This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/JSTARS.2022.3179446, IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing

Multiscale Spatial-Spectral Feature Extraction Network for Hyperspectral Image Classification

Zhen Ye, Cuiling Li, Qingxin Liu, Lin Bai, *Member, IEEE*, and James E. Fowler, *Fellow, IEEE*

***Abstract*—Convolutional neural networks have garnered increasing interest for the supervised classification of hyperspectral imagery. However, images with a wide variety of spatial land-cover sizes can hinder the feature-extraction ability of traditional convolutional networks. Consequently, many approaches intended to extract multiscale features have emerged; these techniques typically extract features in multiple parallel branches using convolutions of differing kernel sizes, with concatenation or addition employed to fuse the features resulting from the various branches. In contrast, the present work explores a multiscale spatial-spectral feature-extraction network that operates in a more granular manner. Specifically, in the proposed network, a multi-branch structure expands the convolutional receptive fields through the partitioning of input feature maps, applying hierarchical connections across the partitions, cross-channel feature fusion via pointwise convolution, and depthwise 3D convolutions for feature extraction. Experimental results reveal that the proposed multiscale spatial-spectral feature-fusion network outperforms other state-of-the-art networks at supervised classification of hyperspectral imagery while being robust to limited training data.**

***Index Terms*—Convolutional neural networks (CNNs); feature fusion; multiscale feature extraction; hyperspectral-image (HSI) classification.**

I. Introduction

THE recent renaissance of neural networks (NNs) for machine learning has led to increasing interest in NN-driven supervised classification for hyperspectral imagery (HSI). Initial focus centered exclusively on spectral features, often treating hyperspectral pixel vectors as 1D sequence data fed into a recurrent neural network (RNN) to effectuate classification (e.g., [1], [2]). However, the realization that the incorporation of spatial features can improve HSI classification beyond spectrum-exclusive methods has inspired increasing interest in spatial-spectral methods for HSI classification (e.g., [3], [4]). Recent efforts along these lines have included, for example, the use of superpixels [5]–[8], graph convolutional networks [9], transformers [10], and multimodal classifiers [11]. Alternatively, there is increasing interest in NN-based spatial-spectral classifiers that are built upon convolutional neural networks (CNNs), with two general strategies existing in the literature. The first strategy combines separate 1D and 2D CNNs, with the 1D network extracting information in the spectral dimension while the 2D network captures information in the spatial dimensions (e.g., [12], [13]). The second strategy deploys 3D CNNs to extract spatial-spectral features jointly (e.g., [14]–[22]).

The most straightforward paradigm in this second category comprises simply a cascade of multiple 3D-CNN layers for spatial-spectral feature extraction followed by a fully connected (FC) layer implementing the classification—this is essentially the approach taken by [14]–[16], for example, while [17] follows a similar framework but with the addition of a variety of skip connections to yield a spectral-spatial residual network (SSRN). Alternatively, several techniques combine 2D and 3D CNNs; such methods include the hybrid spectral network (HybridSN) of [18] which follows a cascade of 3D-CNN layers with a single 2D-CNN layer intended to enhance the extraction of spatial features. The mixed-CNN (MCNN) network of [19] essentially follows the same 3D-2D approach as HybridSN but follows the 2D CNN with covariance pooling to extract second-order information from features before rendering the final classification decision, while [20] also adopted a similar 3D-2D hybrid strategy, albeit with multiple 2D layers. Alternatively, [21] employs both 3D and 2D CNNs simultaneously in multiple parallel branches followed by fusion of the resulting features from the respective branches. Finally, [22] reverses the typical order, applying 2D CNNs first in order to simplify the features before following with 3D CNNs.

Although these various 3D-CNN-driven classifiers can achieve quite effective performance in many scenarios, they are hindered in the common situation in which land-cover regions vary significantly in size of spatial coverage. That is, a large diversity of spatial land-cover sizes can impede the extraction of spatial features. While increasing the number of 3D-CNN layers—which effectively enlarges the range of the receptive fields of the CNNs—may alleviate this issue to a certain extent, a deeper network entails significantly more computation and memory and, importantly, is often more difficult to train.

As an alternative to using ever-deeper networks for HSI classification, there has been increasing recent interest in multiscale network architectures (e.g., [23]–[35]) which permit enlarging the range of the receptive fields of a CNN-based network without incurring the computational and training issues entailed by simply adding more CNN layers. Typically,

This work was supported by a Key Research & Development Program of Shaanxi Province (No. 2020GY-060), a Xi'an Science & Technology Project (No. 2020KJRC0126, 202012), a National Key Research & Development Program of China (No. 2020YFC1512002), and the National Natural Science Foundation of China (Grant Nos. 61601059 and 41601344). (*Corresponding author: Lin Bai).*

Z. Ye, C. Li, Q. Liu, and L. Bai are with the School of Electronics and Control Engineering, Chang'an University, Xi'an 710064, China (e-mail: e-mail: yezhen525@126.com; 2020132046@chd.edu.cn; 2019132074@chd.edu.cn; linbai@chd.edu.cn).

J. E. Fowler is with the Department of Electrical and Computer Engineering, Mississippi State University, Mississippi State, MS 39762 USA. (e-mail: fowler@ece.msstate.edu).

such a multiscale network consists of, in essence, multiple parallel branches of feature extraction in conjunction with some form of feature fusion prior to classification, with the branches being designed to independently extract features at differing scales. Commonly, each branch implements a cascade of one or more convolutional layers with feature summation or concatenation providing the fusion of features from the multiple branches. One of the first uses of this multi-branch paradigm for multiscale feature extraction in HSI classification was the multiscale 3D deep CNN (M3D-DCNN) of [23] which features multiple parallel convolutional branches—each extracting features at a different scale via convolutional kernels of different sizes—coupled with summing of the branch outputs to provide feature fusion. While multi-branch network architectures permit multiscale feature extraction without the difficulties associated with a deeper, single-branch network, the number of different scales is effectively limited by the number of branches, with each additional branch contributing additional network complexity and associated computation and memory burden.

In contrast to this multi-branch paradigm for multiscale feature extraction, [36] proposes a network architecture that widens the range of CNN receptive fields in a more granular manner by splitting the channels of the input feature map into multiple partitions and applying hierarchical connections among the partitions. In this fashion, a large number of scales can be implemented since the partitioning keeps computation in check while the partition-to-partition connections alleviate the vanishing gradients that often impede the training of deep networks. The large number of scales thus achieved effectively permits a fine-grained multiscale representation, as opposed to the more coarse-grained structure of the typical multi-branch architecture. Inspired by [36], [37] proposes a similar fine-grained multiscale feature extraction for remote-sensing imagery. We note that, being intended for 2D imagery, both [36] and [37] employ 2D CNNs within each partition.

