Deep Feature Fusion via Two-Stream Convolutional Neural Network for Hyperspectral Image Classification

Xian Li, *Student Member, IEEE*, Mingli Ding, and Aleksandra Pižurica, *Senior Member, IEEE*

***Abstract*—The representation power of Convolutional Neural Network (CNN) models for hyperspectral image (HSI) analysis is in practice limited by the available amount of the labelled samples, which is often insufficient to sustain deep networks with many parameters. We propose a novel approach to boost the network representation power with a two stream 2D-CNN architecture. The proposed method extracts simultaneously spectral features, local spatial and global spatial features with two 2D-CNN networks, and makes use of channel correlations to identify the most informative features. Moreover, we propose a layer-specific regularization and a smooth normalization fusion scheme to adaptively learn the fusion weights for the spectral-spatial features from the two parallel streams. An important asset of our model is simultaneous training of the feature extraction, fusion and classification processes with the same cost function. Experimental results on several hyperspectral data sets demonstrate the efficacy of the proposed method compared to the state-of-the-art in the field.**

***Index Terms*—Feature fusion, convolutional neural networks (CNN), hyperspectral image (HSI) classification, squeeze-and-excitation.**

I. Introduction

HYPERSPECTRAL remote sensing remains to be one of the key technologies for the Earth observation, and also one of the most demanding and challenging ones for data processing and analysis [1, 2]. Captured with hundreds of contiguous and narrow spectral bands, hyperspectral images (HSIs), enable more accurate discrimination between different materials in the scene than conventional panchromatic and multi-spectral remote sensing images [3]. Hence, the technology has became widely adopted in a range of applications including defence and security [4], agriculture [5], geology [6], ocean [7] and environmental monitoring [8].

While different materials can typically be distinguished based on their spectral signatures, scene classification based on spectral information alone is often not accurate enough. Various factors, such as spatial variability of spectral signatures [9], and spectral noise increase the intraclass variability. If the interclass variability is small, it is difficult to differentiate one class from another [10]. With the improvement of spatial resolution in HSI, it becomes natural to make use of the spatial information as well [11]. For example, knowing that adjacent pixels in homogeneous areas are likely to belong to the same class, we can improve the results in precise mapping. It is generally agreed that combined spectral-spatial classification improves the accuracy significantly compared to spectral classification alone [12].

Feature extraction and feature fusion are the two crucial steps in spectral-spatial classification. Various approaches have been proposed to incorporate spatial context into feature extraction, using e.g., segmentation [13, 14], morphological filters [15], Markov Random Field (MRF) models [16], and texture features [17]. State-of-the-art feature extraction approaches include multiple kernel learning [18], sparse representation [19–23], and active learning [24]. Recent explosion of deep learning has transformed feature extraction. Instead of hand-crafting features based on domain-specific expert knowledge and a lot of parameter tuning, new deep learning approaches learn automatically a hierarchical feature representation that is optimally suited for complex classification and recognition tasks.

Deep learning models for feature extraction from HSIs can be grouped in four main categories: models employing stacked auto-encoders (SAEs) [25, 26], deep belief networks (DBNs) [27, 28], recurrent neural networks (RNNs) [29, 30], and Convolutional Neural Networks (CNNs) [31–38]. Compared with the other deep learning models, CNNs facilitate extraction of spatial features since they can operate directly on image patches, without flattening them to one dimension. Besides, CNNs reduce hugely the number of learning parameters compared to fully connected networks with the same number of hidden units, with their local receptive fields and shared-weights architecture, which is the main reason for their dominance in image/video processing.

Spectral-spatial feature extraction and classification based on CNN methods can be generally divided into two categories. The first category extracts jointly spectral-spatial features using 3D filtering. For instance, Chen *et al.* [31] proposed a 3D-CNN model with a large receptive field in the spectral domain and a small receptive field in the spatial domain to extract the integrated spectral-spatial features. Similarly, in the 3D-CNN framework of Li *et al.* [39], the spectral-spatial features are extracted simultaneously, taking full advantage of the structural characteristics of the 3D HSI data. Zhong *et al.* [40]

This work was partially supported by the China Scholarship Council, and by the Research Foundation Flanders (FWO) under the grant G.OA26.17N.

X. Li is with the School of Instrumentation Science and Engineering, Harbin Institute of Technology, 150001 Harbin, China, and also with the Department of Telecommunications and Information Processing, UGent-GAIM, Ghent University, 9000 Ghent, Belgium (e-mail: xianli0511@gmail.com).

M. Ding is with the School of Instrumentation Science and Engineering, Harbin Institute of Technology, 150001 Harbin, China (e-mail: dingml@hit.edu.cn).

A. Pižurica is with the Department of Telecommunications and Information Processing, imec-UGent-GAIM, Ghent University, 9000 Ghent, Belgium (e-mail: Aleksandra.Pizurica@UGent.be)

introduced residual learning to 3D-CNN to consecutively learn discriminative features from abundant spectral signatures and spatial contexts in HSIs. However, 3D-CNN feature extraction and classification methods often exploit shallow networks to avoid overfitting due to an additional filter in the spectral dimension compared to 2D-CNN. This limits their ability in exploiting the available spectral-spatial information, and the resulting classification maps tend to be oversmoothed [41].

The second large category of feature extraction methods extracts the spectral features and the spatial features separately, and fuses them subsequently. Most of spatial feature extraction methods are CNN-based methods inspired by computer vision models, while the spectral feature extraction methods are more diverse including balanced local discriminant embedding [42], SAE [43], and stacked denosing autoencoder (SdAE) [44]. As opposed to the above-described methods, which apply different architectures in their spatial and spectral stream, several recent works including [41, 45] formulated unified approaches to spectral-spatial feature extraction, with an end-to-end training strategy and a uniform objective function. Spectral feature extraction in all these methods is based on 1D-CNN.

Next to the spectral and spatial feature extraction, feature fusion is another key step in the classification task. CNN-based methods typically use one or more fully connected layers with a rectified linear unit (ReLU) non-linear activation function to fuse the extracted features [31, 34, 44, 46]. For example, Song *et al.* [46] proposed a deep feature fusion network by introducing residual learning to increase the network depth. The features extracted from multiple (low-level, middle-level, and high-level) layers as complementary information were fused by global average pooling (GAP) and the fully connected layers with ReLU. We hypothesize that the fused features using the ReLU in the fully connected layers tend to blow up (the output range is [0, inf]), due to which some detail features may be lost.

Although the above described CNN-based methods demonstrated huge success in HSI processing, two important challenges remain. Firstly, a large number of labelled training samples is required to obtain a satisfactory performance. In practice, a limited amount of training data and unbalanced samples constrain the network depth and width, reducing the feature extraction capability. Some strategies like data augmentation [34, 47, 48] and transfer learning [45, 49] are adopted to alleviate this problem to a certain extent, but the inherent limitation of the models remains a limiting factor for the network performance. The second challenge is how to exploit the spectral and spatial information more effectively. Although various approaches have been proposed, this question remains relevant, both theoretically and practically. In [34], it was pointed out that a single input architecture has strong limitations in heterogeneous area, and thus the authors proposed using multiple inputs based on six diverse-region to better extract spectral-spatial features. This led to an improved performance compared to most of single input methods [32, 47, 49], but the diverse regions construction is time-consuming and each region employs similar shallow networks.

We address the challenges mentioned above, and propose a novel two-stream spectral and spatial feature extraction, feature fusion, and classification architecture based on 2D-CNN. Specifically, we develop a deep learning framework, which extracts simultaneously local and global spatial-spectral features via two streams that operate in parallel. The first stream is a shallow 2D-CNN that extracts spectral and local spatial correlation features from a relatively small image patch. The second stream is a deep 2D-CNN, which extracts more complex global spatial structure from a relatively large image patch. Hence, the complete network extracts spectral, local spatial and global spatial features. Inspired by squeeze-and-excitation (SE) networks [50] that were recently introduced in the field of computer vision, we introduce a related SE module to further enhance the feature extraction capability of the two streams. In the fusion stage, our method learns adaptively the fusion weights to form joint features. The output labels are then predicted by a softmax layer.

The main contributions of this paper are as follows:

1) We propose a novel two-stream CNN architecture for HSI classification, which extracts spectral, *local* spatial and *global* spatial information in parallel. In this approach, the feature extraction, fusion, and classification are trained in an end-to-end manner under a unified objective function.

2) We introduce an effective approach for improving the spectral-spatial feature extraction capability of the two parallel streams based on interchannel correlations and the so-called squeeze-and-excitation (SE) concept. This is especially important in practice where the actual depth of the feature extraction streams is limited by the available amount of training data. To this end, we derive a formal approach for incorporating the SE concept into HSI spectral-spatial classification.

3) We propose a layer-specific regularization and smooth normalization fusion scheme, which adaptively controls the fusion weights and better fuses the spectral-spatial features.

4) We embed a 2D-CNN into the feature extraction stream. Different from conventional spectral feature extraction streams which were always based on 1D-CNN or other 1D methods, our proposed feature extraction stream operates on small image patches and extracts simultaneously spectral and local spatial features. Moreover, we combine shallow and deep networks to extract optimally both spectral and spatial information content. This configuration effectively makes use of multi-scale spectral-spatial information and fuses features at different depths.

The rest of this paper is organized as follows. Section II reviews the basic concepts of CNN, and ideas behind residual learning and SE approaches. Section III introduces the proposed method. A thorough experimental evaluation and a discussion of the results are given in Section IV and Section V draws the conclusion of this work.

II. Background and Related Works

A. CNN

Among all deep learning models including SAEs, DBNs, and RNNs, CNNs have been by far the most extensively employed in computer vision problems, mainly due to their efficiency with local connections, shared weights, and flexibility, admitting different volumes of neurons.

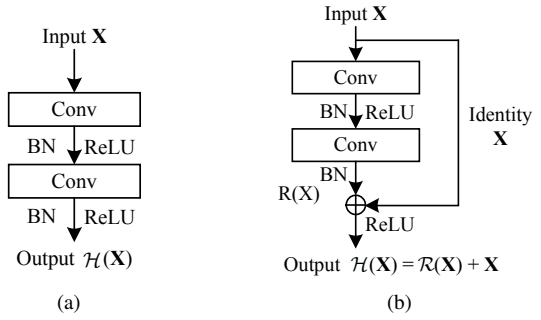


Fig. 1. The architectures of (a) two convolutional layers, (b) residual module.

The basic components of CNN models include convolutional layer, pooling layer, and fully connected layer. The convolutional layer usually contains several convolutional kernels, with which different feature maps are computed and then fed to a nonlinear activation function. Let $\mathbf{X} \in \mathbb{R}^{H' \times W' \times N'}$ be the input of a convolutional layer, where $H' \times W'$ is spatial size, and N' is the number of channels. Given the N convolutional kernels $\mathbf{W}_i$ (bias terms are omitted), the output $\mathcal{F}_i(\mathbf{X}) \in \mathbb{R}^{H \times W \times N}$ is computed as

$$\mathcal{F}_i(\mathbf{X}) = \delta(\mathbf{W}_i * \mathbf{X}) \tag{1}$$

where δ denotes the activation function. ReLU [51] is among the most often used activation functions in CNN models because it offers faster convergence and better performance than the traditional saturated activation functions such as sigmoid or tanh [31].

The role of the pooling layers is to create more general and abstract features by reducing the size of the feature map. They are usually placed between the convolutional layers. After several convolutional layers and pooling layers, one or more fully connected layers are typically employed to combine all the features from the previous layer into more global features. Batch normalization was proposed to effectively alleviate the problem known as internal covariate shift, i.e., changes in the distribution of layer's inputs during training [52]. It enables using larger learning rates and accelerating this way the training process without the risk of divergence.

B. Residual Learning and SE

Deeper CNN-based models are able to approximate target functions with increased non-linearity, and thus to extract more complex features, leading to improved classification. However, deeper models require abundant training data. In practice, a limited amount of high dimensional training samples constrains the depth and the width of CNN models due to various phenomena, such as the Hughes phenomenon [53], overfitting and gradient vanishing [46]. Consequently, the current CNN models for HSI classification are rather small networks [54].

