPN, we design the prior binary attribute for each class of target and devise an ACM to map SAR target images into the attribute space for the attribute classification, which is shown at the bottom of Fig. 2. The class labels of SAR targets are predicted by comprehensively considering the visual feature classification result of the MsPN and the attribute classification result of the ACM. Hence, with the end-to-end training process of the MsPN and the ACM, the classification performance of the MsPN is further enhanced under the joint supervision of the one-hot class label information from a few labeled data and the attribute information from the prior knowledge.

To the best of our knowledge, the attribute information has not been used in the few-shot SAR target classification models before. For the first time, our method tries to combine the attribute classification with the few-shot SAR target classification models to improve the few-shot classification performance.

The design of the prior binary attributes and the ACM is described as follows.

*1) Definition of Prior Binary Attributes:* According to the attributes utilized in previous ZSL studies [54], [57], we define the binary attributes for each type of target according to the following rules.

a) *Distinguishability:* The prior binary attribute of each class must be different, so that the network can be provided with the correct guidance information during the training phase.
b) *Shareability:* An attribute should be contained by multiple classes in order to construct an attribute space shared by multiple classes, in which the samples are mapped and classified by finding the most similar class attributes.

For example, in order to classify otter, polar bear, and zebra, the shared attributes *black*, *white*, *brown*, *stripes*, *water*, and *eat fish* are selected to describe them. Thus, the prior binary attributes of otter, polar bear, and zebra are, respectively, devised as [1,0,1,0,1,1], [0,1,0,0,1,1], and [1,1,0,1,0,0], where "0" means there is no such attribute and "1" is the opposite. Then, a new object can be classified to the three classes by finding whether it has the above attributes.

Thus, according to the rules described above, we define the prior binary attributes of vehicle targets in the MSTAR dataset, where the optical photographs of all classes are depicted in Fig. 7. Following the characteristics of these targets, we select four obvious appearance attributes of *pedrail*, *machine gun*, *cannon*, and *battery* based on the corresponding optical photographs of these vehicle targets. Because of the high appearance similarity between the targets in the MSTAR dataset, the four attributes selected above, as shown in the first four columns in Table I, are not enough to distinguish the ten classes of targets. In [54], Lampert *et al.* pointed out that not only the color and shape of the targets but also the geographic information, which are not observed in the images, can also be utilized. Inspired by this, we add the functional attributes *tank*, *armored personnel carrier*, *armored reconnaissance vehicle*, *rocket launcher*, *air defense unit*, *truck*, and *bulldozer* for the ten classes of MSTAR vehicle targets to expand the prior binary attributes. Additionally, the attribute *derived type* is also added to distinguish the types of T62 and T72, BTR60, and BTR70, which are highly similar in appearance and function. Finally, the designed prior binary attributes of the ten classes of targets in the MSTAR dataset are shown in Table I, where "0" means there is no such attribute and "1" is the opposite.

*2) Attribute Classification:* In order to enhance the meta-learning of the MsPN under limited training samples in the source domain, inspired by the method of [57], we propose an ACM to map SAR target images into the attribute space via the linear combination of the defined prior binary attributes for attribute classification.

As shown in Fig. 2, in a few-shot classification task, after the query samples are classified by the MsPN, we first send the predicted class probabilities of the MsPN together with the prior binary attributes into the ACM to predict the attributes of query samples. Afterward, the similarity scores between the predicted attributes of query samples and the prior binary attribute of each class are calculated to evaluate their similarities. The class labels of query samples are determined by finding the class prior binary attributes with the highest similarity scores to the predicted attributes.

As illustrated in Fig. 2, given a query sample $x_j$, we first employ the MsPN to predict the probabilities $\{p(y_j = k \mid x_j), k \in C_i\}$ that $x_j$ belongs to each of the classes, where $C_i$ is the training label set of the current sampled few-shot classification task. Then, in the ACM, we map the query sample $x_j$ into the attribute space via the linear combination of the prior binary attributes $\{b_k, k \in C_i\}$. The predicted attribute $a_{x_j}$ of $x_j$ is given as follows:

$$a_{x_j} = \sum_k p\left(y_j = k \mid x_j\right) \cdot b_k. \tag{12}$$

After obtaining the predicted attribute $a_{x_j}$ of the query sample $x_j$, in the attribute space, we compute the similarity score between $a_{x_j}$ and the prior binary attribute of each class to evaluate their similarity. The similarity score of $a_{x_j}$ and the prior binary

attribute $b_k$ is given as

$$s\left(a_{x_j}, b_k\right) = \frac{\exp\left(-d_{x_j}^{k}\right)}{\sum_{k'} \exp\left(-d_{x_j}^{k'}\right)} \tag{13}$$

where

$$d_{x_j}^{k} = \left\| a_{x_j} - b_k \right\|_2 . \tag{14}$$

$d_{x_j}^{k}$ is the Euclidean distance between $a_{x_j}$ and $b_k$, which measures the similarity between $a_{x_j}$ and $b_k$.

The similarity loss $L_s$ for attribute $a_{x_j}$ with true class $k$ is

$$L_s = -\log s\left(a_{x_j}, b_k\right) . \tag{15}$$

In the proposed ACM, only the way of attribute prediction is the same as that in [57]. In the attribute classification method proposed in [57], the attribute classification is only used in the test procedure. Specifically, a base classifier is first trained using the training data. Then, in the test stage, the well-trained base classifier is utilized to predict the class probabilities of test data, and then, the obtained class probabilities and the prior attributes of each class are sent into the attribute classification model proposed in [57] to predict the class labels of test data. Differing from the method proposed in [57], we jointly utilize the ACM and the MsPN to predict the class labels of test data. Besides, the ACM and the MsPN are optimized end-to-end, thus improving the classification performance of the MsPN under the supervision of both the label information from a few labeled data and the attribute information from the prior knowledge.

### D. Joint Optimization of the AG-MsPN

By comprehensively considering the predicted class probabilities of the MsPN and the similarity scores of the ACM, the class label that the AG-MsPN predicts for query sample $x_j$ is defined as

$$y_j = \underset{k}{\operatorname{argmax}} \left( p\left(y_j = k \mid x_j\right) + s\left(a_{x_j}, b_k\right) \right) . \tag{16}$$

The joint loss of the proposed AG-MsPN corresponding to the query sample $x_j$ is defined as follows:

$$L = L_c + \lambda L_s \tag{17}$$

where $\lambda$ is a balance parameter.

During the parameter optimization of the AG-MsPN, the backward propagation pass is illustrated by the red-dashed arrows in Fig. 2. It can be observed that the parameter optimization of the feature extractor is influenced by both $L_c$ and $L_s$.

During the minimization of the classification loss $L_c$ of the MsPN, the class probability $p(y_j = k \mid x_j)$ of query sample $x_j$ with true class $k$ should be optimized toward 1. Therefore, according to (9) and (10), the extracted feature $f_t(x_j)$ of $x_j$ is optimized toward the prototype of the $k$th class, which is supervised by the one-hot class label information of query sample $x_j$.

As can be seen from Fig. 2 that when minimizing the similarity loss $L_s$ of the ACM, the similarity score $s(a_{x_j}, b_k)$ is optimized toward 1. This means that the predicted attribute $a_{x_j}$ of query sample $x_j$ is pulled toward the prior binary attribute $b_k$. In order to pull $a_{x_j}$ closer toward $b_k$, as can be seen in (12), the class probability $p(y_i = k \mid x_j)$ of the MsPN should be optimized toward 1. Therefore, the classification performance of the MsPN is promoted under the additional supervision of the attribute information of query sample $x_j$. In other words, the optimization of the MsPN is guided by the attribute information, that is why we call it the AG-MsPN.

In summary, under the joint supervision of the one-hot class label information and the attribute information of targets, the classification performance of the proposed AG-MsPN has been effectively improved.

## V. Experiments

### A. Dataset

In this section, we evaluate our method on the MSTAR benchmark dataset [9]. There are ten types of ground targets in the MSTAR dataset, and the images were collected by a SAR sensor with a resolution of 0.3 m $\times$ 0.3 m. To eliminate the influence of background on the target classification, we crop the complex-valued SAR images from 128 $\times$ 128 pixels to 64 $\times$ 64 pixels. Subsequently, the SD is used for the cropped complex-valued SAR data to obtain the subband images. All the SAR images used for training and testing are processed by L2 normalization. The target classes and the sample number of the MSTAR dataset are shown in Table II.

To evaluate the performance of our proposed AG-MsPN, we implement the classification experiments for the three classes of targets, according to the experimental settings in [25]. Therefore, in this article, the three-class target classification is regarded as our target task, where the target classes and the sample numbers of the support set and test set in the target domain are shown in Table III. Meanwhile, we use the samples of the remaining seven classes in the MSTAR dataset as the training set of the source domain, as illustrated in Table IV, to assist the three-class target classification in the target domain. The training set and the support set have the same depression angle of 17°, but the samples in the test set are acquired under the depression angle of 15°. Notice that the serial numbers of the same targets in the support set and test set are different, which increases the difficulty of classification.

### B. Implementation Details

In our proposed method, we set the structure of convolutional blocks and downsampling blocks of each layer to be the same, and the balance parameter $\lambda$ is set to 1 according to the parameter sensitivity analysis shown later in Section V-D. We employ the ADAM optimization algorithm [66] to optimize our network. The model is trained by 5000 iterations in total, where the learning rate starts from 0.001 and decreases by 1/10 for every 1000 iterations.

*Meta-Training Procedure:* In the meta-training procedure, to fully leverage the label information of the support set in the target domain, we follow the training method of [67] to combine the support set and the training set to optimize the parameters of AG-MsPN.

**Algorithm 1:** Meta-Training Procedure of the AG-MsPN in the $N$-Way $K$-Shot Case.

**Input**: The training set $D_{\text{tr}}$ in the source domain, the augmented support set $S^A$ in the target domain, $D_{\text{all}} = S^A \bigcup D_{\text{tr}}$, the training set $D_{\text{all}}^k$ of class $k$, $D_{\text{all}}^k \subset D_{\text{all}}$, the label set $C_{\text{all}}$ corresponding to $D_{\text{all}}$, the predefined prior binary attributes $B_{\text{all}} = \{b_k, k \in C_{\text{all}}\}$, the maximum number $n$ of training iterations, the number $K$ of support samples per class in the selected support set, the number $K_q$ of query samples per class in the selected query set, $RandomSample(A, B)$ represents a set composed of $B$ elements which are randomly sampled from set $A$ without repetition.

