remote sensing

MDPI

Article

Attention-Based Residual Network with Scattering Transform Features for Hyperspectral Unmixing with Limited Training Samples

Yiliang Zeng [1,2,3,*], Christian Ritz [3], Jiahong Zhao [3] and Jinhui Lan [1,2]

[1] Department of Instrument Science and Technology, School of Automation and Electrical Engineering, University of Science and Technology Beijing, Beijing 100083, China; lanjh@ustb.edu.cn
[2] Beijing Engineering Research Center of Industrial Spectrum Imaging, Beijing 100083, China
[3] School of Electrical, Computer and Telecommunications Engineering, University of Wollongong, Wollongong, NSW 2522, Australia; critz@uow.edu.au (C.R.); jz262@uowmail.edu.au (J.Z.)
* Correspondence: ylzeng@ustb.edu.cn

Received: 27 December 2019; Accepted: 24 January 2020; Published: 26 January 2020

Abstract: This paper proposes a framework for unmixing of hyperspectral data that is based on utilizing the scattering transform to extract deep features that are then used within a neural network. Previous research has shown that using the scattering transform combined with a traditional K-nearest neighbors classifier (STFHU) is able to achieve more accurate unmixing results compared to a convolutional neural network (CNN) applied directly to the hyperspectral images. This paper further explores hyperspectral unmixing in limited training data scenarios, which are likely to occur in practical applications where the access to large amounts of labeled training data is not possible. Here, it is proposed to combine the scattering transform with the attention-based residual neural network (ResNet). Experimental results on three HSI datasets demonstrate that this approach provides at least 40% higher unmixing accuracy compared to the previous STFHU and CNN algorithms when using limited training data, ranging from 5% to 30%, are available. The use of the scattering transform for deriving features within the ResNet unmixing system also leads more than 25% improvement when unmixing hyperspectral data contaminated by additive noise.

Keywords: hyperspectral unmixing; scattering transform; attention mechanism; ResNet; limited training samples

1. Introduction

Hyperspectral image unmixing aims at extracting information about multiple target endmembers from spectral curves that include hundreds of wavebands, which efficiently removes the limit of low spatial resolution of current hyperspectral satellite sensors and retrieves the information of interest in complex ground object conditions, leading to extensive application prospects [1–6]. Unmixing algorithms based on supervised learning are one of the key research directions for hyperspectral image unmixing [7–9]. With the rapid development of artificial intelligence, the structure of deep neural networks based on supervised learning has been recently applied to hyperspectral images [10,11]. However, existing deep learning approaches require a large amount of prior training data to achieve accurate results, while it is difficult to obtain the ground truth for the composition of mixtures of hyperspectral remote sensing images [12]. Therefore, a crucial challenge faced currently is to increase the accuracy of hyperspectral unmixing based on supervised learning when utilizing limited training data.

Recently, deep neural networks that possess great advantages in terms of gaining deep structural features have become the mainstream algorithm in the field of computer vision and image processing, achieving excellent results in various tasks [13–17]. Compared with conventional methods, the deep

learning approaches can automatically obtain high-order features according to the input data, and these algorithms have been used within hyperspectral unmixing research. An autoencoder (AE) cascade approach [18], which concatenates a marginalized denoising autoencoder and a non-negative sparse autoencoder, was employed to solve the unmixing problem in noisy environments. Stacked non-negative sparse autoencoders (NNSAE) [19] for hyperspectral unmixing have been proposed in order to reduce the effect of the outliers and low signal-to-noise ratio scenarios of hyperspectral data. In [20], a weakly-supervised unmixing network, named WU-Net, is proposed. The endmember network is introduced to correct the weights of another SAE unmixing network towards a more accurate unmixing performance. However, the network heavily depends on the endmember extraction results. In this paper, it also shows that the lack of necessary prior knowledge could result in unsatisfactory unmixing results when utilizing the autoencoder-like deep network models. In [10], an end-to-end hyperspectral unmixing method based on the convolutional neural network (CNN) was proposed to extract features and obtain the abundance percentages. The research showed that the multilayer perceptron (MLP) architecture presents a better performance in unmixing than traditional methods. In [21], CNNs were used for the joint optimization of spectral unmixing and subpixel mapping stages. Long-short-term memory (LSTM) networks were also proposed in this article to achieve unmixing, which perform better than the encoder–decoder approach for linear mixtures. Deep spectral convolution networks (DSCNs) were improved in [22] to compute high-level representations, which are then combined with a multinomial mixture kernel to estimate abundances. Although these algorithms have brought good results compared with conventional methods, they are all deep-learning-based approaches and thus require lots of data to train in order to guarantee high accuracy for the obtained model.

However, it is difficult to obtain labeled data for hyperspectral remote sensing images in real-world cases, which leads to the problem of limited training data. In order to solve this problem in hyperspectral image classification, two methods are usually used: generating congeneric data that can be utilized for training as well as constructing network structures suitable for training with a small amount of data. Generative adversarial networks (GAN)-based model was used for semisupervised hyperspectral image classification with a limited number of training samples in [23]. Different transfer learning strategies were presented for limited-samples-based HIS classification and achieved competitive performances [24,25]. In [26], a supervised spectral–spatial residual network (SSRN) was proposed and was shown to be effective in the case of small training samples for hyperspectral image classification. A collaborative representation-based method was applied to deal with the small training set using deep features, which was obtained by deep autoencoder with sample similarity regularization term [27]. In [28], a novel tensor-based model, named rank-1 feedforward neural network (rank-1 FNN), was proposed to train both linear and nonlinear classifiers for hyperspectral data classification. It shows the outperformance in the cases where a small number of training samples is available. In [29], a simplified deep learning model named multigrained network (MugNet) is proposed for limited-sample-based HIS classification, in which the multigrained scanning strategy was utilized to represent the spectral and spatial relationships in different grains. Hyperspectral images classification based on dense convolutional networks with spectral-wise attention mechanism (MSDN-SA) was proposed [30]. It learnt features at different scales and used the spectral-wise attention mechanism to enable accurate training with relatively small training sets. However, in the research area of unmixing, only a few existing articles pay attention to the limited training data problem. The literature [31] discussed the convex optimization problems in hyperspectral unmixing, and constrained sparse regression based on alternating direction algorithms was also proposed for unmixing with limited samples.

The problem caused by limited data also becomes a key reason why applying deep learning to hyperspectral unmixing can be difficult. To provide solutions to the issue, the scattering transform and the attention mechanism are considered to improve the accuracy of hyperspectral unmixing using deep learning algorithm under the condition of limited training samples in this paper. In [32], scattering transform has been shown to possess a better capability for feature extraction than the CNN network,

which achieves high-order features of the hyperspectral data in a cascade manner and has an explicit physical meaning. Additionally, neural network using attention mechanism has the ability to focus on specific parts of information in the feature space [30,33], which is helpful in learning both spatial and spectral features in HSIs.

In this paper, we construct a novel attention-based residual network with scattering transform features (ARNST) learning architecture for hyperspectral unmixing with limited training samples. The major contributions in this paper include three aspects, as follows:

1. A novel network model is proposed, which is a combination of the scattering transform and a deep neural network, such as the CNN and the ResNet. The scattering transform extracts deep-level features from hyperspectral images and the resulting high-order information is processed through neural networks.
2. Hyperspectral unmixing using ResNet and attention-based ResNet are introduced. The attention mechanism is helpful in paying attention to important features in HSIs during learning.
3. Under the condition of limited training data, the proposed approach with only a few parameters to be configured can achieve more accurate results than state-of-the-art methods.
4. When unmixing HSI images corrupted by additive noise, the proposed approach utilizes the scattering transform combined with deep learning to reduce the effect of noise, which is shown to be more robust in terms of suffering a smaller reduction in accuracy compared to the CNN applied directly to the HSI images, which requires retraining with noisy data to achieve satisfactory results.

The rest of this paper is organized as follows. In Section 2, the proposed ARNST method is introduced in detail. In Section 3, we present and discuss the experimental results. Finally, conclusions and future work are described in Section 4.

2. Methods

The proposed ARNST network architecture, as shown in Figure 1, consists of three parts: (1) scattering transform layers, which obtain scattering transform features from original hyperspectral images; (2) deep neural network feature extraction layers, which extract features based on residual networks and attention modules; (3) fully connected layers, which predict the abundance maps for each pixel.