In general, residual learning can increase the network depth [55], and improve hereby HSI classification [40, 46]. Let $\mathcal{H}(\mathbf{X})$ denote the desired mapping. In the traditional approach, like in Fig. 1 (a), each few stacked layers learn this desired mapping. The idea of residual learning illustrated in Fig. 1 (b) is to learn instead the residual $\mathcal{R}(\mathbf{X}) = \mathcal{H}(\mathbf{X}) - \mathbf{X}$. It is easier to optimize this residual than the original mapping. In the extreme case where $\mathcal{H}$ is the identity operator, it is obviously easier to force the function $\mathcal{R}(\mathbf{X}) = 0$ than $\mathcal{H}(\mathbf{X}) = \mathbf{X}$ [55]. With the residual learning modules, the main training task of a deeper network is simplified into training of multiple residual functions, which facilitates the training process and increases the network depth.

The concept of squeeze-and-excitation networks (SENets) [50] has been recently introduced in the field of computer vision to enhance the feature extraction capability of the network by emphasizing automatically informative features and suppressing the less useful ones. The SE module consists of the squeeze part and the excitation part. The squeeze part squeezes the spatial information from each feature channel into a single number by global average pooling. This way, the collection of N channels $\mathcal{F}_i(\mathbf{X})$ is transformed into a vector with N elements. The excitation part uses two fully connected layers to learn channel-wise correlations $\mathbf{e}_i \in \mathbb{R}^{N \times 1}$:

$$\mathbf{e}_i = \sigma(\mathbf{W}'' \delta(\mathbf{W}' \mathcal{A}(\mathcal{F}_i(\mathbf{X})))) \tag{2}$$

where $\mathcal{A}$ denotes the global average pooling operation. δ and σ are the ReLU and the sigmoid, respectively. $\mathbf{W}' \in \mathbb{R}^{\frac{N}{r} \times N}$ and $\mathbf{W}'' \in \mathbb{R}^{N \times \frac{N}{r}}$ are the weights of the two fully connected layers, respectively. A reduction ratio r is to adjust the capability and computational cost. $\mathcal{F}_i(\mathbf{X})$ is then rescaled by $\mathbf{e}_i$, promoting this way more informative feature channels.

III. Proposed Method

A. Overall Architecture

Here we propose a novel two-stream CNN architecture for HSI analysis. The two streams that operate in parallel as shown in Fig. 2, extract simultaneously local and global spatial-spectral features. Specifically, the local feature extraction stream is a shallow network that takes as input all spectral bands of a HSI and extracts spectral and *local* spatial correlation features from small image patches. In parallel, the global feature extraction stream is a deep network that takes as input much less (several to a dozen) principal components of a HSI and extracts more complex *global* spatial structure features from large image patches. The outputs of the two streams are fused using fully connected layers, and the output labels are predicted by a softmax layer.

The main novelties and differences compared to the earlier related spectral-spatial learning architectures are the following. Firstly, earlier reported spectral feature extraction streams were always based on 1D vectors, and when employing CNN those were 1D-CNN networks (e.g., in [31, 45, 56]). Our feature extraction stream is instead embedded into a 2D-CNN, which operates on small image patches and extracts simultaneously spectral and local spatial features. An important advantage of this approach is that it leads to an elegant mathematical formulation with a unique objective function, as it will be shown in Section III-D.

Secondly, we effectively improve the spectral-spatial feature extraction capability by incorporating the squeeze-and-excitation (SE) concept into the two parallel streams. This is especially important in practice where the actual depth of the

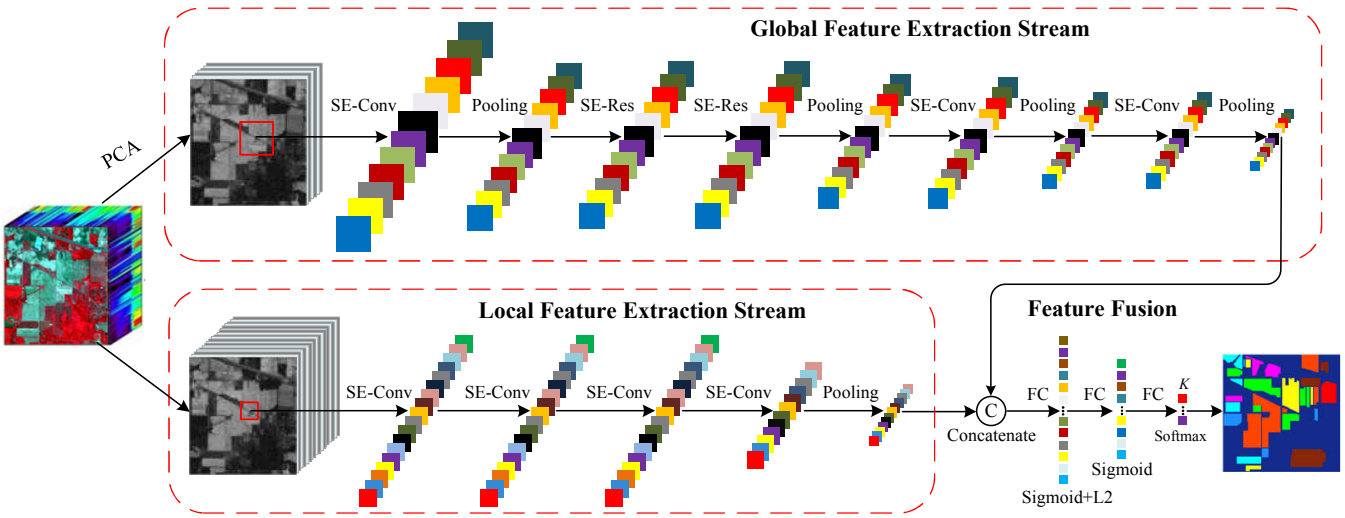


Fig. 2. The overall architecture of the proposed method. SE-Conv denotes a convolutional layer incorporating the SE module, and SE-Res denotes a residual learning module incorporating the SE module. The global feature extraction stream extracts global spatial features from relatively large image patches extending over several most important principal components. The local feature extraction stream extracts spectral and local spatial features from relatively small image patches that extend over all spectral bands.

feature extraction streams is limited by the available amount of the training data. To this end, we shall derive a formal approach for incorporating the SE concept into spectral-spatial classification, as detailed next.

Thirdly, we propose a layer-specific regularization and a smooth normalization fusion scheme, which adaptively controls the fusion weights and better fuses the spectral-spatial features. Finally, while in most of the previous methods, including [42, 44], feature extraction and classifier parts were trained separately and based on different objective functions, in our framework, feature extraction, fusion, and classification processes are trained simultaneously in an end-to-end training manner from scratch. This unified training is one of the important advantages of our two stream 2D-CNN framework.

B. Local Feature Extraction Stream

The local feature extraction stream in our architecture (see the bottom of Fig. 2) employs a shallow 2D-CNN to extract spectral and local spatial correlation features simultaneously. The input is a small image patch extending over all the spectral bands and containing thus local spatial information as well as abundant spectral information. While current spectral feature extraction models based on 1D vectors [31, 42–45, 56] omit spatial information, our local stream not only extracts spectral and local spatial information, but also makes use of spatial information to learn the spectral-band correlations and to boost thereby the feature extraction capability. We accomplish this by incorporating SE similar to [50] but instead of processing RGB images as there, we now employ the SE concept to enhance the feature extraction from a rich spectral content.

The main component of our local feature extraction stream is a convolution layer incorporating the SE module that we denote as SE-Conv. Let $\mathbf{E}_i \in \mathbb{R}^{H\times W\times N}$ denote the channel-wise correlations of a SE module, and $\mathbf{E}_i(:,:,k) = e_i^k \cdot \mathbf{1}_{H\times W}$. Here e_i^k is the k-th element of the correlation vector $\mathbf{e}_i = [e_i^1, e_i^2, ..., e_i^N]^T$ (see equation (2)), and $\mathbf{1}_{H\times W}$ is an H-by-W matrix of all ones. Combining with equation (1), we define SE-Conv as follows:

$$\tilde{\mathcal{F}}_i^l(\mathbf{X}) = \delta(\mathbf{W}_i^l * \mathbf{X}) \cdot \mathbf{E}_i^l \tag{3}$$

where $\mathbf{W}_i^l$ and $\mathbf{E}_i^l$ are the kernels and the channel-wise correlations for the i-th SE-Conv layer of the local stream.

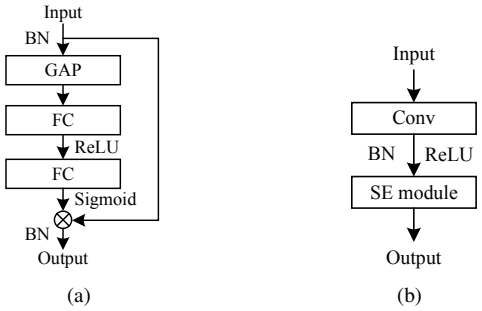


Fig. 3. An illustration of (a) The SE module, (b) The SE-Conv module.

The idea of SE-Conv is to emphasize useful spectral bands and to suppress less useful spectral bands. This way SE-Conv enhances the spectral and local spatial feature extraction capability of the local stream. Fig. 3 shows the architectures of SE and SE-Conv. Specifically, we employ $m > 1$ consecutive SE-Conv modules to extract spectral and local spatial features. A max pooling layer in the end reduces the spatial size and yields more general features at a higher level. Let $\mathbf{I}_l \in \mathbb{R}^{P\times P\times B}$ be the input of the local stream with a relatively small window size of $P \times P$. B denotes the number of HSI spectral bands. The output feature vector of the local stream is

$$\mathbf{y}_l = \mathcal{M}(\tilde{\mathcal{F}}_m^l(\mathbf{I}_l) \cdots \tilde{\mathcal{F}}_2^l(\mathbf{I}_l)\tilde{\mathcal{F}}_1^l(\mathbf{I}_l)) \tag{4}$$

where $\mathcal{M}$ denotes the max pooling operation. We do not use any max pooling layers in between SE-Conv in order to preserve the detail information.

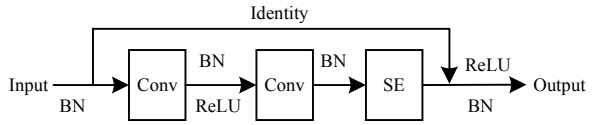


Fig. 4. An illustration of the SE-Res module.

C. Global Feature Extraction Stream

The global feature extraction stream in our model (see the top of Fig. 2) aims to extract global spatial features from relatively large image patches that extend over a relatively small number of principal components (several to a dozen). The current spatial feature extraction models based on 2D-CNN [41, 44, 45, 49, 57] are largely constrained by limited training data. Our proposed spatial feature extraction stream incorporates SE and residual learning concepts to enhance spatial feature extraction capability and network depth. We do not extract the main spectral feature in this stream because it would not only increase computational cost, but also result in spectral features redundancy. An ablation study regarding the number of principal components is given in Section IV-F.