**Output**: The well-trained parameters $\theta$ of the AG-MsPN.

Initialize the parameters $\theta$ of the AG-MsPN;
**for** $i = 1 : n$ **do**
  $C_i \leftarrow RandomSample(C_{\text{all}}, N)$; ▷Select the label set of the few-shot classification task
  **for** $k$ in $C_i$ **do**
    $S_k \leftarrow RandomSample(D_{\text{all}}^k, K)$; ▷Select support set of class $k$
    $Q_k \leftarrow RandomSample(D_{\text{all}}^k \setminus S_k, K_q)$; ▷Select query set of class $k$
  **end for**
  $S^i = \bigcup_{k=1}^{N} S_k$; ▷Selected support set in the $i$th iteration
  $Q^i = \bigcup_{k=1}^{N} Q_k$; ▷Selected query set in the $i$th iteration
  $T_i = S^i \bigcup Q^i$; ▷Selected few-shot classification task in the $i$th iteration
  $B^i = \{b_k, k \in C_i\}$, where $B^i \subset B_{\text{all}}$; ▷Select prior binary attributes in the $i$th iteration
  Send the samples in $T_i$ and the binary attributes in $B^i$ into the AG-MsPN;
  Compute the prototypes $\{m_k, k \in C_i\}$ using the support samples in $S^i$ by (8);
  **for** $j = 1 : N \times K_q$ **do**
    Compute the class probabilities $\{p(y_j = k \mid x_j), k \in C_i\}$ of query sample $x_j$ by (10), where $x_j \in Q^i$;
    Compute the classification loss $L_c$ of $x_j$ by (11);
    Predict the attribute $a_{x_j}$ of $x_j$ by (12);
    Compute the similarity scores $\{s(a_{x_j}, b_k), b_k \in B^i\}$ by (13);
    Compute the similarity loss $L_s$ of $x_j$ by (15);
    Compute the joint loss $L_j$ of $x_j$ by (17);
    $j \leftarrow j + 1$;
  **end for**
  Update $\theta \leftarrow \theta - \alpha \nabla_\theta \frac{1}{N \times K_q} \sum_j L_j$, where $\alpha$ is the learning rate;
**end for**

**Algorithm 2:** Meta-Test Procedure of the AG-MsPN in the $N$-Way $K$-Shot Case.

**Input**: Randomly selected support set $S^C$ and the test set $D_{te}$ in the target domain, the label set $C_{te}$ corresponding to $D_{te}$, the predefined prior binary attributes $B_{te} = \{b_k, k \in C_{te}\}$ of the classes in $C_{te}$, and the number $n_{te}$ of test samples in $D_{te}$.

**Output**: The class labels of the test samples predicted by the AG-MsPN.

The samples in $S^C$ and $D_{te}$ are fed into the AG-MsPN;
Compute prototypes $\{m_k, k \in C_{te}\}$ using the support samples in $S^C$ according to (8);
**for** $j = 1$ to $n_{te}$ **do**
  Compute the class probabilities $\{p(y_j = k \mid x_j), k \in C_{te}\}$ of $x_j$ by (10), where $x_j \in D_{te}$;
  Predict the attribute $a_{x_j}$ of $x_j$ by (12);
  Compute the similarity scores $\{s(a_{x_j}, b_k), b_k \in B_{te}\}$ by (13);
  Predict the class label of $x_j$ by (16);
  $j \leftarrow j + 1$;
**end for**

As can be seen from Table III, the sample number of each class in the support set is more than 200. To imitate the few-shot classification in the target domain, the current $K$-shot support set $S^C$ is randomly selected from the total support set shown in Table III. Specifically, in the 3-way $K$-shot few-shot classification experiment, we randomly select $K$ samples per class from the support set shown in Table III to serve as the current support set $S^C$. Afterwards, for each sample in $S^C$, we randomly select three angles from the interval [0°, 360°), and then rotate the sample clockwise according to these selected angles. Combining the original samples in $S^C$ and their rotated versions, we obtain the augmented support set $S^A$. Thus, the training set shown in Table IV and the augmented support set $S^A$ are jointly utilized to train the AG-MsPN.

The details of meta-training is shown in Algorithm 1.

*Meta-Test Procedure:* In the meta-test procedure, the selected current $K$-shot support set $S^C$ is directly used to calculate the prototypes of the three classes in the target domain. The same procedure as meta-training is utilized to predict the class labels of test data, which is shown in Algorithm 2.

### C. Performance Comparison

In 3-way $K$-shot experiments, we set the number $K$ to 5, 10, and 20. To eliminate the randomness of the experimental results, we conduct the experiments 20 times and reselect the current $K$-shot support set in each experiment. Table V displays the average results of the following methods.

1) *A-ConvNet [2]:* It uses the convolutional layers to replace the fully connected layers to reduce the number of free parameters and is trained with the augmented training set.
2) *The method proposed in [11] (denoted by DA-Net):* DA-Net is a CNN that uses various data augmentation methods to expand the training set, such as translation, pose synthesis, etc.
3) *The ARGN [25] based on transfer learning:* The network is trained with sufficient data in the source domain and aims to transfer the shared angular-related information from the source domain to the target domain to solve the SAR target classification with limited labeled data.

TABLE II
TEN CLASSES OF TARGETS IN THE MSTAR DATASET

| Target | BMP2 | | | BTR70 | T72 | | | BTR60 | 2S1 | BRDM2 | D7 | T62 | ZIL131 | ZSU23/4 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Class | SNC21 | SN9563 | SN9566 | C71 | SN132 | SN812 | SNS7 | k10yt7532 | b01 | E-71 | 92v13015 | A51 | E12 | d08 |
| 17° | 0 | 233 | 0 | 233 | 232 | 0 | 0 | 256 | 299 | 298 | 299 | 299 | 299 | 299 |
| 15° | 196 | 195 | 196 | 196 | 196 | 195 | 191 | 195 | 274 | 274 | 274 | 273 | 274 | 274 |

TABLE III
TARGET DOMAIN: THREE CLASSES OF TARGETS IN THE MSTAR DATASET

| Target | BMP2 | | | BTR70 | T72 | | |
|---|---|---|---|---|---|---|---|
| Class | SNC21 | SN9563 | SN9566 | C71 | SN132 | SN812 | SNS7 |
| Support set | 0 | 233 | 0 | 233 | 232 | 0 | 0 |
| Test set | 196 | 195 | 196 | 196 | 196 | 195 | 191 |

TABLE IV
SOURCE DOMAIN: SEVEN CLASSES OF TARGETS IN THE MSTAR DATASET

| Target Class | BTR60 | 2S1 | BRDM2 | D7 | T62 | ZIL131 | ZSU23/4 |
|---|---|---|---|---|---|---|---|
| Training set | 256 | 299 | 298 | 299 | 299 | 299 | 299 |

TABLE V
CLASSIFICATION ACCURACIES OF EXPERIMENTS IN THE 3-WAY $K$-SHOT CASE

| Methods | 5-shot | 10-shot | 20-shot |
|---|---|---|---|
| A-ConvNet [2] | 57.51% | 68.04% | 78.82% |
| DA-Net [11] | 58.47% | 65.13% | 73.50% |
| ARGN [25] | 53.49% | 67.44% | 82.73% |
| PN [30] | 75.12% | 78.37% | 81.08% |
| RN [31] | 72.54% | 76.66% | 77.88% |
| Meta-Baseline [68] | 77.18% | 79.41% | 83.25% |
| MSAR [35] | 75.76% | 78.57% | 82.08% |
| **Ours** | **82.03**% | **87.57**% | **92.68**% |

4) *The PN [30] and the RN [31] based on meta-learning:* The PN maps samples into the embedding space and classifies them by computing the Euclidean distances between the sample feature and the prototype of each class. The RN learns a relation module to measure the relationships between sample features and the class prototypes for classification.
5) *Meta-Baseline [68]:* Meta-Baseline aims to learn a metric space, in which the sample features of the same class are clustered, and the class labels of the samples are determined via finding the class prototype with the highest cosine similarity to the samples. Compared with the PN, Meta-Baseline makes the following two aspects of improvements. First, it uses the pretraining strategy. Second, it introduces a learnable parameter to scale the calculated cosine similarities in the meta-training procedure, to enhance the performance of few-shot classification.
6) *The few-shot SAR target classification network proposed by Fu* et al. *[35] (named as MSAR).* MSAR is an optimization-based meta-learning framework. The pretraining is first utilized to obtain the transferrable knowledge from the metatraining set of the source domain. Then, the parameters of MSAR are fine-tuned on the metatraining set according to the training method in [69], where the transfer operation proposed in [70] and the hard task mining strategy proposed in [35] are also merged to facilitate the classification performance improving of MSAR.