In this paper, a multiscale spatial-spectral feature extraction network—which we call MS^{2}FENet—is proposed to extract spatial-spectral features for HSI classification. For multiscale feature extraction, MS^{2}FENet comprises feature reuse within a multi-branch structure that expands the CNN receptive fields. In so doing, MS^{2}FENet adapts the fine-grained multiscale feature extraction of [37] to HSI data in order to provide more granular multiscale features than those of the coarse-grained multi-branch architectures (such as [23]–[35]) common in the literature. Accordingly, since MS^{2}FENet is designed for HSI volumes, we replace the 2D CNNs in [37] with 3D CNNs. However, in order to circumvent the excessive computational and memory burdens associated with conventional 3D convolution, we adopt depthwise 3D convolutions for the multiscale feature-extraction 3D CNNs. Additionally, rather than simply concatenating features between the multiscale branches as in [37], MS^{2}FENet employs a convolution-driven feature fusion to combine information across channels more effectively than simple feature addition or concatenation. The primary contributions of this manuscript are as follows:

1) Multiscale spatial-spectral features are extracted from HSI via a multi-branch structure capitalizing on feature reuse and depthwise 3D convolution. The multiscale nature of the feature extraction is finely grained due to channel-wise partitioning coupled with multi-branch connections that alleviate computational and training difficulties of the more common coarse-grained multi-branch approaches in existing literature. Our approach is, to the best of our knowledge, the first to apply such fine-grained multiscale feature extraction to the HSI-classification problem.
2) Within the fine-grained multiscale feature extraction, features are combined by a feature-fusion process driven by pointwise convolutions in order to effectuate multi-branch connections in a manner more sophisticated than simple feature addition or concatenation.
3) The proposed MS^{2}FENet is evaluated via a battery of experimental results in which it is compared to existing state-of-the-art CNN-based approaches for HSI classification. These experimental results reveal that, not only does MS^{2}FENet outperform competing approaches, it also provides highly stable performance as the size of the training set becomes very small.

The remainder of this paper is organized as follows. Sec. II describes the proposed MS^{2}FENet and its constituent components in detail. An overview of the datasets and experimental setup is presented in Sec. III, while Sec. IV discusses experimental results as well as the influence of various parameters in the MS^{2}FENet design. Finally, we make concluding remarks in Sec. V.

II. The Proposed MS^{2}FENet Architecture

In this section, we describe the structure of MS^{2}FENet, an overview of which is shown in Fig. 1. The network is composed of several multiscale spatial-spectral feature-extraction (MS^{2}FE) modules, in which multiscale-fusion-convolution (MFC) layers extract multiscale spatial-spectral features. In the MFC layers, feature reuse and feature fusion are conducted via a multi-branch structure in order to expand convolutional receptive fields, while the feature-fusion block combines information from different channels by fusing multiscale features. Moreover, in the multi-branch structure of each MFC layer, depthwise 3D convolution extracts spatial-spectral features while reducing the number of parameters in the convolution. Below, we describe each of the MS^{2}FENet components in detail, starting with an overview of the overall MS^{2}FENet architecture in Sec. II-A and a detailed description of the MFC layer in Sec. II-B. Additionally, we compare and contrast depthwise 3D convolution to conventional 3D convolution in Sec. II-C while describing the feature-fusion block in Sec. II-D.

A. The Overall Network Structure

The overall structure of the proposed MS^{2}FENet is shown in Fig. 1. First, $X \in \mathbb{R}^{S\times M\times N}$ is a hyperspectral data cube, where S, M, and N are the number of spectral bands, spatial height, and spatial width, respectively of the cube. Principal component analysis (PCA) [38] is used to reduce the spectral dimension of the original image to B principal

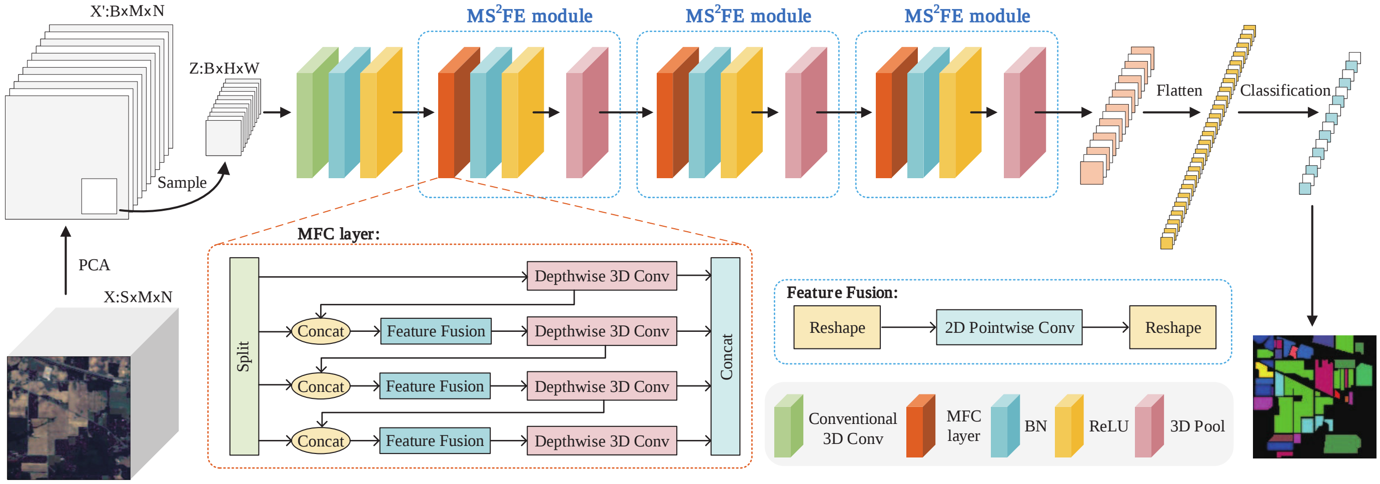


Fig. 1. The overall structure of MS^{2}FENet including MS^{2}FE modules, MFC layers, and feature-fusion blocks.

components (PCs) such that $X' \in \mathbb{R}^{B\times M\times N}$. Since a single pixel cannot be sent into the CNN for feature extraction, we sample a small cube $Z \in \mathbb{R}^{B\times H\times W}$ spatially surrounding the current pixel to be classified from X' such that the class label determined for Z is the label assigned to the central pixel. The Z cube then proceeds through a conventional 3D convolution, followed by batch-normalization (BN) [39] and rectified-linear-unit (ReLU) [40] layers, to expand the number of channels of the sampled data and to realize the extraction of preliminary spatial-spectral features. The size of the conventional 3D convolution kernel is 7×3×3, where 7 is the spectral dimension, and 3×3 is the spatial dimension.

