Meta-Learning Based Hyperspectral Target Detection Using Siamese Network

Yulei Wang, *Member, IEEE*, Xi Chen, Fengchao Wang, Meiping Song, and Chunyan Yu

***Abstract*— When predicting data for which limited supervised information is available, hyperspectral target detection methods based on deep transfer learning expect that the network will not require considerable retraining to generalize to unfamiliar application contexts. Meta-learning is an effective and practical framework for solving this problem in deep learning. This article proposes a new meta-learning based hyperspectral target detection using Siamese network (MLSN). First, a deep residual convolution feature embedding module is designed to embed spectral vectors into the Euclidean feature space. Then, the triplet loss is used to learn the intraclass similarity and interclass dissimilarity between spectra in embedding feature space by using the known labeled source data on the designed three-channel Siamese network for meta-training. The learned meta-knowledge is updated with the prior target spectrum through a designed two-channel Siamese network to quickly adapt to the new detection task. It should be noted that the parameters and structure of the deep residual convolution embedding modules of each channel in the Siamese network are identical. Finally, the spatial information is combined, and the detection map of the two-channel Siamese network is processed by the guiding image filtering and morphological closing operation, and a final detection result is obtained. Based on the experimental analysis of six real hyperspectral image datasets, the proposed MLSN has shown its excellent comprehensive performance.**

***Index Terms*— Deep learning, hyperspectral imagery, meta-learning, Siamese network, target detection.**

I. Introduction

HYPERSPECTRAL image (HSI) is a 3-D cube containing rich spatial and spectral information with hundreds of narrow and contiguous wavebands by an imaging spectrometer. Each pixel in the HSI contains a contiguous spectrum whose characteristic is related to the materials contained therein. Thanks to their high spectral resolution, HSIs have been applied and played an essential role in civilian search and rescue [1], agricultural production [2], [3], military application [4], urban planning [5], and so on. Target detection has attracted more and more attention in these fields, and it has become an urgent need for developing accurate and effective target detection algorithms.

Hyperspectral target detection is a detection method that identifies and locates the targets with similar spectral characteristics as the prior target spectrum, mainly at the pixel/subpixel level. Many target detection algorithms have been proposed in the literature. In [6], the spectral angle mapper (SAM) is proposed to detect the targets by evaluating the spectral angle between the spectrum of each pixel in the image and the prior target spectrum of interest. The spectral information divergence (SID) is used in [7] to identify the targets by the probability difference between spectral features. Both the SAM and SID are direct and straightforward target detectors. The target detector based on constrained energy minimization (CEM) [8] constructs a finite impulse response (FIR) filter. It constrains the characteristics of the target to be detected with specific gain while minimizing the influence of the background. Various of improved algorithms based on the CEM algorithm are proposed afterward. The hierarchical CEM (hCEM) [9] method uses a structure with different layers of CEM detectors to preserve the target and suppress the background through a layer-by-layer filtering process, and the detection performance is gradually improved. Algorithms such as the adaptive coherence/cosine estimator (ACE) [10], [11] and the adaptive matched filter detector (AMF) [12] are designed according to the hypothesis testing method of the Gaussian distribution hypothesis. Subspace-based hyperspectral target detection algorithms have also been proposed, such as the orthogonal subspace projection (OSP) [13] detector proposed by Chein-I Chang and the matched subspace detector (MSD) designed in [14]. Some detectors based on sparse representation have also been proposed successively, such as the sparse target detector (STD) proposed in [15] and the detector based on joint sparse and cooperative representation (CSCR) proposed by Li [16]. In order to get rid of the constraints of model assumptions, a tree-structured encoding [17] method is proposed to eliminate the influence of model assumptions on detection performance. The ensemble learning-based hyperspectral target detection methods have also been proposed. Methods such as ensemble-based information retrieval with mass estimation [18] and ensemble-based cascaded constrained energy minimization (E-CEM) [19] improve both the generalization and nonlinear discrim-

Manuscript received November 14, 2021; revised March 22, 2022; accepted April 8, 2022. Date of publication April 22, 2022; date of current version May 13, 2022. The work of Yulei Wang was supported in part by the National Nature Science Foundation of China under Grant 61801075; in part by the China Postdoctoral Science Foundation under Grant 2020M670723; in part by the Open Research Funds of State Key Laboratory of Integrated Services Networks, Xidian University under Grant ISN20-15; and in part by the Fundamental Research Funds for the Central Universities under Grant 3132022232. The work of Meiping Song was supported by the National Nature Science Foundation of China under Grant 61971082. *(Corresponding author: Meiping Song.)*

Yulei Wang is with the Center of Hyperspectral Imaging in Remote Sensing (CHIRS), Information Science and Technology College, Dalian Maritime University, Dalian 116026, China, and also with the State Key Laboratory of Integrated Services Networks, Xi'an 710000, China (e-mail: wangyulei@dlmu.edu.cn).

Xi Chen, Fengchao Wang, Meiping Song, and Chunyan Yu are with the Center of Hyperspectral Imaging in Remote Sensing (CHIRS), Information Science and Technology College, Dalian Maritime University, Dalian 116026, China (e-mail: xi_chen@dlmu.edu.cn; 1796745999@qq.com; smping@163.com; yuchunyan1997@126.com).

Digital Object Identifier 10.1109/TGRS.2022.3169970

inate capabilities of hyperspectral target detectors and obtain higher detection accuracy and stability.

Due to the strong generalization and deep extraction of advanced semantic features, deep learning has been gradually applied in HSI processing, such as band selection [20], classification [21], unmixing [22], and super-resolution reconstruction [23]. In recent years, deep learning-based hyperspectral target detection algorithms have gradually been proposed. In [24], a shallow neural network structure in which the 2-D convolutional layer and maximum pooling layer are alternately connected is used to transform the spectral vectors of the target and neighboring pixels into the 2-D matrix and feed it to the network for learning. The convolutional neural network-based target detection (CNNTD) model is proposed by Li [25]. It makes use of a known source data with label information, where samples of the same class and samples between different classes are matched into pixel pairs as training samples to train a 1-D deep convolutional neural network (CNN), and then, the trained 1-D deep CNN model is transferred to detect targets. Since deep CNN requires a large amount of data for training, whereas there are very few training data available for hyperspectral target detection, it is necessary to expand the training data. For this purpose, the deep network-based hyperspectral target detection (HTD-Net) detector is proposed in [26], and the U-autoencoder (AE) structure is designed with the U-net [27] idea to generate potential target samples. According to the known target samples, the background samples significantly different from the target are found by linear prediction algorithm. Then, the target pixel is paired with target pixel and background pixel, respectively, to expand the training samples to train a 16-layer 1-D deep CNN. In [28], the target pixel is subtracted from the background pixels of different classes, and the background pixels of different classes are subtracted from each other to expand the training dataset to train the 30-layer 1-D CNN. In addition to the method of CNN-based target detection, there are also detection methods that use the idea of the generative adversarial network (GAN) [29]. Background learning based on a target suppression constraint (BLTSC) detector is proposed in [30], and a variant of GAN is used to the Adversarial AE (AAE) [31] to reconstruct the background. The CEM algorithm is used to coarse filter the HSI to obtain the background samples, feeding the background samples to the AAE to learn until convergence. Target suppression constraint loss is added to the loss function to suppress the AAE reconstruction target. The AAE will reconstruct the HSI while inputting the original HSI. The reconstructed HSI has good background reconstruction, and the targets would be found with the large reconstruction errors. Variational AE (VAE) [32] is also used for target detection. It performs spectral regularization for the VAE through a designed spectral regularization unsupervised network [33] so that the hidden nodes could better characterize the spectral information of the HSI. A weighted map is then obtained by weighting the feature maps outputted by the hidden nodes, the background in the weighted map is suppressed by the morphological opening operation, and the detection result was finally obtained by using the guide image filter to smooth the image. Due to the low spatial resolution of HSI, many of the target pixels in the HSI image are subpixels. In order to better detect the subpixels, a two-stream CNN [34] is designed to learn the discriminative ability of the difference between target and background spectra. It simulates the subpixels and expands training samples by finding the typical background pixels and mixing them with *a priori* target pixel. Furthermore, a semisupervised domain adaptive few shot learning detector is proposed in [35] to solve the problems of limited training samples and sensor-dependent transferability. It adopts the ideas of metric learning and domain adaptation to adaptively transfer the measurement of spectral similarity in the embedded space (obtained by the prototype network [36]) from the data in the source domain to the target domain by generating adversarial training.

To overcome the problems of weak generalization ability and adaptation to new tasks of the transferred models of hyperspectral target detection algorithms based on deep transfer learning and limited training samples for training deep neural networks, this article proposes a new meta-learning based hyperspectral target detection using Siamese network (MLSN). By introducing the idea of meta-learning [37], the meta-trained deep residual convolutional feature embedding (DRFE) module can learn how to discriminate similarities and differences between spectra and can be quickly adapted to hyperspectral target detection tasks. Moreover, the meta-training is performed on tasks constructed in the form of triples on a known labeled dataset, which solved the problem of limited training samples. Experiments on real hyperspectral datasets show that the proposed MLSN algorithm has achieved good comprehensive performance.

The remainder of this article is organized as follows. Section II gives a detailed description of the proposed MLSN method. In Section III, experimental results and analysis of six real HSIs are presented. Finally, the conclusions are drawn in Section IV.

II. Proposed Method

Most of the current deep learning models are typically trained from scratch for specific tasks. Adaptive methods based on deep learning have achieved great success in many fields. However, there are also limitations. For example, the successes are mainly in areas where large amounts of data can be collected or simulated and large amounts of computing resources can be used. When the data to be used are inherently small, and no large amounts of computing resources are available, or the computing resources are expensive, this kind of algorithm often fails to work [37]. It is hoped that the network does not require extensive retraining to be generalized to unfamiliar application tasks. Meta-learning has always been an effective and practical framework to solve such problems in deep learning [38]. Meta-learning methods are roughly divided into three categories [37]: model-based methods (or black box methods) [39], optimization-based methods [40], and metric-based methods (or nonparametric methods) [36], [41], [42].