Table V shows the quantitative comparison of seven different methods. From Table V, we can see that our proposed network achieves the best classification performance in all the comparison methods under all 3-way $K$-shot settings, which demonstrates the superiority of our proposed method. Compared with A-ConvNet [2] and DA-Net [11], our method obtains a performance improvement of 24.52% and 23.56% in the 3-way 5-shot scenario, respectively. This is mainly because the DA-Net and the A-ConvNet employ the augmentation strategy to solve the classification problem under limited SAR data. But the augmented data still lacks diversity, so they cannot learn better feature representations. As for the A-ConvNet, we can see that the classification accuracy drops rapidly when there are only five samples. Although decreasing the network parameters can alleviate the overfitting caused by the limited labeled data, the alleviation is limited. In contrast, the methods based on meta-learning (i.e., PN [30], RN [31], Meta-Baseline [68], MSAR [35], and our proposed method) can achieve higher accuracy in the case of 5-shot, which confirms that the meta-learning method is more suitable for few-shot SAR target classification. Besides, in all scenarios, our method outperforms the PN and the RN by at least 6% and 9%, respectively. Compared with the metric-based meta-learning methods PN and RN, due to the introduced pretraining strategy and a learnable scale parameter to scale the similarities between the sample features and the class prototypes, the classification performance of Meta-Baseline is slightly better than the PN and the RN, but still worse than ours. This is mainly because our method comprehensively considers the characteristics of SAR images. We not only leverage the rich amplitude information and the phase information contained in complex-valued SAR data to obtain better feature representations, but also devise the MsPN and the ACM to further improve the classification performance with few training samples. For ARGN, although the data of the source domain are used to facilitate the classification, it does not use the episodic training method, resulting in poor generalization of the ARGN to the novel classes in the target domain. Although the hard task mining and the transfer operation are used in MSAR, the classifier learned by the limited training samples in the source domain still cannot generalize well enough to the

TABLE VI
CLASSIFICATION RESULTS OF EXPERIMENTS WITH THE SAME SUPPORT SET

| Methods | 5-shot | | | 10-shot | | | 20-shot | | |
|---|---|---|---|---|---|---|---|---|---|
| | Set1 | Set2 | Set3 | Set1 | Set2 | Set3 | Set1 | Set2 | Set3 |
| A-ConvNet [2] | 55.44±5.60 | 57.82±3.27 | 58.10±6.45 | 62.90±7.44 | 67.70±6.07 | 66.48±1.40 | 70.61±1.11 | 80.36±6.95 | 77.12±6.20 |
| DA-Net [11] | 56.49±1.48 | 55.85±1.78 | 60.41±1.73 | 66.26±2.14 | 67.22±1.56 | 62.54±1.60 | 70.85±2.48 | 69.37±1.02 | 71.91±1.67 |
| ARGN [25] | 56.37±4.89 | 60.86±1.76 | 59.88±4.89 | 64.84±5.43 | 61.68±5.59 | 59.16±3.38 | 78.49±4.01 | 79.49±3.25 | 78.43±5.34 |
| PN [30] | 76.95±3.86 | 74.16±3.47 | 75.62±4.20 | 80.05±3.17 | 78.76±3.11 | 79.30±3.33 | 84.35±1.45 | 84.62±2.11 | 85.31±1.58 |
| RN [31] | 71.04±3.27 | 74.33±3.68 | 75.36±3.67 | 78.44±4.36 | 77.48±3.16 | 75.63±4.45 | 79.93±2.81 | 77.43±3.88 | 80.19±3.48 |
| Meta-Baseline [68] | 76.88±2.69 | 75.95±3.92 | 77.80±3.05 | 81.65±2.29 | 78.98±3.71 | 80.76±2.25 | 84.96±1.76 | 85.95±1.95 | 86.17±2.30 |
| MSAR [35] | 74.71±2.67 | 76.52±2.81 | 75.84±2.48 | 77.98±2.18 | 77.06±2.25 | 78.24±1.94 | 81.61±1.25 | 82.76±1.97 | 80.44±1.49 |
| **Ours** | **83.61±2.60** | **81.08±2.42** | **84.16±1.79** | **90.60±2.06** | **89.26±1.01** | **93.39±1.21** | **94.58±0.90** | **92.18±0.97** | **93.16±0.86** |

The average accuracy (%) of 20 training models is shown followed by the standard deviation (%). Set1, Set2, and Set3 represent three randomly selected support sets in each few-shot scenario.

unseen test data. Thus, its classification accuracy is lower than ours in the 5-shot, 10-shot, and 20-shot cases.

Table V shows the average performance of our proposed AG-MsPN and other comparison methods on 20 different randomly selected support sets. To further verify that our method is better than other comparison methods with the same support set, the following experiments are implemented. Specifically, we first fix three sets of support sets in each few-shot scenario. Then, we train each comparison model for 20 times and calculate its average test accuracy and standard deviation. As shown in Table VI, the classification performance of our proposed AG-MsPN is absolutely superior compared with other methods for the same support set, in all three few-shot scenarios. In addition, the robustness of our proposed AG-MsPN outperforms other approaches. Due to the severe data limitation in the 5-shot scenario, the robustness of our method also degrades to a certain extent.

### *D. Parameter Sensitivity Analysis*

The subband number in SD, the layers to be fused in MsFF, and the balance parameter $\lambda$ in the joint loss function shown in (17) are the main parameters affecting the performance of the AG-MsPN. The sensitivity analysis about the above three parameters is provided as follows. All the experiments mentioned below are performed with the same training set, support set, and test set, which are shown in Tables III and IV.

*1) Number of Subbands:* In order to discuss the influence of the number of decomposed subbands on our proposed method, we show the performance comparison result in Fig. 10 with different numbers of subbands. Specifically, we select to divide the azimuth spectrum and range spectrum into two, three, four, and five equal parts without overlap, respectively. Thereby, four, six, eight, and ten decomposed subband images can be obtained for each complex-valued SAR image, respectively. As demonstrated in Fig. 10, the classification accuracy achieves the highest when the spectrum is divided into three parts along the range and azimuth directions, respectively. When the spectrum is divided into four and five parts, in the condition of 3-way 5-shot, the classification accuracies of the AG-MsPN drop to 78.15% and 76.25%, respectively. As the number of divided

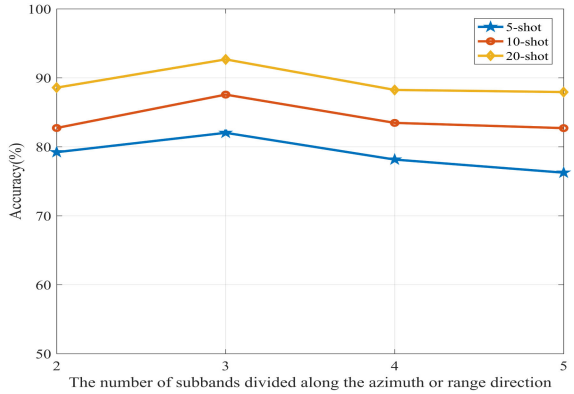


Fig. 10. Performance of the AG-MsPN with different number of subbands.

subspectrums increases, the target information contained in each subspectrum is more incomplete; thus, the detailed information of the target contained in each subband image becomes fewer. Therefore, the useful information of the SAR targets learned by the AG-MsPN is reduced, leading to a decrease in the classification accuracy. When we divide the spectrum into two parts, the backscattering variation of the targets is not fully reflected due to the insufficient decomposition. Hence, the classification accuracy drops to 79.22% in the case of 5 shots. Based on the above analysis, we finally choose to divide the azimuth spectrum and range spectrum into three parts, respectively, to obtain six subband images.

*2) Layers to be Fused in MsFF:* Since the ways of MsFF are flexible, we investigate the impact of different MsFF methods on the performance of few-shot SAR target classification. For simplicity, we use the token MsFF($i, j$) to represent the MsFF from the $i$th to the $j$th layers mentioned in Section IV-B, and MsFF(0) denotes the feature extraction without the MsFF. The experiments are carried out in the 3-way $K$-shot scenario where $K = 5$, 10, and 20, and the performance comparison results are depicted in Fig. 11. From Fig. 11, we can see that the models with the MsFF [i.e., MsFF(1,4), MsFF(2,4) and MsFF(3,4)] have a significant enhancement in classification performance compared to the model without the MsFF [MsFF(0)], and MsFF(1,4) achieves the highest classification accuracy. In the

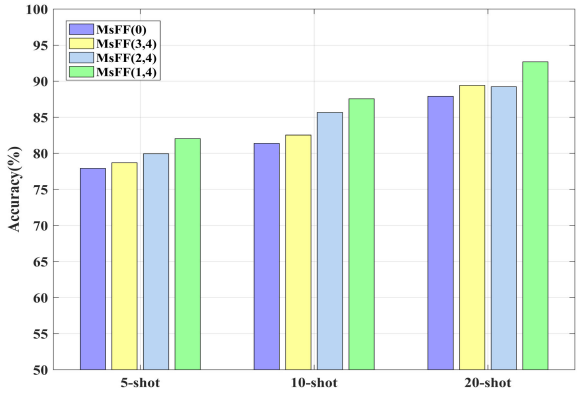


Fig. 11. Performance of the AG-MsPN with different feature fusion methods on the MSTAR dataset. MsFF$(i, j)$ represents to fuse the features from the $i$th layer to the $j$th layer. MsFF(0) means that only the output features from the fourth layer are used without any fusion strategy.

TABLE VII
CLASSIFICATION ACCURACIES (%) UNDER DIFFERENT VALUES OF $\lambda$

| $\lambda$ | 0 | 0.01 | 0.05 | 0.1 | 0.5 | 1 | 5 | 10 |
|---|---|---|---|---|---|---|---|---|
| 5-shot | 81.62 | 81.75 | 81.66 | 81.12 | 81.91 | **82.03** | 81.73 | 82.01 |
| 10-shot | 85.93 | 87.14 | 86.51 | 86.92 | 86.25 | **87.57** | 86.92 | 86.89 |
| 20-shot | 91.02 | 92.35 | 92.28 | 92.56 | **93.02** | 92.68 | 91.94 | 92.54 |

5-shot, 10-shot, and 20-shot cases, the classification accuracies of MsFF(1,4) are about 4%, 6%, and 5% higher than those of the MsFF(0), respectively. The above experimental results confirm that the detailed information extracted by the low-level layers benefits the classification. Thus, in the proposed AG-MsPN, we choose the way of MsFF(1,4) to fuse the features from the first to the fourth layers, to improve the classification performance.

*3) Balance Parameter $\lambda$:* In order to determine the appropriate value for the balance parameter $\lambda$ in (17), we set $\lambda = 0, 0.01, 0.05, 0.1, 0.5, 1, 5, 10$ to gradually increase the influence of the similarity loss of the ACM in the total loss. We performed the experiments 20 times to eliminate the randomness of the experimental results. Table VII displays the average results. It is noticed from Table VII that, under the condition of 5-shot and 10-shot, the classification performance achieves the best when $\lambda = 1$. However, in the 20-shot scenario, the accuracy at $\lambda = 0.5$ is 93.02%, which is a little higher than that at $\lambda = 1$. Since the performance improvement at $\lambda = 0.5$ is not significant, we finally set $\lambda = 1$.