The core components of this stream are SE-Conv with max pooling (denoted as MP-SE-Conv) and SE-based residual learning (denoted as SE-Res). Building on SE-Conv from equation (3), we define MP-SE-Conv as

$$\tilde{\mathcal{F}}_i^{g_1}(\mathbf{X}) = \mathcal{M}(\delta(\mathbf{W}_i^{g_1} * \mathbf{X}) \cdot \mathbf{E}_i^{g_1}) \tag{5}$$

where $\mathbf{W}_i^{g_1}$ and $\mathbf{E}_i^{g_1}$ are the kernels and the channel-wise correlations for the i-th MP-SE-Conv layer. $\mathcal{M}$ is the max pooling operation. The idea of MP-SE-Conv is to yield more robust features by identifying more or less informative spatial-channels and reducing the spatial size using max pooling. We define SE-Res as

$$\tilde{\mathcal{F}}_i^{g_2}(\mathbf{X}) = \delta((\mathbf{W}_{i,2}^{g_2} * \delta(\mathbf{W}_{i,1}^{g_2} * \mathbf{X})) \cdot \mathbf{E}_i^{g_2} + \mathbf{X}) \tag{6}$$

where $\mathbf{W}_{i,1}^{g_2}$ and $\mathbf{W}_{i,2}^{g_2}$ are the two kernels for the i-th SE-Res layer of the global stream, respectively. $\mathbf{E}_i^{g_2}$ is corresponding channel-wise correlations. The idea of SE-Res is to learn more complex global spatial features by enhancing the feature extraction capability and increasing the network depth. Let $\mathbf{I}_g \in \mathbb{R}^{P\times P\times PC}$ be the input of this stream with a relatively large window size of $P\times P$, and *PC* is the number of principal components. The output feature vector of the global stream is

$$\mathbf{y}_g = \tilde{\mathcal{F}}_3^{g_1}(\mathbf{I}_g)\tilde{\mathcal{F}}_2^{g_1}(\mathbf{I}_g)\mathcal{M}[\tilde{\mathcal{F}}_n^{g_2}(\mathbf{I}_g)\cdots\tilde{\mathcal{F}}_1^{g_2}(\mathbf{I}_g)]\tilde{\mathcal{F}}_1^{g_1}(\mathbf{I}_g) \tag{7}$$

where n is the number of SE-Res modules. $\mathcal{M}$ is the max pooling operation. An ablation study regarding the number of SE-Conv and SE-Res is given in Section IV-E. The basic structure of the SE-Res module is illustrated in Fig. 4. The SE module is inserted before and not after the shutcut connection. This is based on the fact that the main training process of the residual learning module is to train the residual function, and thus the SE module can better boost the representative power of the residual learning module when training.

D. Feature Fusion Scheme and Classification

Having extracted spectral, and local and global spatial features, we need to fuse them adaptively. The current deep learning feature fusion methods (e.g., in [34, 44, 46]) employ fully connected layers with ReLU. We propose instead a layer-specific regularization and smooth normalization fusion scheme. We define the fusion scheme as follows:

$$\mathbf{y} = \sigma(\mathbf{W}_2^f\sigma(\mathbf{W}_1^f(\mathbf{y}_l||\mathbf{y}_g) + \lambda\|\mathbf{W}_1^f\|_F^2)) \tag{8}$$

where $||$ denotes the operation of concatenating. $\mathbf{W}_1^f$ and $\mathbf{W}_2^f$ are the kernels of the two fully connected layers, respectively. $\|\ \ \|_F^2$ is the Frobenius norm, and λ is the regularization parameter, which adjusts all the fusion weights and further decides the degree of features fusion. An ablation study regarding λ is given in Section IV-C. σ is sigmoid activation function. We choose sigmoid (that smoothly normalizes the fused features to [0, 1]) instead of ReLU to avoid the blow up phenomenon (feature values in [0, inf]). This choice preserves more detailed features and facilitates the following classification. A L2 kernel regularizer term $\lambda\|\mathbf{W}_1^f\|_F^2$ is added in the fusion layer to enable adaptive adjustment of the fusion weights alone. With this layer-specific regularization, instead of a common regularizer on all network weights like in [31], we avoid overfitting.

Finally, the fused features are fed into the last fully connected layer with K nodes (classes) following a softmax function to generate the predicted probability vector. The cross entropy objective function is computed as

$$\mathcal{L} = -\frac{1}{T}\sum_{j=1}^{T}\sum_{k=1}^{K}\mathbf{t}_k^j\log(\frac{e^{\mathbf{W}_k\mathbf{y}^j+b_k}}{\sum_{i=1}^{K}e^{\mathbf{W}_i\mathbf{y}^j+b_i}}) \tag{9}$$

where T is the total number of training samples. $\mathbf{t}_k^j$ is the k-th value (i.e., 0 or 1) of the one-hot encoding ground truth for the j-th training sample. $\mathbf{W}_k$ and b_k are the weights and bias for the k-th unit in this layer, respectively. $\mathbf{y}^j$ is the input of the j-th training sample. We optimize (9) by using the mini-batch Adadelta [58] optimizer. Observe that the proposed two-stream network has a unique objective function and is trained in an end-to-end training manner from scratch. Thus, the local feature extraction stream and the global feature extraction steam interact during the training process through this unique objective function. This is an important asset of the proposed approach compared to most of the earlier reported ones including [42–44].

IV. Experimental results and analysis

The proposed method is implemented in Keras[1] and TensorFlow[2] deep learning framework with Python language. All the experiments were repeated ten times with different randomly selected training data, and the average results over the ten runs with standard deviations are reported. Three objective performance indexes are used for evaluation: overall accuracy (OA), average accuracy (AA), and Kappa coefficient (κ).

[1]https://keras.io/

[2]https://www.tensorflow.org/

TABLE I
THE NUMBERS OF TRAINING AND TESTING SAMPLES FOR INDIAN PINES, PAVIAU, AND SALINAS IMAGES.

Indian Pines				PaviaU				Salinas			
No.	Classes	Training	Testing	No.	Classes	Training	Testing	No.	Classes	Training	Testing
1	Corn-notill	50	1378	1	Asphalt	50	6581	1	Brocoli green weeds_1	50	1959
2	Corn-mintill	50	780	2	Meadows	50	18599	2	Brocoli green weeds_2	50	3676
3	Corn	50	187	3	Gravel	50	2049	3	Fallow	50	1926
4	Grass-pasture	50	433	4	Trees	50	3014	4	Fallow_rough_plow	50	1344
5	Grass-trees	50	680	5	Painted metal sheets	50	1295	5	Fallow_smooth	50	2628
6	Hay-windrowed	50	428	6	Bare soil	50	4979	6	Stubble	50	3909
7	Soybean-notill	50	922	7	Bitumen	50	1280	7	Celery	50	3529
8	Soybean-mintill	50	2405	8	Self-blocking bricks	50	3632	8	Grapes_untrained	50	11221
9	Soybean-clean	50	543	9	Shadows	50	897	9	Soil_vinyard_develop	50	6153
10	Wheat	50	155					10	Corn_senesced_green_weeds	50	3228
11	Woods	50	1215					11	Lettuce_romaine_4wk	50	1018
12	Buildings-Grass-Trees	50	336					12	Lettuce_romaine_5wk	50	1877
13	Stone-Steel-Towers	50	43					13	Lettuce_romaine_6wk	50	866
14								14	Lettuce_romaine_7wk	50	1020
15								15	Vinyard_untrained	50	7218
16								16	Vinyard_vertical_trellis	50	1757
-	Total	650	9505	-	Total	450	42326	-	Total	800	53329

A. Data Set Description and Parameter Setting

The experiments were conducted on three well-known HSI data sets: Indian Pines, University of Pavia (PaviaU) and Salinas. The Indian Pines data set is captured by the Airborne Visible Infrared Imaging Spectrometer (AVIRIS) sensor over the agricultural Indian Pines site in northwestern Indiana in 1992. It contains 145×145 pixels with 220 spectral bands covering the spectral range from 0.4 to 2.5 μm with a spatial resolution 20 m. It contains 16 ground-truth classes, out of which we select 13 large classes with more than 50 training samples. The PaviaU data set is gathered by the ROSIS-03 sensor over an urban area surrounding the University of Pavia, Pavia, Italy. It consists of 610×340 pixels with 9 classes and 103 spectral bands covering the spectral range from 0.43 to 0.86 μm with a spatial resolution of 1.3 m. The Salinas data set is collected by the AVIRIS sensor over the area of Salinas Valley, CA, USA. It composes of 512×217 pixels with 224 spectral bands covering the spectral range from 0.4 to 2.5 μm with spatial resolution of 3.7 m, 20 water absorption bands were removed. The numbers of training and testing samples for the three HSIs are listed in Table I.

We randomly select 50 labelled samples per class for training. Out of these, 10% are randomly selected and regarded as the validation set. We determine the hyper-parameters based on the classification performance in the validation set. The remaining labelled samples are used as the test set to evaluate the classification performance. The estimated optimal values of the hyper-parameters are as follows. The optimal initial learning rate: 0.3 for PaviaU image and 1 for other two images. $\lambda = 0.03$ for Salinas image and $\lambda = 0.02$ for other two images. An ablation study regarding λ is given in Section IV-C. The optimal number of principal components in the global stream is 3 for Indian Pines image and 10 for the other two images, and an ablation study regarding the number of principal components is shown in Section IV-F. The number of training epochs and batch size are empirically set to 400 and 50. The reduction ratio r of the SE module is empirically set to 1. The main network architecture of the proposed method is shown in Table II. The same network architecture is used in all the reported experiments, with all the test images.

TABLE II
THE PROPOSED NETWORK ARCHITECTURE. BN AND AF REFER TO BATCH NORMALIZATION, ACTIVATION FUNCTION. FOR THE TYPE OF LAYERS, *C*, *M*, *R*, AND *FC* REPRESENT THE CONVOLUTIONAL, MAX POOLING, RESIDUAL LEARNING, AND FULLY CONNECTED LAYERS, RESPECTIVELY. $3 \times 3 \times 128$ MEANS THAT A CONVOLUTIONAL LAYER WITH 128 KERNELS WITH 3×3 KERNEL SIZE.

Nets	No.	Type	Convolution	BN	SE	Padding	AF
	1		Input shape is $7 \times 7 \times B$				
	2	*C*	$1 \times 1 \times 192$	Yes	Yes	Yes	ReLU
Local	3-4	*C*	$3 \times 3 \times 192$	Yes	Yes	Yes	ReLU
	5	*C*	$3 \times 3 \times 128$	Yes	Yes	Yes	ReLU
	6	*M*	-	No	No	No	No
	1		Input shape is $27 \times 27 \times PC$				
	2	*C*	$3 \times 3 \times 128$	Yes	Yes	Yes	ReLU
	3	*M*	-	No	No	No	No
	4-5	*R*	$3 \times 3 \times 128$	Yes	Yes	Yes	ReLU
Global	6	*M*	-	No	No	No	No
	7	*C*	$3 \times 3 \times 128$	Yes	Yes	Yes	ReLU
	8	*M*	-	No	No	No	No
	9	*C*	$3 \times 3 \times 128$	Yes	Yes	Yes	ReLU
	10	*M*	-	No	No	No	No
	1	FC	200	No	No	No	sigmoid
Fusion	2	FC	100	No	No	No	sigmoid
	3	FC	class	No	No	No	softmax

B. Comparisons with the State-of-the-art Method

We compare the performance of the proposed method with several state-of-the-art CNN-based methods for HSI classification. The reference methods are divided into four groups: 1) spectral and local spatial feature extraction (SLSFE) methods: CNN combined with MRF (CNN-MRF) [48] and CNN with pixel-pair features (CNN-PPF) [47]; 2) global spatial feature extraction (GSFE) methods: deep feature fusion (DFFN) [46] and deformable CNN for HSIs classification (DHCNet) [57]; 3) feature fusion methods: diverse region-based CNN (DR-CNN) [34] and DFFN [46]; 4) methods optimized for small-scale training data: multi-grained network (MugNet) [59] and 3D-CNN combined with residual learning (SSRN) [40]. The

TABLE III
COMPARISON OF THE CLASSIFICATION ACCURACIES AMONG THE PROPOSED METHOD AND THE BASELINES USING INDIAN PINES IMAGE.