In our work, the scattering transforms are capable of extracting stable and enhanced high-order features in a process that has explicit physical meanings, so that the distinguishable information of original limited hyperspectral samples is enriched. Based on the scattering transform feature maps, the low-complexity residual network is utilized for training. The combination of features and the lower layers has been performed to capture fine features. Moreover, attention modules including channel attention and spatial attention are also utilized to obtain information of interest, which is to enhance relevant features and restrain irrelevant features, so that the key parameters of feature information are kept. Finally, fully connected layers with a soft-max activation function are used to obtain the abundance maps. Thus, this paper combines the above advantages, proposing a method that constructs a network to further extract feature information, improve the accuracy of trained models, and ensure the stability of the network when the resources for training are limited.

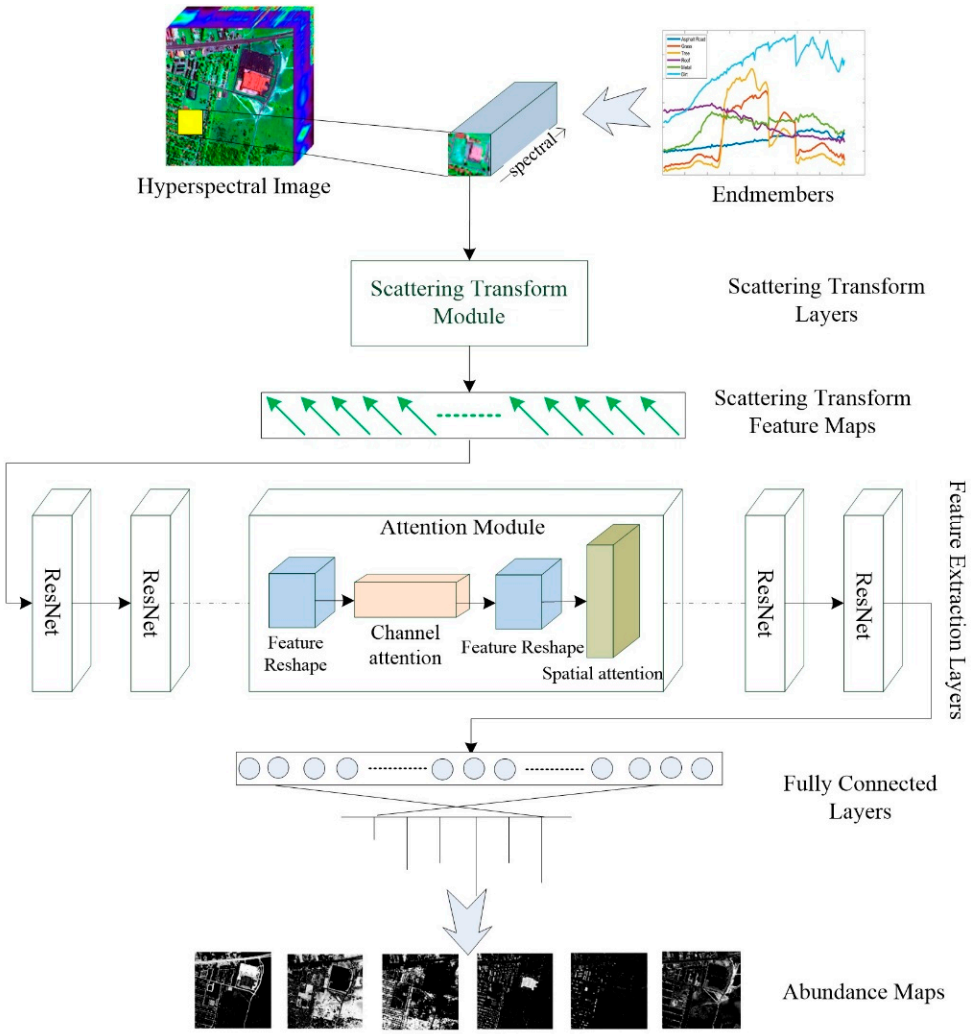

Figure 1. Network architecture of the proposed attention-based residual network with scattering transform features (ARNST).

2.1. Scattering Transform Module

The scattering transform structure with known filters has shown a great potential in lots of different applications, such as feature extraction, classification, unmixing and recognition. Given the vth pixel spectrum as $r_v \in \Re^{1\times l}$, the spectral mixture model f can be simply described as:

$$\begin{aligned} r_v &= f(A_v, X, \varepsilon_v) \\ s.t. &\begin{cases} a_{vk} \geq 0 \\ \sum_{k=1}^{n} a_{vk} = 1 \end{cases} \end{aligned} \tag{1}$$

where $A_v = [a_{v1}, \ldots, a_{vk}, \ldots, a_{vn}] \in \Re^{1\times n}$ is the abundance fractions and a_{vk} denotes the abundance of the kth endmember. $X = [x_1, x_2, \ldots, x_n]^T \in \Re^{n\times l}$ is the endmember matrix of n endmembers and l is the number of bands of hyperspectral data. $\varepsilon_v = [\varepsilon_{v1}, \ldots, \varepsilon_{vk}, \ldots, \varepsilon_{vn}] \in \Re^{1\times n}$ represents the error vector. $a_{vk} \geq 0$ is the abundance non-negativity constraint, while $\sum_{k=1}^{n} a_{vk} = 1$ is the sum-to-one constraint of the abundances.

According to the literature [32], the scattering transform expression of the spectral mixture model can be written as:

$$S(r) = \{S_0(r), S_1(r), \ldots, S_m(r)\} = f(A_v, X, \varepsilon_v) \quad s.t. \begin{cases} a_{vk} \geq 0 \\ \sum_{k=1}^{n} a_{vk} = 1 \end{cases} \tag{2}$$

where $S(r) \in \Re^{1\times1\times L}$ is a collection of the scattering transform coefficient outputs from the zero-order to the mth-order and represents the main information at low frequency bands of the input signal. L is the length of $S(r)$, and r is the spectral vector input.

In summary, as shown in Figure 1, $S_0(r), S_1(r), \ldots, S_m(r)$ are calculated using wavelet or Fourier functions in the scattering transform layers, and then the scattering transform feature maps $S(r)$ can be obtained, which will be used as the input of following deep learning neural networks. Translation invariance, local deformation stability, energy conservation and strong antinoise ability are the main advantages of the scattering transform.

2.2. Deep Neural Network Feature Extraction Module

Due to limited obtainable labels of the ground truth in practical applications of hyperspectral remote sensing images, it is hard for existing algorithms to achieve ideal results. The scattering transform has been shown to extract deep features in a similar way to that of the CNN but utilizes a fixed transform rather than a learnt transform (kernel) as used in the CNN. Moreover, scattering transforms have few characteristic parameters and high stability, which brings better unmixing results than the CNN under conditions of limited training samples. However, state-of-the-art deep learning algorithms usually require a large amount of labeled data for training to have good results. The single scattering transform features also cannot lead to ideal unmixing results when the training data are extremely limited. For example, when unmixing Urban hyperspectral data using 5% data for training, the summation of the root-mean-square error (RMSE) of CNN is 0.5717, while the summated RMSE of scattering transform is 0.4738, both of which are relatively inaccurate. Therefore, there is a need for deeply exploring the capability of hyperspectral feature extraction when the training data are limited, so that the performance of hyperspectral unmixing can be improved.

Previous research [26,30,34] has shown that the ResNet [35] with hundreds of layers can provide effective features and achieve state-of-the-art performance in hyperspectral image classification. This paper focuses on the combination of the scattering transform and the deep neural network, aiming at sufficiently making use of the advantages of both to extract the feature information of interest, realizing accurate abundance estimation under limited training samples.

(1) Residual network based on scattering transform features

Let scattering transform coefficients of vth pixel $S(r) \in \Re^{1\times1\times L}$ be reshaped to three dimensions, i.e., $S \in \Re^{H\times W\times C}$, where H, W and C represent the three dimensions. $S \in \Re^{H\times W\times C}$ is the input of the deep neural network.

One of the key parts within feature extraction layers is the convolution model. The input S or the feature maps F^S are convolved with convolution kernels to obtain the feature maps as follows:

$$F_l^S = F_{l-1}^S * W_l + b_l, \tag{3}$$

where F_{l-1}^S and F_l^S represent the input and output of the lth convolution layer, respectively. When $l = 0$, $F_l^S = S$. Additionally, $*$ refers to the convolution operator. W_l and b_l are the weights and biases of the lth convolution layer.

