Few-Shot Open-Set Recognition of Hyperspectral Images with Outlier Calibration Network

Debabrata Pal[1, 2], Valay Bundele[1], Renuka Sharma[1], Biplab Banerjee[1], Yogananda Jeppu[2]

[1]Indian Institute of Technology, Bombay, [2]Honeywell Technology Solutions, India

{deba.iitbcsre19, valaybundele, renuka.10078, getbiplab, yvjeppu}@gmail.com

Abstract

We tackle the few-shot open-set recognition (FSOSR) problem in the context of remote sensing hyperspectral image (HSI) classification. Prior research on OSR mainly considers an empirical threshold on the class prediction scores to reject the outlier samples. Further, recent endeavors in few-shot HSI classification fail to recognize outliers due to the 'closed-set' nature of the problem and the fact that the entire class distributions are unknown during training. To this end, we propose to optimize a novel outlier calibration network (OCN) together with a feature extraction module during the meta-training phase. The feature extractor is equipped with a novel residual 3D convolutional block attention network (R3CBAM) for enhanced spectral-spatial feature learning from HSI. Our method rejects the outliers based on OCN prediction scores barring the need for manual thresholding. Finally, we propose to augment the query set with synthesized support set features during the similarity learning stage in order to combat the data scarcity issue of few-shot learning. The superiority of the proposed model is showcased on four benchmark HSI datasets. [1]

1. Introduction

In the modern era, with the rapidly growing population, land-cover classification plays an indispensable role in the strategic management of the available natural resources [18]. Amongst different remote sensing technologies, HSI plays a crucial part in land-cover classification for its enriched spectral-spatial characteristics. There exist several machine learning and deep learning-based endeavors for HSI classification [5, 23, 24, 37]. However, HSI data suffer from the 'curse of dimensionality' [5] given a large number of bands and the availability of small training data for certain land-cover classes. As a pragmatic alternative, few-shot learning (FSL) [6, 27, 31] has become popular due to better generalization capability from limited training samples [17, 21, 36]. Nevertheless, the existing HSI classification

models mainly focus on the closed-set learning setup where the classes observed during training and testing are indistinguishable. Closed-set learning overestimates the training classes and fails to detect any unseen class test sample which may appear in real-life applications.
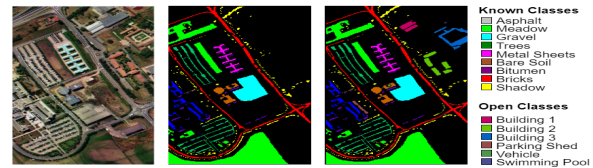

Figure 1. (Left) The original image of the University of Pavia HSI dataset is displayed. (Middle) Land cover maps of nine known classes are annotated as per the actual ground truth. (Right) Six more unknown classes are annotated along with known classes, which often get misclassified to one of the known classes.

We further motivate the problems through one of the benchmark HSI datasets in Fig. 1. The dataset has annotations for nine land-cover classes. Still, the labeling is highly inconsistent across the classes with severe data imbalance, which poses significant challenges in designing robust supervised learning models. It can also be seen from the middle figure of Fig. 1 that a large portion of the scene is unlabeled (black pixels) [1, 18]. This is because of the fact that labeling is costly and many of the classes have very marginal differences in the spectral dimensions. Let us now consider two scenarios for deploying a model trained on this data: i) on another geographical area, ii) on the same area but at a future time. In both cases, this is highly likely that the model will encounter novel class samples due to spatial or temporal changes. It is evident from these discussions that the paradigms of classification with less supervision and open-set learning are very much relevant to HSI classification. However, less emphasis has been given to it.

Closed-set learning misclassifies these unknown samples into one of the known classes. Again, efforts by applying a threshold on the softmax probability in closed-set learning only convey the training class prediction uncertainty [3]. Contemporary Open-Set Recognition (OSR) methods apply a threshold on the distance of test samples from the train-

---

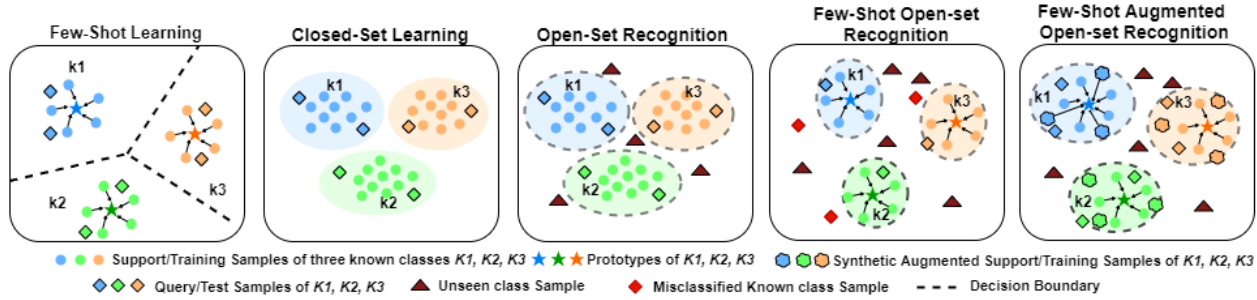[1]The code is available at https://github.com/DebabrataPal7/OCN

Figure 2. By learning from large samples, OSR develops a firm discriminative boundary to reject the outliers. Due to the availability of few samples, Contemporary FSOSR can not learn distribution density like OSR. It causes many known class samples to get misclassified as outliers. Our FSOSR with an augmented query set approach amplifies known class prototype representation. It squeezes the known class query distance from the prototype compared to an outlier's with equivalent latent feature embedding.

ing class prototypes or on the prediction probability post calibrating with a distribution [3, 7, 18]. To reject the outliers, OSR methods [2, 25] alternatively seek to maximize the data density given a large set of training samples. This limits OSR applicability over HSI datasets due to limited training samples, class imbalance, and the spectrally fine-grained nature of the closed-set and open-set classes.

Our focus in this paper is FSOSR, where we have access to very few labeled samples for the known classes, and no information regarding the outliers is present during training. The existing FSOSR methods function by updating the model parameters in metric space to distinguish unseen samples from seen classes along with the classification of seen class samples [12, 16, 26]. However, it generates an uncertain prototype radius [12, 16] where both the known and unknown class query can reside equidistant from the prototype, impacting an outlier to get misclassified as a known class sample. There exists limited research progress on FSOSR, and majorly they focus on applying empirical threshold [16, 12] which is non-trivial to set in general optimally. To the best of our knowledge, there does not exist any FSOSR method for HSI in the true sense, and the only existing method uses more than twenty samples per class [18] whereas a typical FSOSR system should consider not more than five samples classwise.