Meta-learning is usually understood as learning to learn [37]. During basic learning, the internal learning

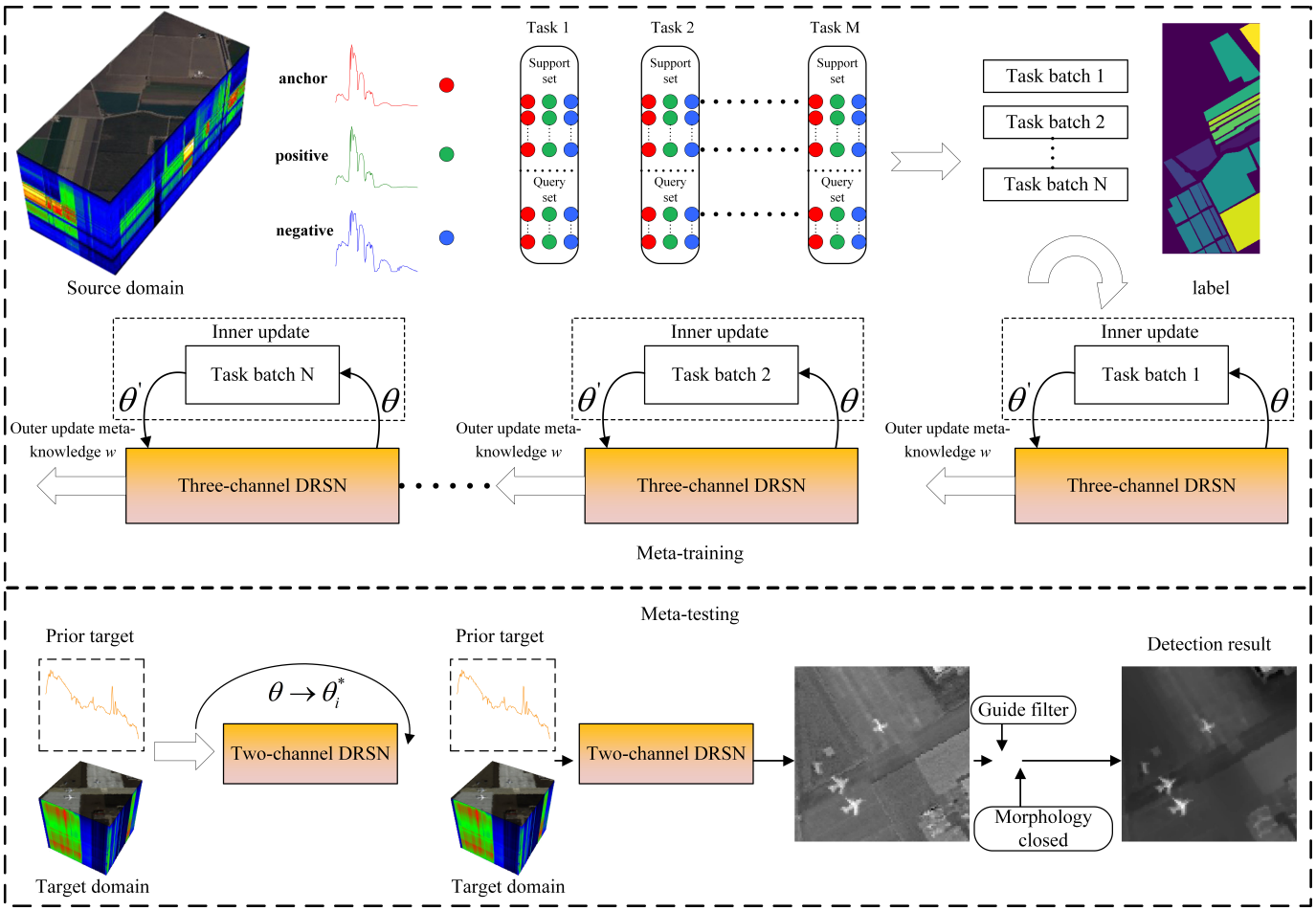

Fig. 1. MLSN algorithm block diagram.

algorithms can solve tasks defined by datasets and goals [43]. During the meta-learning period, the external algorithm will update the internal learning algorithm so that the learned model can improve the performance of the external algorithm. The performance of deep CNN-based hyperspectral target detection is often affected by the fact that the available training samples are often limited, resulting in performance degradation. Meta-learning can be used to solve the problem of lacking training samples in deep learning and applied to hyperspectral target detection. The flowchart of the proposed algorithm is shown in Fig. 1.

A. Deep Residual Convolution Feature Embedding Module

Through the DRFE module, each pixel spectrum is embedded into the Euclidean feature space. The formed feature vector has high-level semantic feature information for HSI target detection. The designed DRFE contains seven 1-D convolutional layers and 40 1×3 convolution kernels with a step size of 1. The pooling layer is replaced by using a 1-D convolution layer, which has 40 1×3 convolution kernels with a step size of 2, and residual connections are added between convolutional layers [44] with a better preservation of the gradient. As a result, it would be beneficial to extract more advanced semantic feature information and distinguish the similarities and differences between spectra if the network is deeper. The penultimate layer of the network uses a 1-D convolutional layer with a convolution kernel size of 1×1, a step size of 1, and the number of convolution kernels is 1, where the number of channels can be changed without changing the dimension of the spectrum. Finally, the final spectral embedding feature vector is obtained through a fully connected layer.

As shown in Fig. 2, the DRFE inputs a spectral vector with a dimension of $1 \times d$. In order to extract depth features for a discriminant learning of intraclass similarity and interclass dissimilarity, seven convolution layers and three pooling layers consisting of convolution layers with a step size of 2 are used to obtain richer spectral features of the original pixel spectrum. Moreover, the feature information can be retained more by adding a residual connection between the convolution layer and the pooling layer. Finally, the final feature embedding vector is obtained to discriminate the spectral difference through a 1×1 convolutional layer and a fully connected layer.

B. Triplet Loss

Embedding is represented by $f_\theta(\mathbf{x}) \in \mathbb{R}^d$, which embeds the pixel spectrum into a d-dimensional Euclidean feature space. Furthermore, the embedding is limited to exist on the d-dimensional hypersphere with the constraint of $\|f_\theta(\mathbf{x})\|_2 = 1$. In order to make the network learn to

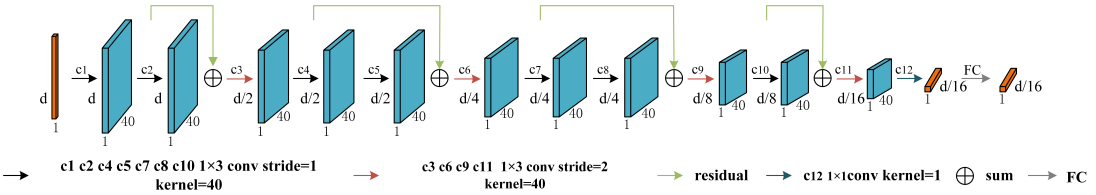

Fig. 2. Deep residual convolution feature embedding module (DRFE).

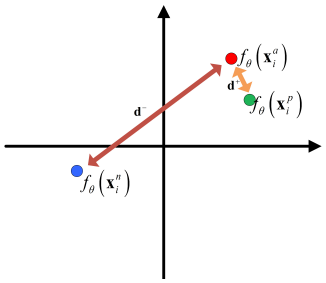

Fig. 3. Pixel spectra embedded in the Euclidean feature space schematic.

distinguish the similarity between the pixel spectra, it should be ensured that $\mathbf{x}_i^p$ (positive samples) from the same class as $\mathbf{x}_i^a$ (anchor) is as close to $\mathbf{x}_i^a$ as possible and $\mathbf{x}_i^n$ (negative samples) from a different class is as far away from $\mathbf{x}_i^a$ as possible in the Euclidean feature space, respectively. The distance between the anchor and a positive sample in the embedding feature space, representing the intraclass spectral similarity between pixels from the same class, is defined as follows:

$$d^+ = \left\| f_\theta\left(\mathbf{x}_i^a\right) - f_\theta\left(\mathbf{x}_i^p\right) \right\|_2^2. \tag{1}$$

The distance between the anchor and a negative sample in the embedding feature space, representing the interclass spectral dissimilarity between pixels from different classes, is as follows:

$$d^- = \left\| f_\theta\left(\mathbf{x}_i^a\right) - f_\theta\left(\mathbf{x}_i^n\right) \right\|_2^2. \tag{2}$$

The schematic of the embedding feature space is shown in Fig. 3. Triple loss [45], [46] encourages the positive samples to constantly close to the anchor and the negative samples to constantly move away from the anchor in the Euclidean feature space, respectively. When $d^- = d^+ + \text{margin}$, the value of the triplet loss is zero, which is expected. In other cases, the triplet loss would be a nonzero value, and it will be optimized and get closer and closer to zero through continuous training and learning. Therefore, in order to make the distance of spectral pixels from the same class as small as possible and the distance of spectral pixels from different classes as large as possible in the Euclidean feature space, this process can be formulated as

$$d^+ + \lambda < d^- \quad \forall\left(\mathbf{x}_i^a, \mathbf{x}_i^p, \mathbf{x}_i^n\right) \in D_{\text{source}} \tag{3}$$

where λ is a constant margin between positive and negative sample pair. $D_{\text{source}} = \left\{\left(x_1^a, x_1^p, x_1^n\right), \ldots, \left(x_N^a, x_N^p, x_N^n\right)\right\}$ is the set of all possible triples in the training set. The loss function minimized by training is

$$L = \sum_{i=1}^{N} \max\left\{0, \left|\mathbf{d}^+ + \lambda - \mathbf{d}^-\right|\right\}. \tag{4}$$