Classes	CNN-MRF	CNN-PPF	DR-CNN	MugNet	SSRN	DFFN	DHCNet	**Local**	**Global**	**Proposed**
1	76.54±4.42	85.61±3.28	87.82±3.18	88.37±2.65	**93.78±3.81**	89.78±3.35	92.10±2.83	91.02±3.13	92.05±3.33	92.67±3.28
2	87.41±3.85	84.58±4.85	95.23±3.31	94.03±2.30	90.71±7.21	96.49±1.72	96.71±1.66	95.22±3.50	97.46±2.83	**98.27±1.34**
3	98.93±0.99	60.55±7.68	99.04±1.10	99.20±1.08	89.86±5.87	**100±0**	**100±0**	99.09±1.05	99.84±0.35	99.89±0.33
4	94.43±2.28	94.46±3.82	97.07±1.49	97.27±1.97	97.98±1.21	96.79±1.55	97.51±1.69	98.06±1.25	**98.80±1.44**	98.59±1.42
5	94.74±1.85	98.95±0.90	99.53±0.32	**99.85±0.15**	98.08±1.11	98.32±1.65	98.74±1.72	99.40±0.55	99.49±0.59	99.75±0.37
6	98.53±1.33	99.95±0.10	**99.98±0.08**	99.84±0.34	99.21±2.07	99.95±0.10	99.93±0.15	99.95±0.10	**99.98±0.08**	**99.98±0.08**
7	85.24±3.33	82.23±5.11	90.97±3.03	95.00±2.32	80.36±11.18	93.92±3.22	**95.12±1.72**	94.76±2.23	94.20±3.15	94.70±2.84
8	72.42±3.83	92.56±2.68	78.83±3.15	92.50±2.66	94.19±2.98	90.26±3.14	92.01±1.67	85.50±3.29	94.00±2.81	**95.37±1.40**
9	81.34±3.70	80.21±6.29	94.30±3.25	96.45±1.37	80.45±11.48	95.56±1.80	**96.85±1.70**	94.73±3.27	95.87±1.58	96.57±1.54
10	99.94±0.20	99.17±0.94	**100±0**	99.68±0.33	96.29±3.70	99.94±0.20	99.94±0.20	**100±0**	**100±0**	**100±0**
11	92.41±2.06	98.68±1.30	96.21±1.62	99.56±0.23	99.14±0.67	98.21±1.33	96.48±2.26	95.37±2.26	**99.60±0.64**	99.39±0.78
12	95.92±1.90	67.79±4.49	99.08±2.11	98.21±1.34	88.61±5.26	99.73±0.37	99.43±1.00	97.74±2.69	99.94±0.12	**99.97±0.09**
13	99.77±0.70	85.24±9.54	99.54±1.40	98.60±1.14	73.52±8.01	99.07±1.55	98.60±1.55	99.77±0.70	**100±0**	**100±0**
AA(%)	90.59±0.62	86.92±1.11	95.20±0.47	96.81±0.56	90.94±1.41	96.77±0.61	97.19±0.39	96.20±0.45	97.79±0.35	**98.09±0.28**
OA(%)	84.26±0.94	88.17±0.94	90.58±1.01	94.95±0.97	91.59±1.83	94.41±1.16'	95.22±0.80	93.00±0.88	96.17±0.67	**96.75±0.44**
$\kappa \times 100$	82.09±1.03	86.49±1.08	89.25±1.13	94.20±1.10	90.39±2.05	93.59±1.32	94.52±0.91	91.99±0.99	95.60±0.76	**96.26±0.51**

TABLE IV
COMPARISON OF THE CLASSIFICATION ACCURACIES AMONG THE PROPOSED METHOD AND THE BASELINES USING THE PAVIAU IMAGE

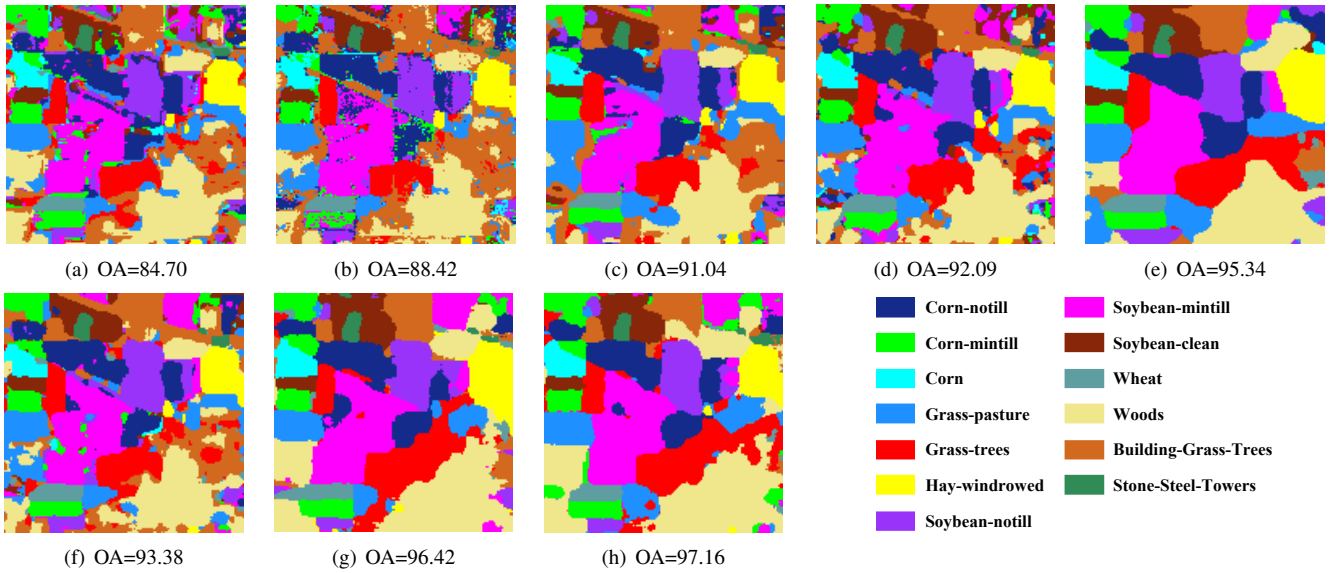
Classes	CNN-MRF	CNN-PPF	DR-CNN	MugNet	SSRN	DFFN	DHCNet	**Local**	**Global**	**Proposed**
1	84.54±1.85	97.33±1.42	93.73±3.83	84.07±2.09	**99.56±0.58**	96.63±1.99	97.46±1.73	95.96±1.41	95.03±2.08	97.07±2.09
2	89.77±1.41	97.58±1.14	97.76±1.76	98.42±0.57	**99.58±0.29**	95.50±2.66	98.52±1.39	96.59±1.36	98.28±0.75	99.16±0.50
3	83.04±1.19	84.64±3.21	94.72±0.96	97.33±1.01	89.99±6.71	97.73±2.84	98.74±0.88	95.84±1.37	98.77±1.57	**99.38±0.90**
4	97.13±0.75	78.93±7.65	98.42±0.47	97.47±0.61	97.36±3.97	92.57±2.51	95.20±1.31	96.70±1.16	97.94±1.31	**98.37±0.59**
5	99.70±0.37	99.73±0.33	99.92±0.11	99.90±0.10	**100±0**	98.88±1.24	99.45±0.94	**100±0**	**100±0**	**100±0**
6	87.66±1.96	76.29±6.99	97.63±1.25	98.00±0.49	92.41±4.33	98.29±1.65	99.39±0.67	96.80±2.71	**99.80±0.29**	99.72±0.46
7	92.73±1.29	89.94±6.55	98.73±0.75	99.11±0.43	94.16±5.47	99.70±0.43	99.45±0.53	99.73±0.22	99.70±0.67	**99.96±0.10**
8	79.51±3.36	86.67±6.44	97.80±1.25	96.01±1.31	92.49±2.77	**98.19±1.14**	98.14±1.27	95.98±1.78	96.85±2.54	97.96±1.56
9	98.65±0.94	99.16±1.33	99.89±0.15	99.29±0.49	99.88±0.24	94.96±1.12	96.22±1.89	**99.98±0.05**	99.79±0.36	99.88±0.17
AA(%)	90.30±0.36	90.03±1.24	97.62±0.51	96.62±0.33	96.16±1.57	96.94±0.66	98.06±0.44	97.51±0.43	98.46±0.39	**99.05±0.33**
OA(%)	88.61±0.50	90.83±1.74	97.16±0.97	95.89±0.39	97.14±0.96	96.35±1.36	98.21±0.63	96.71±0.61	97.95±0.53	**98.82±0.40**
$\kappa \times 100$	85.11±0.60	88.07±2.20	96.24±1.27	94.53±0.51	96.22±1.25	95.19±1.77	97.62±0.83	95.64±0.80	97.29±0.69	**98.43±0.53**

TABLE V
COMPARISON OF THE CLASSIFICATION ACCURACIES AMONG THE PROPOSED METHOD AND THE BASELINES USING THE SALINAS IMAGE

Classes	CNN-MRF	CNN-PPF	DR-CNN	MugNet	SSRN	DFFN	DHCNet	**Local**	**Global**	**Proposed**
1	99.81±0.29	99.96±0.08	99.96±0.04	64.91±3.30	**100±0**	99.81±0.26	99.96±0.06	99.21±1.08	**100±0**	**100±0**
2	97.51±1.24	99.60±0.31	99.59±0.27	99.90±0.10	99.76±0.65	99.83±0.26	99.97±0.07	99.91±0.17	**99.98±0.05**	99.95±0.10
3	98.94±1.17	96.86±2.97	99.63±0.13	99.63±0.18	99.65±0.34	99.95±0.16	99.93±0.22	99.87±0.21	99.95±0.09	**99.99±0.02**
4	99.30±0.69	97.04±1.61	99.95±0.06	99.87±0.28	98.85±1.03	98.79±1.41	99.69±0.22	99.90±0.11	99.81±0.49	**99.99±0.03**
5	97.87±0.60	99.23±0.55	98.75±0.89	99.34±0.37	**99.90±0.18**	98.93±1.14	99.72±0.21	99.50±0.51	99.23±0.30	99.50±0.44
6	99.83±0.30	99.80±0.31	**99.99±0.02**	98.98±0.18	99.99±0.03	99.75±0.31	99.91±0.17	**99.99±0.01**	99.77±0.65	99.92±0.27
7	98.67±0.96	99.64±0.77	99.92±0.07	99.31±0.63	**99.99±0.02**	99.86±0.14	99.72±0.19	99.98±0.04	99.81±0.23	**99.99±0.02**
8	76.69±2.49	84.83±2.65	77.25±10.99	**99.40±0.34**	89.73±5.65	97.36±1.56	95.09±2.52	67.23±19.03	96.49±2.81	96.61±2.70
9	98.77±0.42	99.29±0.33	**99.96±0.04**	89.23±2.12	99.74±0.15	99.78±0.32	99.70±0.37	99.84±0.18	99.89±0.26	99.94±0.19
10	94.88±1.72	89.00±4.25	96.39±0.59	99.12±0.92	97.84±1.37	99.21±0.55	99.77±0.35	97.09±0.92	99.67±0.28	**99.75±0.19**
11	99.08±0.78	89.99±6.91	**99.97±0.05**	94.72±1.56	98.13±2.59	99.22±0.70	99.62±0.43	99.74±0.39	99.47±0.61	99.78±0.34
12	99.99±0.02	98.79±0.81	**100±0**	99.29±0.54	99.65±0.46	99.57±0.62	99.87±0.17	99.99±0.02	99.54±0.98	99.58±1.15
13	99.75±0.35	98.35±1.59	**100±0**	99.85±0.22	99.95±0.08	99.38±0.84	99.95±0.14	99.84±0.32	99.76±0.32	99.93±0.18
14	98.25±0.82	95.33±2.41	**100±0**	98.43±0.69	98.42±1.42	99.89±0.21	99.70±0.53	99.67±0.40	99.94±0.10	99.91±0.14
15	82.43±3.55	75.02±7.19	90.38±5.19	97.80±1.14	85.41±4.26	97.50±1.75	98.73±0.97	87.96±5.22	99.01±1.23	**99.16±1.05**
16	97.44±1.97	98.96±0.79	99.35±0.19	94.02±2.22	99.85±0.28	99.92±0.26	99.92±0.12	99.39±0.42	99.80±0.29	**99.99±0.04**
AA(%)	96.18±0.41	95.11±0.94	97.57±0.51	95.86±0.16	97.93±0.23	99.30±0.19	99.45±0.20	96.82±1.04	99.51±0.19	**99.63±0.20**
OA(%)	91.66±0.68	91.80±1.48	93.55±1.87	79.74±1.60	95.36±0.94	98.86±0.24	98.67±0.56	91.18±3.63	98.98±0.53	**99.09±0.56**
$\kappa \times 100$	90.73±0.75	90.87±1.64	92.84±2.05	74.35±1.75	94.82±1.06	98.73±0.27	98.52±0.63	90.23±3.97	98.87±0.59	**98.99±0.62**

DFFN method can be regarded as a global spatial feature extraction method and also as a feature fusion method because it uses a relatively large image patch size and fuses the features extracted from different hierarchical layers. The parameters of the reference methods are set to the default values indicated in their original works. For a fair comparison, we use in all experiments the same number of PCA components and the same patch size for DHCNet [57] and for our global stream. To demonstrate the effectiveness of the local stream and the global stream, we also test the networks that only contain the local stream and the global steam.