## E. Ablation Study

In this section, a series of ablation experiments are conducted on the MSTAR dataset to further analyze the contribution of our proposed method to few-shot SAR target classification. All the experiments mentioned below are performed with the same training set, support set, and test set, which are shown in Tables III and IV.

*1) Contributions of SD, MsFF, and ACM:* To further evaluate the effectiveness of the complex-valued SAR image SD, the MsFF strategy, and the added ACM, we conducted the following experiments.

TABLE VIII
VERIFICATION OF THE CONTRIBUTIONS OF DIFFERENT COMPONENTS IN THE PROPOSED AG-MSPN

| Methods | 5-shot | 10-shot | 20-shot |
|---|---|---|---|
| Baseline | 75.67% | 79.86% | 85.11% |
| Baseline+SD | 76.25% | 80.99% | 86.72% |
| Baseline+SD+MsFF | 81.62% | 85.93% | 91.20% |
| **Baseline+SD+MsFF+ACM** | **82.03**% | **87.57**% | **92.68**% |

a) *Baseline:* Only the PN is utilized to classify the SAR images; the complex-valued SAR image SD, the MsFF strategy, and the attribute classification are not used here.
b) *Baseline+SD:* The SD of the complex-valued SAR images is introduced into Baseline.
c) *Baseline+SD+MsFF:* On the basis of Baseline+SD, we add the MsFF strategy to improve the feature extractor of the PN.
d) *Baseline+SD+MsFF+ACM:* The SD, the MsFF, and the attribute classification are added to Baseline at the same time, which is our proposed AG-MsPN.

The ablation experiments mentioned above are also performed in the 3-way $K$-shot scenario, where $K$ is set to 5, 10, and 20, and the selected support set of the target domain and the training set in the source domain are jointly used to train the models. The experimental results are reported in Table VIII. In Table VIII, the classification accuracies of Baseline are higher than those of the PN in Table V. This phenomenon is due to that we follow the original method in [30] to get the results in Table V, that is, only the training set of the source domain is used to train the PN. It is obvious that the classification performance of Baseline+SD+MsFF+ACM achieves the best in 5-shot, 10-shot, and 20-shot scenarios. Compared with Baseline, the classification accuracy of Baseline+SD is 86.72% in the 20-shot case, which is about 1.6% higher than Baseline. This indicates that with the various backscattering behaviors of the targets in subband images, the SAR targets can be expressed in more detail. Thereby, the network acquires more useful information about the targets from limited labeled data, and the classification accuracy is enhanced. Due to the added MsFF strategy, the classification performance of Baseline+SD has been dramatically improved from 86.72% to 91.20% in the 20-shot case. Since the low-level detailed information and the high-level global information are fused gradually, the extracted features become more discriminative, leading to a higher classification accuracy of SAR targets. In the 3-way 20-shot case, the classification accuracy of Baseline+SD+MsFF+ACM is about 1.4% higher than that of Baseline+SD+MsFF, which indicates that with the additional supervision of the attribute information from the prior knowledge, the model parameters have been further optimized to obtain a better feature embedding.

*2) Contributions of Newly Added CA and Tran in MsFF:* The MsFF used in the AG-MsPN is based on the feature fusion method of [64]. To verify the effect of the two newly added

TABLE IX
CONTRIBUTIONS OF THE CA MODULE AND THE TRANSFORMATION BLOCK

| Methods | 5-shot | 10-shot | 20-shot |
|---|---|---|---|
| MsFF0 | 73.13% | 80.75% | 87.77% |
| MsFF0+CA | 74.99% | 82.22% | 89.01% |
| **MsFF0+CA+Tran** | **82.03**% | **87.57**% | **92.68**% |

TABLE X
EFFECTIVENESS VERIFICATION OF OUR DEFINED 12-D PRIOR BINARY ATTRIBUTES

| Methods | $N_1$ | $N_2$ | $N_3$ | $N_4$ | $N_5$ | $N_6$ |
|---|---|---|---|---|---|---|
| Accuracy (%) | 89.77 | 89.26 | 89.37 | 88.80 | 90.02 | 89.92 |
| Methods | $N_7$ | $N_8$ | $N_9$ | $N_{10}$ | $N_{11}$ | $N_{12}$ |
| Accuracy (%) | 89.83 | 85.62 | 90.57 | 86.86 | 86.17 | 85.48 |

The token $N_i, i = 1, 2, \ldots 12$, mentioned above means to remove the $i$th attribute.

elements, i.e., the CA module after the input, and the transformation block before the output, as shown in Fig. 8(b), some ablation experiments have been implemented as follows.

Table IX reports the average performance of each model on 20 different randomly selected support sets, where MsFF0 denotes that only the feature downsampling and concatenation in [64] are used in our method, and CA and Tran represent the channel attention and the transformation block, respectively.

From Table IX, we can see that MsFF0 has the lowest classification accuracy. After utilizing the CA, the classification accuracy of MsFF0 is improved by approximately 1.8%, 1.5%, and 1.2% in the 5-shot, 10-shot, and 20-shot cases. This indicates that the generated CA map effectively weights the SAR images of different channels. Thus, the channels with critical discriminative features are emphasized, while the ones that contribute less to classification are suppressed. Compared with MsFF0 and MsFF0+CA, MsFF0+CA+Tran achieves the highest classification accuracy in 5-shot, 10-shot, and 20-shot cases. This phenomenon demonstrates that the multi-scale features obtained by simply merging all features from different network layers contain redundant information, which leads to degraded classification performance. Through the learning of the transformation block, the important information in the fused multi-scale features is further selected, where the complementary information extracted by different layers of the CNN is retained, and the redundant information is removed at the same time.

*3) Exploration of Prior Binary Attribute Design:* To evaluate the effectiveness of our defined 12-D prior binary attributes, we conducted a series of experiments on attribute selection. Specifically, we remove a certain attribute in turn from the defined 12-D prior binary attributes and then train and test our AG-MsPN using the remaining 11-D prior binary attributes in the 3-way 20-shot scenario, aiming to prove the necessity of the 12-D attributes we have defined. Table X reports the results of the above experiments, where the token $N_i, i = 1, 2, \ldots 12$, means to remove the $i$th attribute. It can be seen from Table X that the classification performance of AG-MsPN degrades to different degrees using the different 11-D attributes. The classification performance of the AG-MsPN is worst when removing the 12th attribute *derived type*. From Table I, we know that after the 12th attribute *derived type* is removed, the 11-D prior binary attributes of BTR60 and BTR70 are the same, as well as T62 and T72. Therefore, when the samples of BTR70 and BTR60 or T72 and T62 are selected within a training iteration, the ACM fails to distinguish them, resulting in the poor few-shot classification performance of the AG-MsPN. For further validation, in Fig. 12, we also show the confusion matrices of the AG-MsPN on the test classes after removing a certain attribute. From the confusion matrices, we can clearly see that when the attribute *derived type* is removed, the classification accuracy of the AG-MsPN for BTR70 and T72 is reduced to 89.64% and 91.11%, respectively, reaching the lowest. The above phenomenon confirms our view that the mappings of BTR70 and T72, learned by the AG-MsPN, are slightly worse than those of the other types under the condition of removing the attribute *derived type*.

In summary, the AG-MsPN has the best classification result when using the designed 12-D prior binary attributes; thereby, the necessity of the 12-D attributes is proved. To further show the effectiveness of the designed 12-D attributes more intuitively, we use Baseline+SD+MsFF and Baseline+SD+MsFF+ACM, which are shown in Table VIII, to extract the features of test data. Then, t-SNE [71] is applied to visualize them in Fig. 13. It is obvious that the features of the same class in Fig. 13(b) are closer to each other, and the boundaries between different classes are clearer, compared with Fig. 13(a). This phenomenon verifies that the designed 12-D attributes yield a better feature distribution in the feature space, thereby improving the classification performance.

*4) Effectiveness of Attributes on the Obscured Test Data:* In real scenes, the vehicle targets we are interested in might be obscured by surrounding objects, resulting in some components of targets invisible in SAR images (e.g., the cannon, battery, etc.). In this case, some attributes in the defined 12-D prior binary attributes do not exist in the captured SAR images. In order to verify that the defined prior binary attributes can still contribute to the classification of the obscured SAR images, in this section, we conduct the comparison experiments in the condition of obscuration. Specifically, we perform the random obscuration on the three classes of test images (shown in Table III) and train the AG-MsPN on the training images with no obscuration to directly classify the obscured test images.

The process of generating the obscured test images [72] is shown in Fig. 14. Given a complex-valued SAR image, we first randomly select a pixel from the specified area of $30 \times 30$ pixels [as shown in Fig. 14(b)]. With this pixel as the center, the surrounding area of $54 \times 54$ pixels is selected, which is denoted as $obspart$. Then, we perform a threshold segmentation on the complex-valued SLICY image $x$ in the MSTAR dataset to set the pixel values of the target region to 1 and set the background ones to 0, to obtain the binary segmented SLICY image $x_{\text{seg}}$, where the SLICY image is acquired at an azimuth angle of 260.95° with a size of $54 \times 54$. Then, the obscured $54 \times 54$ region $obspart'$ of complex-valued SAR image can be defined as follows:

$$obspart' = obspart \times (I - x_{\text{seg}}) + x \times x_{\text{seg}} \tag{18}$$