C. Meta-Training Three-Channel Deep Residual Convolution Siamese Network (Three-Channel DRSN)

The source domain HSI $\mathbf{P}_s \in \mathbb{R}^{H_s \times W_s \times B_s}$ has C classes of spectral pixels, and the triplet is used as the form of the training set in the meta-training process. There are many known labeled HSI datasets by different sensors, and meta-training chooses the known labeled HSI captured by the same sensor as the HSI to be detected for training. In the meta-training stage, five classes are randomly selected from the source domain HSI with four spectra from each class to constitute a task $T_{\text{source}} = \left\{D_{\text{source}}^{\text{tr}}, D_{\text{source}}^{\text{te}}\right\}$ in the form of triplets. The support set $D_{\text{source}}^{\text{tr}}$ is formed by the above randomly selected five classes with two spectral pixels for each class. Since each class has two samples, if one sample is selected as an anchor, then the other one is set as the positive sample, and the negative sample should be selected from the other classes to form a triplet. A lot of triplet sets could be formed in this way in the support set $D_{\text{source}}^{\text{tr}}$. The query set $D_{\text{source}}^{\text{te}}$ is constructed in the same way as the support set, but should meet the condition that $D_{\text{source}}^{\text{tr}} \cap D_{\text{source}}^{\text{te}} = \emptyset$. For traditional supervised deep learning, the training process can train a prediction model $\hat{y} = f_\theta(\mathbf{x})$ with parameter θ by solving the equation

$$\theta^* = \arg\min_\theta \sum_{i=1}^{N} L\left(f_\theta\left(\mathbf{x}_i^a\right), f_\theta\left(\mathbf{x}_i^p\right), f_\theta\left(\mathbf{x}_i^n\right); \theta\right) \tag{5}$$

where L is the triplet loss function shown in (4). In the meta-learning process, training samples are obtained from many different tasks in order to learn that a general learning algorithm can generalize to various tasks, and ideally, each new task is better than the previous one. Specifically, the tasks are extracted from the distribution $p(T)$, and M tasks $D_{\text{source}} = \left\{T_{\text{source}}^i\right\}_{i=1}^M$ are constructed. The meta-goal is to find the public parameters that can be applied to multiple tasks, named meta-knowledge. The process can be formulated as

$$w^* = \arg\min_w \sum_{i=1}^{M} L_i\left(T_{\text{source}}^i; w\right). \tag{6}$$

In the training stage, most of the mainstream meta-learning algorithms optimize the meta-parameter w based on gradient descent. In order to solve the meta-training problem in (6), the meta-training steps are usually transformed into a bilevel optimization problem [38]. Bilevel optimization refers to a hierarchical optimization problem, in which one optimization is constrained by the other optimization [37]. Then, the meta-training can be formalized as

follows:

$$w^* = \arg\min_{w} \sum_{i=1}^{M} L\left(T^i_{\text{source}}; \theta_i^*(w), w\right) \quad \text{s.t. } \theta_i^*(w) = \arg\min_{\theta} L_i\left(T^i_{\text{source}}; \theta, w\right) \quad \forall i. \tag{7}$$

Internally gradient updates are performed for each task T^i_{source} by

$$\theta_i' = \theta - \alpha \nabla_\theta L_i\left(D^{\text{tr}(i)}_{\text{source}}, \theta_i, w\right). \tag{8}$$

Then, externally update the meta-knowledge w by

$$w = w - \beta \nabla_w \sum_i L\left(D^{\text{te}(i)}_{\text{source}}, \theta_i', w\right). \tag{9}$$

The internal optimization only uses the support set of each task, while the external optimization uses the query set of each task and is performed through the loss obtained on a batch of tasks. The stochastic gradient descent (SGD) is used to train the three-channel deep residual convolution Siamese network (DRSN), where parameters α and β are the learning rate. The rectified linear units are used as the nonlinear activation function in the model. The meta-training process is shown in Algorithm 1.

Algorithm 1 Meta-Training Process

Input: source dataset $D_{source} \sim p(T)$, learning rate α, β, three-channel DRSN f_θ, number of iterations I

Output: good initialize parameters θ, which is meta- knowledge w

1: randomly initialize θ
2: **for** $i \le I$ **do**
3: sample M tasks from $p(T)$ as a batch, each task

$$T^k_{source} = \left\{D^{tr(k)}_{source}, D^{te(k)}_{source}\right\} \text{ and } D^{tr(k)}_{source} \cap D^{te(k)}_{source} = \emptyset$$

4: **for** $j \le M$ **do**
5: use $D^{tr(j)}_{source}$ in task T^j_{source} to calculate

$$\nabla_\theta L_j\left(D^{tr(j)}_{source}, \theta_j, w\right)$$

6: update parameters using gradient descent:

$$\theta_j' = \theta - \alpha \nabla_\theta L_j\left(D^{tr(j)}_{source}, \theta_j, w\right)$$

7: $j = j + 1$
8: **end for**
9: use the loss generated by $D^{te(k)}_{source}$ of each task in a batch and update the meta-knowledge:

$$w = w - \beta \nabla_w \sum_j L\left(D^{te(i)}_{source}, \theta_j', w\right)$$

10: $i = i + 1$
11: **end for**

The three-channel DRSN is shown in Fig. 4. The intraclass similarity corresponds to (1) and the interclass dissimilarity corresponds to (2), and the discriminate module corresponds to (4). By conducting meta-training on the constructed tasks, the network will make the anchor continuously closer to the positive sample and further away from the negative sample in the Euclidean feature space. Through continuous training, the network is expected to learn a shared representation from various tasks and, finally, learn to distinguish the intraclass similarities and interclass dissimilarities of spectra. Once the network learns to distinguish the similarities and dissimilarities between spectra, it can be applied to HSI target detection using the learned meta-knowledge. It should be noted that, in the three-channel DRSN, the DRFE module in each channel has the same structure and model parameters.

D. Meta-Testing Target Detection Process

The HSI to be detected is represented as $\mathbf{P}_t \in \mathbb{R}^{H_t \times W_t \times B_t}$ with its prior target spectrum $\mathbf{x}^t_*$, and the spectral vector of each pixel is represented as $\mathbf{P}_t = \left\{\mathbf{x}^t_i\right\}^{H_t \times W_t}_{i=1}$. Before target detection, the meta-knowledge of the DRFE module is updated using the prior target spectrum $\mathbf{x}^t_*$. The upper branch and lower branch channels are fed into the prior target spectrum and $\mathbf{P}_t$, respectively, and the loss function of the meta-knowledge updated two-channel DRSN is

$$L_f = \frac{1}{N} \sum_{i=1}^{N} \left\| \frac{1}{\pi} \arccos \frac{f_\theta\left(\mathbf{x}^t_*\right)^T \cdot f_\theta\left(\mathbf{x}^t_i\right)}{\left\| f_\theta\left(\mathbf{x}^t_*\right)\right\|_2 \cdot \left\| f_\theta\left(\mathbf{x}^t_i\right)\right\|_2} - 1 \right\|^2_2. \tag{10}$$

In the meta-testing stage, the learned meta-knowledge w^* is updated to obtain the best parameters of the model for a given detection task (such as task i). This process can be formulated as follows:

$$\theta_i^* = \arg\min_{\theta} L_{fi}\left(\mathbf{x}^t_*, \theta | w^*\right). \tag{11}$$

After updating the meta-knowledge, the parameters for a specific detection task are changed from θ to θ_i^* to adapt to the corresponding detection task. It should be noted that the parameter updating process only updates those of the last fully connected layer in the DRFE module, and the parameters of the convolutional layer are frozen without updating. Through the meta-knowledge update, it is hoped that the DRFE module will increase the difference between the *a priori* target spectrum and the background pixel spectral embedding feature vector.

In the meta-testing stage, the two-channel DRSN is used for target detection. The DRFE module of the two channels is learned through meta-training and then updated through meta-knowledge. Using the learned meta-knowledge, the two-channel DRSN will discriminate the spectral similarity and dissimilarity between pixels. Similar to the target spectrum means it is likely to be a target, while dissimilar from the target spectrum means that it is likely to be a background pixel. The structure of the two-channel DRSN for target detection is shown in Fig. 5. The prior target spectrum is inputted into the upper branch channel, and the pixel spectrum of the HSI under test is inputted into the lower branch channel. The corresponding embedding feature vectors are obtained through the upper and lower channels, respectively, and cosine similarity is used to judge the similarity between the two embedding feature vectors, shown as follows:

$$k_i = \frac{f_{\theta_i^*}\left(\mathbf{x}^t_*\right)^T \cdot f_{\theta_i^*}\left(\mathbf{x}^t_i\right)}{\left\| f_{\theta_i^*}\left(\mathbf{x}^t_*\right)\right\|_2 \cdot \left\| f_{\theta_i^*}\left(\mathbf{x}^t_i\right)\right\|_2}. \tag{12}$$

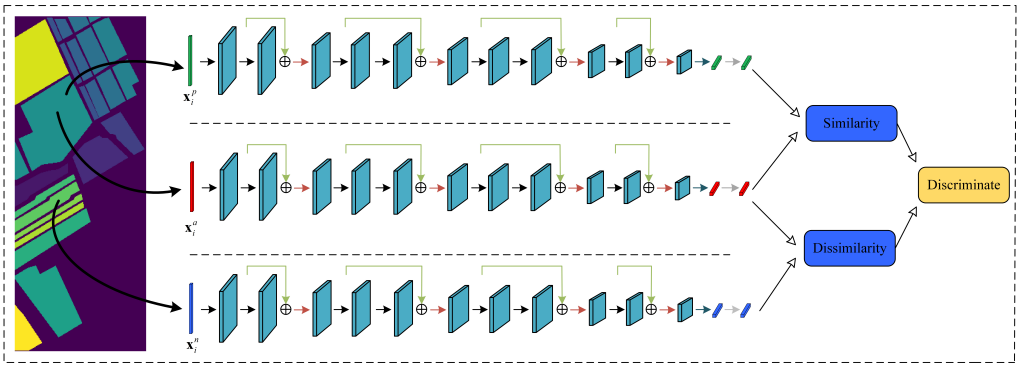

Fig. 4. Meta-training three-channel deep residual convolution Siamese network (three-channel DRSN) structure.