However, CNN can cause the gradient vanishing problem as the depth of the network increases, particularly for the high-frequency space where scattering transform coefficients become smaller and smaller, which leads to higher probabilities of occurring gradient vanishing. In order to remove this

effect, this paper adopts the residual network, which is trained in the network to ensure the efficiency of the network. Defining the residual function to be learnt as $\chi(F_{l-1}^S) = F_l^S - F_{l-1}^S$, we have:

$$F_l^S = \chi(F_{l-1}^S) + F_{l-1}^S. \tag{4}$$

(2) Scattering transform attention mechanism module

Due to limited obtainable ground truth of hyperspectral images, the algorithm needs to pay much attention to key feature information in the network, so that the accuracy of training can be improved. Thus, the attention mechanism is introduced to address this issue. The convolution block attention module (CBAM) [36] is utilized to recalibrate scattering transform feature maps in the ResNet model. In the attention blocks, channel-wise attention and spatial-based attention are both utilized to train scattering transform features in a three-dimensional structure. Therefore, this method is called scattering transform attention mechanism, which is targeted at enhancing useful scattering transform features and suppressing less useful information. The basic structure of the scattering transform attention mechanism is illustrated in Figure 2.

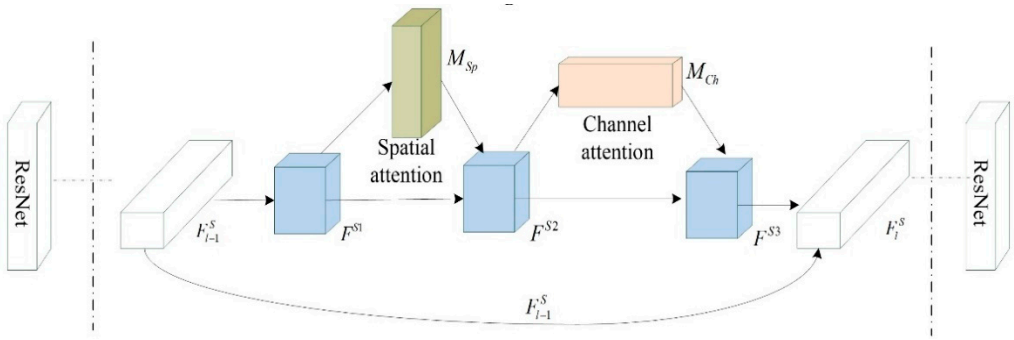

Figure 2. Basic structure of the scattering transform attention mechanism.

F_{l-1}^S will be reshaped to an appropriate dimension $F^{S1} \in \Re^{h \times w \times c}$, which is beneficial for the input of the spatial attention network. The 2D spatial attention map is defined as $M_{Sp} \in \Re^{h \times w \times 1}$, while the 1D channel attention map is defined as $M_{Ch} \in \Re^{1 \times 1 \times c}$. Considering the feature map F_{l-1}^S, the CBAM module can calculate the attention weights in two independent dimensions (spatial and channel), which are then multiplied with F_{l-1}^S to implement the detailing of the feature map. The 2D spatial attention map is used for searching for the mostly focused information in the spatial dimension, while the channel attention is to search for the focal point along the channel axis.

According to [36], average pooling and max pooling operations are applied to the spatial module along the channel axis, and these outputs are concatenated to generate a feature descriptor, which is then used to obtain the spatial attention map M_{Sp} using convolution layers with a filter $f^{7\times7}$, whose size is 7×7, and the sigmoid activation function σ.

$$M_{Sp}(F^{S1}) = \sigma(f^{7\times7}([F_{avg}^{S1}, F_{\max}^{S1}])), \tag{5}$$

where F_{avg}^{S1} and $F_{\max}^{S1}$ represent the average pooling operator and max pooling operator, respectively.

Therefore, the output of the spatial attention module can be described as:

$$F^{S2} = M_{Sp}(F^{S1}) \otimes F^{S1}, \tag{6}$$

where $\otimes$ denotes element-wise multiplication.

For the channel module, both max-pooling outputs and average-pooling outputs are utilized with a shared network that is composed of the multi-layer perceptron (MLP) with one hidden layer. The two

output feature vectors are merged using element-wise summation, and then the sigmoid activation function is utilized to gain the channel attention map:

$$M_{Ch}(F^{S2}) = \sigma(\omega * F^{S2}_{avg} + \omega * F^{S2}_{\max}), \tag{7}$$

where ω is the shared MLP weights.

After that, the output of the channel attention module can be described as:

$$F^{S3} = M_{Ch}(F^{S2}) \otimes F^{S2}. \tag{8}$$

Finally, the output of the whole scattering transform attention mechanism is:

$$F^{S}_{l} = F^{S3} + F^{S}_{l-1}. \tag{9}$$

Comparing Equation (4) with Equation (9), the proposed attention-based residual network with scattering transform features can be achieved, and the residual function can be expressed as:

$$\chi(F^{S}_{l-1}) = F^{S3}. \tag{10}$$

In most cases, the input of the scattering transform attention mechanism is not required to be reshaped, which means that $F^{S1} = F^{S}_{l-1}$, and thus the residual function can be finalized as:

$$\chi(F^{S}_{l-1}) = M_{Ch}([M_{Sp}(F^{S}_{l-1}) \otimes F^{S}_{l-1}]) \otimes [M_{Sp}(F^{S}_{l-1}) \otimes F^{S}_{l-1}]. \tag{11}$$

By calculating the ARNST module, we can realize the goal of further extracting high-order features and eliminating inefficient feature information when there are few training samples.

After executing the last feature extraction layer, the final feature maps F^{S}_{k} can be obtained, followed by achieving the fully connected network. In addition, in order to make sure that the final output can satisfy the abundance non-negativity constraint and sum-to-one constraint, a soft-max activation function is used in the final output layer. The operation principle is illustrated in Figure 3, in which the input is the feature maps based on scattering transform deep residual network features, and the output is the abundance maps.

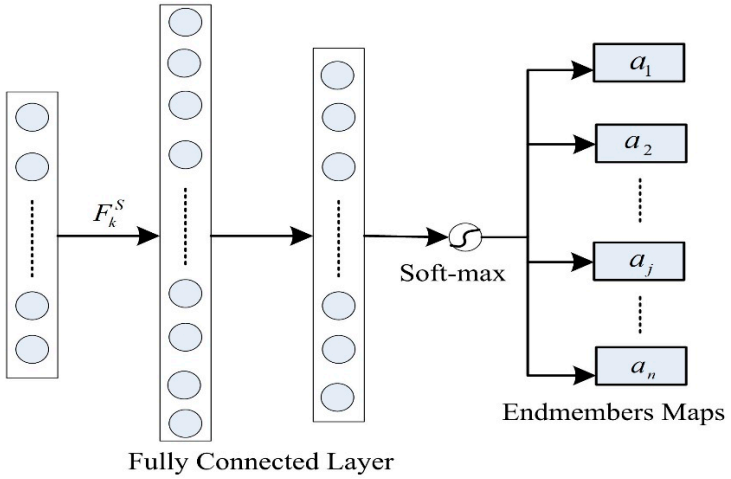

Figure 3. Operation principle of the fully connected layer with scattering transform feature maps.

3. Experimental Results

In this section, we introduce three public datasets used in our experiments. The hyperspectral unmixing performance of the proposed attention-based residual network method with scattering

transform features is presented by comparing with other approaches, which are based on the deep network architecture. All the experiments are implemented with NVIDIA Quadro P4200 GPU, Tensorflow-gpu [37], scikit-learn [38] and Keras [39] with Python 3.6.

3.1. Description of Hyperspectral Datasets

To evaluate the effectiveness of the proposed method, three widely-used real-world hyperspectral data sets, namely Urban, Jasper Ridge and Samson, are selected. These datasets can be openly accessed online [40,41].

(1) Urban: the first dataset contains 307 × 307 pixels and 162 effective spectral bands in the range of 0.4 to 2.5 μm. There are six endmembers in the dataset, including "Asphalt Road", "Grass", "Tree", "Roof", "Metal" and "Dirt".

(2) Jasper Ridge: the second dataset has 100 × 100 pixels, while each pixel contains 198 effective spectral bands with the wavelength ranging from 0.38 to 2.5 μm. There are four endmembers latent in this dataset, including "Road", "Dirt", "Water" and "Tree".

(3) Samson: the third dataset contains 95 × 95 pixels and 156 channels covering the wavelengths from 401 to 889 nm. There are three target endmembers in the dataset, including "Rock", "Tree" and "Water".

3.2. Experimental Setup

(1) Setup for limited training samples

In our experiment, the training set and cross-validation set are generated from the ground truth data, while the testing set is composed of the remaining ground truth samples. As our work focuses on the problem of limited training samples, small training ratios of ground truth data are considered, which are selected from approximately 30% to 0.5% downwards. In this paper, each of the selected datasets corresponding to each ratio is divided into a training set (80% of this set of samples) and a cross-validation set (remaining 20% of these sample).