Next, the proposed MS^{2}FE module is applied three times to extract multiscale spatial-spectral features. Each MS^{2}FE module includes MFC, BN, and ReLU layers, along with a 3D average-pooling layer. In the MFC layers, feature reuse and feature fusion are implemented via a multi-branch structure to expand the receptive fields of depthwise 3D convolution, while the feature-fusion block combines information from the various channels by fusing multiscale features. Consequently, the MFC layer extracts discriminative multiscale spatial-spectral features. The MFC layer does not downsample in order to facilitate the concatenation of the multiscale feature maps; rather, 3D average pooling realizes downsampling at the end of the MS^{2}FE module. We note that the BN and ReLU layers within the MS^{2}FE module are intended to help to suppress gradient disappearance and to expedite training.

After the MS^{2}FE modules, a flattening operation is used to convert the feature map into a vector, which is sent to an FC layer and ultimately to a softmax classifier to obtain the final class predicted label. Table I gives a detailed summary of the proposed network in terms of layer type as well as input and output feature-map sizes for the Houston dataset as an example (see Sec. III-A for a description of this dataset). It can be seen that the output size of the last FC layer is 15, which matches the number of land-cover classes of the Houston dataset.

TABLE I
NETWORK LAYER SIZES FOR THE HOUSTON DATASET

	Layer	Input size	Output size
1	Input	-	144×349×1905
2	PCA	144×349×1905	30×349×1905
3	Sample	30×349×1905	30×15×15
4	Conventional 3D Conv	30×15×15	8×24×13×13
5	MFC layer	8×24×13×13	20×24×13×13
6	3D Pooling	20×24×13×13	20×12×6×6
7	MFC layer	20×12×6×6	50×12×6×6
8	3D Pooling	50×12×6×6	50×6×2×2
9	MFC layer	50×6×2×2	128×6×2×2
10	3D Pooling	128×6×2×2	128×3×2×2
11	Flatten	128×3×2×2	1536
12	FC	1536	15

B. The MFC Layer

Each MS^{2}FE module comprises MFC, BN, and ReLU layers followed by a 3D average-pooling layer. Among these, the MFC layer serves as the heart of the MS^{2}FE module in that it is designed specifically to extract multiscale spatial-spectral features of HSIs in a manner that is more finely grained than that of other multi-branch architectures common in the literature. Specifically, the MFC layer we propose here adapts the multiscale network of [37], originally proposed for three-band remote-sensing images, to HSI. In doing so, we adapt the 2D CNNs used in [37] to 3D. While conventional 3D convolutions can extract spatial-spectral features, such conventional 3D convolutions entail an enormous number of network parameters when applied to HSI data. Accordingly, we use depthwise 3D convolution within each branch to extract spatial-spectral features, which reduces the number of convolution parameters as well as the resulting complexity of the network while still expanding the receptive fields of convolution. We discuss depthwise 3D convolution in more detail below in Sec. II-C.

Each MFC layer splits the input feature map into 4 partitions along the channel dimension. As an example, Fig. 2 illustrates

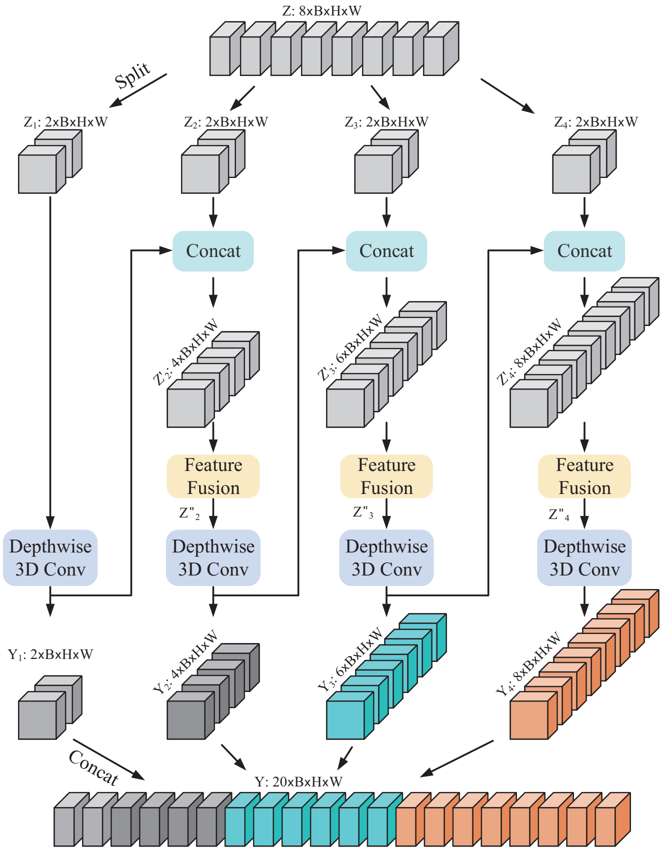


Fig. 2. The architecture of the proposed MFC layer for an 8-channel input feature map.

the operation of an MFC layer for an input feature map with 8 channels (i.e., the first MFC layer in Table I). That is, in Fig. 2, the input feature map Z is partitioned by the split function $\mathcal{S}(\cdot)$; i.e.,

$$\{Z_1, Z_2, Z_3, Z_4\} = \mathcal{S}(Z), \tag{1}$$

such that the C channels of Z are partitioned into Z_1, Z_2, Z_3, and Z_4 with $C/4$ channels each[1]. Then, first input partition Z_1 is subjected to depthwise 3D convolution to extract spatial-spectral features, yielding the output feature map Y_1,

$$Y_1 = \mathcal{D}(Z_1), \tag{2}$$

where $\mathcal{D}(\cdot)$ denotes depthwise 3D convolution.