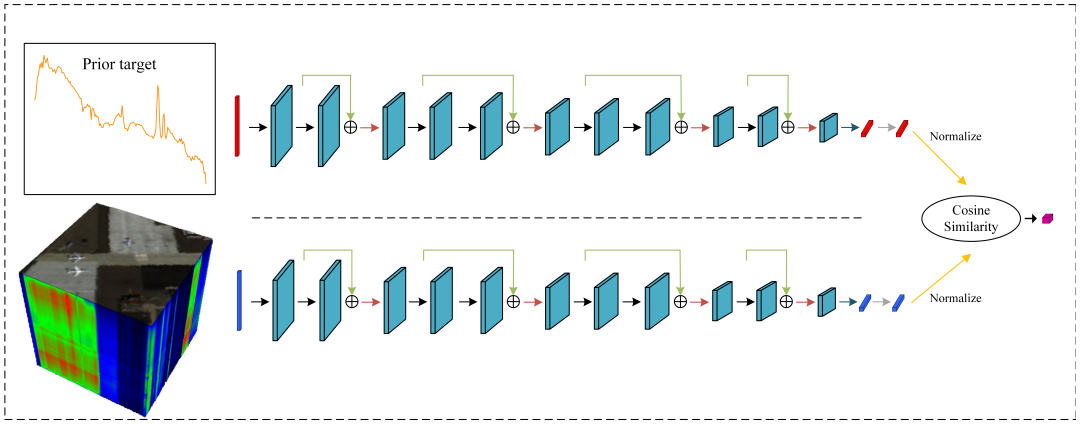

Fig. 5. Meta-testing two-channel deep residual convolution Siamese network (two-channel DRSN) structure.

The closer the value is to 1, the more similar the two feature vectors in the Euclidean feature space, and the more likely that $\mathbf{x}_i^t$ is a target. $\mathbf{K} = \{k_i\}_{i=1}^{H_t \times W_t}$ is the final detection map outputted by the two-channel DRSN.

E. Joint Spatial Information for Target Detection

The HSI is a 3-D cube with both spectral information and spatial information, but the two-channel DRSN detects targets only by the spectral features, while the spatial information is ignored. Using the spatial information to modify the detection map obtained by the two-channel DRSN could further improve the detection accuracy.

The guided image filtering [47] is chosen in this article to make use of the spatial information, and it is performed on the detection map $\mathbf{K}$ obtained in Section II-D. The guiding image is obtained by the principal component analysis (PCA) performed on the detected HSI $\mathbf{P}_t$, and the first principal component is selected as the guiding image. A general linear shift variable filtering process is defined, including a guiding, an input image $\mathbf{K}$, and an output image $\mathbf{Q}$ of the joint spatial–spectral detection result. The filtering output of pixel i is represented as a weighted average, shown as follows:

$$\mathbf{Q}_i = \sum_j \mathbf{W}_{ij}(\mathbf{I})\mathbf{K}_j \tag{13}$$

and the filter kernel weight $\mathbf{W}_{ij}(\mathbf{I})$ can be expressed as

$$\mathbf{W}_{ij}(\mathbf{I}) = \frac{1}{|e|^2} \sum_{k:(i,j)\in e_k} \left(1 + \frac{(\mathbf{I}_i - \mu_k)(\mathbf{I}_j - \mu_k)}{\sigma_k^2 + \varepsilon}\right) \tag{14}$$

where e_k is the window centered at the kth pixel, the window size is $(2r+1) \times (2r+1)$ (r is the radius of the window), and μ_k and σ_k^2 are the mean and variance of the guiding, respectively. $|e|$ is the number of pixels in e_k, and ε is a penalty value. $\mathbf{I}_i$ and $\mathbf{I}_j$ refer to the values of two adjacent pixels in the guiding image. After the guided image filter, the detection map $\mathbf{K}$ can be smoothed, and the boundary of the target region can be maintained.

Finally, the morphological closing operation is performed on the detection map $\mathbf{Q}$ after the guided image filtering, with dilation and erosion in sequence, connecting the discontinuous regions of the target. It can be formulated as follows:

$$\mathbf{Q}_{\text{final}} = \mathbf{Q} \cdot \mathbf{B} = (\mathbf{Q} \oplus \mathbf{B})\Theta\mathbf{B} \tag{15}$$

where $\mathbf{Q}_{\text{final}}$ is the final joint spatial–spectral detection result, $\mathbf{B}$ is a matrix of 2∗2 with all 1 elements, $\oplus$ represents the dilation operation, and Θ represents the erosion operation.

III. Experimental Results and Analysis

A. Meta-Training Hyperspectral Dataset

1) Salinas Dataset: The Salinas dataset was collected by the airborne visible light infrared imaging spectrometer (AVIRIS) sensor over the Salinas Valley in California, USA, with 224 bands. The spatial resolution is 3.7 m with the original image size of 512×217pixels. There are a total of 16 classes in the image scene, including vegetables, bare soil, vineyards, and so on. The pseudocolor image and ground truth are shown in Fig. 6(a) and (b).

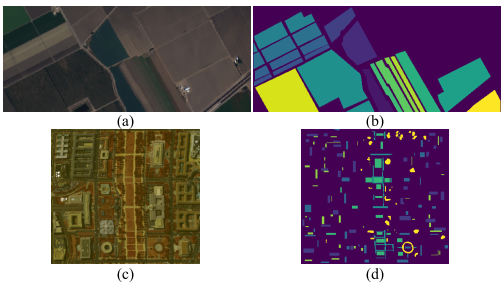

Fig. 6. Meta-training hyperspectral datasets. (a) Salinas pseudocolor image. (b) Salinas ground truth. (c) Washington DC pseudocolor image. (d) Washington DC ground truth.

2) Washington DC Dataset: The Washington DC dataset is an HSI acquired by an airborne sensor of the hyperspectral digital image collection experiment (HYDICE). The image contains 210 bands in the visible and near-infrared ranges from 400 to 2500 nm, and the original size is 1208×307. A portion of the original image with a size of 208×307 pixels is used in this article for experiments. There are a total of nine classes, including roofs, streets, gravel roads, grasslands, numbers, water, shadows, and so on. The pseudocolor image and ground truth are shown in Fig. 6(c) and (d).

B. Meta-Testing Hyperspectral Dataset

1) HYDICE Dataset: The HYDICE dataset was collected by the HYDICE sensor in an urban area in California, USA. The spatial size of the original HSI is 307×307. It has 210 bands, with a wavelength range of 400–2500 nm, and the spectral resolution is 10 nm. A portion of the original HSI image with a size of 80×100 pixels is used in this article for target detection, and each pixel corresponds to a region of $2 \times 2\text{m}^2$. Due to the influence of dense water vapor and atmosphere, bands 1–4, 76, 87, 101–11, 136–153, and 198–210 are removed, and 162 bands are retained for hyperspectral target detection. The roof and the car in this HSI dataset are the targets, and there are a total of 21 target pixels, as shown in Fig. 7(a).

2) San Diego Dataset: The dataset of San Diego was captured by the AVIRIS of the San Diego airport area in California, USA. The original image size is 400×400, the spatial resolution of the image is 3.5 m, and the spectral resolution is 10 nm. In the experiment, a portion of the original San Diego data with a size of 100×100 is named AVIRIS1, and another portion with a size of 120×120 is named AVIRIS2, corresponding to Fig. 7(b) and (c), respectively. After removing the low signal-to-noise ratio and water absorption bands (1–6, 33–35, 97, 107–113, 153–166, and 221–224), 189 bands remain, with a wavelength range of 400–2500 nm. The targets are all airplanes. There are 134 target pixels in AVIRIS1 and 58 target pixels in AVIRIS2.

3) El Segundo Dataset: The El Segundo dataset was captured by the AVIRIS sensor in the El Segundo area of California, USA. The wavelength range is 400–2500 nm, the spatial resolution of each pixel is 7.1 m, there are 224 bands in total, the original size of the image is 250×300, and a size of 100×100 is intercepted in the experiment. Target is the facilities of the refinery, such as oil storage tanks and towers, with a total of 715 target pixels, as shown in Fig. 7(d).

4) Beach and Urban Datasets: The Beach and Urban datasets are captured by AVIRIS sensors. The sizes of the Beach and Urban are $90 \times 90 \times 188$ and $100 \times 100 \times 204$ after discarding noisy bands, respectively. The Beach dataset was captured over Cat Island, and the spatial resolution of each pixel is 17.2 m. The Urban dataset was captured over the Texas coast, USA, and the spatial resolution is 17.2 m per pixel. Target pixels in the beach and urban datasets are 19 and 67, respectively. The pseudocolor image and reference detection map of these two datasets are shown in Fig. 7(e) and (f).

C. Evaluation Indicators

In order to study the performance of the proposed MLSN algorithm, the receiver operating characteristic (ROC) curve, the area under the curve (AUC) value, and the target background separability box plot are used to measure the performance of the algorithms.

The ROC curve has been widely used in the evaluation of target detection performance in hyperspectral remote sensing images. After the detection result is obtained, the target detection probability and false alarm probability are calculated through a given detection threshold, and different detection probabilities and false alarm probabilities are obtained by changing the threshold, so as to obtain the ROC curve for quantitative analysis of detectability. The calculation formula of detection probability and false alarm probability is

$$P_d = \frac{N_d}{N_t}, \quad P_f = \frac{N_f}{N_{\text{total}}} \tag{16}$$

where N_d is the number of target pixels under a certain threshold, N_t is the total number of target pixels in the real HSI ground truth, N_f is the number of background pixels that are falsely detected as target pixels, and N_{total} is the total number of pixels in the HSI to be detected.