The details of the training, cross-validation and testing pixels in the Urban dataset are listed in Table 1. As shown in the table, there is a total of 33,153 pixels corresponding to the "Asphalt road" endmember (last column of Table 1) among all the 307 × 307 = 94,249 pixels of the training dataset. For example, when choosing 5% as the training ratio, it means that 3438 and 860 pixels out of the total of 94,249 pixels are utilized for training and cross-validation, respectively, among which 1378 pixels contain the endmember "Asphalt road", and 3256 pixels include the endmember "Grass". Meanwhile, the other 94,249 – 3438 – 860 = 89,951 pixels are used for testing.

Table 1. Number of the training, cross-validation and testing pixels in the Urban dataset.

	Training Ratio					**Total Pixels**
	20%	**10%**	**5%**	**1%**	**0.5%**	
Total training pixels	14,736	7122	3438	491	245	94,249
Total validation pixels	3684	1781	860	123	62	
Total testing pixels	75,829	85,346	89,951	93,635	93,942	
Endmembers	**Number of training and validation pixels for each endmember**					
Asphalt road	9630	4015	1378	171	82	33,153
Grass	11,216	6135	3256	471	233	61,978
Tree	7251	4499	2328	318	159	48,244
Roof	8350	4354	1899	263	134	36,303
Metal	5677	2768	1068	141	66	18,446
Dirt	13,966	6515	3008	390	192	56,082

The training and testing parameters of the datasets of Jasper Ridge and Samson are also set in alignment with the same principle. In all comparative experiments, the training data selected are identical.

(2) Performance evaluation approach

To assess the unmixing performance of the proposed method, the root-mean-square error (RMSE) [42] is used to evaluate the difference between ground truth abundance maps and predicted results.

The performance of the proposed ARNST is compared with two state-of-the-art contrastive methods, namely, the scatter transform framework for hyperspectral unmixing (STFHU) [29] and the CNN [10], which are also based on deep network and perform better than other supervised methods. Meanwhile, the scattering transform plus CNN (STCNN), ResNet, attention-based ResNet (AResN) and ResNet with scattering transform features (RNST) are used for unmixing for the first time.

For CNN and STCNN, four convolution layers and four sampling sublayers are used in the network. The sizes of the convolution layers are set to be 1×5, 1×4, 1×5 and 1×4, while their feature maps are configured to be 3, 6, 12 and 24, respectively [10]. The scattering transform parameters of the Urban dataset are set to be $J = 2$ and $m = 3$, while the parameters of the Jasper Ridge and Samson datasets are set as $J = 3$ and $m = 2$. The parameters of CNN and scattering transform are selected the same as the previous references, where the effectiveness of parameters has been proven in different experiments. The training phase for CNNs in our paper is 500 epochs, while the training phases for ResNet-based approaches are 100 epochs. For the parameter of the number of epochs, the larger value usually obtains the better accuracy. In fact, the training phases for the proposed approaches with 500 epochs can only achieve slightly higher results than 100 epochs. Therefore, we selected 100 epochs in this paper. This also shows that our proposed method can achieve better results using a smaller number of training epochs.

(3) Computational cost of proposed method

The main structures of the ARNST consist of scattering transform, attention mechanism and ResNet. The computational cost should be computed with three parts.

According to [43], the complexity of scattering coefficients is $O(n \log n)$, and the computational cost of self-attention in each layer is $O(n^2 * d)$ [44], where n is the cardinality of the training set and d is the dimension of each sample. In addition, the complexity of neural networks [45] is usually considered as $O(n^5)$. When combining them together, the ResNet plays a major role in complexity calculation. Therefore, the computational cost of proposed ARNST is $O(n^5)$.

(4) ARNST network implementation details

Now we consider the Urban data with 162 spectral bands as an example. The architecture details of the proposed ARNST are described in Table 2.

K refers to the size of the convolving kernel. Forming the ARNST framework, the input of the proposed network is the spectrum with the size of $1 \times 1 \times 162$. Firstly, the scattering transform layer with parameters $m = 2$, $j = 3$ is employed to extend the input and extract the scattering features with the size of $1 \times 1 \times 648$, which is then reshaped to $9 \times 9 \times 8$. Next, several residual network blocks are used to extract deep features from scattering transform feature maps. Each residual block includes four parts: the module of convolutions with different kernel filters with batch normalization and ReLU (Conv-BN-ReLU), channel attention module, spatial attention module and addition module. The kernel filters of convolutions of residual blocks are set as 3×3 in this paper. The number of filters in the residual network layers of Block1, Block2 and Block3 are set to 16, 32 and 64, respectively. It is worth mentioning that "$1 \times 1 \times 16$, 81" represents that an attention weight of 81 is obtained by channels, while "$9 \times 9 \times 1$, 16" means that an attention weight of 16 is obtained by the spatial attention module. Finally, a flattened layer and fully connected layers transform the previous feature maps into an output feature vector with a size of 6, which is the number of endmembers.

Table 2. Architecture details of the proposed ARNST for the Urban dataset.

Layers	K	Network Structure	Output Size
Input	-	-	$1 \times 1 \times 162$
Scattering Transform	$m = 2, j = 3$	Scattering transform	$1 \times 1 \times 648$
Feature Reshape	-	Reshape	$9 \times 9 \times 8$
Residual Block 1	3×3	Conv-BN-ReLU	$9 \times 9 \times 16$
	-	Channel attention	$1 \times 1 \times 16, 81$
	7×7	Spatial attention	$9 \times 9 \times 1, 16$
	-	Add	$9 \times 9 \times 16$
Residual Block 2	3×3	Conv-BN-ReLU	$5 \times 5 \times 32$
	-	Channel attention	$1 \times 1 \times 32, 25$
	7×7	Spatial attention	$5 \times 5 \times 1, 32$
	-	Add	$5 \times 5 \times 32$
Residual Block 3	3×3	Conv-BN-ReLU	$3 \times 3 \times 64$
	-	Channel attention	$1 \times 1 \times 64, 9$
	7×7	Spatial attention	$3 \times 3 \times 1, 64$
	-	Add	$3 \times 3 \times 64$
AveragePooling2D	8×8	Pooling-Flatten	576
FC layer	1×1	FC-softmax	6

As the whole proposed network is carried out in an end-to-end manner, the parameters and effective paths can be learnt automatically. Therefore, the implementation details will be carried out in a similar way for different datasets.

For a fair comparison, the same training and testing samples are utilized for all methods, and all algorithms are executed five times. The average results can reduce the error involved by random selection effects. In addition, the network structures are set to the same width and depth.

3.3. Results for the Urban Dataset

To validate the robustness of the algorithms to noise, Gaussian white noise with a power of 20 dB is added to the samples. Figure 4 shows the comparison of the spectral curves of the original and noisy data, with the blue curve being the original and the red being the one with additive noise. The effects of the noise are evident from the random spikes appearing in the spectrum. Results are evaluated for noisy test samples by utilizing the trained model, which is based on the original data.

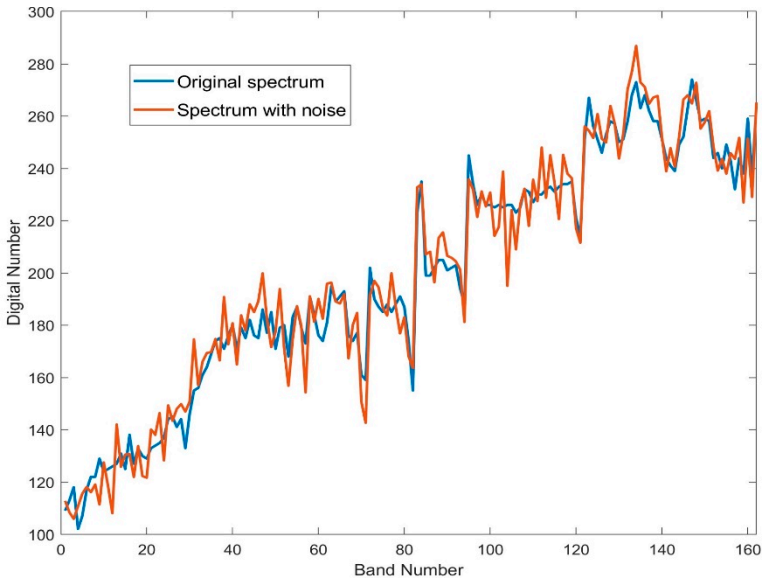

Figure 4. Comparison of the spectral curves of the original and noisy in Urban data.