Next, we concatenate Y_1 to the second input partition Z_2; this channelwise concatenation effectively realizes feature reuse. The resulting concatenated feature map Z_2' is then sent to the feature-fusion block to fuse multiscale information, yielding feature map Z_2'' which is, in turn, sent to depthwise 3D convolution to extract multiscale spatial-spectral features. That is,

$$Z_i' = \mathcal{C}(Y_{i-1}, Z_i) \tag{3}$$
$$Z_i'' = \mathcal{F}(Z_i') \tag{4}$$
$$Y_i = \mathcal{D}(Z_i''), \tag{5}$$

where $\mathcal{C}(\cdot)$ is channelwise concatenation, and $\mathcal{F}(\cdot)$ is feature fusion (discussed below in Sec. II-D). This process is repeated for $i \in \{2, 3, 4\}$ to process the input feature maps of all partitions. Finally, all the output feature maps are concatenated as

$$Y = \mathcal{C}(Y_1, Y_2, Y_3, Y_4) \tag{6}$$

to yield the final MFC feature map, Y.

In this process, we employ feature reuse to increase the information interaction between different partitions. That is, consider the output Y_1 of the first partition—on the one hand, the output Y_1 of the first partition is initially concatenated to the input Z_2 of the second partition, while, on the other hand, Y_1 is also concatenated to the output of the other partitions to obtain the final output Y; thus, the network uses feature Y_1 multiple times. The MFC layer employs similar feature reuse for Y_2 and Y_3 as well.

Furthermore, the architecture enlarges the receptive field in a multiscale manner. For example, consider the second partition—the concatenated feature map Z_2' is fed into feature fusion and depthwise 3D convolution within the second partition. Consequently, in the depthwise 3D convolution within the second partition, the receptive field of the feature map of half the channels (i.e., those from Y_1) is larger than that of the other half. Accordingly, the MFC layer achieves multiscale spatial-spectral feature extraction in a fine-grained manner by expanding the receptive field of convolution.

More specifically, a traditional multiscale feature extraction (such as the M3D-DCNN of [23]) employs multiple parallel branches, each with a CNN of a fixed kernel size, such that, by using different kernel sizes in the different branches, extraction of features with multiple scales is accomplished. In contrast, in the MFC layer as proposed here, each partition uses the same CNN kernel size. The multiscale nature of MFC arises instead from the fact that the output of each CNN is fed into the subsequent partition (i.e., feature reuse) which expands the effective receptive field of the subsequent CNN, thereby providing multiscale feature extraction. Take a cascade of two consecutive $3 \times 3 \times 3$ convolutions as an example: the receptive field of the first convolution is $3 \times 3 \times 3$ (each output value addresses 27 values in the input feature map), while the receptive field of the second convolution kernel is effectively $9 \times 9 \times 9$ (each output value addresses $9^3 = 729$ values in the original input feature map). The receptive field of the second convolution is larger than that of the preceding convolution even though the kernel itself is the same size; thus, the receptive field is enlarged without the cost of actually using a larger kernel in the CNN. Additionally, the partitioning along the channel dimension in MFC further keeps computation in check. In the traditional "coarse-grained" approach of using parallel branches with different kernel sizes, computational issues limit the number of branches to typically no more than three. MFC faces no such limitation, as MFC could, in theory, have as many partitions as channels—this is what makes MFC much more fine-grained in its multiscale nature than the traditional approach.

As can be seen in Fig. 1, in the overall MS^{2}FENet, there are three MFC layers (one in each MS^{2}FE module). In the

[1] If C is not evenly divisible by 4, then Z_2, Z_3, and Z_4 each receive $\lfloor C/4 \rfloor$ channels while Z_1 receives the remaining $C - 3\lfloor C/4 \rfloor$ channels.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/JSTARS.2022.3179446, IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing

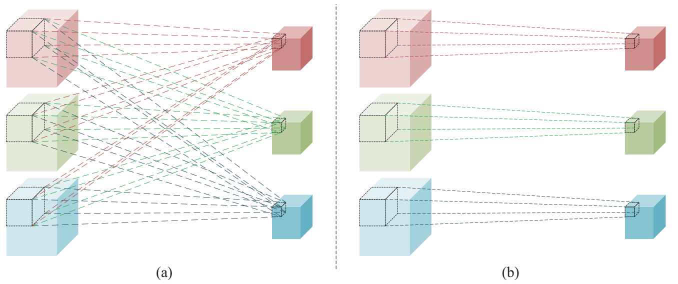


Fig. 3. (a) Conventional 3D convolution, (b) depthwise 3D convolution.

TABLE II
COMPARISON OF MFC-LAYER COMPLEXITY FOR CONVENTIONAL AND DEPTHWISE 3D CONVOLUTION

	Parameters	FLOPs
Conventional	$\frac{15}{8}C^2bhw + \frac{29}{16}C^2B^2$	$\frac{15}{8}C^2bhwBHW + \frac{29}{16}C^2B^2HW$
Depthwise	$\frac{5}{2}Cbhw + \frac{29}{16}C^2B^2$	$\frac{5}{2}CbhwBHW + \frac{29}{16}C^2B^2HW$

first MFC layer, a kernel size of $5 \times 3 \times 3$ is used for each depthwise 3D convolution; the second and third MFC layers both use $3 \times 3 \times 3$ kernels. We note that, although the size of the convolution kernel is the same spatially within an MFC layer, multiscale feature extraction is nonetheless realized by the changing receptive field of the convolution kernel that occurs due to feature reuse as described above.

Finally, we observe that, since each MFC layer concatenates feature maps along the channel dimension, all tensor dimensions (except the channel dimension) must be the identical. Consequently, the stride of depthwise 3D convolution is set to 1 such that there is no downsampling of the feature maps within the depthwise 3D convolutions of the MFC layer. To compensate, at the end of each MS^{2}FE module, we apply 3D average pooling to downsample the output feature maps.

C. Depthwise 3D Convolution

In the spatial domain of HSIs, there is local correlation between neighboring pixels. Accordingly, a convolution kernel of spatial size of $k \times k$ is better able to extract spatial features and represent spatial correlation as k becomes larger. However, in conventional 3D convolution, each kernel must be convolved over each channel of the input feature map, as is illustrated in Fig. 3(a). Thus, although conventional 3D convolution can extract rich spatial-spectral features, the number of network parameters entailed is very large. In order to reduce network complexity, depthwise convolution is widely used for 2D CNNs. Likewise, we adopt depthwise 3D convolution for the MFC layer proposed here. That is, while conventional 3D convolution is used in MS^{2}FENet for preliminary feature extraction (i.e., layer 4 in Table I), we deploy depthwise 3D convolution for the remaining convolutions within the MFC layers.