This research proposes a unified architecture comprising three modules; salient feature learning from HSI patches, enhancing the known class distribution through augmentation, and automatic outlier recognition using a decision network. For learning better fine-grained salient features from limited samples, we extend the traditional CBAM [33] to support the spectral, spatial, and channel dimensions of HSI cubes and propose Convolutional Block Attention 3D (CBAM3D) layer. We incorporate them in-between Conv3d layers in our feature extraction network R3CBAM. This helps in attenuating irrelevant features with simultaneous amplification of relevant salient features. To further reinforce prototype representation and mitigate sampling bias,

we propose to augment query set with samples generated by Variational Autoencoder (VAE) [15]. We illustrate in Fig. 2 how our augmentation strategy helps to learn known class data density in few-shot context. Finally, OCN is introduced to reject outliers by optimizing cross-entropy loss in each episode of *meta-training* as shown in Fig.3. This helps in gaining transferable knowledge to reject outliers during *meta-testing* without defining any threshold. Overall, our contributions are summarized as follows:

- To the best of our knowledge, ours is among the first endeavors to use the meta-learning-based FSOSR for HSI datasets that do not require any threshold.
- Our model consists of two novel sub-modules, i) an *OCN* to automatically reject the outliers ii) a residual feature extraction network, *R3CBAM* with newly proposed *CBAM3D* layer to learn salient spectral-spatial features from few samples. We also introduce a feature augmentation paradigm to better estimate the known class distributions.
- We perform extensive experiments on four benchmark HSI datasets, namely, Indian Pines, Pavia, Salinas, and Houston-2013. On average, our method beats the next best FSOSR alternative by 7% ClosedOA, 10% OpenOA, and 20% AUROC on these datasets.

2. Related Work

Few-shot open-set recognition: Metric-based [27, 28, 31] and optimization-based [6] closed-set FSL does not enrich a model to recognize the outliers. Finding an optimal threshold is crucial for FSOSR as insufficient training data creates sampling bias in learning known class distributions. Outliers are rejected if the largest probability is small [16]. In case of transformation inconsistency of modified prototypes, unknowns are detected based on a certain threshold in [12]. A threshold of 0.5 on cumulative distribution function is used to detect HSI outliers in [18]. In [1] discrimination threshold is varied on reconstruction error under the Receiver Operating Characteristic (ROC) curve.
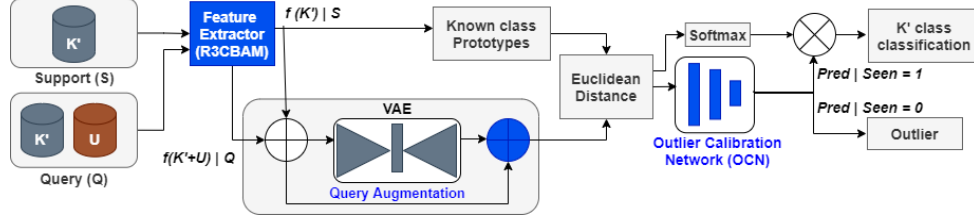
Figure 3. Visualization of our method. Input support set corresponding to known ($\mathcal{K}'$) class samples and a query set corresponding to known ($\mathcal{K}'$), pseudo-unknown ($\mathcal{U}$) samples are passed through feature extraction network, R3CBAM, in each episode. Support features are augmented with query features and passed through VAE to generate pseudo-known samples. Finally, features are classified as seen or unseen based on OCN prediction. For seen class prediction, $\mathcal{K}'$ class samples are classified based on their distance from prototypes $\mathcal{P}$.

In contemporary OSR approaches, the model parameters are not optimized to cope with an empirical threshold to reject outliers. Alternatively, we propose to follow a generative-discriminative approach, where class representations are enriched in a low-data regime with a pseudo support set. Thanks to OCN, we have overcome the requirement of manual rejection thresholding.

Threshold computation in open-set recognition: OSR aims to detect outliers by forcing the model to give a high rise of entropy posterior distributions. Existing OSR approaches trained on large-scale dataset can broadly be categorized as discriminative [2, 3, 9, 25], generative [35, 19, 20], and combination of both [22]. OpenMax [3] detects outliers by thresholding on uncertainty after applying Extreme Value Theory on the SoftMax generated known class probabilities. A contamination ratio is considered globally as well as per class to define a threshold [9]. An optimal threshold is computed in [20] by inspecting histogram between match reconstruction errors and non-match counterpart after passing samples through an autoencoder.

Residual attention-aware network engineering: To extract strong representative-discriminative feature, network engineering has been evolved from Universal Approximation theorem [11] to state-of-the-art models with increasing width [29], depth, residual connection [10], cardinality [34] and attention [13, 30, 4]. Introducing skip connections around few layers and adding to main path alleviates vanishing or exploding gradient problem [17] which can occur in high parameterised conv3D layers. Using conv3D layers, HSI spectral-spatial feature correlation can jointly be learnt [24, 37, 17], but it takes 5-dimensional input. There exists a disjoint effort on attention-based [8] and 3D residual convolution based HSI classification [37, 17]. To learn spectral-spatial cubic salient features, we extend [33] by paying attention in depth dimension and created CBAM3D layer to stitch with conv3d layer connected in residual topology.

3. Proposed Method

3.1. Preliminaries

At each HSI pixel location, we form 3D patches comprising spectral and spatial features. FSOSR is a supervised meta-learning approach. During *meta-training*, the model learns the ability to reject the outliers along with the classification of known class samples using an episodic learning strategy. During *meta-testing*, it employs the learned model to classify the outliers and the known class samples correctly. To this end, each dataset is split into two disjoint sets of classes, base classes for training and target classes for testing. In HSI datasets, the sample distributions for the outliers have a high degree of identical reflectivity patterns with those of the known classes. To comply with this issue, we select a subset of base classes as known or seen while the rest act as the pseudo-unknown in each episode during the *meta-training* phase. Let $\mathcal{K} = \{1, 2, \cdots, K\}$ and $\mathcal{U}$ define the set of known classes and the pseudo-unknown classes, where $|\mathcal{K}| = K$ denote the cardinality of $\mathcal{K}$.

By definition, an episode consists of a support and query set. Mathematically, the support set is defined as $\mathcal{S} = \{(x_i^s, y_i^s)\}_{i=1}^{m|\mathcal{K}'|}$ where $|\mathcal{K}'| = K'$ classes are randomly sampled from $\mathcal{K}$ ($\mathcal{K}' \subset \mathcal{K}$) and is termed as K'-way m-shot since there exist exactly m training samples per $\mathcal{K}'$ classes. The query set, $\mathcal{Q} = \{(x_j^q, y_j^q)\}_{j=1}^{N|\mathcal{K}|}$ contains N samples from each of the $\mathcal{K}$ classes, where samples corresponding to $\mathcal{K}'$ act as inliers while the remaining class samples are treated as $\mathcal{U}$. x^s, x^q represent HSI patch cubes and $y^s \in \mathcal{K}', y^q \in \mathcal{K}$. We propose to generate synthetic samples by VAE corresponding to $\mathcal{K}'$ and augment it with $\mathcal{Q}$ for better estimating class density. We represent $\mathcal{Q}(\mathcal{K}')$ as known-class query set and $\mathcal{Q}(\mathcal{U})$ as unknown-class query set, where $\mathcal{U} = \mathcal{K} - \mathcal{K}'$ and $'-'$ defines set difference operation. The feature extraction network (f_θ) with learnable parameters θ gains transferable knowledge by learning to correctly classify $\mathcal{Q}(\mathcal{K}')$. Also, the proposed OCN (u_ϕ) with learnable parameters ϕ learns the ability to identify $\mathcal{Q}(\mathcal{U})$.