Table 3 shows the summations of the RMSEs for different training ratios applied to the original and noisy Urban dataset. For the original data, it is clear that the CNN, STFHU and STCNN set of methods result in a lower performance than the set of four ResNet-based approaches. For training ratios of 20%, 10% and 5%, the average difference of the two sets of methods in terms of the summation of the RMSEs is approximately 0.2. When 1% or 0.5% of data are utilized for training, this difference tends to be larger, ranging from 0.4 to 0.7. To be more specific, the CNN method performs the worst for training ratios of 5% and higher, while the STFHU method results in the worst performance for training ratios less than 5%. The combination of the scattering transform and the CNN (STCNN) results in some improvement for training ratios of less than 5% but the resulting RMSEs are still much higher than the ResNet based methods. The separate inclusion of the attention mechanism to ResNet (AResN) and the scattering transform (RNST) both result in lower errors, while the proposed ARNST, which combines both, presents the best hyperspectral unmixing results among all seven considered algorithms across all conditions of training ratios. Moreover, the improvement caused by ARNST shows increasing trends as the training ratio decreases, with the 0.1715 using 20% data for training being the smallest error tested on the original data.

Table 3. Summations of RMSE with small training ratios for the Urban hyperspectral dataset.

	Training Ratio	CNN	STFHU	STCNN	ResNet	AResN	RNST	ARNST
Original	20%	0.4324	0.2930	0.3577	0.1896	0.1716	0.1864	0.1715
	10%	0.4761	0.4125	0.4159	0.2356	0.2222	0.2324	0.2158
	5%	0.5717	0.4738	0.4964	0.3217	0.2913	0.2997	0.2817
	1%	0.8162	0.9208	0.7756	0.5456	0.4913	0.5395	0.4382
	0.5%	1.0803	1.2889	1.0246	0.6095	0.5246	0.5836	0.5129
Noisy	20%	1.1002	0.3011	1.1302	0.2599	0.2511	0.2315	0.2227
	10%	1.1291	0.4192	1.1924	0.3148	0.2970	0.2717	0.2643
	5%	1.3143	0.4812	1.2826	0.3835	0.3776	0.3522	0.3365
	1%	1.1435	0.9308	1.3196	0.6138	0.5382	0.5667	0.4706
	0.5%	1.7159	1.3014	1.1140	0.6504	0.5683	0.5986	0.5468

When adding noise to the original data, the robustness of the algorithms to noise can be validated. It is axiomatic that the CNN and STCNN result in summations of RMSE that are larger than one under all considered conditions, with the 1.7159 resulting from the CNN being the largest error in the whole table. The STFHU previously proposed by our team shows accurate results when the training ratio is no less than 5%, which are equivalent to the performance of the ResNet-based approaches. However, due to the extremely small training ratio, the normal scattering transform method cannot unmix hyperspectral images properly either, and thus the errors increase to be similar to those of the CNN-based methods. The ResNet leads to large improvements of the unmixing accuracy, and the difference of the summation of RMSE to the CNN-based approaches ranges from approximately 0.5 to 0.9. The AResN and RNST methods further enhance the results by incorporating the attention mechanism and the scattering transform,respectively. The ARNST that includes both the attention mechanism and the scattering transform achieves the most satisfied performance under all considered training ratios, with 0.2227 being the smallest summation of RMSE in all conditions of noisy hyperspectral data unmixing. In addition, the increase in the ARNST performance compared to other comparative methods becomes larger when the training ratio is smaller.

Figure 5 shows the ground truth and estimated abundance maps for corresponding endmembers using the proposed method and comparative methods with 5% training ratio. It can be clearly observed that the ResNet-based methods (images of rows 4 to 7 of Figure 5) always achieve abundance maps that are visually much more similar to the original Urban hyperspectral abundance maps with no noise (images of row 1 of Figure 5) compared with the CNN-based approaches (images 2 to 3 of Figure 5). This corresponds with the RMSE results observed in Table 3.

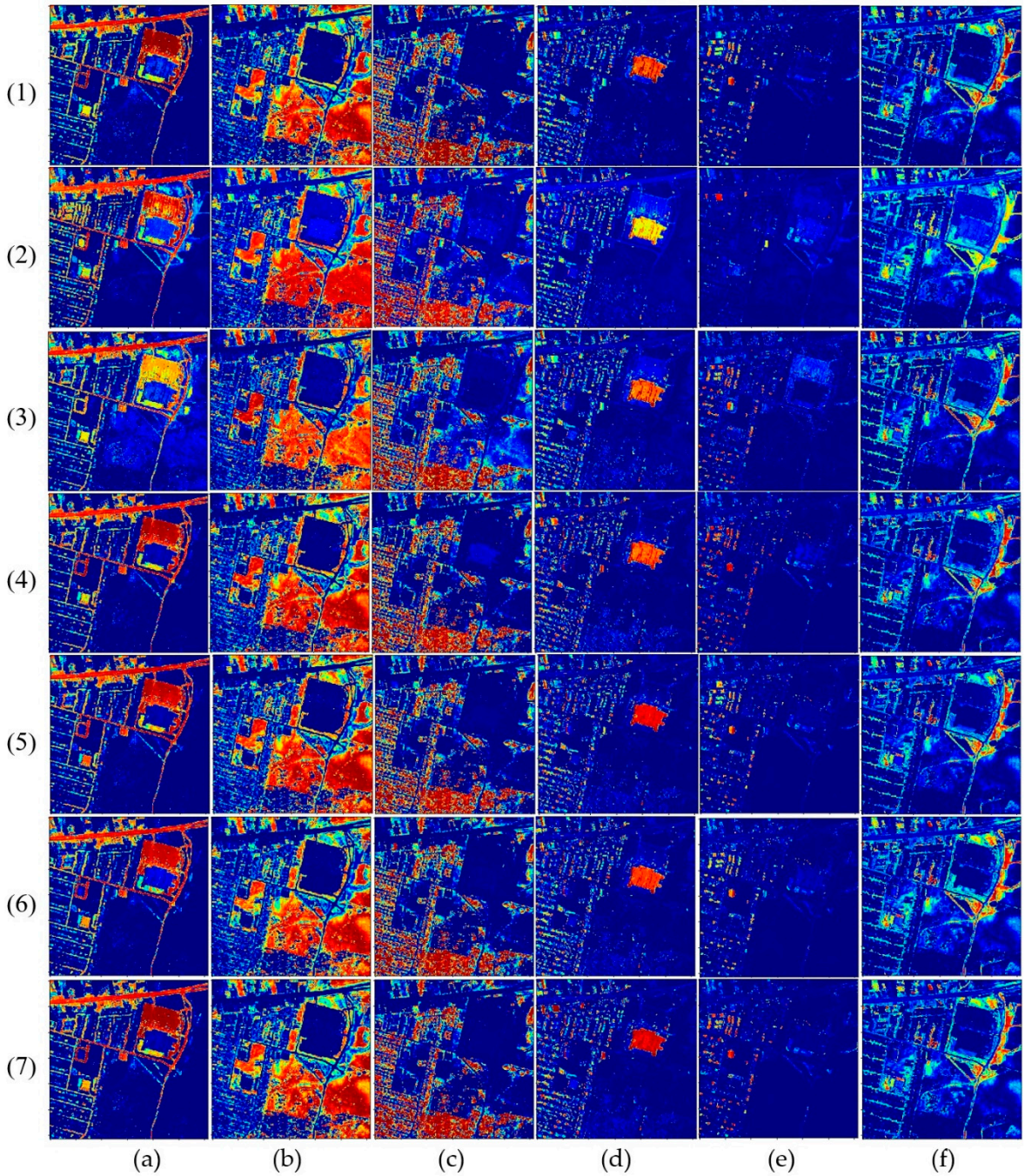

Figure 5. Ground truth and estimated abundance maps of its endmembers using the proposed method and contrastive methods with a 5% training ratio. (**1**) Ground truth; (**2**) CNN; (**3**) STFHU; (**4**) STCNN; ResNet; (**5**) AResN; (**6**) RNST; (**7**) ARNST; (**a**) Asphalt Road; (**b**) Grass; (**c**)Tree; (**d**)Roof; (**e**)Metal; and (**f**) Dirt.

Figure 6 shows the estimated maps for the endmember "Metal" using the noisy data when the training ratio is 5%. There are large differences between the ground truth results (Figure 6a and the results from the CNN-based methods (Figure 6b,d)). This corresponds with the results of Table 3, where RMSE results for the noisy data are far greater than 1. It can also be seen that abundance maps resulting from the STResN method (Figure 6e) are visually more similar to the original ground truth abundance maps (Figure 6a) than the abundance maps resulting from the ResNet method (Figure 6g). This indicates that the scattering transform is able to suppress noise and thus is helpful for stable feature representation of noisy data.