The value of AUC is the size of the area under the ROC curve. For the traditional ROC curve of detection probability and false alarm probability, the value of AUC is between 0.5 and 1, and its value can be quantified as the evaluation index of algorithm accuracy. In the case of AUC $>$ 0.5, the closer the value is to 1, the better performance of the target detection algorithm.

The box plot of target-background separability can measure the degree of separation between target and background. According to the label of the HSI to be detected, the corresponding values of target and background are taken out in the detection result, and the box plot of target and background is drawn. Red represents the target, and green represents the background. The larger the interval between the red box and the green box is, the narrower the green box is, which indicates that the target is separated from the background, and the background is suppressed well.

D. Experimental Setup

The experiment is divided into two parts: meta-training and meta-testing. The three-channel DRSN is meta-trained

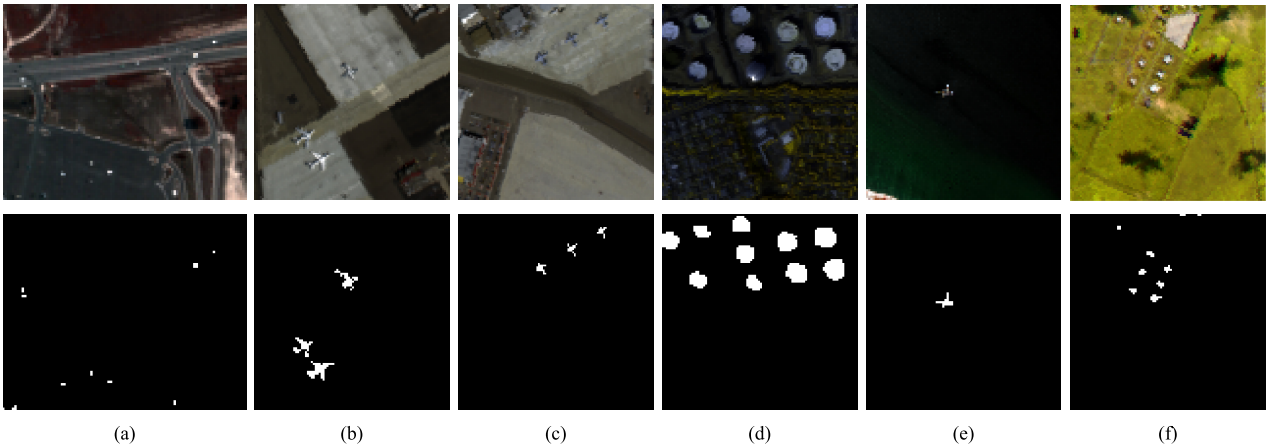

Fig. 7. Meta-testing hyperspectral datasets. (a) HYDICE Dataset. (b) AVIRIS1. (c) AVIRIS2. (d) El Segundo Data set. (e) Beach. (f) Urban.

through the training HSI datasets with known labels and sensor types. In the meta-testing stage, the learned meta-knowledge is updated in the two-channel DRSN using *a priori* target spectrum so that the updated meta-knowledge can quickly adapt and be more suitable for the target detection task of the HSI scene to be detected.

In the process of meta-training, the pixel spectrum from the same type of sensor is randomly extracted in the way of five-way-two-shot to form tasks in the form of a triplet. Each task is constructed into a task batch according to the batch size. The hyperparameters of the meta-training process include the batch size, the inner learning rate α, the outer learning rate β, the margin, and the number of iterations, where, in our experiments, the batch size is set to 4, the inner learning rate is set to 10^{-2}, the outer learning rate is set to 10^{-3}, the margin is set to 1, and the number of iterations is set to 60 000, while, for the meta-testing process that is a task of hyperspectral target detection, one HYDICE sensor dataset and five AVIRIS sensor datasets are used to test the target detection performance of the proposed MLSN algorithm. A two-channel DRSN is used to update the learned meta-knowledge w^* using *a priori* target spectrum to adapt to the new target detection tasks quickly. The hyperparameters' epoch, batch size, and learning rate are set to 5, 1000, and 0.001, respectively, for the meta-knowledge update process. The upper branch channel inputs the prior target spectrum in the HSI to be detected, and the lower branch channel inputs the spectrum of each pixel in the same HSI. By comparing with the embedding feature vector of the prior target spectrum in the upper branch channel, the similarity map between each pixel and the prior target is obtained. The local window radius r of guided image filtering corresponds to the HSI dataset to be detected in Fig. 7 and is set to 2, 8, 2, 8, 4, and 2, respectively. The penalty value of guided image filtering is set to 0.04 for all datasets in the experiments.

In order to investigate the performance of MLSN, three traditional target detection algorithms and two deep learning target detection algorithms are used for comparison. Three traditional algorithms include ACE, CEM, and CSCR, and on the other hand, two deep learning-based methods are CNNTD and BLTSC, respectively.

The parameters of the algorithms are set as follows. For the CSCR algorithm, the sizes of the double window (w_{out}, w_{in}) are set to (11,3) for Fig. 7(a), (c), (e), and (f), while they are set to (11,5) for Fig. 7(b) and (d). For the CNNTD algorithm, when detecting the data of Fig. 7(a) captured by the HYDICE sensor, the Washington DC dataset of Fig. 6(c) captured by the same sensor is used for pretraining the model, and the knowledge learned in the transfer is used for target detection; when detecting the remaining hyperspectral datasets captured by the AVIRIS sensor, the CNNTD method uses the Salinas hyperspectral dataset collected by the same sensor, as shown in Fig. 6(a), to train the model and uses the trained model to detect the AVIRIS hyperspectral datasets. The learning rate is 1e-3, the batch size is 256, and the epoch is 50. For the BLTSC algorithm using a GAN, the coarse detection uses a classic CEM detector, the learning rate is set to 1e-4, and the epoch is set to 500. The experimental hardware configuration is Intel Core i7-10875h eight-core CPU, NVIDIA GeForce RTX 2080 GPU. The three traditional algorithms (ACE, CEM, and CSCR) are implemented on MATLAB 2017b platform. The deep learning-based CNNTD and the proposed MLSN methods are implemented using Python 3.8.3 and PyTorch version 1.60. The BLTSC algorithm uses MATLAB 2017b to implement coarse detection and background search, and is then implemented using Python 3.6.2 and TensorFlow version 1.80 for AAE to reconstruct the background with suppressed targets.

E. Results and Analysis

The above six algorithms (including three traditional algorithms of ACE, CEM, and CSCR, two state-of-the-art deep learning-based algorithms, CNNTD and BLTSC, and the proposed MLSN) are conducted on six hyperspectral datasets, and the detection results are shown in Figs. 8–13.

For these six hyperspectral datasets, the two classic target detectors, ACE and CEM, can detect most of the target pixels, but there are many false detected pixels, where some target pixels are falsely detected as the background, and some background pixels are falsely detected as the target. The CSCR detector can detect most targets, but the background suppression is not good, and the targets are not significantly

Fig. 8. HYDICE Dataset detection map. (a) ACE. (b) CEM. (c) CSCR. (d) CNNTD. (e) BLTSC. (f) MLSN.

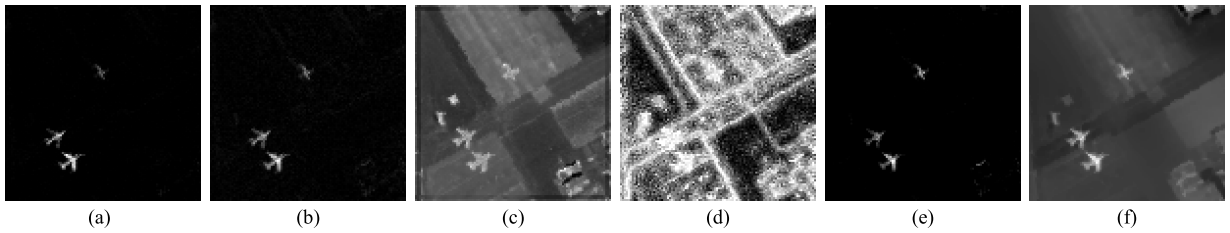

Fig. 9. AVIRIS1 Dataset detection map. (a) ACE. (b) CEM. (c)CSCR. (d) CNNTD. (e) BLTSC. (f) MLSN.

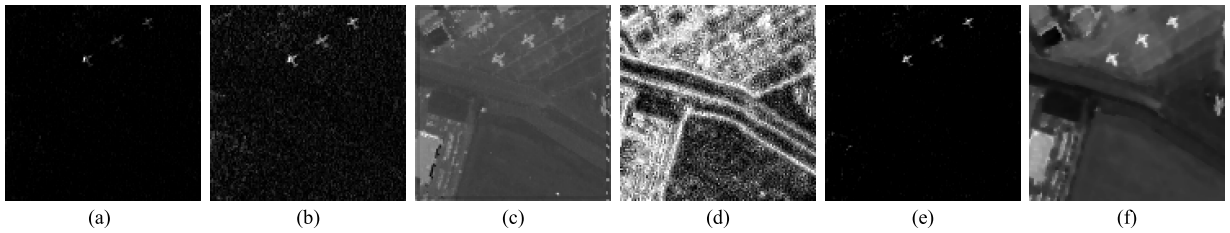

Fig. 10. AVIRIS2 Dataset detection map. (a) ACE. (b) CEM. (c) CSCR. (d) CNNTD. (e) BLTSC. (f) MLSN.