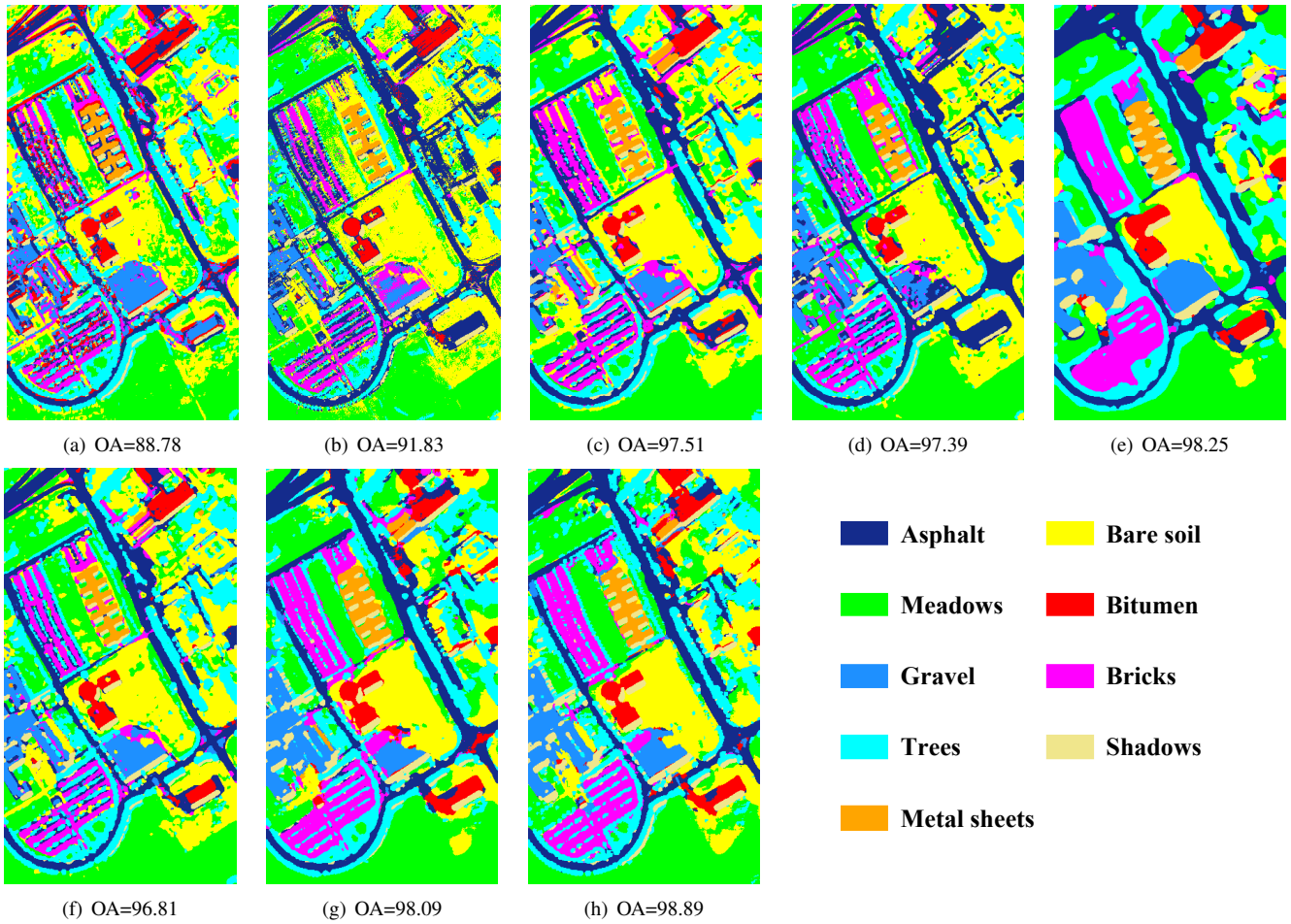
Tables III–V report the class-specific accuracy, AA, OA,



(a) OA=84.70 (b) OA=88.42 (c) OA=91.04 (d) OA=92.09 (e) OA=95.34

(f) OA=93.38 (g) OA=96.42 (h) OA=97.16

Fig. 5. Full classification maps on the Indian Pines image obtained by (a) CNN-MRF, (b) CNN-PPF, (c) DR-CNN, (d) SSRN, (e) DHCNet, (f) the local stream, (g) the global stream, (h) the proposed method.



(a) OA=88.78 (b) OA=91.83 (c) OA=97.51 (d) OA=97.39 (e) OA=98.25

(f) OA=96.81 (g) OA=98.09 (h) OA=98.89

Fig. 6. Full classification maps on the PaviaU image obtained by (a) CNN-MRF, (b) CNN-PPF, (c) DR-CNN, (d) SSRN, (e) DHCNet, (f) the local stream, (g) the global stream, (h) the proposed method.

and κ of all the methods for Indian Pines, PaviaU and Salinas images. As can be observed, the proposed method yields the best OA, AA and κ with a significant improvement over the reference methods for the three HSIs. For instance, in Tables III, the proposed method achieves OA 96.75%, with gains of 12.49%, 8.58%, 6.17%, 1.80%, 5.16%, 2.34%, and

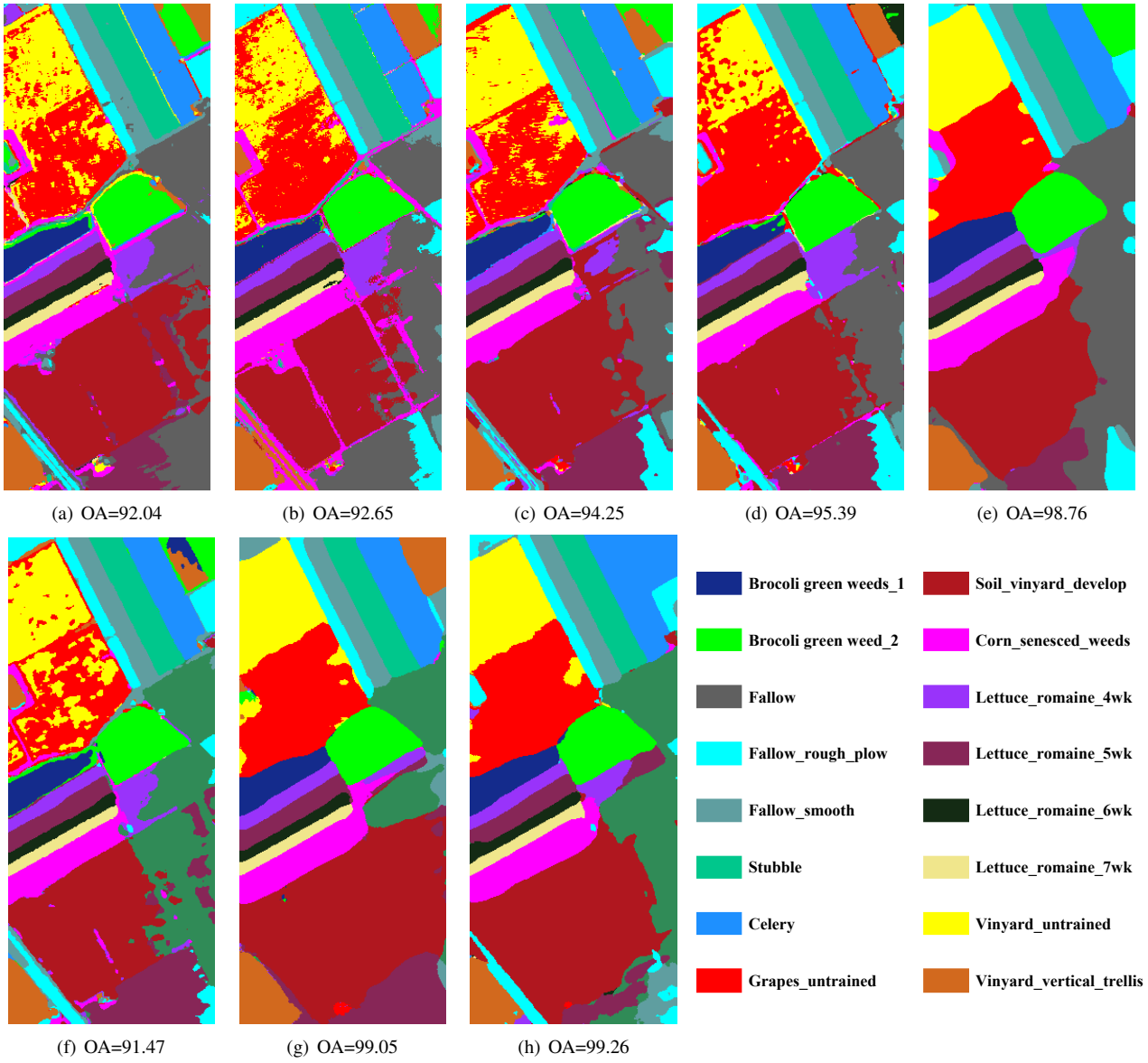


Fig. 7. Full classification maps on the Salinas image obtained by (a) CNN-MRF, (b) CNN-PPF, (c) DR-CNN, (d) SSRN, (e) DHCNet, (f) the local stream, (g) the global stream, (h) the proposed method.

1.53% over CNN-MRF, CNN-PPF, DR-CNN, MugNet, SSRN, DFFN, and DHCNet methods, respectively. The other two HSIs have similar classification results. Obviously, the GSFE methods constantly perform better than the SLSFE methods due to exploiting more spatial context information. For the comparison of the SLSFE methods, the local stream of our proposed method performs comparable in Salinas image and even performs better in the other two images than the CNN-MRF and the CNN-PPF methods in terms of classification performance. In addition, the global stream of our proposed method yields comparable OA in the PaviaU image and yields better in Indian Pines and Salinas images over the DFFN and the DHCNet methods, which demonstrates the strong feature extraction capability of the two feature extraction streams.

Compared with the feature fusion methods DFFN and DR-CNN, our proposed method yields again better classification performance. Also, our proposed method yields better classification performance compared to MugNet and SSRN designed for small-scale training data. It is also evident that the proposed method yields better accuracy than any of its two streams alone. This is because the local stream extracts the spectral and the local spatial features that are complementary to the global spatial features extracted in the second stream. Thus the proposed two-stream method has more robust feature representation power and better generalization ability. In terms of the class-specific accuracy, the proposed method performs best or yields comparable results to the best ones in most of the classes for all the three images. Only in several classes this is not the case. For instance, in the Salinas image, some ‘Grapes_untrained’ samples were misclassified as ‘Vinyard_untrained’ due to their huge spectral similarity and the large within-class variation in their spectral reflectance.

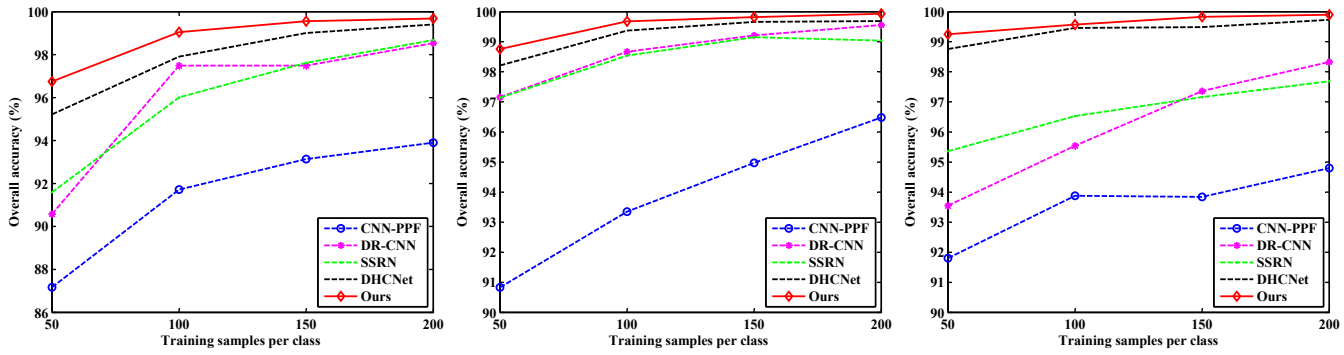


Fig. 8. OA of different methods with different numbers of training samples per class (a) Indian Pines image, (b) PaviaU image, (c) Salinas image.