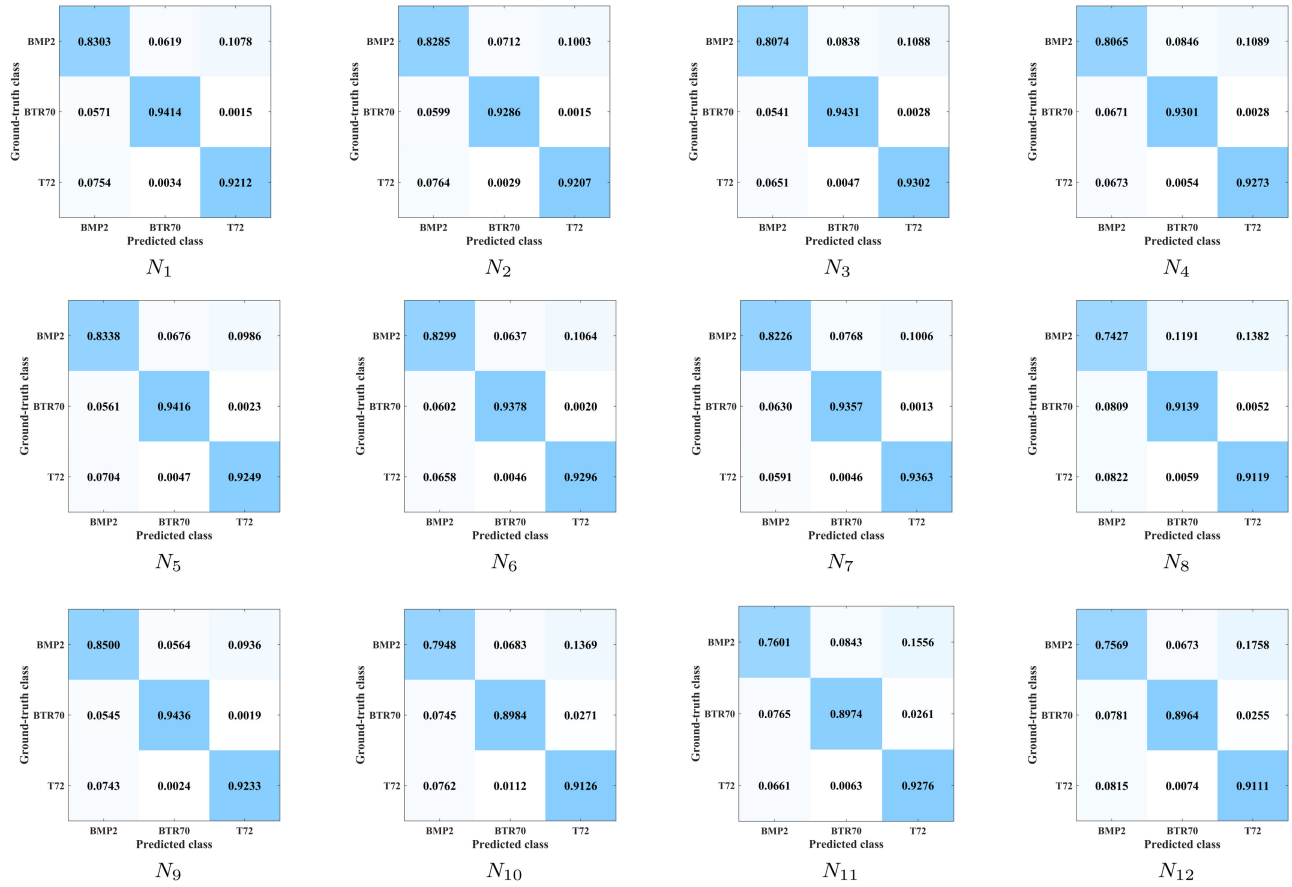


Fig. 12. Confusion matrices of the AG-MsPN on the test classes after removing a certain attribute.

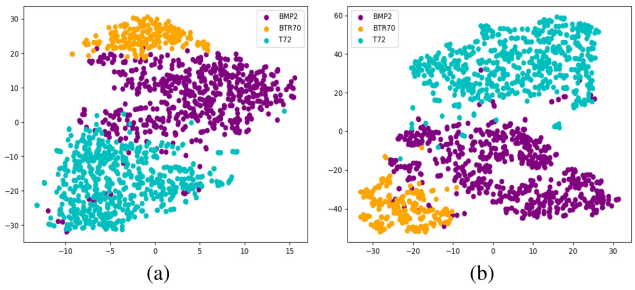


Fig. 13. Feature visualization of test data. (a) Features extracted by Baseline+SD+MsFF in the 3-way 20-shot case. (b) Features extracted by Baseline+SD+MsFF+ACM in the 3-way 20-shot case.

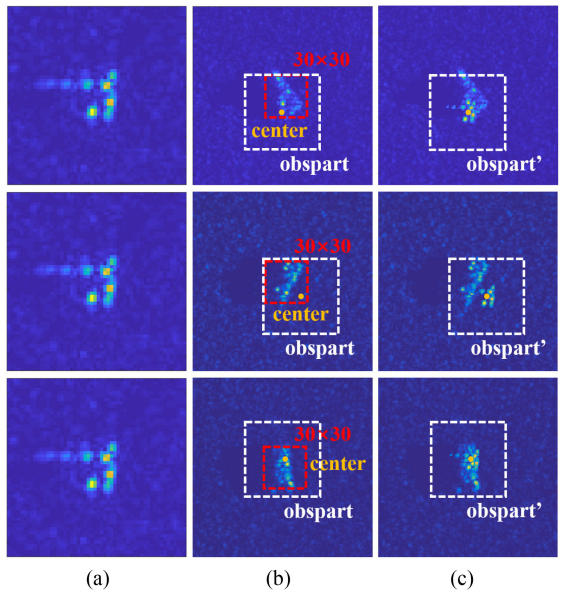


Fig. 14. Diagram of synthesizing the obscured SAR data. (a) SLICY image. (b) Original image. (c) Obscured image.

where $I$ represents a matrix of size $54 \times 54$ and the value of each element is set to 1.

Finally, we replace $obspart$ by $obspart'$ in the original complex-valued SAR image to obtain the obscured complex-valued SAR image, which is shown in Fig. 14(c).

Following the above obscuration process, we randomly select 450, 900, and all of the samples (1365) from the test set to generate the obscured test images. Fig. 15 displays the comparison results under three different numbers of obscured test images, where MsPN indicates that only the multi-scale prototypical network is used to implement the test. In addition, other methods are also compared and shown in Fig. 15. Since all the models are trained on the unobscured dataset, we can see that the classification accuracies of each model decrease to various degrees when testing the obscured SAR data. Moreover, as the number

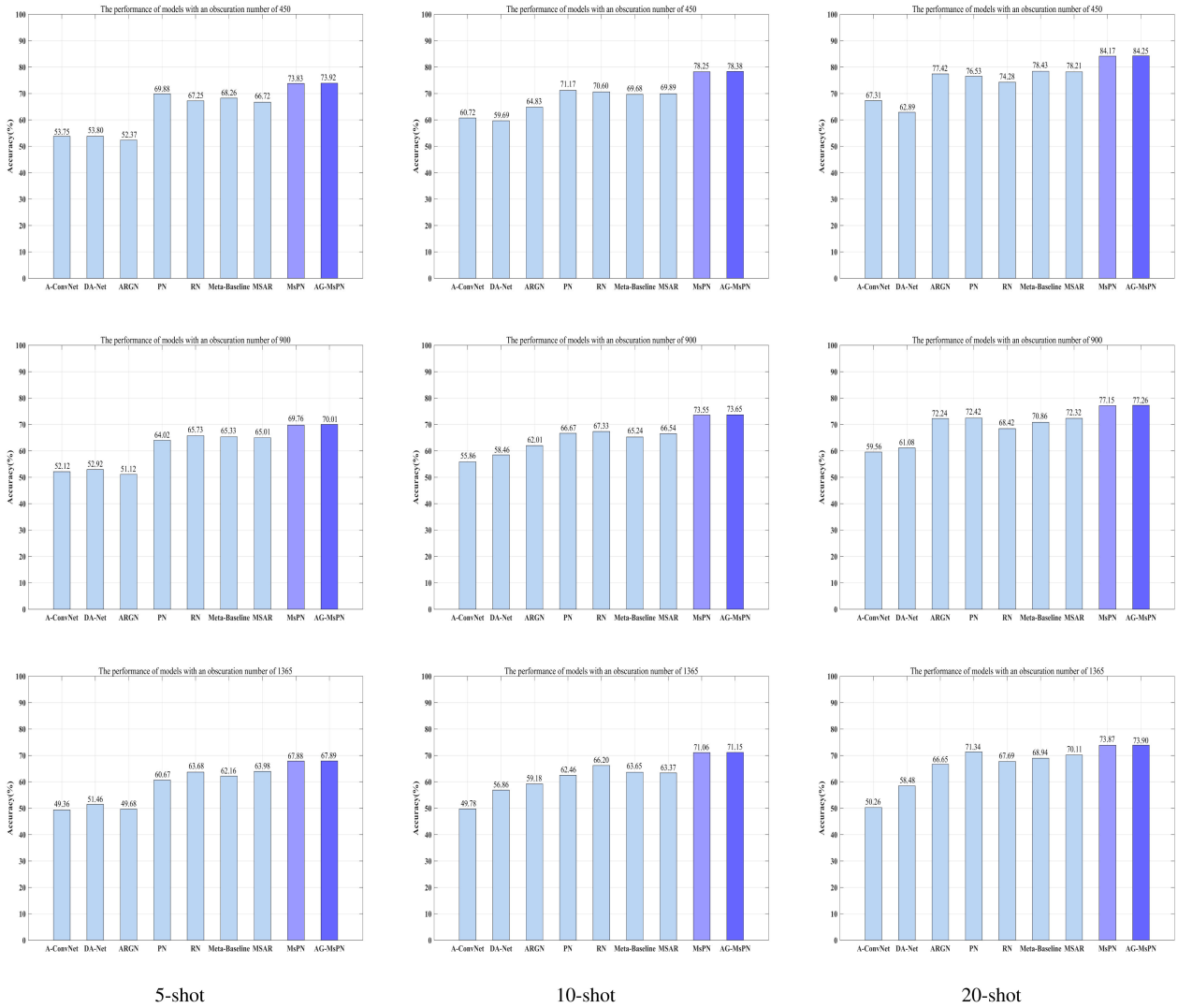


Fig. 15. Classification performance of our model and the other comparison models with different number of obscured images.

of obscured samples in the test set increases, the classification performance of the models gradually decreases. Obviously, the classification performance of our proposed AG-MsPN is the best. For instance, the classification accuracy of the AG-MsPN is 84.25% with 450 obscured test images in the 20-shot case, which is higher than the other methods without obscuration, as shown in Table V, implying that the MsPN and prior binary attributes we designed can maintain better performance on the obscured test data and allow a good generalization ability to realistic scenarios of the proposed method. Compared to the MsPN, the AG-MsPN achieves the better performance regardless of the 5-shot, 10-shot, or 20-shot scenario. It means that the classification results of the MsPN can still be improved by further mapping the obscured test data into the attribute space for classification. Although the gain from the ACM becomes smaller when the obscuration is severe compared to that without the obscuration, the designed SD and multi-scale feature extraction methods still enable our model to maintain a better classification performance than other methods, so that the proposed AG-MsPN can still be generalized to obtain a better classification result in the presence of obscuration.