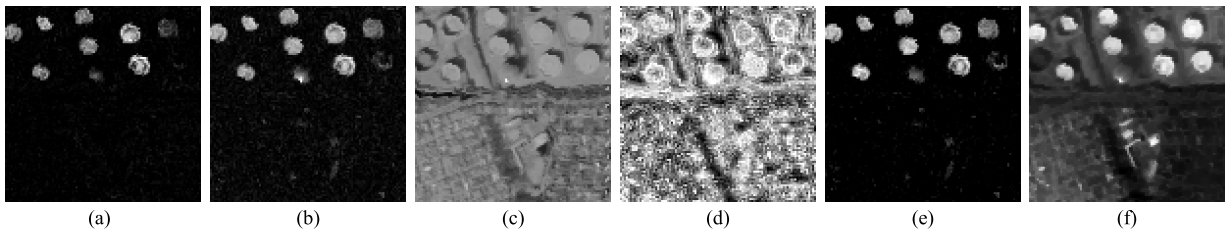

Fig. 11. El Segundo Dataset detection map. (a) ACE. (b) CEM. (c) CSCR. (d) CNNTD. (e) BLTSC. (f) MLSN.

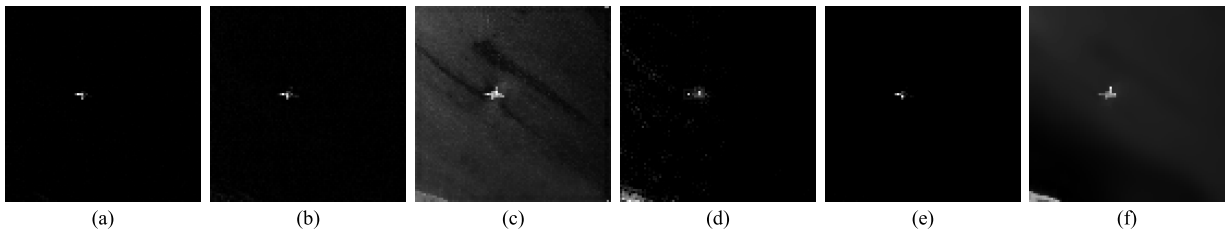

Fig. 12. Beach Dataset detection map. (a) ACE. (b) CEM. (c) CSCR. (d) CNNTD. (e) BLTSC. (f) MLSN.

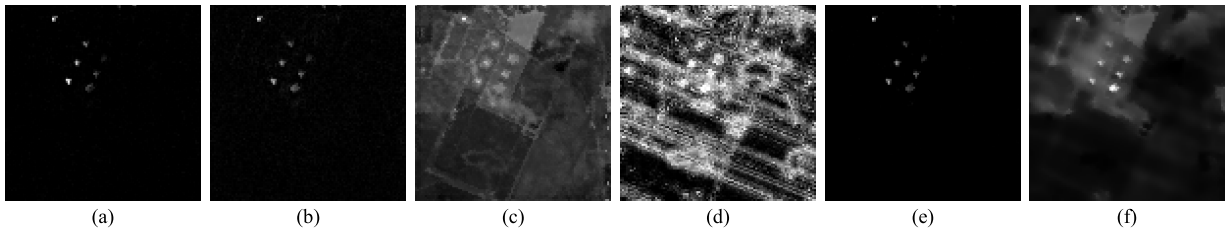

Fig. 13. Urban Dataset detection map. (a) ACE. (b) CEM. (c) CSCR. (d) CNNTD. (e) BLTSC. (f) MLSN.

separated from the background. The detection map obtained by the CNNTD detector is the worst, where a large number of background pixels are falsely detected as the targets, and the targets are not separated from the background. For the BLTSC detector, some reliable background pixels are found from the coarse detection results of the CEM detector. The reliable background pixels are then used as training samples to train AAE. The trained network will reconstruct the background of the HSI to be detected. The distance weight map is obtained by comparing the reconstructed background with the original

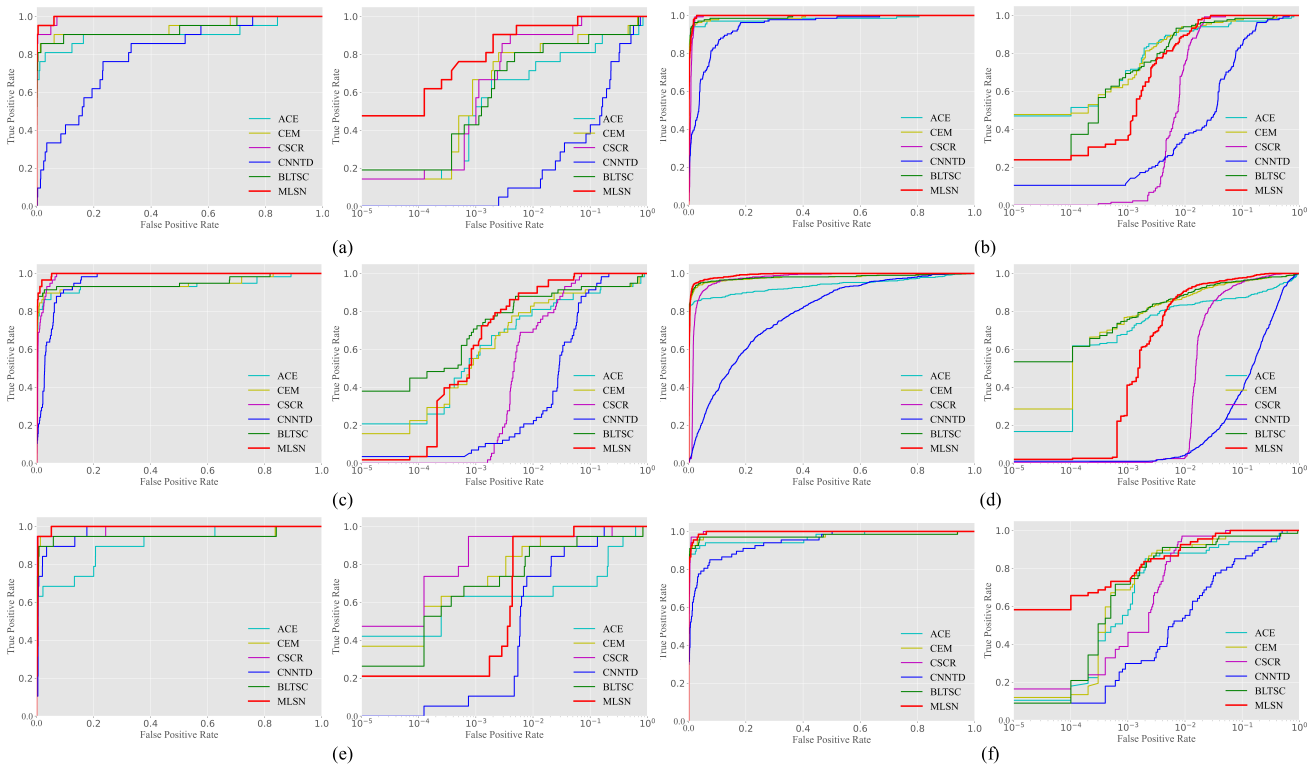

Fig. 14. ROC curve of the six target detectors for different HSI datasets. (a) HYDICE. (b) AVIRIS1. (c) AVIRIS2. (d) El Segundo. (e) Beach. (f) Urban.

TABLE I
AUC VALUES FOR THE COMPARED METHODS ON DIFFERENT DATASETS

Method	HYDICE	AVIRIS1	AVIRIS2	EI Segundo	Beach	Urban
ACE	0.90952	0.98153	0.93897	0.93561	0.90670	0.96909
CEM	0.94157	0.99078	0.94566	0.97851	0.95440	0.97633
CSCR	0.99314	0.99168	0.98753	0.97042	0.98706	0.99668
CNNTD	0.80182	0.94355	0.95803	0.78767	0.97585	0.94645
BLTSC	0.93666	0.99195	0.95115	0.98112	0.95120	0.97631
MLSN	**0.99654**	**0.99667**	**0.99621**	**0.99189**	**0.99452**	**0.99691**

HSI pixel by pixel. It can be seen from the detection map that BLTSC has corrected most of the background pixels falsely detected as target pixels in the CEM detector. However, the quality of background reconstruction depends entirely on the performance of the CEM detector. When the coarse filter detector is not good, it will have a significant impact on the detection performance. The detection map obtained by the proposed MLSN algorithm has the most significant difference between targets and background pixels, and the edge shape of the target remains the best.

To assess different algorithms quantitatively, the ROC curves are plotted for comparison. Fig. 14 shows the ROC curves of different algorithms using different HSI datasets, where each subfigure includes all algorithms for the same HSI data. In order to compare the minor performance difference, each subfigure provides the original ROC curve on the left and a zoom-in ROC curve with a range of 0–0.1 on the right of each subfigure with better observation. For the HYDICE dataset, the ROC curve of the proposed MLSN detector has always been above the curves of other detectors with the best performance. For the AVIRIS1, AVIRIS2, El Segundo, and Beach datasets, although the proposed MLSN method is sometimes lower than the detection probability of other detectors in the zoom-in curves, the proposed MLSN method is the first to reach 1 from the overall ROC curve. For the Urban dataset, the ROC curve of the proposed MLSN detector first rises between 0 and 0.1 with a higher detection probability and then slightly lower than the ROC curve of the CSCR algorithm.