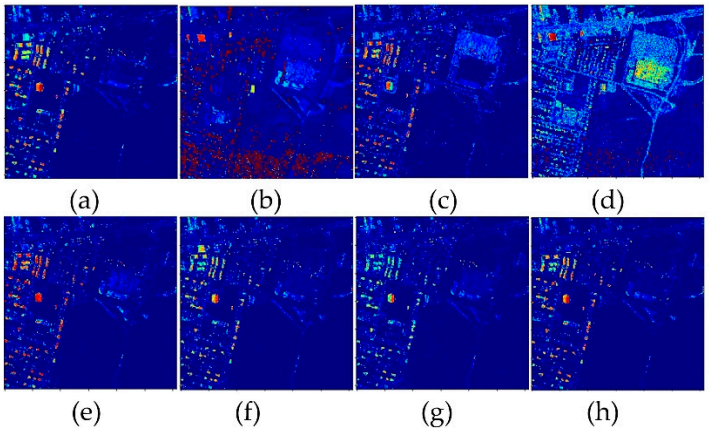

Figure 6. Estimated maps of the endmember "Metal" when 5% samples of the noisy data are utilized for training. (**a**) Ground truth; (**b**) CNN; (**c**) STFHU; (**d**) STCNN; (**e**) ResNet; (**f**) AResN; (**g**) RNST; and (**h**) ARNST.

3.4. Results for the Jasper Ridge Dataset

Figure 7 compares original and noisy versions of spectral curves of the Jasper Ridge hyperspectral dataset, again showing the impact of additive Gaussian noise.

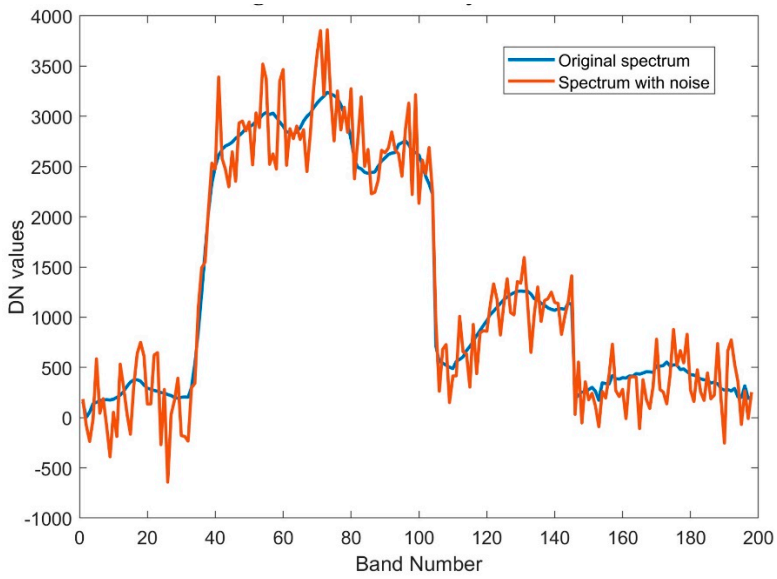

Figure 7. Comparison of spectral curves of the original and noisy data in the Jasper Ridge dataset.

The experimental results for the Jasper Ridge hyperspectral dataset are listed in Table 4. It can be found that considering the processing of original data when the training samples are limited, ResNet, AResN and STResN all lead to similar experimental results that are significantly better than the results obtained for the STFHU and CNN approaches. ARNST achieves the best result among all comparative algorithms, and the summations of RMSE improve from 0.1986 to 0.1087 when comparing with the CNN approach when the training ratio is 30% (a 45.26% increase in accuracy). In the meantime, considering the training ratios of 20% and 10%, the performance of ARNST improves by 50% and 59%, respectively, which indicates that the improvement resulting from the proposed solution becomes larger as the percentage of data used for training decreases. Moreover, it is obvious that, compared to the other approaches, the results of ARNST are also more stable when the training ratio changes.

Table 4. Summations of RMSE with small training ratios for the Jasper Ridge dataset.

	Training Ratio	CNN	STFHU	STCNN	ResNet	AResN	RNST	ARNST
Original	30%	0.1986	0.4856	0.1725	0.1330	0.1162	0.1238	0.1087
	20%	0.2246	0.4952	0.1837	0.1473	0.1240	0.1436	0.1106
	10%	0.3237	0.5647	0.2009	0.1685	0.1487	0.1567	0.1320
	5%	0.3360	0.6026	0.2295	0.2058	0.1795	0.2046	0.1432
Noisy	30%	1.9703	0.5509	1.9539	1.8460	1.6932	0.2967	0.2943
	20%	1.9239	0.5531	2.2657	1.8541	1.8152	0.3356	0.3055
	10%	1.8982	0.5720	2.0489	1.9129	1.7395	0.3866	0.3657
	5%	1.8149	0.6067	2.0297	1.7578	1.7047	0.4102	0.3766

Considering the noisy data, Figure 8 illustrates the estimated abundance maps of the "Dirt" endmember with a 5% training ratio. The estimated maps obtained by training the original data using CNN (Figure 8b), ResNet (Figure 8e) and AResN (Figure 8f) are all unsatisfactory, with the corresponding summations of the RMSE being greater than 1.7. Although the STCNN result (Figure 8d) is not ideal, it can distinguish different endmembers to some extent. In comparison, all methods based on scattering transform features, including STFHU (Figure 8c), RNST (Figure 8g) and ARNST (Figure 8h) all perform well in unmixing, which shows that the scattering transform has advantages in suppressing noise.

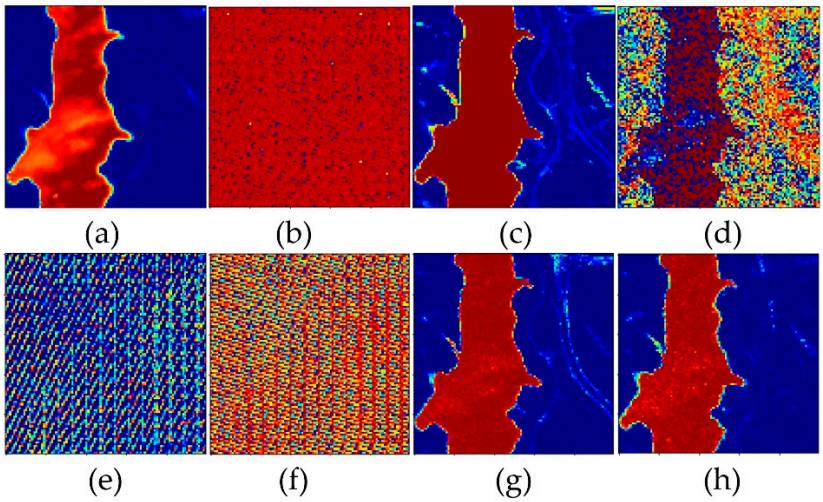

Figure 8. Estimated maps of the endmember "Dirt" when 30% samples of the noisy data are utilized for training. (**a**) Ground truth; (**b**) CNN; (**c**) STFHU; (**d**) STCNN; (**e**) ResNet; (**f**) AResN; (**g**) RNST; and (**h**) ARNST.

When we compare of the summations of RMSE of unmixing noisy images using the Urban and Jasper Ridge datasets under a training ratio of 20%, it can be observed that the CNN is not good at adapting to noise and cannot perform well when unmixing noisy data. Since the scattering transform approaches are robust and stable when representing features, the STFHU and the proposed ARNST have ability to reduce the effects of Gaussian noise, and thus achieve satisfactory results. The proposed ARNST obtains the smallest summations of RMSE, which are 0.2227 for Urban and 0.3055 for Jasper Ridge. Thus, the involvement of the scattering transform in ARNST brings robust performances against noise.

When we compare of noisy hyperspectral unmixing performances of the STFHU, AResN and ARNST under different training ratios, the proposed method leads to better performances than STFHU across all conditions, with the average percentage of improvement being more than 25%. Therefore, the ARNST shows better hyperspectral unmixing results than the scattering transform approach after combining the attention mechanism and the ResNet. In addition, the AResN algorithm performs well

for the Urban dataset but shows dissatisfactory results for Jasper Ridge. This also proves that the proposed ARNST can obtain more stable results than other comparative algorithms.

3.5. Results for the Samson Dataset

The Samson dataset is also used to validate the proposed algorithms. The summations of RMSE are listed in Table 5. It can be seen that ARNST method achieves the best results among all comparative methods, which are 0.0751 using 20% of data for training and 0.5255 for the 10% case. The CNN-based approaches both lead to large errors, while the ResNet and the attention mechanism bring more accurate unmixing results than the two algorithms based on CNN. The STFHU without using the proposed methods cannot work well either.