TABLE VI
OA OBTAINED BY SEVERAL MULTI-STREAM METHOD AND A GRAPH CONVOLUTIONAL NETWORKS METHOD. THE RESULTS OF THE PROPOSED METHOD ARE IN BRACKETS.

Image	Method	Training set	OA
Indian Pines	SdAE-CNN	60% per class	98.65% (**99.95%**)
	Multi-CNN	5% per class	79.11% (**96.55%**)
	MMFN	3% per class	91.81% (**93.58%**)
	MDGCN	30 samples per class	93.47% (**94.14%**)
PaviaU	SdAE-CNN	20% per class	97.50% (**99.97%**)
	Multi-CNN	1% per class	90.75% (**97.88%**)
	MMFN	3% per class	99.40% (**99.78%**)
	CSFF	200 samples per class	98.53% (**99.93%**)
	MDGCN	30 samples per class	95.68% (**96.09%**)
Salinas	Multi-CNN	1% per class	88.62% (**99.42%**)
	MMFN	3% per class	98.37% (**99.91%**)
	CSFF	200 samples per class	98.90% (**99.90%**)
PaviaC	SSRN	1% per class	99.51% (**99.60%**)
	CSFF	200 samples per class	99.75% (**99.85%**)
	DHCNet	50 samples per class	98.29% (**99.40%**)
Grss_dfc _2013	SSRN	3% per class	93.90% (**94.90%**)
	MugNet	20 samples per class	90.82% (**92.13%**)
	DHCNet	50 samples per class	95.66% (**96.75%**)

Apart from quantitative analysis, Fig. 5–7 show the full classification maps. Visually, they are consistent with the results reported in Tables III–V. Obviously, the SLSFE methods (e.g., CNN-MRF, CNN-PPF and the local stream) exhibit noisier estimations than the GSFE methods (DR-CNN, DHCNet, and the global stream). Furthermore, the proposed method presents more similar results to the reference map exhibiting smoother appearance than other reference methods because of more robust spectral and spatial features. In addition, the feature fusion strategy effectively combines the advantages of the both streams, e.g., the regions of Meadows and Bare Soil in Fig. 6.

To comprehensively validate the proposed architecture, we also compare the proposed method with several state-of-the-art multi-stream fusion methods: SdAE-CNN [44], Multi-CNN [60], MMFN [61], CSFF [62], a hierarchical architecture MugNet [59] and a very recent graph convolutional method MDGCN [63]. To validate the robustness on more data sets, we include two additional data sets: Pavia Center[3] (PaviaC) and Grss_dfc_2013[4]. The optimal values of the hyper-parameters of the proposed method for these two data sets are the same as for the PaviaU data set. The results of the reference methods are taken from the original works. To validate the performance with a different sample partitioning method, we also compare with SSRN [40] and DHCNet [57]. The results are given in Table VI, where for the proposed method we show in brackets the results obtained with the same sample partitioning as in the corresponding reference method. As can be observed in Table VI, the proposed method yields the best OA for all the HSIs compared to all the reference methods. It is worth mentioning that the proposed method exhibits robust classification performance for balanced (e.g., 30 samples per class) and unbalanced (e.g., 1% per class) training samples.

To verify the generalization ability of the proposed method on different numbers of training samples, 50, 100, 150, and 200 samples per class are randomly chosen as training data for three HSIs. For Indian Pines image, following the references including [32, 34, 47], we choose 8 large classes when the number of training samples are larger than 50. Fig. 8 shows the OA for the proposed method and four kinds of reference methods: (i) spectral and local spatial feature extraction: CNN-PPF [47], (ii) feature fusion based: DR-CNN [34], (iii) optimized for small-scale training data: SSRN [40], and (iv) global spatial feature extraction: DHCNet [57] . Clearly, all the methods yield better classification performance as the number of training samples increase. The proposed method consistently provides superior OA compared to the reference methods for three HSIs. Especially when the number of the labelled training data is limited, the proposed method has obvious advantage in terms of classification performance over the reference methods.

C. Analysis on Feature Fusion Scheme

To validate the proposed feature fusion scheme, we compare it with ReLU, ReLU with L2, and sigmoid under the same settings as in Tables III–V. The results in Table VII show that both sigmoid and ReLU with L2 regularizer (where λ equals 0.02, 0.02, and 0.03 for Indian Pines, PaviaU, and

[3]Available online: http://www.ehu.eus/ccwintco/index.php/Hyperspectral_Remote_Sensing_Scenes#Pavia_Centre_scene

[4]Available online: http://www.grss-ieee.org/community/technical-committees/data-fusion/.

TABLE VII
THE EFFECT OF FEATURE FUSION SCHEMES ON OA FOR THREE HSIS.

Schemes	Indian Pines	PaviaU	Salinas
ReLU	95.64±1.04	96.24±0.83	98.76±0.69
ReLU+L2	96.29±0.75	97.34±0.78	98.94±0.72
sigmoid	96.65±0.69	98.41±0.44	98.69±0.84
sigmoid+L2	96.75±0.44	98.82±0.40	99.09±0.56

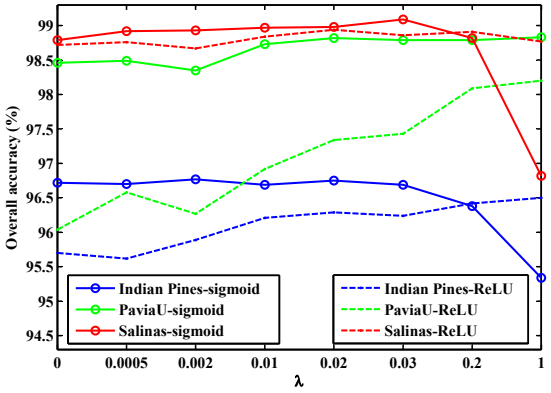


Fig. 9. The effect of λ on OA for sigmoid and ReLU cases.

TABLE VIII
THE EFFECT OF r ON OA FOR THREE HSIS.

SE	Indian Pines	PaviaU	Salinas
Non-SE	96.34±0.58	98.07±0.44	98.81±0.86
SE(r=1)	96.75±0.44	98.82±0.40	99.09±0.56
SE(r=4)	96.62±0.63	98.66±0.53	99.01±0.59
SE(r=8)	96.69±0.64	98.42±0.44	98.98±0.70
SE(r=16)	96.53±0.51	98.36±0.57	98.97±0.63
SE(r=128)	96.72±0.52	98.16±0.64	98.89±0.79

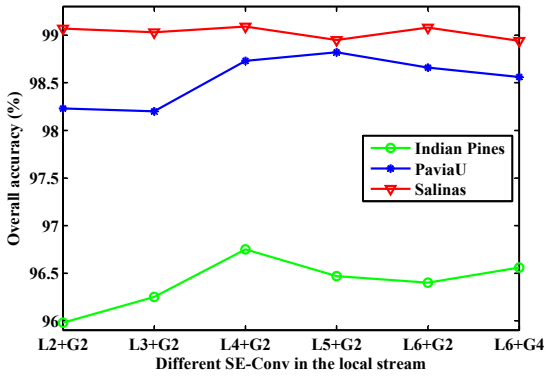


Fig. 10. The effect of the number of SE-Conv modules on OA for three HSIs.

Salinas, respectively) yield better classification performance than without L2 regularizer. Hence, L2 regularizer effectively controls the degree of feature fusion. The scheme with the sigmoid and L2 regularizer performs the best and shows indeed an improvement in OA over ReLU (that was used in earlier reported feature fusion schemes), with gains of 1.11%, 2.58%, and 0.33% for Indian Pines, PaviaU, and Salinas images, respectively. The results indicate that the combined sigmoid and L2 regularizer scheme in the fully connected layers effectively fuses the spectral, the local spatial, and the global spatial features extracting from the two streams, and forms more discriminative and robust features.

Fig. 9 further shows the OA for sigmoid and ReLU versus different $\lambda \in \{0, 0.0005, 0.002, 0.01, 0, 02, 0.03, 0.2, 1\}$ for three HSIs. Obviously, the sigmoid with the optimal λ performs better than ReLU with the optimal λ, especially for Indian Pines and PaviaU images. The proposed feature fusion scheme yields stable OA values within a certain range of λ. In general, the OA initially increases and then declines as λ increases. The main reason is that a smaller λ underfits the degree of feature fusion and results in some redundant features, while a larger λ overfits the fusion degree and results in a loss of some useful features, degrading the classification performance.

D. Analysis of the SE module

To verify the effectiveness of the SE module, we compare the performance of the proposed method without the SE module to the version with the SE module for different $r \in \{1, 4, 8, 16, 128\}$. The results are reported in Table VIII. Clearly, the SE module with different r consistently performs better than without the SE module in terms of OA for three HSIs. The reason is that the SE module enhances the network feature representation and further improves the classification performance of HSI. The results in Table VIII reveal that the OA does not increase monotonically as r decreases for Indian Pines image. A possible reason is that the SE module overfits the feature channel-wise correlations. By contrast, for PaviaU and Salinas images, a large r slightly degrades the OA, which means it underfits the feature channel-wise correlations.

E. Analysis of the Network Depth

We combined a shallow network in the local stream and a deep network in the global stream to extract more robust features (spectral, local spatial and global spatial features) of HSIs. The network depths for the two streams are thus the two key hyperparameters. We fix the other parameters under the same settings as in Tables III–V. As shown in Fig. 10 (L4+G2 denote 4 SE-Conv modules in the local stream and 2 SE-Res modules in the global stream, respectively), the results on the PaviaU and the Indian Pines images first improve significantly when the number of SE-Conv modules increases (because they have many small and local regions) and then degrade slightly due to excessive depth and overfitting. By contrast, the result on the Salinas image tends to relatively stable with increasing the network depth because it has many large smooth regions.

Fig. 11 shows the effect of the number of SE-Res modules n in the global stream and the proposed network ($n = 2$) without the short connection (denoted as noRL). Compared with noRL, the proposed provides better classification performance, demonstrating that SE-Res with residual learning mechanism mitigates overfitting problem when the depth of the global stream increases. Furthermore, the OA indeed increases at first as the number of SE-Res modules increases because deeper network extracts more abstract features, and then the OA decreases due to overfitting caused by excessive network depth and limited training data. Observe that the OA in the

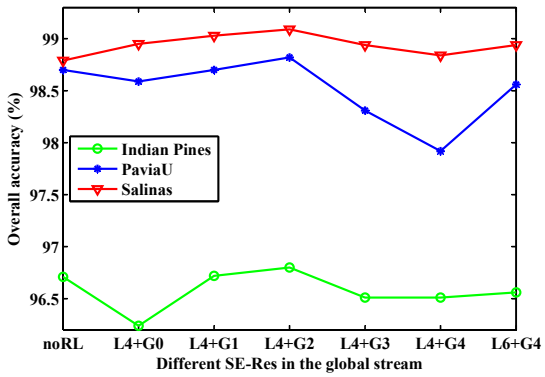


Fig. 11. The effect of the number of SE-Res modules on OA for three HSIs.

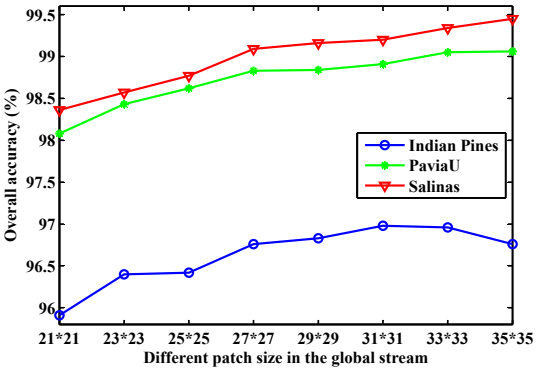


Fig. 13. The effect of the global input patch size on OA for three HSIs.