However, it can be seen from the AUC values that the performance of the proposed MLSN detector is still better. The AUC values of the six methods for the six hyperspectral datasets are shown in Table I. For each hyperspectral dataset, the maximum AUC value detected is highlighted in the form of coarsening, and the AUC values of the proposed MLSN algorithm are always the highest among the six algorithms. The classical ACE detector has a similar detection performance to the CEM detector, but the AUC values of CEM are always higher than those of the ACE algorithm. The CSCR detector has better detection accuracy for each dataset, but the accuracy is lower than that of the proposed MLSN

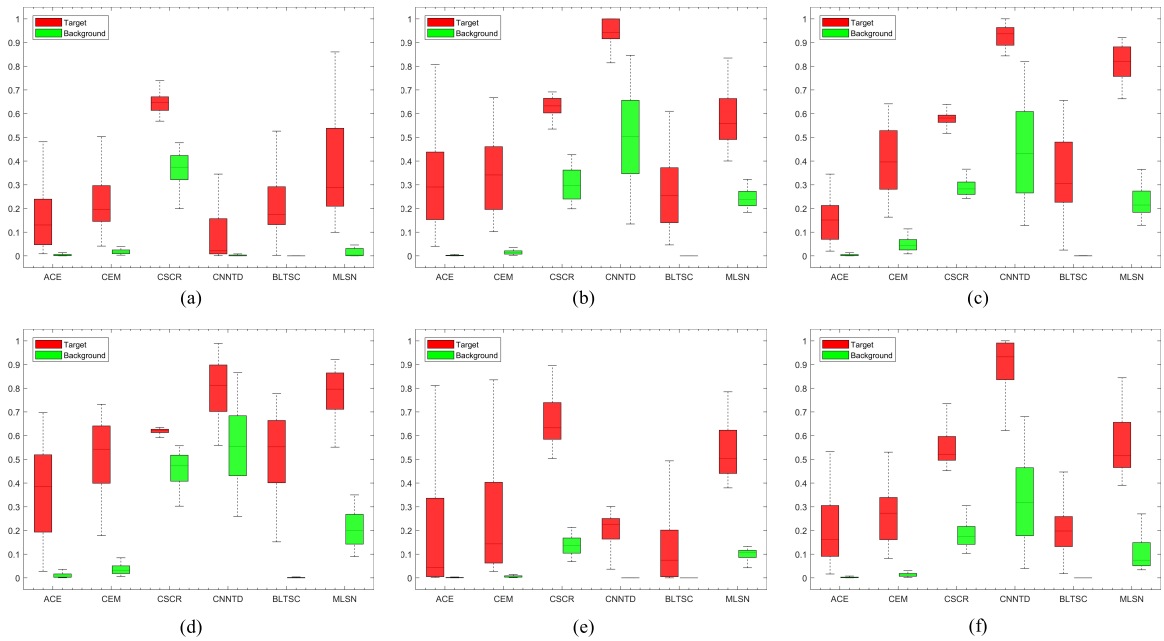

Fig. 15. Box plot of the separability of the target and background. (a) HYDICE. (b) AVIRIS1. (c) AVIRIS2. (d) El Segundo. (e) Beach. (f) Urban.

method, and the target of the CSCR detector is not obviously separated from the background even though the AUC performs well with high values. The CNNTD method based on transfer learning has the worst detection performance in most cases. This might be because the CNNTD algorithm increases the training samples using the subtractions of spectral pixels, losing the spectral details between pixels, and then leading to a decrease in the detection performance. The AUC values of the BLTSC algorithm will be significantly improved in some hyperspectral datasets compared with CEM because it utilizes the CEM detector for coarse filtering detection, then selects reliable background samples to generate adversarial training and reconstruct the background, and adds suppression targets in generating adversarial training to expect relatively pure background. If the coarse detection method is good, the selected background samples will be more reliable. The reconstructed background obtained by AAE and the original HSI to be detected is used to calculate the SAM pixel by pixel. The obtained distance weight map will be with small values of the background and large values of the target pixels. Finally, it is used to correct the detection map of the coarse detector and correct the pixels in the CEM detector that the background pixels are falsely detected as the targets so that the detection accuracy can be improved. However, it is greatly affected by the coarse filter detection method. If the background sample is not well selected, the AUC value will decrease. For example, for the HYDICE, Beach, and Urban datasets, the AUC values of the BLTSC detector are lower than the coarse detection CEM detector. The proposed MLSN method does not rely on searching for pure background pixels. It maintains a good edge on the targets with high detection accuracy and can quickly adapt to new detection tasks. It conducts meta-training on the source domain hyperspectral datasets in the form of a three-channel DRSN. The network learns to distinguish the difference between spectra and updates the learned meta-knowledge with the prior target spectrum so that the learned meta-knowledge can quickly and better adapt to the HSI to be detected. Each pixel in the HSI under test and the prior target spectrum are discriminated by embedding into the Euclidean feature space, and the optimal detection result is obtained.

Finally, in order to compare the separability between the target and the background of different algorithms, the box plot of the target background separability is used for a comparable analysis. The box plots of the target background separability for different algorithms on different datasets are shown in Fig. 15. The red box indicates the distribution range of the target, the green box indicates the distribution range of the background, and the interval between the red box and the green box indicates the degree of separation between the target and the background. It would be better if the red box and the green box have fewer overlapping regions and are far away from each other, which means that the targets and the background are better separated. It is obviously seen that the proposed MLSN algorithm can best separate the target from the background for all six datasets, which further proves the superior comprehensive performance of the proposed MLSN algorithm.

IV. Conclusion

The generalization ability of the transferred model and its adaptation to new tasks, as well as limited training samples for deep neural networks, limit the hyperspectral target detection algorithm based on deep transfer learning. In order to address these issues, a method of MLSN is proposed. By introducing the meta-learning method, the three-channel DRSN structure is designed, and the triplet loss is used for meta-training on the known label dataset. A similar spectral distance in the Euclidean feature space is learned. The DRFE module with the same parameters and structure is then used to form a

two-channel DRSN, and the prior target spectrum is used to update the meta-knowledge so that the model can quickly and better adapt to the new target detection task. The prior target spectrum is inputted into the upper branch channel, and each pixel spectrum of the HSI to be detected is inputted into the lower branch channel. The pixel spectrum is embedded into the Euclidean feature space through the DRFE module, and the spectral similarity is calculated by cosine similarity to obtain the detection map. Since HSI contain abundant spatial information, guided image filtering is used to maintain the edge of the target and smooth the background. Finally, the morphological closing operation is used to connect the target region to obtain the final joint spectral–spatial detection result. The experimental results show that MLSN has a good comprehensive performance.

The main contributions of this article are given as follows.

1) A deep residual convolution feature embedding module (DRFE) is designed to embed the spectrum of pixels into the Euclidean feature space, and it is then used to construct a three-channel deep residual convolution Siamese network (three-channel DRSN) for meta-training so that the DRFE module has the ability to discriminate spectral similarities and differences and then forms a two-channel deep residual convolution Siamese network (two-channel DRSN) for meta-testing so that the meta-knowledge of the DRFE module is suitable for the detection task and further increases the discrimination ability of target and background spectra.
2) The idea of meta-learning is first introduced to HSI target detection so that the DRFE module obtained by meta-training can be generalized to unfamiliar application scenarios without extensive retraining and has better generalization capability when predicting data for which almost no supervised information is available. A good comprehensive performance is obtained by testing on several datasets collected by different sensors.

REFERENCES

[1] M. T. Eismann, A. D. Stocker, and N. M. Nasrabadi, "Automated hyperspectral cueing for civilian search and rescue," *Proc. IEEE*, vol. 97, no. 6, pp. 1031–1055, Jun. 2009.

[2] K. Sendin, P. J. Williams, and M. Manley, "Near infrared hyperspectral imaging in quality and safety evaluation of cereals," *Crit. Rev. Food Sci. Nutrition*, vol. 58, no. 4, pp. 575–590, Mar. 2018.

[3] B. Lu, P. Dao, J. Liu, Y. He, and J. Shang, "Recent advances of hyperspectral imaging technology and applications in agriculture," *Remote Sens.*, vol. 12, no. 16, p. 2659, Aug. 2020.

[4] B. Zhang, W. Yang, L. Gao, and D. Chen, "Real-time target detection in hyperspectral images based on spatial–spectral information extraction," *EURASIP J. Adv. Signal Process.*, vol. 2012, no. 1, p. 142, Dec. 2012.

[5] X. Kang, X. Zhang, S. Li, K. Li, J. Li, and J. A. Benediktsson, "Hyperspectral anomaly detection with attribute and edge-preserving filters," *IEEE Trans. Geosci. Remote Sens.*, vol. 55, no. 10, pp. 5600–5611, Oct. 2017.

[6] F. A. Kruse *et al.*, "The spectral image processing system (SIPS)-interactive visualization and analysis of imaging spectrometer data," *Remote Sens. Environ.*, vol. 44, nos. 2–3, pp. 145–163, 1993.

[7] C.-I. Chang, "An information-theoretic approach to spectral variability, similarity, and discrimination for hyperspectral image analysis," *IEEE Trans. Inf. Theory*, vol. 46, no. 5, pp. 1927–1932, Aug. 2000.

[8] C.-I. Chang and D. Heinz, "Constrained subpixel target detection for remotely sensed imagery," *IEEE Trans. Geosci. Remote Sens.*, vol. 38, no. 3, pp. 1144–1159, May 2000.

[9] Z. Zou and Z. Shi, "Hierarchical suppression method for hyperspectral target detection," *IEEE Trans. Geosci. Remote Sens.*, vol. 54, no. 1, pp. 330–342, Jan. 2016.

[10] D. Manolakis, D. Marden, and G. A. Shaw, "Hyperspectral image processing for automatic target detection applications," *Lincoln Lab. J.*, vol. 14, no. 1, pp. 79–116, 2003.

[11] S. Kraut and L. L. Scharf, "The CFAR adaptive subspace detector is a scale-invariant GLRT," *IEEE Trans. Signal Process.*, vol. 47, no. 9, pp. 2538–2541, Sep. 1999.

[12] F. C. Robey, D. R. Fuhrmann, E. J. Kelly, and R. Nitzberg, "A CFAR adaptive matched filter detector," *IEEE Trans. Aerosp. Electron. Syst.*, vol. 28, no. 1, pp. 208–216, Jan. 1992.

[13] C.-I. Chang, "Orthogonal subspace projection (OSP) revisited: A comprehensive study and analysis," *IEEE Trans. Geosci. Remote Sens.*, vol. 43, no. 3, pp. 502–518, Mar. 2005.

[14] L. L. Scharf and B. Friedlander, "Matched subspace detectors," *IEEE Trans. Signal Process.*, vol. 42, no. 8, pp. 2146–2157, Aug. 1994.