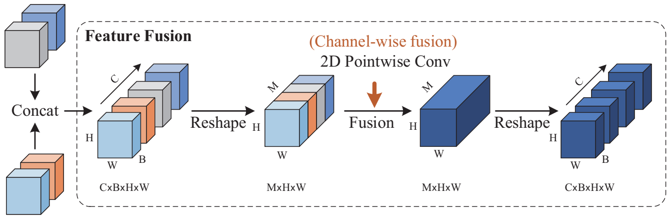


Fig. 4. The structure of feature-fusion block. The 4D feature map of size $C \times B \times H \times W$ is reshaped into a 3D feature map of size $M \times H \times W$, where $M = CB$.

Depthwise 3D convolution is illustrated in Fig. 3(b) wherein it can be seen that depthwise 3D convolution applies one filter per input channel; accordingly, the number of channels in the input and output feature maps is the same. If the size of the MFC-layer input feature map is $C \times B \times H \times W$, and the kernel size is $b \times h \times w$, then Table II compares the complexity, in terms of numbers of parameters as well as floating-point operations per second (FLOPs), of the MFC layer using both depthwise and conventional 3D convolution. It can be seen that depthwise 3D convolution results in significantly reduced MFC-layer complexity. For example, for a $5 \times 3 \times 3$ kernel and an $8 \times 24 \times 13 \times 13$ input feature map, depthwise 3D convolution results in an MFC layer with $67,716$ parameters and 1.5×10^7 flops, while conventional 3D convolution entails $72,216$ parameters and 3.3×10^7 flops; note in particular that conventional convolution requires over twice as many flops in this example.

D. Feature Fusion

In traditional multi-branch multiscale feature extraction in existing literature (e.g., [23], [27], [34]), features extracted from different branches are simply added or concatenated. In contrast, however, the feature-fusion block in our proposed MFC layer fuses information from different convolution channels to achieve a more fine-grained interaction among the channels. To achieve this inter-channel information interaction, we use 2D pointwise convolution across a 3D feature map as illustrated in Fig. 4.

In more detail, there are three steps shown in Fig. 4, including two reshape operation and a 2D pointwise convolution. First, the feature maps from two MFC partitions are concatenated along the channel dimension prior to the feature-fusion block. Then, the resulting 4D feature map is reshaped into a 3D feature map in order to accommodate 2D pointwise convolution. Specifically, before entering the feature-fusion block, we fuse the channel dimension C and spectral dimension B of the input feature map into one dimension M, with $M = CB$. After reshaping, the size of the resulting 3D feature map is $M \times H \times W$ which is suitable to 2D convolution. This 2D pointwise convolution (i.e., 2D convolution with M kernels of size 1×1) serves to fuse information across the M channels. Finally, the fused feature map is reshaped again into 4D in preparation for the subsequent depthwise 3D convolution to come in the MFC layer. The feature-fusion block is used in the last three partitions of MFC layer. For the three partitions, the size of the feature map before entering the feature-fusion

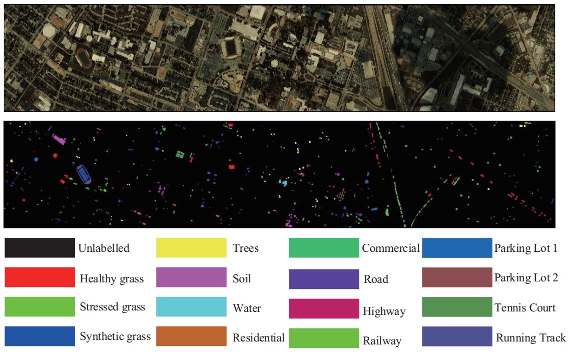


Fig. 5. False-color image (upper) and ground-truth map (lower) of the Houston dataset.

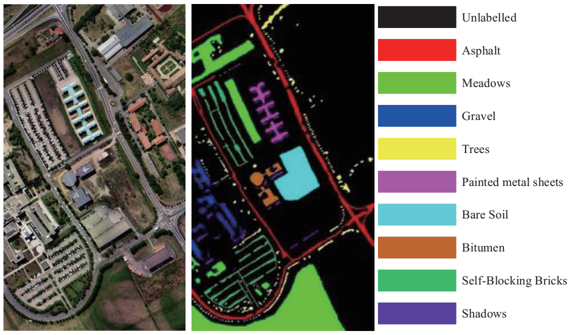


Fig. 6. False-color image (left) and ground-truth map (right) of the Pavia University dataset.

block is the same as that after the output of the feature-fusion block.

III. Datasets and Experimental Setup

A. Data Description

We have selected three widely-used datasets: Houston, Pavia University, and Botswana, which were acquired by different sensors. The Houston dataset[2] was collected by the Compact Airborne Spectrographic Imager (CASI) over the campus of the University of Houston and the neighboring urban area in June 2012. The dataset has a high spatial resolution of 349×1905 pixels. There are 15 classes, and the dataset is composed of 144 spectral bands. A false-color image and ground-truth map are shown in Fig. 5.

The Pavia University dataset [41] consists of a portion of data collected by the Reflective Optics Spectrometer Imaging System (ROSIS) sensor in Pavia, Italy. The Pavia University dataset contains 9 classes. The dataset contains 103 usable spectral bands after removing 12 noise bands; the spatial size of the dataset is 610×340, while the spatial resolution is 1.3 m. Fig. 6 shows the false-color image and ground-truth for Pavia University.

The NASA EO-1 Hyperion sensor obtained the Botswana dataset[3] above the Okavango Delta. There are 145 spectral bands after uncalibrated and noisy bands are removed. The spatial size of the Botswana dataset is $1,476 \times 256$ with 30-m spatial resolution, and it has 14 identified classes. Its false-color image and ground-truth map are shown in Fig. 7.