After *meta-training*, we fine-tune f_θ parameters based on the available samples of a subset of target classes to adapt to the new classification environment quickly. Finally, at the *meta-testing* phase, an unlabeled query sample from the target class is first classified by OCN, to find whether it belongs to a known or unknown class. If the OCN labels the sample as known, its class is estimated based on the Euclidean distance from the learned prototypes.
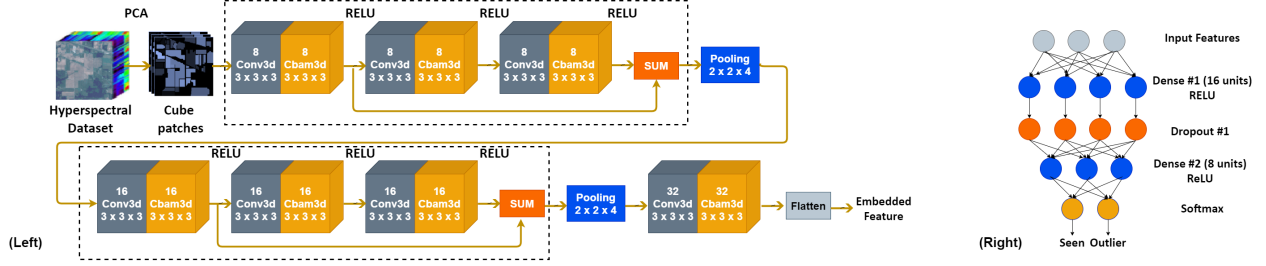
Figure 4. (Left) The architecture of the proposed attention-aware feature extraction network, R3CBAM, is to learn the complex salient spectral-spatial features of HSI data. Our proposed CBAM3D layers are stitched after each conv3D layer for paying attention to relevant 3D HSI features. (Right) We introduce OCN architecture to reject the outliers.

3.2. Overview of the proposed model components

Here we first discuss the R3CBAM feature encoder model together with the proposed CBAM3D attention layer, followed by a discussion on OCN.

Feature extractor R3CBAM: The architecture of the proposed R3CBAM encoder (f_θ) for HSI feature extraction and learning the known class prototypes is shown in Fig.4 (Left). At each spatial pixel location of HSI data, 3D spectral-spatial patches $I \in \mathbb{R}^{H \times W \times B}$ are formed, where H, W, B indicate the height, width, and spectral bands, respectively. Principal component analysis (PCA) [32] is applied to preserve only the b significant bands, where $b << B$. The backbone of f_θ is based on the 3D CNN architecture, primarily consisting of 3D residual convolutional layers [17] which can learn improved spectral-spatial features. Two residual blocks, as shown in dashed boxes in Fig.4 (Left), are comprised of three successive units of Conv3D and the proposed CBAM3D layers. There also exists a skip connection in each residual block [10]. After each residual block, 3D max-pooling layer reduces the computational complexity and aggregates spectral-spatial relationship. Due to the presence of a higher number of bands b as compared to H and W, stride across spectral dimension is chosen as 4 whereas it is taken as 2 for spatial dimension. Finally, one Conv3D-CBAM3D layer is used to extract the compact salient feature globally corresponding to the input 3D patch before flattening the features.

CBAM3D attention layer: Attention learning has been proved to highlight the insightful visual features in a CNN model. In this respect, the CBAM model, proposed in [33], disentangle the spectral and spatial attention learning modules. However, CBAM is designed for RGB images. But, to exploit the spectral-spatial relationship of HSI data, we need a module that could also refine features along the spectral dimension. To this note, we propose a novel 3D attention module, CBAM3D, built upon CBAM [33], for highlighting the important regions in the spectral-spatial feature maps. The proposed CBAM3D module can infer attention maps both along the channel dimension and the spectral-spatial dimension and is illustrated in Fig.5.
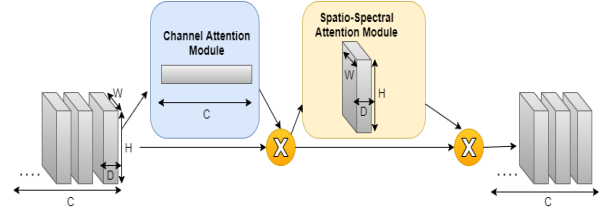

Figure 5. Visualization of the proposed CBAM3D layer. Input 3D feature is passed initially through the channel attention module to learn inter-channel salient features. Using the residual connection, features are refined with a channel attention map. Finally, this feature is refined with a spatio-spectral attention module to amplify the relevant features across the spatial and spectral dimensions.

A given feature map output $F \in \mathbb{R}^{C \times H \times W \times D}$ from Conv3d layer is passed through CBAM3D to sequentially infer 1D channel attention map $\mathcal{A}_c \in \mathbb{R}^{C \times 1 \times 1 \times 1}$ followed by a spatio-spectral attention map $\mathcal{A}_{ss} \in \mathbb{R}^{1 \times H \times W \times D}$, where C, H, W, D refer to number of channels, height, width, and depth of HSI patches respectively. The mathematical formulation of refinement of feature maps using proposed CBAM3D is represented by (1), where $\otimes$ symbolizes element-wise product with necessary broadcasting.

$$F' = \mathcal{A}_c(F) \otimes F,$$
$$F'' = \mathcal{A}_{ss}(F') \otimes F' \qquad (1)$$

Channel attention module explores the inter-channel salient features by squeezing spatial and depth dimensions. This module helps in identifying the channels with important features. Using 3D GlobalAveragePooling and 3D GlobalMaxPooling layers, overall latent statistics (F^c_{avg3d}) and significant features (F^c_{max3d}) are extracted, respectively. These global features are passed individually through a shared multi-layer perceptron (MLP) with one hidden layer of $\mathbb{R}^{C/r \times 1 \times 1 \times 1}$ activation units, where r is the parameter reduction ratio. Finally, the outputs of the MLP are added element-wise to produce $\mathcal{A}_c$ in (2), where σ denotes the sigmoid activation to keep attention map within $[0, 1]$.