[15] Y. Chen, N. M. Nasrabadi, and T. D. Tran, "Sparse representation for target detection in hyperspectral imagery," *IEEE J. Sel. Topics Signal Process.*, vol. 5, no. 3, pp. 629–640, Jun. 2011.

[16] W. Li, Q. Du, and B. Zhang, "Combined sparse and collaborative representation for hyperspectral target detection," *Pattern Recognit.*, vol. 48, no. 12, pp. 3904–3916, Dec. 2015.

[17] X. Sun *et al.*, "Target detection through tree-structured encoding for hyperspectral images," *IEEE Trans. Geosci. Remote Sens.*, vol. 59, no. 5, pp. 4233–4249, Sep. 2020.

[18] X. Sun *et al.*, "Ensemble-based information retrieval with mass estimation for hyperspectral target detection," *IEEE Trans. Geosci. Remote Sens.*, vol. 60, pp. 1–23, 2022.

[19] R. Zhao, Z. Shi, Z. Zou, and Z. Zhang, "Ensemble-based cascaded constrained energy minimization for hyperspectral target detection," *Remote Sens.*, vol. 11, no. 11, p. 1310, 2019.

[20] W. Xie, J. Lei, J. Yang, Y. Li, Q. Du, and Z. Li, "Deep latent spectral representation learning-based hyperspectral band selection for target detection," *IEEE Trans. Geosci. Remote Sens.*, vol. 58, no. 3, pp. 2015–2026, Mar. 2020.

[21] Q. Liu, L. Xiao, J. Yang, and Z. Wei, "CNN-enhanced graph convolutional network with pixel-and superpixel-level feature fusion for hyperspectral image classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 59, no. 10, pp. 8657–8671, Oct. 2020.

[22] D. Hong *et al.*, "Endmember-guided unmixing network (EGU-Net): A general deep learning framework for self-supervised hyperspectral unmixing," *IEEE Trans. Neural Netw. Learn. Syst.*, early access, May 28, 2021, doi: 10.1109/TNNLS.2021.3082289.

[23] J.-F. Hu, T.-Z. Huang, L.-J. Deng, T.-X. Jiang, G. Vivone, and J. Chanussot, "Hyperspectral image super-resolution via deep spatiospectral attention convolutional neural networks," *IEEE Trans. Neural Netw. Learn. Syst.*, early access, Jun. 9, 2021, doi: 10.1109/TNNLS.2021.3084682.

[24] X. Liu, C. Wang, Q. Sun, and M. Fu, "Target detection of hyperspectral image based on convolutional neural networks," in *Proc. 37th Chin. Control Conf. (CCC)*, Jul. 2018, pp. 9255–9260.

[25] W. Li, G. Wu, and Q. Du, "Transferred deep learning for hyperspectral target detection," in *Proc. IEEE Int. Geosci. Remote Sens. Symp. (IGARSS)*, Jul. 2017, pp. 5177–5180.

[26] G. Zhang, S. Zhao, W. Li, Q. Du, Q. Ran, and R. Tao, "HTD-net: A deep convolutional neural network for target detection in hyperspectral imagery," *Remote Sens.*, vol. 12, no. 9, p. 1489, 2020.

[27] O. Ronneberger, P. Fischer, and T. Brox, "U-Net: Convolutional networks for biomedical image segmentation," in *Proc. Int. Conf. Med. Image Comput.-Assist. Intervent.*, 2015, pp. 234–241.

[28] J. Du and Z. Li, "Hyperspectral target detection with CNN using subtraction model," in *Proc. 2nd IEEE Adv. Inf. Manage., Communicates, Electron. Autom. Control Conf. (IMCEC)*, May 2018, pp. 2330–2335.

[29] M. Mirza and S. Osindero, "Conditional generative adversarial nets," 2014, *arXiv:1411.1784*.

[30] W. Xie, X. Zhang, Y. Li, K. Wang, and Q. Du, "Background learning based on target suppression constraint for hyperspectral target detection," *IEEE J. Sel. Top. Appl. Earth Observ. Remote Sens.*, vol. 13, pp. 5887–5897, 2020.

[31] A. Makhzani, J. Shlens, N. Jaitly, I. Goodfellow, and B. Frey, "Adversarial autoencoders," 2015, *arXiv:1511.05644*.

[32] D. P Kingma and M. Welling, "Auto-encoding variational Bayes," 2013, *arXiv:1312.6114*.

[33] W. Xie, J. Yang, J. Lei, Y. Li, Q. Du, and G. He, "SRUN: Spectral regularized unsupervised networks for hyperspectral target detection," *IEEE Trans. Geosci. Remote Sens.*, vol. 58, no. 2, pp. 1463–1474, Oct. 2019.
[34] D. Zhu, B. Du, and L. Zhang, "Two-stream convolutional networks for hyperspectral target detection," *IEEE Trans. Geosci. Remote Sens.*, vol. 59, no. 8, pp. 6907–6921, Aug. 2021.
[35] Y. Shi, J. Li, Y. Li, and Q. Du, "Sensor-independent hyperspectral target detection with semisupervised domain adaptive few-shot learning," *IEEE Trans. Geosci. Remote Sens.*, vol. 59, no. 8, pp. 6894–6906, Oct. 2021.
[36] J. Snell, K. Swersky, and R. Zemel, "Prototypical networks for few-shot learning," in *Proc. 31st Int. Conf. Neural Inf. Process. Syst.*, 2017, pp. 4080–4090.
[37] T. M. Hospedales, A. Antoniou, P. Micaelli, and A. J. Storkey, "Meta-learning in neural networks: A survey," *IEEE Trans. Pattern Anal. Mach. Intell.*, early access, May 11, 2021, doi: 10.1109/TPAMI.2021.3079209.
[38] D. Mandal, S. Medya, B. Uzzi, and C. Aggarwal, "Meta-learning with graph neural networks: Methods and applications," 2021, *arXiv:2103.00137*.
[39] N. Mishra, M. Rohaninejad, X. Chen, and P. Abbeel, "A simple neural attentive meta-learner," in *Proc. Int. Conf. Learn. Represent. (ICLR)*, 2018. [Online]. Available: https://arxiv.org/abs/1707.03141
[40] C. Finn, P. Abbeel, and S. Levine, "Model-agnostic meta-learning for fast adaptation of deep networks," in *Proc. Int. Conf. Mach. Learn. (ICML)*, 2017, pp. 1126–1135.
[41] O. Vinyals, C. Blundell, T. Lillicrap, K. Kavukcuoglu, and D. Wierstra, "Matching networks for one shot learning," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 29, 2016, pp. 3630–3638.
[42] 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/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 1199–1208.
[43] K. Fu, T. Zhang, Y. Zhang, Z. Wang, and X. Sun, "Few-shot SAR target classification via metalearning," *IEEE Trans. Geosci. Remote Sens.*, vol. 60, pp. 1–14, 2021.
[44] S. Targ, D. Almeida, and K. Lyman, "ResNet in ResNet: Generalizing residual architectures," 2016, *arXiv:1603.08029*.
[45] F. Schroff, D. Kalenichenko, and J. Philbin, "FaceNet: A unified embedding for face recognition and clustering," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2015, pp. 815–823.
[46] G. Kertész and I. Felde, "One-shot re-identification using image projections in deep triplet convolutional network," in *Proc. IEEE 15th Int. Conf. Syst. Syst. Eng. (SoSE)*, Jun. 2020, pp. 597–602.
[47] K. He, J. Sun, and X. Tang, "Guided image filtering," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 35, no. 6, pp. 1397–1409, Jun. 2013.

Yulei Wang (Member, IEEE) was born in Yantai, Shandong, China, in 1986. She received the B.S. and Ph.D. degrees in signal and information processing from Harbin Engineering University, Harbin, China, in 2009 and 2015, respectively.

She is awarded by the China Scholarship Council in 2011 as a joint Ph.D. Student to study in Remote Sensing Signal and Image Processing Laboratory, University of Maryland, Baltimore, MD, USA, for two years. She is an Associate Professor with the Hyperspectral Imaging in Remote Sensing (CHIRS), Information Science and Technology College, Dalian Maritime University, Dalian, China. Her research interests include hyperspectral image processing and vital signs signal processing.

Xi Chen was born in Kuitun, Xinjiang Uygur Autonomous Region, China, in 2000. He received the B.E. degree in electronic information engineering from Dalian Maritime University, Dalian, China, in 2020. He is pursuing the M.S. degree in information and communication engineering with the Information Science and Technology College, Dalian Maritime University.

His research interests include hyperspectral target detection and deep learning.

Fengchao Wang was born in Dalian, Liaoning, China, in 1997. He received the B.E. degree in electronic information engineering from Dalian Maritime University, Dalian, in 2020, where he is pursuing the M.S. degree in information and communication engineering with the Information Science and Technology College.

His research interests include hyperspectral anomaly detection and deep learning.

Meiping Song received the Ph.D. degree from the College of Computer Science and Technology, Harbin Engineering University, Harbin, China, in 2006.

From 2013 to 2014, she was a Visiting Associate Research Scholar with the Remote Sensing Signal and Image Processing Laboratory, University of Maryland, Baltimore, MD, USA. She is an Associate Professor with the College of Information Science and Technology, Dalian Maritime University, Dalian, China. Her research includes remote sensing and hyperspectral image processing.

Chunyan Yu received the B.S. and Ph.D. degrees in environment engineering from Dalian Maritime University, Dalian, China, in 2004 and 2012, respectively.

In 2004, she joined the College of Computer Science and Technology, Dalian Maritime University. From 2013 to 2016, she was a Post-Doctoral Fellow with the Information Science and Technology College, Dalian Maritime University. From 2014 to 2015, she was a Visiting Scholar with the College of Physicians and Surgeons, Columbia University, New York, NY, USA. She is an Associate Professor with the Information Science and Technology College, Dalian Maritime University. Her research interests include image segmentation, hyperspectral image classification, and pattern recognition.