Table 5. Summations of RMSE with small training ratios for the Samson hyperspectral dataset.

	Training Ratio	**CNN**	**STFHU**	**STCNN**	**ResNet**	**AResN**	**RNST**	**ARNST**
Original	30%	0.1383	0.3738	0.1221	0.0986	0.0800	0.0595	0.0439
	20%	0.1658	0.8253	0.1466	0.1168	0.0866	0.1047	0.0751
	10%	0.8930	1.6998	0.7241	0.7416	0.9706	0.9819	0.5255
	5%	0.9393	1.8950	1.2546	1.1362	0.9849	1.2985	0.8546
Noisy	30%	1.1581	0.7345	0.9467	0.9284	0.8741	0.1785	0.1320
	20%	1.7946	0.8983	1.8845	1.1437	1.2148	0.3514	0.3372
	10%	1.6485	1.8237	1.7618	1.6752	1.7553	1.2196	1.0743
	5%	2.0059	1.9029	1.9864	1.7114	1.8828	1.3821	1.3608

3.6. Results when Training on Noisy Data

In order to validate the robustness of algorithms to different input data, this paper also attempts to train the models using the noisy training data before utilizing the noisy data for testing, leading to the results in Figure 9. All the considered algorithms achieve more accurate unmixing results by training using the noisy data compared to the results obtained when training the original data. It is noted that the greatest improvement is obtained for the CNN-based approaches, which indicates that these methods are not as robust to noise when compared to the other methods. The ARNST achieves the smallest summation of RMSE, proving that the proposed approach possesses the best robustness to noise. It is also noted that the scattering-transform-based approaches (ARNST, RNST and STFHU) achieve results that are most similar to those obtained when testing noisy data using models trained with original (non-noisy) data). This indicates that the scattering transform helps in ensuring that the unmixing system is robust to noise.

From Tables 3–5 and considering all algorithms, the summation of RMSE becomes larger as the amount of training samples becomes smaller, which demonstrates the requirement to have a sufficient amount of training samples to be able to accurately train the models of these approaches. Obviously, the proposed ARNST can provide the best and most stable performance in all cases. There are two key reasons that the proposed method achieves better results than other contrastive methods. Firstly, the scattering transform possesses a multilayer structure which generates highly descriptive features of the hyperspectral images. In addition, the scattering transform is able to distinguish noise from original spectral signal, which reduces the effect of endmember variability. Secondly, the attention mechanism in deep learning networks helps to focus on the interesting and important information of scattering transform features using less training data. This will further suppress the noise information. Therefore, the combination of scattering transform features and attention-based deep learning network within the proposed ARNST approach can effectively address the problem of hyperspectral image unmixing using neural networks trained on limited ground truth data and with endmember variability.

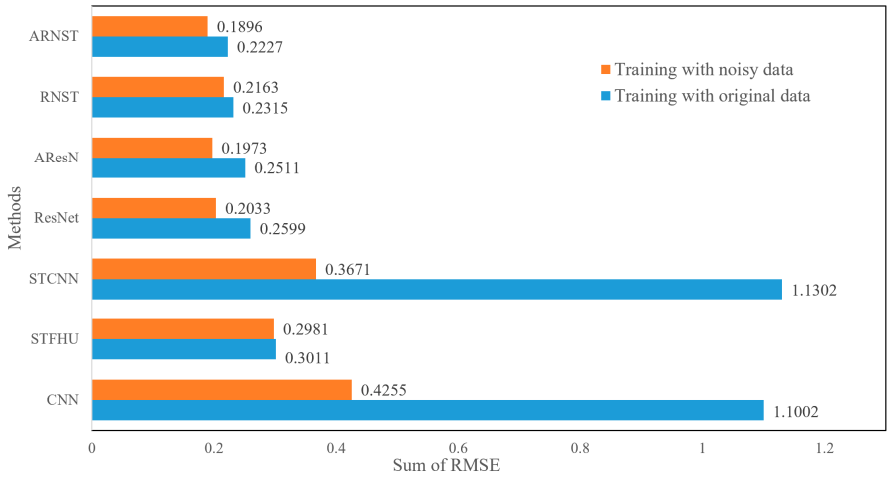

Figure 9. Performance comparisons of hyperspectral unmixing training with the original data and noisy data.

4. Conclusions

In this paper, a novel attention-based residual network framework with scattering transform features is proposed for hyperspectral data unmixing with limited training samples. The scattering transform is utilized to extract high-order deep features that benefit the robustness of trained models of the residual neural networks. Furthermore, the inclusion of attention mechanism into the model helps to focus the residual neural network on the feature information of interest and results in maximized performance. The proposed framework makes full use of the advantages of the scattering transform, the residual neural network, and the attention mechanism. The resulting ARNST possesses the capability of suppressing noise and provides significantly improved accuracy of hyperspectral unmixing when extremely limited training data are available, which is helpful in real-world applications where the data are usually corrupted by additive noise and the access to lots of labeled training data is not practical. Experiments on public datasets have demonstrated that the proposed framework achieves more accurate and stable results than state-of-the-art algorithms in hyperspectral unmixing when noise is present and the training ratio is extremely low. When comparing with CNN and STFHU approaches, the proposed ARNST approach can obtain at least 40% and 25% increases in performance for original and noisy hyperspectral datasets, respectively.

Future work includes further improving the spatial–spectral joint 3D network structure to provide more accurate hyperspectral unmixing results under extreme conditions.

Author Contributions: Conceptualization, Y.Z. and C.R.; methodology, Y.Z. and C.R.; software, Y.Z.; validation, Y.Z., J.Z., and C.R.; writing—original draft preparation, Y.Z. and J.Z.; writing—review and editing, C.R. and J.L.; funding acquisition, Y.Z. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by the National Natural Science Foundation of China under grant number 61801018, the China Scholarship Council under grant number 201906465009, the Advance Research Program under grant number 6140452010101, and the Fundamental Research Funds for the Central Universities under grant number FRF-BD-19-002A.

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

References

1. Landgrebe, D. Hyperspectral image data analysis. *IEEE Signal. Process. Mag.* **2002**, *19*, 17–28. [CrossRef]
2. Bioucas-Dias, J.M.; Plaza, A.; Camps-Valls, G.; Scheunders, P.; Nasrabadi, N.; Chanussot, J. Hyperspectral remote sensing data analysis and future challenge. *IEEE Geosci. Remote Sens. Mag.* **2013**, *1*, 6–36. [CrossRef]
3. Bioucas-Dias, J.M.; Plaza, A.; Dobigeon, N.; Parente, M.; Du, Q.; Gader, P.; Chanussot, J. Hyperspectral unmixing overview: Geometrical, statistical, and sparse regression-based approaches. *IEEE J. Sel. Top. Appl. Earth Obs. Remote Sens.* **2012**, *5*, 354–379. [CrossRef]