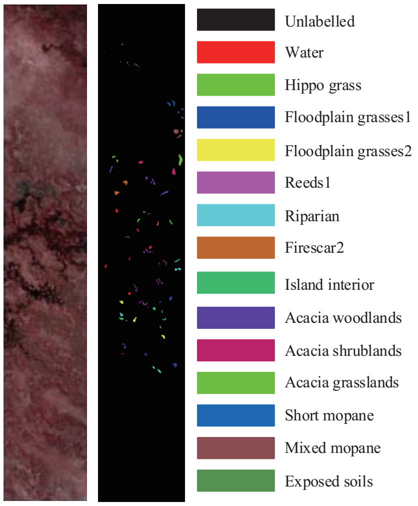


Fig. 7. False-color image (left) and ground-truth map (right) of the Botswana dataset.

B. Parameter Settings

A parameter of paramount importance to network performance is the training ratio, which we set to 5% unless otherwise stated. For a training ratio of 5%, we randomly select 5% of each dataset as a training/validation set with the remaining 95% of the dataset designated as the test set. During training of the proposed network, we divide the training/validation set randomly into two equal-size parts—one part is used to train the network, while the other part is a validation set used to tune the network hyperparameters, which are set as follows.

As illustrated in Fig. 1, the original S-band HSI dataset is reduced in dimension to B PCs via PCA before being spatially sampled to an $H \times W$ window size and being fed into 3D convolution, BN, ReLU, and a cascade of MS^{2}FE modules. Setting $H = W$, in Fig. 8, we investigate performance in terms of overall accuracy (OA) of classification on the three datasets over wide ranges of B and W values to determine the best settings of these two parameters.

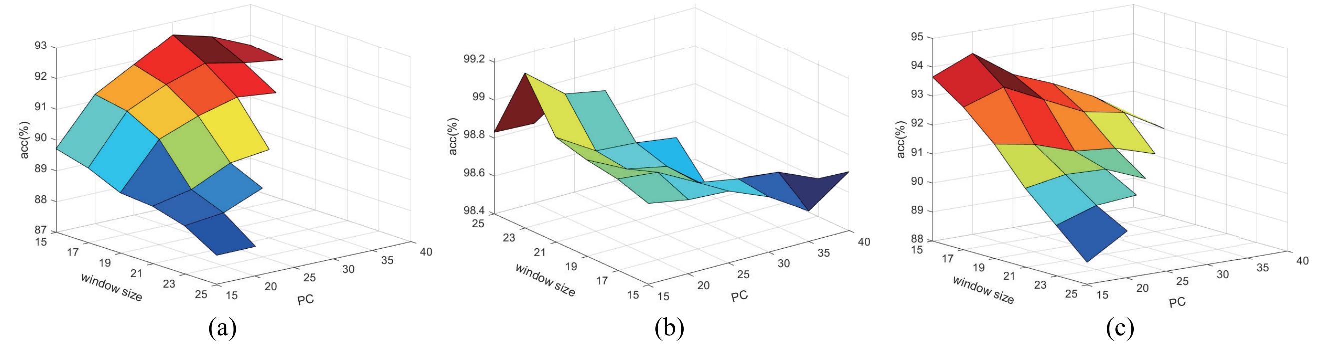
As can be seen in Fig. 8, for Houston, the network reaches peak OA performance for $B = 30$ PCs and a window size of 15×15. For Pavia University and Botswana, peak performance is achieved for $B = 15$ and $B = 20$, respectively, with window sizes of 23×23 and 15×15, respectively. Note that B and W impact the computational complexity of the proposed MS^{2}FE module, with larger B and W rendering the network more complex.

C. Experimental Setup

All experiments are conducted on a system with two NVIDIA Geforce RTX 2080Ti GPUs. Implementation is in Pytorch on Ubuntu 19.04. The network uses the Adam optimizer with cross-entropy loss, and the learning rate is 0.001

[2] https://hyperspectral.ee.uh.edu/?page_id=459

[3] http://www.ehu.eus/ccwintco/index.php?title=Hyperspectral_Remote_Sensing_Scenes



(a) (b) (c)

Fig. 8. OA performance for varying number of PCs B and window size W for (a) Houston, (b) Pavia University, (c) Botswana.

over 100 epochs. Importantly, no data augmentation is used. We note that the MS^{2}FENet source code is available at https://github.com/licuiling-chd/MS2FENet.

To gauge performance, we calculate OA, average accuracy (AA), and κ coefficient. OA is the ratio of the number of correct predictions made by the network to the overall number of pixels in all test sets, while AA is the average value of the ratio of the correct classifications to the total size of each class. The calculation of the κ coefficient is based on the confusion matrix, which is an indication of consistency. To avoid any bias induced by random sampling, we conduct ten trials and report average results along with standard deviation.

IV. Results and Analysis

A. *Comparison to State-of-the-Art Methods*

We now compare the performance of the proposed MS^{2}FENet to several state-of-the-art CNN-based classifiers for HSI from recent literature. Namely, we compare to the techniques of [16] and [15] which both employ a single-branch cascade of conventional 3D convolutions and which we refer to as "3DCNN [16]" and "3DCNN [15]," respectively. We also compare to the SSRN of [17], which is likewise a single-branch cascade of conventional 3D convolutions but with added skip connections to form a residual network; to the HybridSN of [18], which follows a single-branch cascade of 3D-CNN layers with a single 2D-CNN layer; to the MCNN of [19] which is similar to HybridSN but follows the 2D CNN with covariance pooling to extract second-order information from features; and to the SPRN of [42], which deploys parallel 2D residual CNNs on multiple partitions of spectral bands followed by feature fusion via pointwise convolution. Finally, we compare to a competing multiscale method—namely, the M3D-DCNN of [23]—which features multiple parallel convolutional branches for feature extraction coupled with summing of the branch outputs to effectuate feature fusion. We choose M3D-DCNN as representative of coarse-grained multiscale feature-extraction approaches (e.g., [23]–[35]) in contrast to the fine-grained multiscale nature of the MS^{2}FENet proposed here.

1) Quantitative Analysis: Tables III–V present the OA, AA, and κ performance of the classification methods under comparison, while Fig. 9 shows confusion matrices of MS^{2}FENet for the three datasets. It can be observed in Tables III–V that the proposed MS^{2}FENet produces the best OA, AA, and κ compared to the other competing methods. Additionally, it can be seen from the confusion matrices in Fig. 9 that the probability of misclassification for MS^{2}FENet is very small. In fact, for Pavia University, the OA of MS^{2}FENet achieves 100% for the *Asphalt*, *Meadows*, and *Bare Soil* classes.