### *F. Generalization Ability*

To explore the generalization ability of our model, we analyze the classification performance of the proposed AG-MsPN under different settings of the target domain and the source domain in this section. Specifically, we randomly select three different target domains and their corresponding source domains from the ten classes of targets in the MSTAR dataset shown in Table II. Table XI shows the target classes and the sample numbers in each selected target domain, respectively. For the training set of the source domain corresponding to each target domain, the samples at the 17° depression angle of the remaining seven classes are used. The training set for each source domain is not shown here. For simplicity, we use the tokens TD1, TD2, and TD3 to denote the reselected target domains, respectively, and TD0 is applied to represent the target domain in Table III.

TABLE XI
TARGET DOMAINS UNDER THREE DIFFERENT SETTINGS

| Target Class | TD1 | | | | | TD2 | | | TD3 | | |
|---|---|---|---|---|---|---|---|---|---|---|---|
| | BRDM2 | T62 | T72 | | | 2S1 | D7 | T62 | 2S1 | BRDM2 | ZSU23/4 |
| | E-71 | A51 | SN132 | SN812 | SNS7 | b01 | 92v13015 | A51 | b01 | E-71 | d08 |
| Support set | 298 | 299 | 232 | 0 | 0 | 299 | 299 | 299 | 299 | 298 | 299 |
| Test set | 274 | 273 | 196 | 195 | 191 | 274 | 274 | 273 | 274 | 274 | 274 |

TABLE XII
CLASSIFICATION ACCURACIES OF EXPERIMENTS IN THE 3-WAY $K$-SHOT CASE UNDER THREE DIFFERENT TARGET DOMAINS

| Methods | TD1 | | | TD2 | | | TD3 | | |
|---|---|---|---|---|---|---|---|---|---|
| | 5-shot | 10-shot | 20-shot | 5-shot | 10-shot | 20-shot | 5-shot | 10-shot | 20-shot |
| A-ConvNet [2] | 61.80% | 69.34% | 76.78% | 58.36% | 70.00% | 78.33% | 72.79% | 78.25% | 84.04% |
| DA-Net [11] | 62.96% | 68.85% | 76.57% | 59.88% | 68.99% | 76.20% | 73.39% | 77.32% | 83.74% |
| ARGN [25] | 60.11% | 71.72% | 85.47% | 61.09% | 81.98% | 92.09% | 68.53% | 84.50% | 93.31% |
| PN [30] | 85.71% | 88.03% | 90.44% | 89.51% | 91.78% | 93.71% | 89.57% | 90.83% | 91.12% |
| RN [31] | 83.94% | 85.28% | 86.28% | 89.39% | 91.08% | 92.51% | 89.01% | 89.83% | 89.92% |
| Meta-Baseline [68] | 85.78% | 88.77% | 89.94% | 88.92% | 92.70% | 94.16% | 89.66% | 91.08% | 92.63% |
| MSAR [35] | 88.30% | 90.88% | 92.08% | 89.74% | 92.11% | 93.52% | 91.60% | 93.19% | 93.81% |
| **Ours** | **88.43**% | **91.09**% | **95.38**% | **94.60**% | **96.42**% | **98.06**% | **95.00**% | **98.05**% | **98.63**% |

Table XII illustrates the results of our proposed AG-MsPN and other comparison methods under different target domains and the source domains. It can be seen that our proposed AG-MsPN can still maintain the optimal classification performance under different divisions of the target domain and the source domain. Moreover, the performance of the AG-MsPN has a substantial improvement with the change of the target domain. The main reason for this phenomenon is that the classification problems of TD1, TD2, and TD3 are simpler, compared to TD0. In TD0, as shown in Table III, the test set and the support set of BMP2 and T72 have different serial numbers, which makes the classification much more difficult. However, in TD1, TD2, and TD3, only the test images of T72 in TD1 have different serial numbers from the support set. Therefore, the classification accuracy of the AG-MsPN in TD1 condition is lower than that in TD2 and TD3, but still higher than that in TD0. Furthermore, in TD2, the bulldozer D7 has a distinct appearance difference from tank T62 and the rocket launcher 2S1, as shown in Fig. 7, so they can be better classified. Similarly, it is easier to distinguish the truck ZSU23/4 and the other targets in TD3.

To further confirm our above analysis, we use t-SNE to visualize the original test images in TD0, TD1, TD2, and TD3 and display them in Fig. 16(a)–(d). At the same time, the features of test images extracted by the AG-MsPN under different target domains are also visualized in Fig. 16(e)–(h). From the visualization images (a)–(d), it is obvious that the distribution boundaries between different classes of test images gradually become clearer from TD0 to TD3, which indicates that the samples in our reselected target domains TD1, TD2, and TD3 are more easily to be separated than those in TD0. Moreover, as the target domain changes from TD0 to TD3, the extracted features of the same class become more compact, resulting in more distinct boundaries among different classes, which leads to a substantial improvement in the classification performance of the AG-MsPN. In short, our proposed AG-MsPN can maintain excellent generalization capabilities under different settings of the target domains and the source domains.

### G. Model Complexity and Running Time

In this section, the model complexity and the running time during the test procedure have been carefully analyzed. All the experiments are implemented on a DELL workstation with the Intel Xeon Silver 4210R CPU (six cores, 2.40 GHz) and the NVIDIA GeForce RTX 2080 Ti GPU (11-GB memory).

*1) Model Complexity:* For the classification models based on deep learning [73]–[75], the model complexity is usually evaluated via calculating the total number of parameters to be optimized in the network. Thus, following these works [73]–[75], we calculate the total number of parameters in our model and other comparison models to show their model complexity. The results are presented in Table XIII.

*2) Running Time:* In the training stage, since the size of the minibatch and the number of iterations are different for the comparison methods, the running time of the training phase is not compared. To fairly show the running time of the comparison models in the 5-shot, 10-shot, and 20-shot scenarios during the test stage, we feed the samples in the test set shown in Table III into the comparison models with the same batch size (batch size is set to 1). Each model is tested 20 times, and the average running time in the test procedure is shown in Table XIII.

For the ARGN, a feature extractor is first trained during the training procedure and then fixed for feature extraction in the test stage. The test sample features extracted by the trained feature extractor are finally classified by the support vector

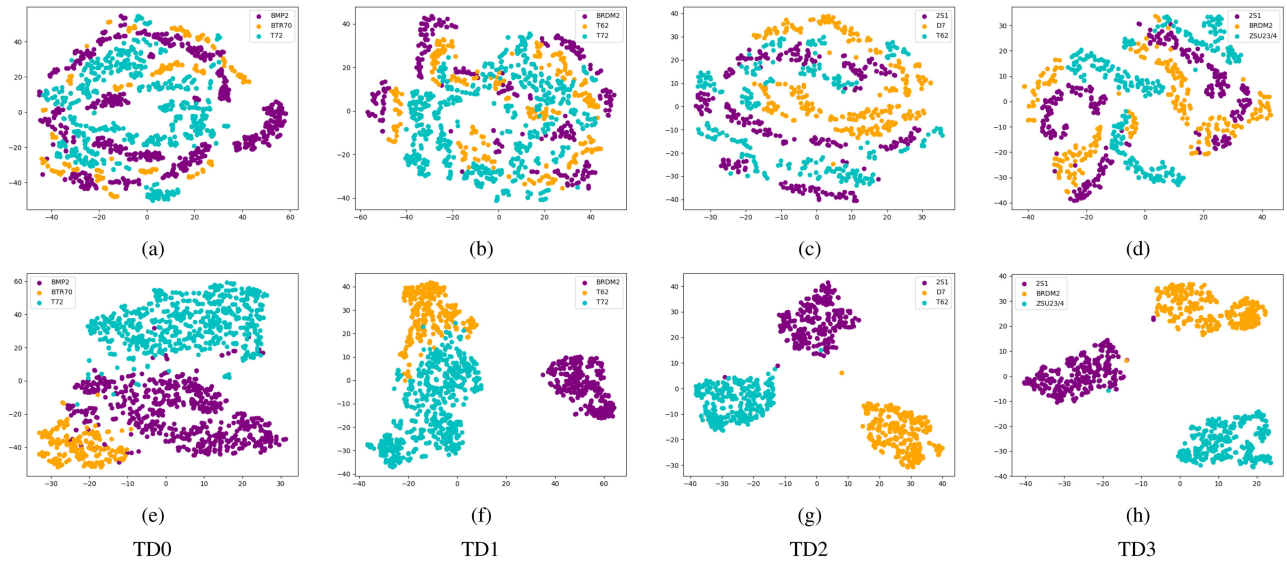


Fig. 16. Visualization of test images and the test features extracted by the AG-MsPN under different target domains. (a)–(d) represent the visualization of test images under TD0, TD1, TD2, and TD3, respectively. (e)–(h) are the feature visualization of the test images extracted by the AG-MsPN under TD0, TD1, TD2, and TD3, respectively.