$$\mathcal{A}_c(F) = \sigma(MLP(F^c_{avg3d}) + MLP(F^c_{max3d})) \qquad (2)$$

Spatio-spectral attention module explores the relationship across spatial and spectral dimensions. Average-pooling and max-pooling are applied independently on F' along the channel dimension to generate global ($F_{avg3d}^{ss} \in \mathbb{R}^{1 \times H \times W \times D}$)) and significant ($F_{max3d}^{ss} \in \mathbb{R}^{1 \times H \times W \times D}$)) descriptors. These descriptors are then concatenated and passed through a conv3d layer to generate $\mathcal{A}_{ss}$ in (3), where $f^{7 \times 7 \times 7}$ symbolizes 3D convolution with kernel $7 \times 7 \times 7$.

$$\mathcal{A}_{ss}(F) = \sigma(f^{7 \times 7 \times 7}[F_{avg3d}^{ss}; F_{max3d}^{ss}]) \qquad (3)$$

Outlier Calibration Network: In order to alleviate defining an empirical threshold for outlier rejection as done traditionally, we propose OCN (u_ϕ) to detect unseen samples. The objective of u_ϕ is to correctly classify known class queries and the VAE generated samples as seen while simultaneously predicting unknown class queries as outliers. The input to u_ϕ is the set of Euclidean distances of the test query from the known class prototypes. Since distances are a measure of the similarity between embeddings, u_ϕ learns to correctly identify the outliers by leveraging the distances in metric space. The architecture of the proposed MLP-based u_ϕ is shown in Fig.4 (Right). OCN has three hidden layers. The first two layers have 16 and 8 hidden units, respectively, with RELU activation. The last layer has two hidden units with a softmax activation function to predict the probability of a sample belonging to the seen or unseen class. Also, a dropout layer with the probability of dropping neurons as 0.4 is added in between the first two layers to regularise the network. By *meta-training*, u_ϕ learns to recognize outliers without the need for any threshold.

3.3. Augmentation with VAE generated samples

It is pretty challenging to learn the known class distribution from limited samples in FSOSR. Due to insufficient knowledge, a known class query mistakenly gets classified as an outlier, whereas an outlier might get recognized as a seen class sample. To improve the known class data density, we generate pseudo-known samples by optimizing a VAE using the known class support and query sets. The encoder (q_ψ) of our VAE with learnable parameters ψ has two dense layers of 32 and 16 units and has RELU activation in each layer. The decoder (p_ν) with learnable parameters ν has three dense layers of 16, 32, and 64 units along with RELU activation. The last layer of the VAE decoder outputs 64 units as R3CBAM generates 64-dimensional features.

3.4. Training and inference protocol

Our framework is built upon the Prototypical Networks [27]. As aforesaid, each episode consists of support set consisting of known ($\mathcal{K}'$) classes and a query set consisting of known and pseudo-unknown classes, $\mathcal{K}$, randomly sampled from the set of base classes. The inputs to our Algorithm

---

Algorithm 1: FSOSR meta-learning steps

Meta-Training Phase:
Input: $f_\theta, u_\phi, q_\psi, p_\nu, \mathcal{S}(\mathcal{K}'), \mathcal{Q}(\mathcal{K}), y^s, y^q$
Output: Updated θ, ϕ, ψ, ν parameters

1 Extract features:
  $\mathcal{S}_f \leftarrow f_\theta(\mathcal{S}(\mathcal{K}')), \mathcal{Q}_f \leftarrow f_\theta(\mathcal{Q}(\mathcal{K}))$;
2 Compute known class prototypes $\mathcal{P}$ from $\mathcal{S}_f$ in (4);
3 Update q_ψ, p_ν by optimizing loss (7) with samples x, where $x \in p_\nu(q_\psi(f_\theta(\mathcal{Q}(\mathcal{K}') \cup \mathcal{S}(\mathcal{K}'))))$;
4 Augment $\mathcal{Q}_f$ with VAE generated samples:
  $\mathcal{Q}_{aug} \leftarrow \mathcal{Q}_f \cup p_\nu(q_\psi(f_\theta(\mathcal{Q}(\mathcal{K}') \cup \mathcal{S}(\mathcal{K}'))))$;
5 Update u_ϕ by optimizing $L_{u_\phi\_outlier}$ (8) for $\mathcal{Q}_{aug}$;
6 Update f_θ by optimizing L_{Total} (9) for $\mathcal{Q}_{aug}$ from classification loss $\mathcal{L}_{f_\theta\_clf}$ (6) and $L_{u_\phi\_outlier}$ (8) ;
7 **return** Updated parameters of $f_\theta, u_\phi, q_\psi, p_\nu$;

Meta-Testing Phase:
Input: $f_\theta, u_\phi, \mathcal{S}(\mathcal{K}'), \mathcal{Q}(\mathcal{K}), y^s, y^q$
Output: Classification of the query samples

8 Extract features: $\mathcal{S}_f \leftarrow f_\theta(\mathcal{S}(\mathcal{K}'))$,
  $\mathcal{Q}_f \leftarrow f_\theta(\mathcal{Q}(\mathcal{K}))$;
9 Compute known class prototypes $\mathcal{P}$ from $\mathcal{S}_f$;
10 Compute Euclidean distances: $\mathcal{D} \leftarrow \|\mathcal{Q}_f - \mathcal{P}\|_2^2$;
11 Pass $\mathcal{D}$ as input to u_ϕ;
12 If u_ϕ classifies a sample as seen;
     Predict class by applying softmax over $\mathcal{D}$;
   Else;
     Classify the sample as an outlier;
13 **return** Predicted class of the query samples;

---

1 are f_θ (R3CBAM), u_ϕ (OCN), $q_\psi, p_\nu, \mathcal{S}(\mathcal{K}'), \mathcal{Q}(\mathcal{K})$, and y^s, y^q representing the label space for $\mathcal{S}(\mathcal{K}'), \mathcal{Q}(\mathcal{K})$.

Estimating the known class prototypes: All the $\mathcal{S}(\mathcal{K}')$ and $\mathcal{Q}(\mathcal{K})$ samples are passed through f_θ where relevant class-specific features are magnified. We represent the extracted features $f_\theta(\mathcal{S}(\mathcal{K}'))$ by $\mathcal{S}_f$ and $f_\theta(\mathcal{Q}(\mathcal{K}))$ by $\mathcal{Q}_f$. For $m > 1$, each known class prototype $\mathcal{P}_k$ for class k is computed in (4) after estimating classwise mean of support embeddings.

$$\mathcal{P}_k = \frac{1}{m} \sum_{x_i^s \in \mathcal{S}^k} f_\theta(x_i^s) \qquad (4)$$

Where $\mathcal{S}^k$ is the support set of kth known class and $\mathcal{P}$ is the collection of all known class prototypes.