2) Visual Comparisons: Figs. 10–12 visualize the classification results of the best trained networks. As can be seen, 3DCNN [16] generates a classification map with a great deal of noise. On the other hand, SPRN generates smoother results, albeit with evident misclassifications for some classes. Compared to the other methods, MS^{2}FENet provides the most accurate and smoothest classification maps for all three datasets. Taking the Houston dataset as an example, class *Parking lot 1* includes parking lots in ground and elevated areas, while *Parking lot 2* corresponds to parked vehicles. The distribution of these two classes is complex, and their corresponding spatial regions vary significantly in size. These aspects hinder the performance of many classification networks for these two classes. In contrast, the proposed MS^{2}FENet clearly outperforms the other networks by a wide margin for these classes. Likewise, the *Highway* and *Railway* classes share a similar complex spatial structure, yet these two classes incur fewer misclassifications from MS^{2}FENet than the other techniques.

3) Stability Analysis: Fig. 13 depicts classification performance as the training ratio varies. It can be seen that classification performance for all methods under comparison degrades with decreasing number of training samples, which is as expected. However, compared to the other techniques, MS^{2}FENet always achieves higher classification accuracy and better stability, even with an exceedingly small training ratio (e.g., 1%).

4) Noise Robustness: Performance in noisy environments is often used as a criterion for gauging remote-sensing image classification, and additive white Gaussian noise is commonly used to model distortions occurring in multiple stages within the HSI-acquisition process. Therefore, we consider verifying stability and generalization ability under the condition of additive white Gaussian noise. In the experiment, noise is added to the original HSIs at different signal-to-noise ratios (SNRs). As shown in Fig. 14, MS^{2}FENet has better noise robustness and generalization ability in the noisy case as

TABLE III
CLASSIFICATION ACCURACY (AVERAGE AND STANDARD DEVIATION) FOR HOUSTON FOR 5% TRAINING RATIO

Class		Methods							
No.	Name	3DCNN [16]	M3D-DCNN	3DCNN [15]	SSRN	HybridSN	MCNN	SPRN	MS^{2}FENet
1	Healthy grass	86.24±4.465	96.23±1.838	96.15±1.543	97.89±1.650	97.01±1.949	97.40±1.023	96.31±3.070	**98.30±0.780**
2	Stressed grass	85.47±3.976	96.51±2.293	96.97±2.390	95.68±4.941	97.43±1.625	96.24±1.574	95.46±5.290	**99.61±0.490**
3	Synthetic grass	92.93±5.376	97.62±1.486	98.29±0.971	99.43±1.140	99.68±0.458	99.74±1.802	99.77±0.240	**100.00±0**
4	Trees	85.29±4.490	97.47±1.200	96.50±1.076	96.53±2.712	95.80±2.521	96.02±3.014	94.65±3.011	**97.11±1.813**
5	Soil	90.49±8.344	97.02±1.530	97.52±1.562	99.57±0.650	98.39±1.020	98.81±1.802	98.54±2.511	**100.00±0**
6	Water	84.75±3.672	85.86±4.649	89.51±2.247	88.74±5.640	97.80±1.990	97.15±2.302	96.88±2.341	**98.21±2.088**
7	Residential	78.09±2.236	85.79±2.219	83.36±2.183	91.57±2.020	89.66±3.667	89.72±3.020	**93.82±3.851**	93.54±1.803
8	Commercial	57.55±4.646	84.84±1.705	80.67±3.723	84.43±5.471	96.39±2.579	96.15±2.130	96.97±2.672	**98.34±0.267**
9	Road	51.26±5.048	83.60±3.017	80.99±2.895	86.40±3.211	92.83±2.891	91.87±3.114	93.64±4.291	**96.16±0.700**
10	Highway	39.66±6.334	82.31±3.145	76.24±8.860	90.13±5.532	95.70±2.532	94.95±2.015	95.73±2.586	**99.91±0.300**
11	Railway	48.86±7.220	81.09±3.667	74.52±4.455	88.97±2.971	95.49±1.025	95.02±1.303	94.18±1.996	**98.24±0.600**
12	Parking lot1	40.19±3.878	82.62±4.444	77.55±6.357	89.81±6.821	95.03±7.362	95.19±4.036	95.32±6.435	**99.01±0.004**
13	Parking lot2	32.16±9.035	83.57±5.443	71.13±5.958	87.83±6.091	94.76±3.842	95.13±2.303	93.72±2.851	**97.13±1.513**
14	Tennis Court	91.47±3.239	95.42±2.602	96.06±1.511	99.31±0.953	97.00±2.366	96.95±1.025	96.75±0.236	**100.00±0**
15	Running Track	96.87±1.976	98.21±1.036	98.06±0.830	99.87±0.380	99.11±1.375	99.56±0.998	99.86±0.293	**100.00±0**
	OA	70.87±6.756	89.52±1.770	87.09±1.589	92.78±0.833	95.08±1.232	95.32±1.110	95.72±1.364	**98.31±0.200**
	AA	71.43±2.347	89.96±1.864	88.04±1.568	93.08±0.880	95.72±1.236	95.96±1.116	95.39±1.213	**98.32±0.203**
	κ	68.48±7.265	88.68±1.909	86.01±1.710	92.20±0.900	95.33±1.329	95.68±1.362	95.01±1.322	**98.17±0.220**