4. Zou, J.; Lan, J.; Shao, Y. A Hierarchical Sparsity Unmixing Method to Address Endmember Variability in Hyperspectral Image. *Remote Sens.* **2018**, *10*, 738. [CrossRef]
5. Parra, L.C.; Spence, C.; Sajda, P.; Ziehe, A.; Müller, K.R. Unmixing Hyperspectral Data. In *Advances in Neural Information Processing Systems*; Available online: http://papers.nips.cc/paper/1714-unmixing-hyperspectral-data.pdf (accessed on 15 January 2020).
6. Iordache, M.D.; Bioucas-Dias, J.M.; Plaza, A. Total variation spatial regularization for sparse hyperspectral unmixing. *IEEE Trans. Geosci. Remote Sens.* **2012**, *50*, 4484–4502. [CrossRef]
7. Heylen, R.; Parente, M.; Gader, P. A review of nonlinear hyperspectral unmixing methods. *IEEE J. Sel. Top. Appl. Earth Obs. Remote Sens.* **2014**, *7*, 1844–1868. [CrossRef]
8. Xu, X.; Shi, Z.; Pan, B. A supervised abundance estimation method for hyperspectral unmixing. *Remote Sens. Letter.* **2018**, *9*, 383–392. [CrossRef]
9. Nguyen, N.H.; Chen, J.; Richard, C.; Honeine, P.; Theys, C. Supervised nonlinear unmixing of hyperspectral images using a pre-image methods. *Eur. Astron. Soc. Publ. Ser.* **2013**, *59*, 417–437. [CrossRef]
10. Zhang, X.; Sun, Y.; Zhang, J.; Wu, P.; Jiao, L. Hyperspectral unmixing via deep convolutional neural networks. *Ieee Geosci. Remote Sens. Lett.* **2018**, *15*, 1755–1759. [CrossRef]
11. Palsson, F.; Sigurdsson, J.; Sveinsson, J.R.; Ulfarsson, M.O. Neural network hyperspectral unmixing with spectral information divergence objective. In Proceedings of the 2017 IEEE International Geoscience and Remote Sensing Symposium (IGARSS), Fort Worth, TX, USA, 23–28 July 2017; pp. 755–758.
12. Plaza, J.; Plaza, A.; Perez, R.; Martinez, P. On the use of small training sets for neural network-based characterization of mixed pixels in remotely sensed hyperspectral images. *Pattern Recogn.* **2009**, *42*, 3032–3045. [CrossRef]
13. Shen, D.; Wu, G.; Suk H, I. Deep learning in medical image analysis. *Ann. Rev. Biomed. Engineer.* **2017**, *19*, 221–248. [CrossRef] [PubMed]
14. Zhao, W.; Du, S. Spectral–spatial feature extraction for hyperspectral image classification: A dimension reduction and deep learning approach. *IEEE Trans. Geosci. Remote Sens.* **2016**, *54*, 4544–4554. [CrossRef]
15. Bayar, B.; Stamm M, C. A deep learning approach to universal image manipulation detection using a new convolutional layer. In Proceedings of the 4th ACM Workshop on Information Hiding and Multimedia Security, Vigo, Spain, 20–22 June 2016; pp. 5–10.
16. Druzhkov P, N.; Kustikova V, D. A survey of deep learning methods and software tools for image classification and object detection. *Pattern Recogn. Image Analys.* **2016**, *26*, 9–15. [CrossRef]
17. Pouyanfar, S.; Sadiq, S.; Yan, Y.; Tian, H.; Tao, Y.; Reyes, M.P.; yengar, S.S. A survey on deep learning: Algorithms, techniques, and applications. *ACM Comput. Surve.* **2019**, *51*, 92. [CrossRef]
18. Guo, R.; Wang, W.; Qi, H. Hyperspectral image unmixing using autoencoder cascade. In Proceedings of the Seventh Workshop on Hyperspectral Image and Signal Processing: Evolution in Remote Sensing (WHISPERS), Tokyo, Japan, 2–5 June 2015; pp. 1–4.
19. Su, Y.; Marinoni, A.; Li, J.; Plaza, J.; Gamba, P. Stacked nonnegative sparse autoencoders for robust hyperspectral unmixing. *IEEE Geosci. Remote Sens. Lett.* **2018**, *15*, 1427–1431. [CrossRef]
20. Hong, D.; Chanussot, J.; Yokoya, N.; Heiden, U.; Heldens, W.; Zhu, X.X. WU-Net: A Weakly-Supervised Unmixing Network for Remotely Sensed Hyperspectral Imagery. In Proceedings of the 2019 IEEE International Geoscience and Remote Sensing Symposium (IGARSS 2019), 28 July–2 August 2019; pp. 373–376.
21. Arun, P.; Buddhiraju, K.; Porwal, A. CNN based sub-pixel mapping for hyperspectral images. *Neurocomputing* **2018**, *311*, 51–64. [CrossRef]
22. Ozkan, S.; Akar, G.B. Improved Deep Spectral Convolution Network For Hyperspectral Unmixing With Multinomial Mixture Kernel and Endmember Uncertainty. *arXiv* **2018**, arXiv:1808.01104.
23. He, Z.; Liu, H.; Wang, Y.; Hu, J. Generative adversarial networks based semi-supervised learning for hyperspectral image classification. *Remote Sens.* **2017**, *9*, 1042. [CrossRef]
24. Yang, J.; Zhao, Y.Q.; Chan, C.W. Learning and transferring deep joint spectral–spatial features for hyperspectral classification. *IEEE Trans. Geosci. Remote Sens.* **2017**, *55*, 4729–4742. [CrossRef]
25. Zhang, H.; Li, Y.; Jiang, Y.; Wang, P.; Shen, Q.; Shen, C. Hyperspectral Classification Based on Lightweight 3-D-CNN With Transfer Learning. *IEEE Trans. Geosci. Remote Sens.* **2019**, *57*, 5813–5828. [CrossRef]
26. Zhong, Z.; Li, J.; Luo, Z.; Chapman, M. Spectral–spatial residual network for hyperspectral image classification: A 3-d deep learning framework. *IEEE Trans. Geosci. Remote Sens.* **2018**, *56*, 847–858. [CrossRef]

27. Ma, X.; Wang, H.; Geng, J. Spectral-spatial classification of hyperspectral image based on deep auto-encoder. *Ieee J. Sel. Top. Appl. Earth Obs. Remote Sens.* **2016**, *9*, 4073–4085. [CrossRef]
28. Makantasis, K.; Doulamis, A.D.; Doulamis, N.D.; Nikitakis, A. Tensor-based classification models for hyperspectral data analysis. *IEEE Trans. Geosci. Remote Sens.* **2018**, *56*, 6884–6898. [CrossRef]
29. Pan, B.; Shi, Z.; Xu, X. MugNet: Deep learning for hyperspectral image classification using limited samples. ISPRS J. Photogramm. *Remote Sens.* **2018**, *145*, 108–119.
30. Fang, B.; Li, Y.; Zhang, H.; Chan, J.W. Hyperspectral Images Classification Based on Dense Convolutional Networks with Spectral-Wise Attention Mechanism. *Remote Sens.* **2019**, *11*, 159. [CrossRef]
31. Bioucas-Dias, J.M.; Figueiredo, M.A. Alternating direction algorithms for constrained sparse regression: Application to hyperspectral unmixing. In Proceedings of the 2010 2nd Workshop on Hyperspectral Image and Signal Processing: Evolution in Remote Sensing, Reykjavik, Iceland, 14–16 June 2010; pp. 1–4.
32. Zeng, Y.; Ritz, C.; Zhao, J.; Lan, J. Scattering Transform Framework for Unmixing of Hyperspectral Data. *Remote Sens.* **2019**, *11*, 2868. [CrossRef]
33. Mei, X.; Pan, E.; Ma, Y.; Dai, X.; Huang, J.; Fan, F.; Du, Q.; Zheng, H.; Ma, J. Spectral-Spatial Attention Networks for Hyperspectral Image Classification. *Remote Sens.* **2019**, *11*, 963. [CrossRef]
34. Cai, Y.; Dong, Z.; Cai, Z.; Liu, X.; Wang, G. Discriminative Spectral-Spatial Attention-Aware Residual Network for Hyperspectral Image Classification. In Proceedings of the 2019 10th Workshop on Hyperspectral Imaging and Signal Processing: Evolution in Remote Sensing (WHISPERS), Amsterdam, The Netherlands, 24–26 September 2019; pp. 1–5.
35. He, K.; Zhang, X.; Ren, S.; Sun, J. Deep residual learning for image recognition. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Las Vegas, NV, USA, 26 June–1 July 2016; pp. 770–778.
36. Woo, S.; Park, J.; Lee, J.Y.; also, K.I. Cbam: Convolutional block attention module[C]. In Proceedings of the European Conference on Computer Vision (ECCV), Munich, Germany, 8–14 September 2018; pp. 3–19.
37. TensorFlow Software. Available online: https://www.tensorflow.org (accessed on 20 July 2019).
38. Scikit-Learn Software. Available online: https://scikit-learn.org (accessed on 20 July 2019).
39. Keras Software. Available online: https://keras.io (accessed on 20 July 2019).
40. Hyperspectral Unmixing Datasets & Ground Truths. Available online: http://www.escience.cn/people/feiyunZHU/Dataset_GT.html (accessed on 10 August 2019).
41. Zhu, F.; Wang, Y.; Xiang, S.; Fan, B.; Pan, C. Structured sparse method for hyperspectral unmixing. *ISPRS J. Photogram. Remote Sens.* **2014**, *88*, 101–118. [CrossRef]
42. Shao, Y.; Lan, J. A Spectral Unmixing Method by Maximum Margin Criterion and Derivative Weights to Address Spectral Variability in Hyperspectral Imagery. *Remote Sens.* **2019**, *11*, 1045. [CrossRef]
43. Andén, J.; Mallat, S. Deep scattering spectrum. *IEEE Trans. Signal Process.* **2014**, *62*, 4114–4128. [CrossRef]
44. Complexity of Self-Attention. Available online: https://www.cnblogs.com/nxf-rabbit75/p/11945195.html (accessed on 15 January 2020).
45. Computational Complexity of Neural Networks. Available online: https://kasperfred.com/series/computational-complexity/computational-complexity-of-neural-networks. (accessed on 15 January 2020).