Multi-task VAE and R3CBAM classification loss: To gain representative knowledge of the distributions concerning the known classes in the low-data regime, we pass the extracted $\mathcal{S}_f$ and the known class query features $f_\theta(\mathcal{Q}(\mathcal{K}'))$ through the VAE and generate pseudo-known class samples. In each episode, we optimize VAE parameters based on the computed gradients from the standard VAE loss (5) and the classification loss of the generated samples (6). We repre-

sent the standard VAE loss (5) as consisting of a reconstruction loss and the KL divergence loss. The VAE reconstruction loss is simply the Euclidean distance between the input and the generated samples. The KL divergence term regularizes the distributions of the latent space.

$$\mathcal{L}_{Vr} = \mathbb{E}_{z \sim q_\psi(z|x)}(\|p_\nu(z) - x\|^2 + KL(q_\psi(z|x) \| \mathcal{N}(0, I))) \quad (5)$$

where, z is the latent space vector, $\mathcal{N}(0, I)$ represents the standard Gaussian distribution, and $x \in p_\nu(q_\psi(f_\theta(\mathcal{Q}(\mathcal{K}') \cup \mathcal{S}(\mathcal{K}'))))$ denotes input with class label y. The classification loss is computed for each x in (6).

$$\mathcal{L}_{Vc} = \mathbb{E}\left[-\log \frac{\exp(-d(x, \mathcal{P}_y))}{\sum_{k=1}^{\mathcal{K}'} \exp(-d(x, \mathcal{P}_k))} \right] \quad (6)$$

where, $\mathcal{P}_y$ is the true class prototype of query x and d is squared Euclidean distance. These loss components (7) are optimized to update VAE parameters in each episode to generate synthetic samples precisely.

$$\mathcal{L}_{VAE} = \mathcal{L}_{Vr} + \mathcal{L}_{Vc} \quad (7)$$

After updating parameters of VAE, we use VAE generated samples to augment the query set. The augmented query set, denoted by $\mathcal{Q}_{aug}$, is basically $\mathcal{Q}_f \cup p_\nu(q_\psi(f_\theta(\mathcal{Q}(\mathcal{K}') \cup \mathcal{S}(\mathcal{K}'))))$. The known class classification loss for optimizing R3CBAM is calculated for each query feature in $\mathcal{Q}_{aug}$, using (6). The optimization with $\mathcal{Q}_{aug}$ helps in learning distribution density of the known classes in metric space from limited supervised data. This loss is denoted by $\mathcal{L}_{f_\theta\_clf}$.

OCN outlier rejection loss: We represent the Euclidean distances of the query features in the augmented query set (Q_{aug}) from the known class prototypes $\mathcal{P}$ by $\mathcal{D}_{aug}$, which is given as input to u_ϕ so that it can learn to classify known class queries and the VAE generated samples as seen and the unknown class query samples as outliers. It makes OCN a threshold-free outlier detector. We use cross-entropy loss (8) to optimize the OCN parameters.

$$\mathcal{L}_{u_\phi\_outlier} = -\sum_{i=1}^{2} t_i log(u_\phi(\mathcal{D}_{aug})_i) \quad (8)$$

where, $t_i \in \{0, 1\}$ is the true label for the seen and unseen class of the query sample and $u_\phi(\mathcal{D}_{aug})_i$ is the OCN output probability for the ith class(seen or unseen).

Total training loss: Finally, a multi-task loss constituting of known class classification loss and outlier prediction loss is used to optimize f_θ parameters in (9).

$$\mathcal{L}_{Total} = \mathcal{L}_{f_\theta\_clf} + \mathcal{L}_{u_\phi\_outlier} \quad (9)$$

Inference Strategy: We first sample an episode from the target set and calculate the Euclidean distances $\mathcal{D}$ of the extracted query features from $\mathcal{P}$. $\mathcal{D}$ is then given as input

to u_ϕ, which classifies the query samples as seen or outlier without using any threshold. If u_ϕ classifies a query as seen, we further apply softmax over $\mathcal{D}$ to predict the class.

4. Experiments

4.1. Datasets and preprocessing

We use four benchmark HSI datasets [2]. **Indian Pines (IP)** was acquired using AVIRIS sensor in northwestern Indiana for 16 land-cover classes and had 145×145 spatial resolution with 220 bands. **Salinas** also has 16 classes and was captured at Salinas Valley, California. It has 512×217 pixels with 204 spectral bands. Using ROSIS sensor, **University of Pavia** dataset was captured for nine classes having 610×610 pixels with 103 bands. We consider six additional classes annotated in [18] for FSOSR. The **Houston-2013** dataset was captured for 15 classes at the University of Houston and it has 349×1905 pixels with 144 bands.

Thirty bands are used for each of IP, Pavia and Salinas and ten for Houston to preserve more than 99% data variance after applying PCA. Then, cubic patches of size $(11, 11, ch)$ are formed at each pixel location using zero padding, where ch denotes the number of spectral bands.

4.2. Evaluation metrics

We compare the proposed method with other FSOSR approaches using Closed Overall Accuracy (ClosedOA), Open Overall Accuracy (OpenOA), and AUROC (Area Under Receiver Operating Characteristics Curve). In OSR, closed-set refers to the collection of seen classes, and open-set relates to the samples from outlier classes. ClosedOA refers to the percentage of seen class samples correctly classified by the model. To evaluate the performance of the model in the open-world settings, OpenOA is defined in (10).

$$OpenOA = \frac{\sum_{k=1}^{\mathcal{K}'+1} TP_k + TN_k}{\sum_{k=1}^{\mathcal{K}'+1} TP_k + TN_k + FP_k + FN_k} \quad (10)$$

where, TP_k, TN_k, FP_k, FN_k are true positive, true negative, false positive, and false negative of the kth class. For OpenOA, $\mathcal{K}'+1$ classes are evaluated considering $\mathcal{K}'$ known classes and all outliers belonging to one additional class.

In the presence of mixed seen class queries and outliers, AUROC measures how well the model detects the outliers at various threshold settings. It should be one for an ideal classifier. Even though our OCN-based FSOSR is a threshold-free approach, high AUROC is observed using our method.

4.3. Experimental Protocol

We select six classes randomly per dataset and keep them aside as target classes for *meta-testing* while the remaining classes are used as base classes in the *meta-training* phase.