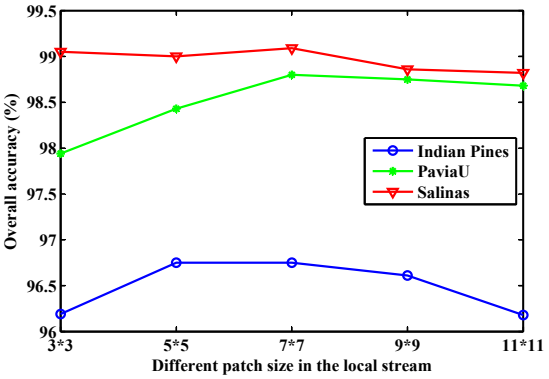


Fig. 12. The effect of the local input patch size on OA for three HSIs.

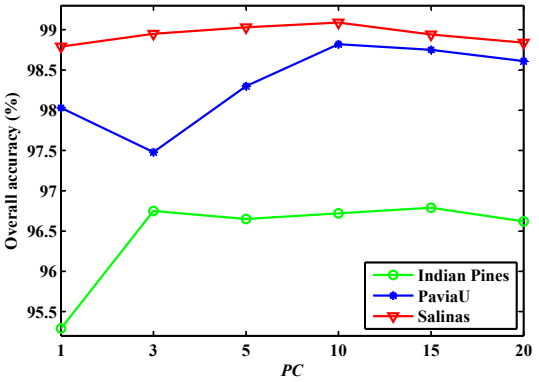


Fig. 14. The effect of the number of principal components for three HSIs.

PaviaU image declines dramatically compared to the other two images when the network depth increases. The main reason is that the PaviaU image has more detailed regions.

Based on the above analysis, the local stream yields better classification performance on images with many small regions (like PaviaU) as the depth in the local stream increases. The global stream yields better classification performance on images with many large regions (e.g., the Salinas image) as the depth in the global stream increases due to extracting global spatial features. In addition, we also test a deeper two-stream CNN (dubbed L6+G4) as shown in Fig. 10 and Fig. 11. The results show that the proposed method performs better than this alternative. The proposed network depth settings (L4+G2) as in Table II demonstrate more robustness and better generalizability on the tested data sets.

F. Analysis of the Patch Size and the Principal Components

In this section, we discuss the effect of different image patch sizes P in the two streams on OA. We keep the same settings as in Tables III–V. We adjust the sizes of the max pooling operations in the two streams for different image patch sizes. Fig. 12 illustrates the OA versus different P in the local stream varying from 3×3 to 11×11 with an interval of 2. The results demonstrate that the OA generally improves at first due to extracting more local spatial features as P increases, and then declines because a large P (e.g., 11×11) cannot effectively extract local spatial features. Fig. 13 shows the OA versus different P in the global stream varying from 21×21 to 35×35 with an interval of 2. Apparently, a large P gets better or comparative classification accuracy, because it contains more global spatial information, but an overlarge P increases the computational cost and memory requirements dramatically. The P equals 27×27 in the global stream as a trade-off between the classification performance and the running time for three HSIs.

Further on, we analyze the effect of the number of principal components in our method under different settings. Fig. 14 (a) shows the OA versus the number of principal components $PC \in \{1, 3, 5, 10, 15, 20\}$ for different HSIs. The OA generally increases and then declines slightly as the number of principal components increases. This is as expected since the first several components contain most of the spatial information. Adding a larger number of principal components results in a redundant spectral information and requires more learning parameters, increasing thereby the computational cost and degrading the classification performance. A sudden drop of OA on PaviaU for $PC = 3$ may be attributed to the fact that some classes (e.g., Asphalt and Bitumen) in this image have huge spectral-spatial similarity when $PC = 3$, which may result in misclassification.

G. Analysis of the Computational Efficiency

A comparative analysis of the processing time and memory requirements for different representative methods is summarized in Table IX. The training and testing time are reported together with the memory required (the maximum value during

TABLE IX
COMPARISON OF COMPUTATIONAL COMPLEXITY ON DIFFERENT METHODS FOR THREE HSIS.

		Indian Pines	PaviaU	Salinas
CNN-PPF	Training (min)	11.4	13.1	30.5
	Testing (s)	6.3	26.9	43.9
	Memory(GB)	14.3	43.7	37.9
DR-CNN	Training (min)	35.6	10.5	41.9
	Testing (s)	110.4	82.2	223.2
	Memory(GB)	9.4	37.6	37.5
SSRN	Training (min)	5.7	3.6	6.7
	Testing (s)	4.5	18.0	25.2
	Memory(GB)	4.6	12.1	14.9
DHCNet	Training (min)	2.6	2.7	2.8
	Testing (s)	5.1	22.6	28.1
	Memory(GB)	2.8	26.4	22.4
Proposed	Training (min)	2.9	2.6	3.9
	Testing (s)	5.6	22.2	28.9
	Memory(GB)	5.1	25.6	23.9

the whole process) for three HSIs. Four kinds of reference methods are used: (i) spectral and local spatial feature extraction: CNN-PPF [47], (ii) feature fusion based: DR-CNN [34], (iii) optimized for small-scale training data: SSRN [40], and (iv) global spatial feature extraction: DHCNet [57]. All experiments are conducted on an Intel Core i7-7820X CUP with an Nvidia TITAN Xp GPU. Compared to PPF-CNN and DR-CNN, the proposed method yields considerably faster training, faster testing (especially compared to DR-CNN) and requires less memory. Compared to SSRN, our method has similar training and testing time but requires more memory, and compared to DHCNet, the time and space complexity are similar, while we obtain better results in terms of accuracy. It can be concluded that the proposed method is not only very competitive in terms of the accuracy, but also computationally efficient relative to the current state-of-the-art.

As an implementation detail, it should be noted that we employ batch normalization layers [52], residual learning mechanism [55] and a clever strategy for terminating the training process[5] and reducing the learning rate[6], which enables us to use a larger initial learning rate. Fig. 15 illustrates the evolution of the training and validation losses and the corresponding learning rate for a particular test image (Indian Pines). Similar trends hold for other test images. It can be seen that the training and the validation losses converge quickly (in around 100 epochs) and terminate in advance (in less than 400 epochs). The large initial learning rate (i.e., 1) decreases quickly, converging (in around 100 epochs) to a stable value.

V. CONCLUSION

In this paper, we proposed a novel two stream spectral and spatial feature extraction and fusion architecture based on 2D-CNN for HSI classification. The proposed method simultaneously extracts spectral, local and global spatial features via a shallow and a deep 2D-CNN networks. Inspired by squeeze-and-excitation networks, we developed a formal approach to enhance the spectral-spatial feature extraction capability

[5]https://keras.io/callbacks/#earlystopping
[6]https://keras.io/callbacks/#reducelronplateau

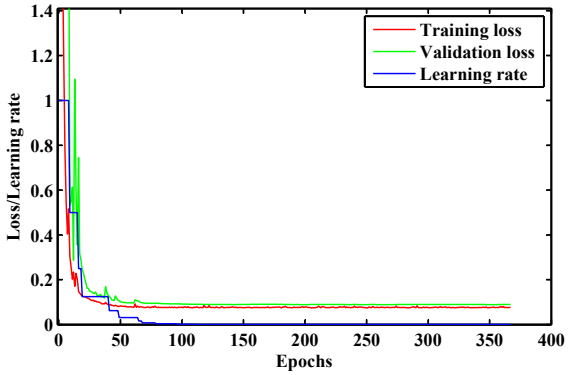


Fig. 15. The training process curves on Indian Pines image.

based on inter-band correlations. This approach improves significantly the classification performance, especially when the amount of the available training data is limited. In addition, we proposed a layer-specific regularization and smooth normalization fusion scheme to adaptively fuse the spectral-spatial features of the two streams. Experimental results on several HSIs demonstrated the state-of-the-art classification performance.

VI. ACKNOWLEDGMENT

The authors would like to thank Dr. W. Song and Dr. B. Pan for providing the software for the DFFN method [46] and the MugNet method [59], respectively. They would also like to thank the Associate Editor and the anonymous Reviewers for their insightful comments and helpful suggestions which have greatly improved this paper.

The authors would like to thank the Hyperspectral Image Analysis group and the NSF Funded Center for Airborne Laser Mapping (NCALM) at the University of Houston for providing the grss_dfc_2013 data set used in this paper, and the data fusion technical committee for their preparation and pre-process for this data set.

REFERENCES

[1] H. Grahn and P. Geladi, *Techniques and applications of hyperspectral image analysis*. John Wiley & Sons, 2007.
[2] J. M. Bioucas-Dias, A. Plaza, G. Camps-Valls, P. Scheunders, N. Nasrabadi, and J. Chanussot, "Hyperspectral remote sensing data analysis and future challenges," *IEEE Geosci. Remote Sens. Mag.*, vol. 1, no. 2, pp. 6–36, 2013.
[3] D. Landgrebe, "Hyperspectral image data analysis," *IEEE Signal Process. Mag.*, vol. 19, no. 1, pp. 17–28, 2002.
[4] B. Du, Y. Zhang, L. Zhang, and D. Tao, "Beyond the sparsity-based target detector: A hybrid sparsity and statistics-based detector for hyperspectral images," *IEEE Trans. Image Process.*, vol. 25, no. 11, pp. 5345–5357, 2016.
[5] B. Park and R. Lu, *Hyperspectral imaging technology in food and agriculture*. Springer, 2015.
[6] R. J. Murphy, S. T. Monteiro, and S. Schneider, "Evaluating classification techniques for mapping vertical geology using field-based hyperspectral sensors," *IEEE Trans. Geosci. Remote Sens.*, vol. 50, no. 8, pp. 3066–3080, 2012.
[7] M. Zhang, C. Hu, M. G. Kowalewski, and S. J. Janz, "Atmospheric correction of hyperspectral gcas airborne measurements over the north atlantic ocean and louisiana shelf," *IEEE Trans. Geosci. Remote Sens.*, vol. 56, no. 1, pp. 168–179, 2017.

[8] B. Zhang, D. Wu, L. Zhang, Q. Jiao, and Q. Li, "Application of hyperspectral remote sensing for environment monitoring in mining areas," *Environ. Earth Sci.*, vol. 65, no. 3, pp. 649–658, 2012.

[9] R. Heylen, A. Zare, P. Gader, and P. Scheunders, "Hyperspectral unmixing with endmember variability via alternating angle minimization," *IEEE Trans. Geosci. Remote Sens.*, vol. 54, no. 8, pp. 4983–4993, 2016.

[10] E. Ibarrola-Ulzurrun, L. Drumetz, J. Marcello, C. Gonzalo-Martín, and J. Chanussot, "Hyperspectral classification through unmixing abundance maps addressing spectral variability," *IEEE Trans. Geosci. Remote Sens.*, 2019.

[11] M. Fauvel, Y. Tarabalka, J. A. Benediktsson, J. Chanussot, and J. C. Tilton, "Advances in spectral-spatial classification of hyperspectral images," *Proc. IEEE.*, vol. 101, no. 3, pp. 652–675, 2013.

[12] L. He, J. Li, C. Liu, and S. Li, "Recent advances on spectral–spatial hyperspectral image classification: An overview and new guidelines," *IEEE Trans. Geosci. Remote Sens.*, vol. 56, no. 3, pp. 1579–1597, 2017.

[13] A. M. Saranathan and M. Parente, "Uniformity-based superpixel segmentation of hyperspectral images," *IEEE Trans. Geosci. Remote Sens.*, vol. 54, no. 3, pp. 1419–1430, 2015.

[14] Q. Gao and S. Lim, "A probabilistic fusion of a support vector machine and a joint sparsity model for hyperspectral imagery classification," *GISCI REMOTE SENS*, no. just-accepted, 2019.

[15] Y. Gu, T. Liu, X. Jia, J. A. Benediktsson, and J. Chanussot, "Nonlinear multiple kernel learning with multiple-structure-element extended morphological profiles for hyperspectral image classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 54, no. 6, pp. 3235–3247, 2016.

[16] P. Ghamisi, J. A. Benediktsson, and M. O. Ulfarsson, "Spectral–spatial classification of hyperspectral images based on hidden Markov random fields," *IEEE Trans. Geosci. Remote Sens.*, vol. 52, no. 5, pp. 2565–2574, 2013.

[17] Y. Qian, M. Ye, and J. Zhou, "Hyperspectral image classification based on structured sparse logistic regression and three-dimensional wavelet texture features," *IEEE Trans. Geosci. Remote Sens.*, vol. 51, no. 4, pp. 2276–2291, 2012.