TABLE XIII
TOTAL NUMBER OF PARAMETERS AND RUNNING TIME OF DIFFERENT MODELS

| Methods | Parameters (M) | Running time (s) | | |
|---|---|---|---|---|
| | | 5-shot | 10-shot | 20-shot |
| A-ConvNet [2] | 0.29 | 3.09 | 3.09 | 3.09 |
| DA-Net [11] | 2.63 | 2.53 | 2.53 | 2.53 |
| ARGN [25] | 0.57 | 1.66 | 1.78 | 2.07 |
| PN [30] | 0.11 | 5.04 | 5.11 | 6.33 |
| RN [31] | 0.19 | 5.12 | 5.56 | 6.42 |
| MSAR [35] | 0.13 | 3.34 | 3.34 | 3.34 |
| Meta-Baseline [68] | 10.99 | 24.83 | 30.32 | 39.30 |
| Ours | 1.36 | 11.59 | 13.16 | 17.40 |

machine [76]. Thus, the running time of the ARGN in the test procedure achieves the shortest, as illustrated in Table XIII. Since the number of convolutional layers in A-ConvNet (five layers) is larger than that of DA-Net (three layers), the running time of the A-ConvNet in the test stage is longer than that of the DA-Net. It is noticed that the running time of the metric-based meta-learning methods (i.e., PN, RN, Meta-Baseline, and ours) is longer than the ordinary deep-learning-based methods (A-ConvNet, DA-Net, and ARGN). This is because the class labels of samples predicted by the metric-based meta-learning methods are obtained via calculating the distances or similarity scores between the sample features and the prototype of each class, which increases the computation burden. Moreover, since the prototype of each class is calculated by the support set in the target domain, more running time is required in the metric-based meta-learning methods (i.e., PN, RN, Meta-Baseline, and ours) when the sample number of the support set increases. For the MSAR based on meta-learning, the class labels of images are predicted by the fully connected layers; therefore, the running time of MSAR is smaller than that of PN, RN, Meta-Baseline, and our model. A great number of parameters in Meta-Baseline leads to a substantial increase in the running time, which is the longest among all the comparison models. In the proposed method, since the feature extractor of the PN is improved to fuse the low-level detailed information and high-level global information, the model complexity of our proposed method is slightly higher than that of the PN, as well as the running time. However, considering the significant performance improvement of our model, the slightly increased model complexity and running time are acceptable.

## VI. CONCLUSION

In this article, we propose an AG-MsPN, in which the complete information contained in the complex-valued SAR data, as well as the prior attribute information of the targets, is fully considered to enhance the performance of few-shot SAR target classification. Specifically, we first perform SD of the complex-valued SAR data to obtain more information about SAR targets. In order to extract more discriminative features to alleviate the intra-class diversity and inter-class similarity problems of SAR images, an MsPN is proposed to adaptively fuse the low-level detailed information and the high-level global information. Besides, we design the prior binary attributes for SAR targets and present an ACM, aiming to enhance the performance of the MsPN under the joint guidance of the one-hot class label information of the few training samples and the attribute information from the prior knowledge. In the 3-way 5-shot, 10-shot, and 20-shot cases, the classification accuracies of our proposed network on the MSTAR dataset surpass those of other comparison methods. In the future, we will further consider the few-shot SAR target classification combining with the simulated data.

## REFERENCES


[1] J. Pei, Y. Huang, W. Huo, Y. Zhang, J. Yang, and T.-S. Yeo, "SAR automatic target recognition based on multiview deep learning framework," *IEEE Trans. Geosci. Remote Sens.*, vol. 56, no. 4, pp. 2196–2210, Apr. 2018.

[2] S. Chen, H. Wang, F. Xu, and Y. Jin, "Target classification using the deep convolutional networks for SAR images," *IEEE Trans. Geosci. Remote Sens.*, vol. 54, no. 8, pp. 4806–4817, Aug. 2016.

[3] W. Wu, H. Li, L. Zhang, X. Li, and H. Guo, "High-resolution PolSAR scene classification with pretrained deep convnets and manifold polarimetric parameters," *IEEE Trans. Geosci. Remote Sens.*, vol. 56, no. 10, pp. 6159–6168, Oct. 2018.

[4] M. Zhang, J. An, D. H. Yu, L. D. Yang, L. Wu, and X. Q. Lu, "Convolutional neural network with attention mechanism for SAR automatic target recognition," *IEEE Geosci. Remote Sens. Lett.*, early access, Nov. 2020. doi: 10.1109/LGRS.2020.3031593.

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

[6] C. Wang, H. Gu, and W. Su, "SAR image classification using contrastive learning and pseudo-labels with limited data," *IEEE Geosci. Remote Sens. Lett.*, to be published, doi: 10.1109/LGRS.2021.3069224.

[7] S. Woo, J. Park, J. Lee, and I. S. Kweon, "CBAM: Convolutional block attention module," in *Proc. 15th Eur. Conf. Comput. Vis.*, 2018, pp. 3–19.

[8] S. Ioffe and C. Szegedy, "Batch normalization: Accelerating deep network training by reducing internal covariate shift," in *Proc. 32nd Int. Conf. Mach. Learn.*, 2015, vol. 37, pp. 448–456.

[9] *The Air Force Moving and Stationary Target Recognition Database*. Accessed on: Mar. 8, 2011, [Online]. Available: https://www.sdms.afrl.af.mil/datasets/mstar/

[10] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, "ImageNet: A large-scale hierarchical image database," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2009, pp. 248–255.

[11] J. Ding, B. Chen, H. Liu, and M. Huang, "Convolutional neural network with data augmentation for SAR target recognition," *IEEE Geosci. Remote Sens. Lett.*, vol. 13, no. 3, pp. 364–368, Mar. 2016.

[12] B. Ding, G. Wen, X. Huang, C. Ma, and X. Yang, "Data augmentation by multilevel reconstruction using attributed scattering center for SAR target recognition," *IEEE Geosci. Remote Sens. Lett.*, vol. 14, no. 6, pp. 979–983, Jun. 2017.

[13] J. Guo, B. Lei, C. Ding, and Y. Zhang, "Synthetic aperture radar image synthesis by using generative adversarial nets," *IEEE Geosci. Remote Sens. Lett.*, vol. 14, no. 7, pp. 1111–1115, Jul. 2017.

[14] G. Fei, Y. Yue, J. Wang, J. Sun, E. Yang, and H. Zhou, "A deep convolutional generative adversarial networks (DCGANs)-based semi-supervised method for object recognition in synthetic aperture radar (SAR) images," *Remote Sens.*, vol. 10, no. 6, 2018, Art. no. 846.

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

[16] Z. Wang and X. Xu, "Efficient deep convolutional neural networks using CReLU for ATR with limited SAR images," *J. Eng.*, vol. 2019, no. 12, pp. 7615–7618, 2019.

[17] M. Amrani and F. Jiang, "Deep feature extraction and combination for synthetic aperture radar target classification," *J. Appl. Remote Sens.*, vol. 11, no. 4, 2017, Art. no. 0 42616.

[18] M. Amrani, F. Jiang, Y. Xu, S. Liu, and S. Zhang, "SAR-oriented visual saliency model and directed acyclic graph support vector metric based target classification," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 11, no. 10, pp. 3794–3810, Oct. 2018.

[19] M. Amrani, K. Yang, D. Zhao, X. Fan, and F. Jiang, "An efficient feature selection for SAR target classification," in *Proc. Pacific Rim Conf. Multimedia*, 2017, pp. 68–78.

[20] Y. Du, J. Liu, W. Song, Q. He, and D. Huang, "Ocean eddy recognition in SAR images with adaptive weighted feature fusion," *IEEE Access*, vol. 7, pp. 1520 23–1520 33, 2019.

[21] Z. Huang, Z. Pan, and B. Lei, "Transfer learning with deep convolutional neural network for SAR target classification with limited labeled data," *Remote Sens.*, vol. 9, no. 9, 2017, Art. no. 907.

[22] C. Zhong, X. Mu, X. He, J. Wang, and M. Zhu, "SAR target image classification based on transfer learning and model compression," *IEEE Geosci. Remote Sens. Lett.*, vol. 16, no. 3, pp. 412–416, Mar. 2019.

[23] W. Zhang, Y. Zhu, and Q. Fu, "Semi-supervised deep transfer learning-based on adversarial feature learning for label limited SAR target recognition," *IEEE Access*, vol. 7, pp. 152 412–152 420, 2019.

[24] M. Rostami, S. Kolouri, E. Eaton, and K. Kim, "Deep transfer learning for few-shot SAR image classification," *Remote. Sens.*, vol. 11, no. 11, 2019, Art. no. 1374.

[25] Y. Sun, Y. Wang, H. Liu, N. Wang, and J. Wang, "SAR target recognition with limited training data based on angular rotation generative network," *IEEE Geosci. Remote Sens. Lett.*, vol. 17, no. 11, pp. 1928–1932, Nov. 2020.

[26] A. Li, T. Luo, Z. Lu, T. Xiang, and L. Wang, "Large-scale few-shot learning: Knowledge transfer with class hierarchy," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2019, pp. 7205–7213.

[27] S. Ravi and H. Larochelle, "Optimization as a model for few-shot learning," in *Proc 5th Int. Conf. Learn. Represent.*, 2017, pp. 1–11.

[28] V. G. Satorras and J. B. Estrach, "Few-shot learning with graph neural networks," in *Proc. 6th Int. Conf. Learn. Represent.*, 2018, pp. 1–13.

[29] C. Finn, P. Abbeel, and S. Levine, "Model-agnostic meta-learning for fast adaptation of deep networks," in *Proc. 34th Int. Conf. Mach. Learn.*, 2017, vol. 70, pp. 1126–1135.

[30] J. Snell, K. Swersky, and R. S. Zemel, "Prototypical networks for few-shot learning," in *Proc. Int. Conf. Neural Inf. Process. Syst.*, 2017, pp. 4077–4087.

[31] F. Sung, Y. Yang, L. Zhang, T. Xiang, P. H. S. Torr, and T. M. Hospedales, "Learning to compare: Relation network for few-shot learning," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2018, pp. 1199–1208.

[32] A. A. Rusu *et al.*, "Meta-learning with latent embedding optimization," in *Proc. 7th Int. Conf. Learn. Represent.*, 2019, pp. 1–17.

[33] L. Wang, X. Bai, and F. Zhou, "Few-shot SAR ATR based on Conv-BiLSTM prototypical networks," in *Proc. Asia-Pacific Conf. Synth. Aperture Radar*, 2019, pp. 1–5.

[34] J. Tang, F. Zhang, Y. Zhou, Q. Yin, and W. Hu, "A fast inference networks for SAR target few-shot learning based on improved Siamese networks," in *Proc. IEEE Int. Geosci. Remote Sens. Symp.*, 2019, pp. 1212–1215.

[35] K. Fu, T. Zhang, Y. Zhang, Z. Wang, and X. Sun, "Few-shot SAR target classification via metalearning," *IEEE Trans. Geosci. Remote Sens.*, to be published, doi: 10.1109/TGRS.2021.3058249.

[36] Y. Li, X. Li, Q. Sun, and Q. Dong, "SAR image classification using CNN embeddings and metric learning," *IEEE Geosci. Remote Sens. Lett.*, to be published, doi: 10.1109/LGRS.2020.3022435.