Table 1. Comparison of 1-shot open-set classification performance on hyperspectral datasets of the proposed R3CBAM and SOTA Methods

Model	Indian Pines			Pavia University			Salinas			Houston-2013		
	ClosedOA	OpenOA	AUROC	ClosedOA	OpenOA	AUROC	ClosedOA	OpenOA	AUROC	ClosedOA	OpenOA	AUROC
OpenMax [3]	43.33±0.63	48.54±0.29	44.44±0.33	52.08±0.53	54.61±0.21	52.22±0.35	51.92±0.52	42.50±0.23	50.33±0.34	37.50±0.73	31.33±0.71	36.11±0.33
RDOSR [1]	51.28±0.34	50.13±0.41	47.29±0.23	50.85±0.27	51.68±0.32	55.45±0.42	59.14±0.35	60.19±0.33	54.23±0.21	58.92±0.27	63.53±0.51	61.08±0.34
MDL4OW [18]	46.15±0.21	46.50±0.23	48.66±0.32	56.66±0.22	54.89±0.21	51.11±0.34	58.33±0.21	57.28±0.31	52.77±0.32	41.66±0.43	43.42±0.34	43.88±0.32
PEELER [16]	71.41±0.31	75.45±0.24	71.84±0.21	57.18±0.35	71.55±0.31	52.39±0.18	65.14±0.53	69.63±0.43	57.78±0.33	46.95±0.41	74.34±0.17	53.75±0.32
SnaTCHer [12]	**89.33±0.11**	81.25±0.23	74.53±0.32	58.50±0.29	75.57±0.34	50.35±0.33	65.94±0.67	**78.91±0.28**	53.88±0.43	58.73±0.35	77.42±0.33	48.05±0.34
R3CBAM [Ours]	80.67±0.63	**82.72±0.34**	**75.93±0.32**	**59.50±0.21**	**84.77±0.34**	**89.96±0.43**	**82.64±0.31**	74.27±0.32	**81.36±0.21**	**60.84±0.32**	**78.93±0.46**	**74.26±0.17**

Table 2. Comparison of 5-shot open-set classification performance on hyperspectral datasets of the proposed R3CBAM and SOTA Methods

| Model	Indian Pines			Pavia University			Salinas			Houston-2013		
	ClosedOA	OpenOA	AUROC	ClosedOA	OpenOA	AUROC	ClosedOA	OpenOA	AUROC	ClosedOA	OpenOA	AUROC
OpenMax [3]	51.92±0.34	58.33±0.51	55.62±0.46	69.17±0.32	58.12±0.53	53.36±0.53	69.23±0.25	58.12±0.35	55.44±0.34	41.66±0.24	35.64±0.35	37.84±0.51
RDOSR [1]	55.98±0.51	55.92±0.45	52.38±0.52	64.74±0.45	64.89±0.34	63.94±0.37	67.80±0.35	66.32±0.51	60.28±0.47	68.82±0.52	66.37±0.31	62.56±0.43
MDL4OW [18]	50.76±0.53	46.96±0.35	64.51±0.51	65.33±0.25	62.66±0.19	48.66±0.41	75.01±0.37	62.66±0.34	72.88±0.22	46.66±0.21	44.42±0.23	46.22±0.33
PEELER [16]	82.81±0.39	87.37±0.34	74.19±0.25	60.71±0.23	72.69±0.53	60.36±0.51	74.20±0.52	75.39±0.31	60.39±0.36	57.98±0.47	75.57±0.57	55.47±0.34
SnaTCHer [12]	92.00±0.51	**89.42±0.45**	76.05±0.55	**74.91±0.35**	78.10±0.38	54.98±0.29	74.21±0.54	83.15±0.39	72.11±0.43	69.24±0.34	83.29±0.52	49.09±0.45
R3CBAM [Ours]	**94.61±0.38**	84.71±0.51	**88.40±0.38**	71.55±0.32	**87.94±0.34**	**91.90±0.45**	**84.88±0.37**	**85.61±0.39**	**88.52±0.34**	**71.97±0.29**	**84.96±0.48**	**88.50±0.47**
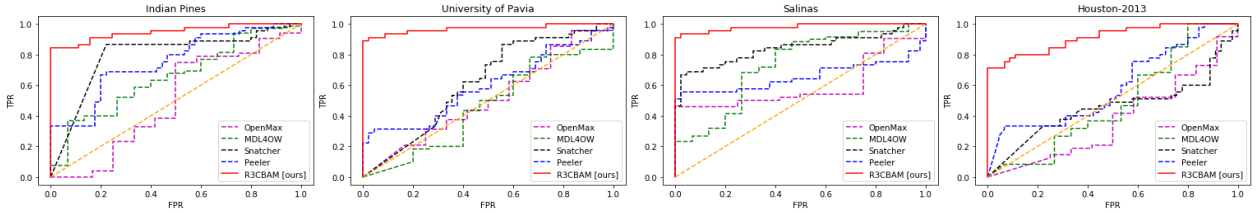
Figure 6. Comparison of Area under Receiver Operating Characteristics Curve (AUROC) for 5-shot FSOSR using different methods on four benchmark HSI datasets. The result shows our method's superior AUROC compared to all other existing FSOSR methods. R3CBAM's performance indicated in the 'Red' envelop shows a high True-Positive Rate (TPR) at a low False-Positive Rate (FPR).

An episodic training strategy is followed to train R3CBAM to learn salient features and OCN to reject outliers. During training, in each episode, we form query set (N = 15) from six randomly chosen base classes and support set is formed with samples from three randomly selected query classes. At test time, we similarly sample episodes from target classes and average the results over 500 episodes to get trustworthy measures of the model performance. We use Adam optimizer [14] with a learning rate of 0.0001 for each of the learning models, namely R3CBAM, VAE and OCN.

4.4. Comparison with the other methods

We compared 1-shot and 5-shot OSR performance against the SOTA methods respectively in Table 1 and 2. OpenMax [3] and MDL4OW [18] are originally experimented with large-scale datasets for performing OSR, but they perform poorly on recognizing outliers from small open-set HSI classes. To compare these methods for meta-learning-based FSOSR, we fitted the Weibull distribution in Prototypical Networks during *meta-training* phase for OpenMax. Similarly, the classification-reconstruction encoder in [18] with Extreme Value Theory is fitted in Prototypical Networks for MDL4OW. RDOSR [1] performs OSR with few-shot supervised samples of HSI datasets in latent space. PEELER [16] performs Mahalanobis distance-based metric-learning and SnaTCHer [12] detects outliers by validating prototype consistency after applying transformation functions in FSOSR context. [16, 12] are evaluated here without any adaptation as those are introduced to tackle the FSOSR task. Both of these two methods slightly perform better for IP dataset on OpenOA metric for 5-shot setting. Rest, our method shows dominating performance on all other HSI datasets in Table 2. Our method performs better than others in a margin of 10% ClosedOA on Salinas, 9% OpenOA on Pavia and 26% AUROC on Houston dataset for 5-shot settings. Also for 1-shot settings in Table 1, we beat others by 17% ClosedOA on Salinas, 9% OpenOA and 35% AUROC on Pavia dataset. This reflects in Fig.6 where our method R3CBAM envelops maximum area under the ROC curve compared to baseline methods projecting high outlier detection performance in few-shot regime.
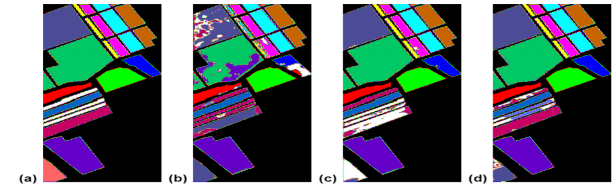