[18] J. Li, P. R. Marpu, A. Plaza, J. M. Bioucas-Dias, and J. A. Benediktsson, "Generalized composite kernel framework for hyperspectral image classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 51, no. 9, pp. 4816–4829, 2013.

[19] Q. Gao, S. Lim, and X. Jia, "Spectral-spatial hyperspectral image classification using a multiscale conservative smoothing scheme and adaptive sparse representation," *IEEE Trans. Geosci. Remote Sens.*, 2019.

[20] S. Huang, H. Zhang, and A. Pizurica, "Semisupervised sparse subspace clustering method with a joint sparsity constraint for hyperspectral remote sensing images," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, 2019.

[21] Q. Gao, S. Lim, and X. Jia, "Improved joint sparse models for hyperspectral image classification based on a novel neighbour selection strategy," *Remote Sensing*, vol. 10, no. 6, p. 905, 2018.

[22] S. Huang, H. Zhang, and A. Pižurica, "A robust sparse representation model for hyperspectral image classification," *Sensors*, vol. 17, no. 9, p. 2087, 2017.

[23] Q. Gao, S. Lim, and X. Jia, "Hyperspectral image classification using joint sparse model and discontinuity preserving relaxation," *IEEE Geoscience and Remote Sensing Letters*, vol. 15, no. 1, pp. 78–82, 2017.

[24] Y. Gu, J. Chanussot, X. Jia, and J. A. Benediktsson, "Multiple kernel learning for hyperspectral image classification: A review," *IEEE Trans. Geosci. Remote Sens.*, vol. 55, no. 11, pp. 6547–6565, 2017.

[25] X. Zhang, Y. Liang, C. Li, N. Huyan, L. Jiao, and H. Zhou, "Recursive autoencoders-based unsupervised feature learning for hyperspectral image classification," *IEEE Geosci. Remote Sens. Lett.*, vol. 14, no. 11, pp. 1928–1932, 2017.

[26] X. Kang, C. Li, S. Li, and H. Lin, "Classification of hyperspectral images by Gabor filtering based deep network," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 11, no. 4, pp. 1166–1178, 2018.

[27] Y. Chen, X. Zhao, and X. Jia, "Spectral–spatial classification of hyperspectral data based on deep belief network," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 8, no. 6, pp. 2381–2392, 2015.

[28] P. Zhong, Z. Gong, S. Li, and C.-B. Schönlieb, "Learning to diversify deep belief networks for hyperspectral image classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 55, no. 6, pp. 3516–3530, 2017.

[29] L. Mou, P. Ghamisi, and X. X. Zhu, "Deep recurrent neural networks for hyperspectral image classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 55, no. 7, pp. 3639–3655, 2017.

[30] R. Hang, Q. Liu, D. Hong, and P. Ghamisi, "Cascaded recurrent neural networks for hyperspectral image classification," *IEEE Trans. Geosci. Remote Sens.*, 2019.

[31] Y. Chen, H. Jiang, C. Li, X. Jia, and P. Ghamisi, "Deep feature extraction and classification of hyperspectral images based on convolutional neural networks," *IEEE Trans. Geosci. Remote Sens.*, vol. 54, no. 10, pp. 6232–6251, 2016.

[32] H. Lee and H. Kwon, "Going deeper with contextual CNN for hyperspectral image classification," *IEEE Trans. Image Process.*, vol. 26, no. 10, pp. 4843–4855, 2017.

[33] Y. Wei, Y. Zhou, and H. Li, "Spectral-spatial response for hyperspectral image classification," *Remote Sens.*, vol. 9, no. 3, p. 203, 2017.

[34] M. Zhang, W. Li, and Q. Du, "Diverse region-based CNN for hyperspectral image classification," *IEEE Trans. Image Process.*, vol. 27, no. 6, pp. 2623–2634, 2018.

[35] S. Li, W. Song, L. Fang, Y. Chen, P. Ghamisi, and J. A. Benediktsson, "Deep learning for hyperspectral image classification: An overview," *IEEE Trans. Geosci. Remote Sens.*, 2019.

[36] Q. Gao, S. Lim, and X. Jia, "Hyperspectral image classification using convolutional neural networks and multiple feature learning," *Remote Sens.*, vol. 10, no. 2, p. 299, 2018.

[37] B. Pan, Z. Shi, and X. Xu, "R-VCANet: a new deep-learning-based hyperspectral image classification method," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 10, no. 5, pp. 1975–1986, 2017.

[38] Y. Zhou and Y. Wei, "Learning hierarchical spectral–spatial features for hyperspectral image classification," *IEEE Trans. Cybern.*, vol. 46, no. 7, pp. 1667–1678, 2015.

[39] Y. Li, H. Zhang, and Q. Shen, "Spectral–spatial classification of hyperspectral imagery with 3d convolutional neural network," *Remote Sens.*, vol. 9, no. 1, p. 67, 2017.

[40] Z. Zhong, J. Li, Z. Luo, and M. Chapman, "Spectral–spatial residual network for hyperspectral image classification: A 3-d deep learning framework," *IEEE Trans. Geosci. Remote Sens.*, vol. 56, no. 2, pp. 847–858, 2018.

[41] Y. Xu, L. Zhang, B. Du, and F. Zhang, "Spectral-spatial unified networks for hyperspectral image classification," *IEEE Trans. Geosci. Remote Sens.*, no. 99, pp. 1–17, 2018.

[42] W. Zhao and S. Du, "Spectral–spatial feature extraction for hyperspectral image classification: A dimension reduction and deep learning approach," *IEEE Trans. Geosci. Remote Sens.*, vol. 54, no. 8, pp. 4544–4554, 2016.

[43] J. Yue, S. Mao, and M. Li, "A deep learning framework for hyperspectral image classification using spatial pyramid pooling," *Remote Sens. Lett.*, vol. 7, no. 9, pp. 875–884, 2016.

[44] S. Hao, W. Wang, Y. Ye, T. Nie, and L. Bruzzone, "Two-stream deep architecture for hyperspectral image classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 56, no. 4, pp. 2349–2361, 2018.

[45] J. Yang, Y.-Q. Zhao, and J. C.-W. Chan, "Learning and transferring deep joint spectral–spatial features for hyperspectral classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 55, no. 8, pp. 4729–4742, 2017.

[46] W. Song, S. Li, L. Fang, and T. Lu, "Hyperspectral image classification with deep feature fusion network," *IEEE Trans. Geosci. Remote Sens.*, vol. 56, no. 6, pp. 3173–3184, 2018.
[47] W. Li, G. Wu, F. Zhang, and Q. Du, "Hyperspectral image classification using deep pixel-pair features," *IEEE Trans. Geosci. Remote Sens.*, vol. 55, no. 2, pp. 844–853, 2017.
[48] X. Cao, F. Zhou, L. Xu, D. Meng, Z. Xu, and J. Paisley, "Hyperspectral image classification with Markov random fields and a convolutional neural network," *IEEE Trans. Image Process.*, vol. 27, no. 5, pp. 2354–2367, 2018.
[49] S. Mei, J. Ji, J. Hou, X. Li, and Q. Du, "Learning sensor-specific spatial-spectral features of hyperspectral images via convolutional neural networks," *IEEE Trans. Geosci. Remote Sens.*, vol. 55, no. 8, pp. 4520–4533, 2017.
[50] J. Hu, L. Shen, and G. Sun, "Squeeze-and-excitation networks," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit (CVPR)*, 2018, pp. 7132–7141.
[51] V. Nair and G. E. Hinton, "Rectified linear units improve restricted boltzmann machines," in *Pro ICML*, 2010, pp. 807–814.
[52] S. Ioffe and C. Szegedy, "Batch normalization: Accelerating deep network training by reducing internal covariate shift," *arXiv preprint arXiv:1502.03167*, 2015.
[53] G. Hughes, "On the mean accuracy of statistical pattern recognizers," *IEEE Trans. Inf. Theory.*, vol. 14, no. 1, pp. 55–63, 1968.
[54] G. Cheng, Z. Li, J. Han, X. Yao, and L. Guo, "Exploring hierarchical convolutional features for hyperspectral image classification," *IEEE Trans. Geosci. Remote Sens.*, no. 99, pp. 1–11, 2018.
[55] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit (CVPR)*, 2016, pp. 770–778.
[56] S. Yu, S. Jia, and C. Xu, "Convolutional neural networks for hyperspectral image classification," *Neural Comput.*, vol. 219, pp. 88–98, 2017.
[57] J. Zhu, L. Fang, and P. Ghamisi, "Deformable convolutional neural networks for hyperspectral image classification," *IEEE Geosci. Remote Sens. Lett.*, 2018.
[58] M. D. Zeiler, "Adadelta: an adaptive learning rate method," *arXiv preprint arXiv:1212.5701*, 2012.
[59] B. Pan, Z. Shi, and X. Xu, "Mugnet: Deep learning for hyperspectral image classification using limited samples," *ISPRS J. Photogramm. Remote Sens.*, vol. 145, pp. 108–119, 2018.
[60] H. Gao, Y. Yang, S. Lei, C. Li, H. Zhou, and X. Qu, "Multi-branch fusion network for hyperspectral image classification," *Knowledge-Based Systems*, vol. 167, pp. 11–25, 2019.
[61] Z. Li, L. Huang, and J. He, "A multiscale deep middle-level feature fusion network for hyperspectral classification," *Remote Sensing*, vol. 11, no. 6, p. 695, 2019.
[62] A. J. Guo and F. Zhu, "A cnn-based spatial feature fusion algorithm for hyperspectral imagery classification," *IEEE Transactions on Geoscience and Remote Sensing*, 2019.
[63] S. Wan, C. Gong, P. Zhong, B. Du, L. Zhang, and J. Yang, "Multi-scale dynamic graph convolutional network for hyperspectral image classification," *arXiv preprint arXiv:1905.06133*, 2019.

Xian Li (S'19) received the M.S. degree from Harbin Institute of Technology, Harbin, China, in 2016, where he is currently pursuing the Ph.D. degree in instrument science and technology with the School of Instrumentation Science and Engineering. He is also a doctoral researcher with the Department of Telecommunications and Information Processing, UGent-GAIM, Ghent University, Belgium, supported by the China Scholarship Council. His research interests include deep learning, hyperspectral remote sensing image analysis.

Mingli Ding received the B.S. and the Ph.D. degrees from Harbin Institute of Technology, Harbin, China, in 2000 and 2005, respectively. From 2009 to 2010, he was a Visiting Scholar with the French National Center for Scientific Research, Toulouse, France. He is currently a Full Professor with the School of Instrumentation Science and Engineering, Harbin Institute of Technology, China.

His research interests include deep learning, image classification, object detection, automation test technology, and information processing.

Aleksandra Pižurica (SM'15) received the Diploma in electrical engineering from the University of Novi Sad, Serbia, in 1994, the Master of Science degree in telecommunications from the University of Belgrade, Serbia, in 1997, and the Ph.D. degree in engineering from Ghent University, Belgium, in 2002.

She is a Professor in statistical image modeling with Ghent University. Her research interests include the area of signal and image processing and machine learning, including multiresolution statistical image models, Markov Random Field models, sparse coding, representation learning, and image and video reconstruction, restoration, and analysis.

Prof. Pižurica served as an Associate Editor for the IEEE TRANSACTIONS ON IMAGE PROCESSING (2012 – 2016), Senior Area Editor for the IEEE TRANSACTIONS ON IMAGE PROCESSING (2016 – 2019) and currently an Associate Editor for the IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS FOR VIDEO TECHNOLOGY. She was also the Lead Guest Editor for the EURASIP Journal on Advances in Signal Processing for the Special Issue "Advanced Statistical Tools for Enhanced Quality Digital Imaging with Realistic Capture Models" (2013). The work of her team has been awarded twice the Best Paper Award of the IEEE Geoscience and Remote Sensing Society Data Fusion contest, in 2013 and 2014. She received the scientific prize "de Boelpaepe" for 2013–2014, awarded by the Royal Academy of Science, Letters and Fine Arts of Belgium for her contributions to statistical image modeling and applications to digital painting analysis.