[37] D. Lu, L. Cao, and H. Liu, "Few-shot learning neural network for SAR target recognition," in *Proc. 6th Asia-Pacific Conf. Synth. Aperture Radar*, 2019, pp. 1–4.

[38] K. Wang, G. Zhang, Y. Xu, and H. Leung, "SAR target recognition based on probabilistic meta-learning," *IEEE Geosci. Remote Sens. Lett.*, vol. 18, no. 4, pp. 682–686, Apr. 2021.

[39] L. Wang, X. Bai, C. Gong, and F. Zhou, "Hybrid inference network for few-shot SAR automatic target recognition," *IEEE Trans. Geosci. Remote Sens.*, vol. 59, no. 11, pp. 9257–9269, Nov. 2021.

[40] M. Spigai, C. Tison, and J. Souyris, "Time-frequency analysis in high-resolution SAR imagery," *IEEE Trans. Geosci. Remote Sens.*, vol. 49, no. 7, pp. 2699–2711, Jul. 2011.

[41] J. Singh and M. Datcu, "SAR target analysis based on multiple-sublook decomposition: A visual exploration approach," *IEEE Geosci. Remote Sens. Lett.*, vol. 9, no. 2, pp. 247–251, Mar. 2012.

[42] Z. Huang, M. Datcu, Z. Pan, and B. Lei, "Deep SAR-Net: Learning objects from signals," *ISPRS J. Photogrammetry Remote Sens.*, vol. 161, pp. 179–193, 2020.

[43] Z. Zhang, H. Wang, F. Xu, and Y. Jin, "Complex-valued convolutional neural network and its application in polarimetric SAR image classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 55, no. 12, pp. 7177–7188, Dec. 2017.

[44] Z. Huang, M. Datcu, Z. Pan, X. Qiu, and B. Lei, "HDEC-TFA: An unsupervised learning approach for discovering physical scattering properties of single-polarized SAR image," *IEEE Trans. Geosci. Remote Sens.*, vol. 59, no. 4, pp. 3054–3071, Apr. 2021.

[45] L. Ferro-Famil, A. Reigber, E. Pottier, and W.-M. Boerner, "Scene characterization using subaperture polarimetric SAR data," *IEEE Trans. Geosci. Remote Sens.*, vol. 41, no. 10, pp. 2264–2276, Oct. 2003.

[46] L. Ferro-Famil, A. Reigber, E. Pottier, and W. Boerner, "Scene characterization using sub-aperture polarimetric SAR data analysis," in *Proc. IEEE Int. Geosci. Remote Sens. Symp.*, 2002, vol. 1, pp. 417–419.

[47] J. C. Souyris, C. Henry, and F. Adragna, "On the use of complex SAR image spectral analysis for target detection: Assessment of polarimetry," *IEEE Trans. Geosci. Remote Sens.*, vol. 41, no. 12, pp. 2725–2734, Dec. 2003.
[48] B. M. Lake, R. Salakhutdinov, J. Gross, and J. B. Tenenbaum, "One shot learning of simple visual concepts," in *Proc. Annu. Meeting Cogn. Sci. Soc.*, pp. 1–6, 2011.
[49] T. Munkhdalai and H. Yu, "Meta networks," in *Proc. Int. Conf. Mach. Learn.*, 2017, vol. 70, pp. 2554–2563.
[50] M. Schuster and K. K. Paliwal, "Bidirectional recurrent neural networks," *IEEE Trans. Signal Process.*, vol. 45, no. 11, pp. 2673–2681, Nov. 1997.
[51] G. Koch, R. Zemel, and R. Salakhutdinov, "Siamese neural networks for one-shot image recognition," in *Proc. Int. Conf. Mach. Learn. Deep Learn. Workshop*, Lille, France, pp. 1–30, 2015, vol. 2.
[52] N. Guberman, "On complex valued convolutional neural networks," *CoRR*, 2016, *arXiv:1602.09046*.
[53] L. Yu, Y. Hu, X. Xie, Y. Lin, and W. Hong, "Complex-valued full convolutional neural network for SAR target classification," *IEEE Geosci. Remote Sens. Lett.*, vol. 17, no. 10, pp. 1752–1756, Oct. 2020.
[54] C. H. Lampert, H. Nickisch, and S. Harmeling, "Learning to detect unseen object classes by between-class attribute transfer," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2009, pp. 951–958.
[55] Z. Akata, F. Perronnin, Z. Harchaoui, and C. Schmid, "Label-embedding for attribute-based classification," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2013, pp. 819–826.
[56] Y. Li, J. Zhang, J. Zhang, and K. Huang, "Discriminative learning of latent features for zero-shot recognition," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2018, pp. 7463–7471.
[57] M. Norouzi *et al.*, "Zero-shot learning by convex combination of semantic embeddings," in *Proc. 2nd Int. Conf. Learn. Represent.*, 2014, pp. 1–9.
[58] A. Frome *et al.*, "DeViSE: A deep visual-semantic embedding model," in *Proc. Int. Conf. Neural Inf. Process. Syst.*, 2013, pp. 2121–2129.
[59] X. Glorot, A. Bordes, and Y. Bengio, "Deep sparse rectifier neural networks," in *Proc. 14th Int. Conf. Artif. Intell. Statist.*, 2011, vol. 15, pp. 315–323.
[60] O. Vinyals, C. Blundell, T. Lillicrap, K. Kavukcuoglu, and D. Wierstra, "Matching networks for one shot learning," in *Proc. Int. Conf. Neural Inf. Process. Syst.*, Barcelona, Spain, 2016, pp. 3630–3638.
[61] T.-Y. Lin, P. Dollär, R. Girshick, K. He, B. Hariharan, and S. Belongie, "Feature pyramid networks for object detection," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2017, pp. 936–944.
[62] K. He, X. Zhang, S. Ren, and J. Sun, "Spatial pyramid pooling in deep convolutional networks for visual recognition," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 37, no. 9, pp. 1904–1916, Sep. 2015.
[63] L.-C. Chen, G. Papandreou, I. Kokkinos, K. Murphy, and A. Yuille, "Semantic image segmentation with deep convolutional nets and fully connected CRFs," in *Proc. Int. Conf. Learn. Represent.*, vol. 40, no. 4, pp. 834–848, 2017.
[64] D. Quan *et al.*, "AFD-Net: Aggregated feature difference learning for cross-spectral image patch matching," in *Proc. IEEE/CVF Int. Conf. Comput. Vis.*, 2019, pp. 3017–3026.
[65] X. Yin, J. Goudriaan, E. A. Lantinga, J. Vos, and H. J. Spiertz, "A flexible sigmoid function of determinate growth," *Ann. Botany*, vol. 91, no. 3, pp. 361–371, 2003.
[66] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," in *Proc. 3rd Int. Conf. Learn. Represent.*, 2015, pp. 1–15.
[67] A. Li, T. Luo, T. Xiang, W. Huang, and L. Wang, "Few-shot learning with global class representations," in *Proc. IEEE/CVF Int. Conf. Comput. Vis.*, 2019, pp. 9715–9724.
[68] Y. Chen, X. Wang, Z. Liu, H. Xu, and T. Darrell, "A new meta-baseline for few-shot learning," *CoRR*, 2020, *arXiv:2003.04390*.
[69] Z. Li, F. Zhou, F. Chen, and H. Li, "Meta-SGD: Learning to learn quickly for few-shot learning," *CoRR*, 2017, *arXiv:1707.09835*.
[70] Q. Sun, Y. Liu, T.-S. Chua, and B. Schiele, "Meta-transfer learning for few-shot learning," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2019, pp. 403–412.
[71] L. van der Maaten and G. Hinton, "Visualizing data using t-SNE," *J. Mach. Learn. Res.*, vol. 9, no. 86, pp. 2579–2605, 2008.
[72] J. Ding, H.-W. Liu, and Y.-H. Wang, "SAR image target recognition based on non-negative sparse representation," *Dianzi Yu Xinxi Xuebao/ J. Electron. Inf. Technol.*, vol. 36, pp. 2194–2200, 2014.
[73] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2016, pp. 770–778.
[74] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," in *Proc. 3rd Int. Conf. Learn. Representations, (ICLR)*, pp. 1–10, 2015, *arXiv:1409.1556*.
[75] C. Szegedy *et al.*, "Going deeper with convolutions," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2015, pp. 1–9.
[76] C.-C. Chang and C.-J. Lin, "LIBSVM: A library for support vector machines," *ACM Trans. Intell. Syst. Technol.*, vol. 2, no. 3, 2011, Art. no. 27.

**Siyuan Wang** received the B.S. degree in electronic information engineering from Zhengzhou University, Zhengzhou, China, in 2018. She is currently working toward the Ph.D. degree with Xidian University, Xi'an, China.

Her research interests include few-shot learning and synthetic aperture radar target recognition.

**Yinghua Wang** (Member, IEEE) received the B.S. degree in information engineering and the Ph.D. degree in control science and engineering from Xi'an Jiaotong University, Xi'an, China, in 2004 and 2010, respectively.

In 2007, she joined the Department of Image and Signal Processing, Telecom Paris, Paris, France, as a Visiting Student. She is currently an Associate Professor with the National Laboratory of Radar Signal Processing, Xidian University, Xi'an. Her research interests include synthetic aperture radar (SAR) automatic target recognition, polarimetric SAR data analysis and interpretation, and SAR image processing.

**Hongwei Liu** (Member, IEEE) received the M.S. and Ph.D. degrees in electronic engineering from Xidian University, Xi'an, China, in 1995 and 1999, respectively.

From 2001 to 2002, he was a Visiting Scholar with the Department of Electrical and Computer Engineering, Duke University, Durham, NC, USA. He is currently a Professor with the National Laboratory of Radar Signal Processing, Xidian University. His research interests include radar automatic target recognition, radar signal processing, and adaptive signal processing.

**Yuanshuang Sun** received the B.S. degree in communication engineering in 2015 from Xidian University, Xi'an, China, where she is currently working toward the Ph.D. degree in information and communication engineering.

Her research interests include synthetic aperture radar target recognition under a small number of samples.