Figure 7. 5-shot FSOSR result comparison of SOTA methods on Salinas dataset (open classes are annotated in 'White' color) (a) Ground Truth (b) PEELER (c) SnaTCHer (d) R3CBAM

Table 3. Ablation study on our attention-aware feature extraction network performance, the usefulness of support set augmentation with VAE and outlier detection capability by newly introduced OCN on four benchmark hyperspectral datasets for 5-shot FSOSR task.

| Model	Indian Pines			Pavia University			Salinas			Houston-2013		
	ClosedOA	OpenOA	AUROC	ClosedOA	OpenOA	AUROC	ClosedOA	OpenOA	AUROC	ClosedOA	OpenOA	AUROC
DRes3DCNN [17]	77.51±0.21	69.37±0.32	70.14±0.35	70.44±0.33	68.11±0.28	67.51±0.26	83.46±0.43	71.36±0.42	77.09±0.42	70.44±0.52	68.61±0.36	75.54±0.23
HybridSN [24]	41.67±0.51	50.38±0.41	52.01±0.35	43.11±0.37	50.77±0.18	47.32±0.24	53.99±0.57	58.49±0.32	48.90±0.38	40.00±0.32	48.33±0.31	54.60±0.49
Without VAE	54.66±0.46	63.66±0.37	56.02±0.25	57.33±0.52	64.33±0.37	50.05±0.18	60.66±0.49	65.16±0.45	54.33±0.29	68.88±0.65	67.22±0.27	59.89±0.35
With threshold (0.5)	70.23±0.35	73.45±0.28	76.58±0.26	67.33±0.51	68.72±0.33	71.11±0.43	69.56±0.45	73.00±0.42	72.22±0.38	45.90±0.41	52.50±0.51	54.44±0.36
R3CBAM [Ours]	**94.61±0.38**	**84.71±0.51**	**88.40±0.38**	**71.55±0.32**	**87.94±0.34**	**91.90±0.45**	**84.88±0.37**	**85.61±0.39**	**88.52±0.34**	**71.97±0.29**	**84.96±0.48**	**88.50±0.47**
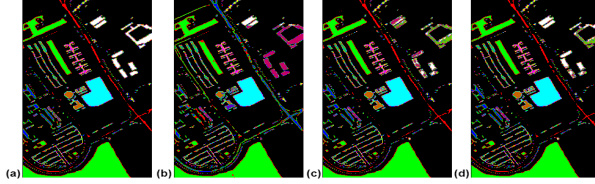
Figure 8. 5-shot FSOSR result comparison of SOTA methods on Pavia dataset (open classes are annotated in 'White' color) (a) Ground Truth (b) PEELER (c) SnaTCHer (d) R3CBAM

Figure 9. Ablation study on feature encoder for 5-shot FSOSR on IP (a) Ground Truth (b) DRes3DCNN (c)HybridSN (d) R3CBAM

The comparison of our method's generated classification maps against the two most competing FSOSR methods i.e., PEELER and SnaTCHer, for the Salinas and Pavia datasets are shown in Fig.7 and Fig.8 respectively. Outliers are annotated in 'White' color. From Fig.8, it can be observed that both PEELER and SnaTCHer misclassify some known class samples as outlier due to their threshold-based mechanism. In contrast, using OCN, our method correctly recognizes these outliers along with better known class sample classification by R3CBAM. Altogether this results in high ClosedOA, OpenOA and AUROC performance of our method. In the case of Pavia, most of the outliers are recognized by our method, causing better OpenOA in Fig.8..

4.5. Further analysis

The results for the ablation studies highlighting the significance of each novelties in our method are presented in Table 3 for the 5-shot setting.

R3CBAM Attention Analysis: In DRes3DCNN [17], residual 3D-CNN learns the spectral-spatial HSI features but does not learn attention maps. A hybrid combination of conv3D and conv2D layers are used in HybridSN [24] without using the residual connection and attention learning. We fit these encoders as f_θ in Algorithm 1 and compare against R3CBAM. Our R3CBAM uses an attention-aware CBAM3D layer connected in residual topology. It boosts OpenOA by 16% for Houston and ClosedOA by 17% for IP as compared to [17]. By jointly using attention learning and residual connections, R3CBAM gains 53% ClosedOA for IP and 37% OpenOA for Pavia as compared to [24].

The classification map ablation based on the feature extractor performance on IP dataset is shown in Fig.9. Many known class samples are misclassified and wrongly recognized as outliers with DRes3DCNN and HybridSN as salient spectral-spatial features of known classes are not learned due to a lack of attention-aware feature learning.
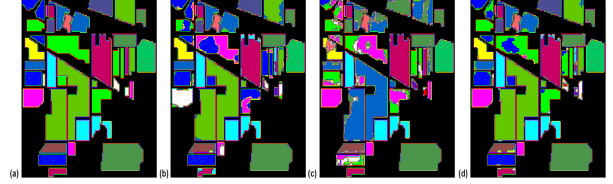
Impact of query augmentation: To see the effect of query set augmentation using VAE generated samples, we conducted experiments by removing VAE from the network. As can be seen from the results in 'Without VAE' in Table 3, R3CBAM shows significantly better performance as it learns a distribution density from limited samples due to the usage of query set augmentation. The improvement in performance by around 40% ClosedOA for IP, 23% OpenOA and 41% AUROC for Pavia dataset is observed.

OCN outlier rejection performance analysis: We compared the effectiveness of the newly introduced OCN with the contemporary threshold-based mechanism in Table 3. We set an arbitrary threshold of 0.5 and samples are classified as outliers if the maximum known class probability is below this value. OCN boosts OpenOA by approximately 32% for Houston and 19% for Pavia showing its excellence in detecting outliers in few-shot learning context.

5. Conclusions

In this work, we propose a novel meta-learning-based FSOSR framework for the HSI datasets consisting of the MLP-based OCN, a residual attention-aware feature extraction network (R3CBAM) and the paradigm that uses of the query set augmentation using VAE for learning dense distribution of the known classes. The performance of our method is better than existing OSR and FSOSR methods in the few-shot learning paradigm for the four real-world hyperspectral datasets. Further performance improvement is observed when the network is trained with more spectral-spatial samples. With superior FSOSR performance on low inter-class variance HSI datasets, our method can easily be adapted to any real-world OSR problems. In the future, we plan to do more fine-level analysis of the open-set classes.

References

[1] Razieh Kaviani Baghbaderani, Ying Qu, Hairong Qi, and Craig Stutts. Representative-discriminative learning for open-set land cover classification of satellite imagery. In *European Conference on Computer Vision*, pages 1–17. Springer, 2020.

[2] Abhijit Bendale and Terrance Boult. Towards open world recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1893–1902, 2015.

[3] Abhijit Bendale and Terrance E Boult. Towards open set deep networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1563–1572, 2016.

[4] Liang-Chieh Chen, Yi Yang, Jiang Wang, Wei Xu, and Alan L Yuille. Attention to scale: Scale-aware semantic image segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3640–3649, 2016.

[5] Yushi Chen, Zhouhan Lin, Xing Zhao, Gang Wang, and Yanfeng Gu. Deep learning-based classification of hyperspectral data. *IEEE Journal of Selected topics in applied earth observations and remote sensing*, 7(6):2094–2107, 2014.

[6] Chelsea Finn, Pieter Abbeel, and Sergey Levine. Model-agnostic meta-learning for fast adaptation of deep networks. In *International Conference on Machine Learning*, pages 1126–1135. PMLR, 2017.

[7] ZongYuan Ge, Sergey Demyanov, Zetao Chen, and Rahil Garnavi. Generative openmax for multi-class open set classification. *arXiv preprint arXiv:1707.07418*, 2017.

[8] Renlong Hang, Zhu Li, Qingshan Liu, Pedram Ghamisi, and Shuvra S Bhattacharyya. Hyperspectral image classification with attention-aided cnns. *IEEE Transactions on Geoscience and Remote Sensing*, 59(3):2281–2293, 2020.

[9] Mehadi Hassen and Philip K Chan. Learning a neural-network-based representation for open set recognition. In *Proceedings of the 2020 SIAM International Conference on Data Mining*, pages 154–162. SIAM, 2020.

[10] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016.

[11] Kurt Hornik. Approximation capabilities of multilayer feedforward networks. *Neural networks*, 4(2):251–257, 1991.

[12] Minki Jeong, Seokeon Choi, and Changick Kim. Few-shot open-set recognition by transformation consistency. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 12566–12575, 2021.

[13] Saumya Jetley, Nicholas A Lord, Namhoon Lee, and Philip HS Torr. Learn to pay attention. *arXiv preprint arXiv:1804.02391*, 2018.

[14] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

[15] Diederik P Kingma and Max Welling. Auto-encoding variational bayes. *arXiv preprint arXiv:1312.6114*, 2013.

[16] Bo Liu, Hao Kang, Haoxiang Li, Gang Hua, and Nuno Vasconcelos. Few-shot open-set recognition using meta-learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8798–8807, 2020.

[17] Bing Liu, Xuchu Yu, Anzhu Yu, Pengqiang Zhang, Gang Wan, and Ruirui Wang. Deep few-shot learning for hyperspectral image classification. *IEEE Transactions on Geoscience and Remote Sensing*, 57(4):2290–2304, 2018.

[18] Shengjie Liu, Qian Shi, and Liangpei Zhang. Few-shot hyperspectral image classification with unknown classes using multitask deep learning. *IEEE Transactions on Geoscience and Remote Sensing*, 59(6):5085–5102, 2020.

[19] Lawrence Neal, Matthew Olson, Xiaoli Fern, Weng-Keen Wong, and Fuxin Li. Open set learning with counterfactual images. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 613–628, 2018.

[20] Poojan Oza and Vishal M Patel. C2ae: Class conditioned auto-encoder for open-set recognition. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2307–2316, 2019.

[21] Debabrata Pal, Valay Bundele, Biplab Banerjee, and Yogananda Jeppu. Spn: Stable prototypical network for few-shot learning-based hyperspectral image classification. *IEEE Geoscience and Remote Sensing Letters*, 2021.

[22] Pramuditha Perera, Vlad I Morariu, Rajiv Jain, Varun Manjunatha, Curtis Wigington, Vicente Ordonez, and Vishal M Patel. Generative-discriminative feature representations for open-set recognition. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 11814–11823, 2020.

[23] Craig Rodarmel and Jie Shan. Principal component analysis for hyperspectral image classification. *Surveying and Land Information Science*, 62(2):115–122, 2002.

[24] Swalpa Kumar Roy, Gopal Krishna, Shiv Ram Dubey, and Bidyut B Chaudhuri. Hybridsn: Exploring 3-d–2-d cnn feature hierarchy for hyperspectral image classification. *IEEE Geoscience and Remote Sensing Letters*, 17(2):277–281, 2019.

[25] Lei Shu, Hu Xu, and Bing Liu. Doc: Deep open classification of text documents. *arXiv preprint arXiv:1709.08716*, 2017.

[26] Yu Shu, Yemin Shi, Yaowei Wang, Tiejun Huang, and Yonghong Tian. p-odn: prototype-based open deep network for open set recognition. *Scientific reports*, 10(1):1–13, 2020.

[27] Jake Snell, Kevin Swersky, and Richard S Zemel. Prototypical networks for few-shot learning. *arXiv preprint arXiv:1703.05175*, 2017.

[28] Flood Sung, Yongxin Yang, Li Zhang, Tao Xiang, Philip HS Torr, and Timothy M Hospedales. Learning to compare: Relation network for few-shot learning. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1199–1208, 2018.

[29] Christian Szegedy, Wei Liu, Yangqing Jia, Pierre Sermanet, Scott Reed, Dragomir Anguelov, Dumitru Erhan, Vincent Vanhoucke, and Andrew Rabinovich. Going deeper with

convolutions. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1–9, 2015.

[30] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *Advances in neural information processing systems*, pages 5998–6008, 2017.

[31] Oriol Vinyals, Charles Blundell, Timothy Lillicrap, Daan Wierstra, et al. Matching networks for one shot learning. *Advances in neural information processing systems*, 29:3630–3638, 2016.

[32] Svante Wold, Kim Esbensen, and Paul Geladi. Principal component analysis. *Chemometrics and intelligent laboratory systems*, 2(1-3):37–52, 1987.

[33] Sanghyun Woo, Jongchan Park, Joon-Young Lee, and In So Kweon. Cbam: Convolutional block attention module. In *Proceedings of the European conference on computer vision (ECCV)*, pages 3–19, 2018.

[34] Saining Xie, Ross Girshick, Piotr Dollár, Zhuowen Tu, and Kaiming He. Aggregated residual transformations for deep neural networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1492–1500, 2017.

[35] Ryota Yoshihashi, Wen Shao, Rei Kawakami, Shaodi You, Makoto Iida, and Takeshi Naemura. Classification-reconstruction learning for open-set recognition. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4016–4025, 2019.

[36] Chengye Zhang, Jun Yue, and Qiming Qin. Global prototypical network for few-shot hyperspectral image classification. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, 13:4748–4759, 2020.

[37] Zilong Zhong, Jonathan Li, Zhiming Luo, and Michael Chapman. Spectral–spatial residual network for hyperspectral image classification: A 3-d deep learning framework. *IEEE Transactions on Geoscience and Remote Sensing*, 56(2):847–858, 2017.