TABLE IV
CLASSIFICATION ACCURACY (AVERAGE AND STANDARD DEVIATION) FOR PAVIA UNIVERSITY FOR 5% TRAINING RATIO

Class		Methods							
No.	Name	3DCNN [16]	M3D-DCNN	3DCNN [15]	SSRN	HybridSN	MCNN	SPRN	MS^{2}FENet
1	Asphalt	93.12±0.163	93.23±0.126	95.31±0.801	97.66±0.392	94.02±0.601	98.36±0.969	99.04±0.935	**100.00±0**
2	Meadows	88.10±0.847	92.91±0.357	95.91±0.652	98.96±0.051	**100.00±0**	99.71±0.068	99.79±0.160	**100.00±0**
3	Gravel	68.30±0.691	85.30±0.954	89.20±1.021	96.77±1.852	98.90±0.045	96.63±0.051	95.12±1.681	**99.47±0.008**
4	Trees	93.26±0.230	97.13±0.210	96.12±0.320	98.44±0.863	**99.00±0.032**	97.10±0.039	98.12±1.115	98.90±0.012
5	Painted Metal Sheets	95.12±0.156	98.78±0.102	98.70±0.091	97.63±0.201	99.64±0.007	99.68±0.012	**99.73±0.340**	99.54±0.007
6	Bare Soil	70.66±0.650	91.88±0.210	96.60±0.224	98.85±0.215	99.32±0.009	99.56±0.302	99.25±0.672	**100.00±0**
7	Bitumen	81.06±0.621	88.52±0.336	90.87±0.784	98.63±0.261	**99.00±0.016**	97.80±0.112	98.89±2.430	98.97±0.014
8	Self-Blocking Bricks	81.25±0.341	92.16±0.475	94.35±0.320	97.02±0.931	98.28±0.078	96.20±0.330	98.18±1.076	**99.12±0.011**
9	Shadows	94.30±0.106	97.64±0.080	98.90±0.087	98.75±0.251	96.79±0.302	95.80±0.124	99.34±0.683	**99.89±0.009**
	OA	83.82±0.945	90.47±1.024	93.79±1.120	97.67±0.161	98.31±0.360	98.55±0.331	98.60±0.175	**99.66±0.045**
	AA	85.36±1.012	91.12±0.824	94.01±0.965	97.64±0.269	97.96±0.225	97.76±0.212	98.67±0.352	**99.39±0.103**
	κ	84.89±0.897	92.56±0.970	93.66±0.811	98.41±0.352	99.01±0.410	97.57±0.301	98.92±0.220	**99.71±0.098**

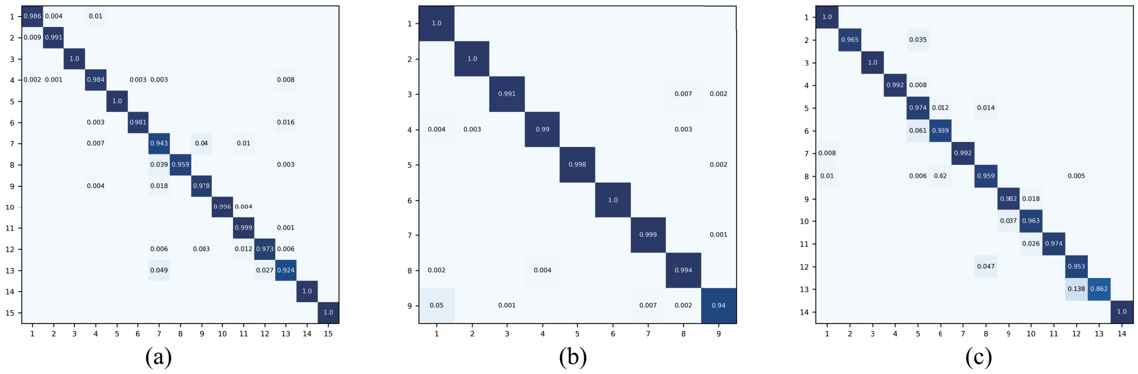


Fig. 9. Confusion matrices for MS^{2}FENet for a training ratio of 5%. (a) Houston, (b) Pavia University, (c) Botswana.

TABLE V
Classification Accuracy (average and standard deviation) for Botswana for 5% Training Ratio

Class		Methods							
No.	Name	3DCNN [16]	M3D-DCNN	3DCNN [15]	SSRN	HybridSN	MCNN	SPRN	MS^{2}FENet
1	Water	95.26±0.084	96.94±0.089	97.56±0.633	98.96±0.121	90.00±1.023	94.80±0.997	97.62±0.980	**100.00±0**
2	Hippo grass	88.40±0.158	89.30±0.021	79.20±2.120	90.10±2.390	94.45±0.902	**97.80±1.021**	95.20±0.762	**97.80±0.120**
3	Floodplain grasses1	79.10±0.987	90.22±0.817	94.12±0.990	97.52±1.830	**100.00±0**	98.80±0.655	**100.00±0**	**100.00±0**
4	Floodplain grasses2	77.78±2.661	90.93±0.941	98.35±0.454	98.58±1.283	89.62±1.650	95.70±0.901	99.69±0.076	**100.00±0**
5	Reeds1	61.56±1.021	82.70±1.654	84.65±1.903	88.67±2.451	94.98±0.840	89.97±0.400	94.00±0.962	**97.16±0.231**
6	Riparian	68.61±0.998	80.98±1.243	82.06±2.014	85.90±1.833	93.64±0.896	**93.91±1.023**	88.06±1.023	92.85±0.905
7	Firescar2	94.65±0.067	95.48±0.523	98.42±0.095	99.07±0.965	99.00±0.004	98.30±0.098	**100.00±0**	99.95±0.002
8	Island interior	88.10±1.099	92.93±0.210	91.60±0.945	92.44±1.304	98.67±0.087	95.82±0.901	**98.97±0.705**	98.85±0.305
9	Acacia woodlands	82.24±4.021	76.71±3.601	90.03±0.912	89.83±1.023	89.62±1.341	96.41±0.909	94.05±0.887	**97.23±0.986**
10	Acacia shrublands	63.93±1.025	79.70±2.014	89.90±1.026	85.68±1.452	90.26±1.262	93.40±1.768	95.95±0.775	**96.87±0.620**
11	Acacia grasslands	61.81±2.386	80.65±1.978	92.40±0.983	93.67±0.987	**96.54±0.640**	96.40±1.402	95.27±1.360	96.36±0.863
12	Short mopane	58.97±3.002	91.00±0.909	93.46±0.504	**97.15±0.390**	96.50±0.872	96.20±1.036	96.53±1.856	96.90±0.703
13	Mixed mopane	69.78±1.578	89.90±1.110	88.81±1.847	93.57±0.858	93.14±1.760	94.10±0.998	92.57±0.763	**94.65±1.023**
14	Exposed soils	80.14±2.641	85.51±1.257	89.70±1.069	89.44±1.382	96.98±0.362	96.30±0.698	90.65±1.740	**100.00±0**
OA		67.16±2.652	83.74±1.875	90.96±1.890	92.20±1.193	93.05±0.874	94.82±0.906	95.19±0.890	**96.42±0.320**
AA		70.26±1.360	84.98±2.044	91.02±1.231	91.93±1.570	93.06±0.962	94.93±1.030	93.99±1.023	**96.39±0.544**
κ		65.30±3.601	83.46±1.399	90.10±1.022	91.27±1.378	93.33±0.680	94.79±0.989	95.28±0.990	**96.28±0.630**

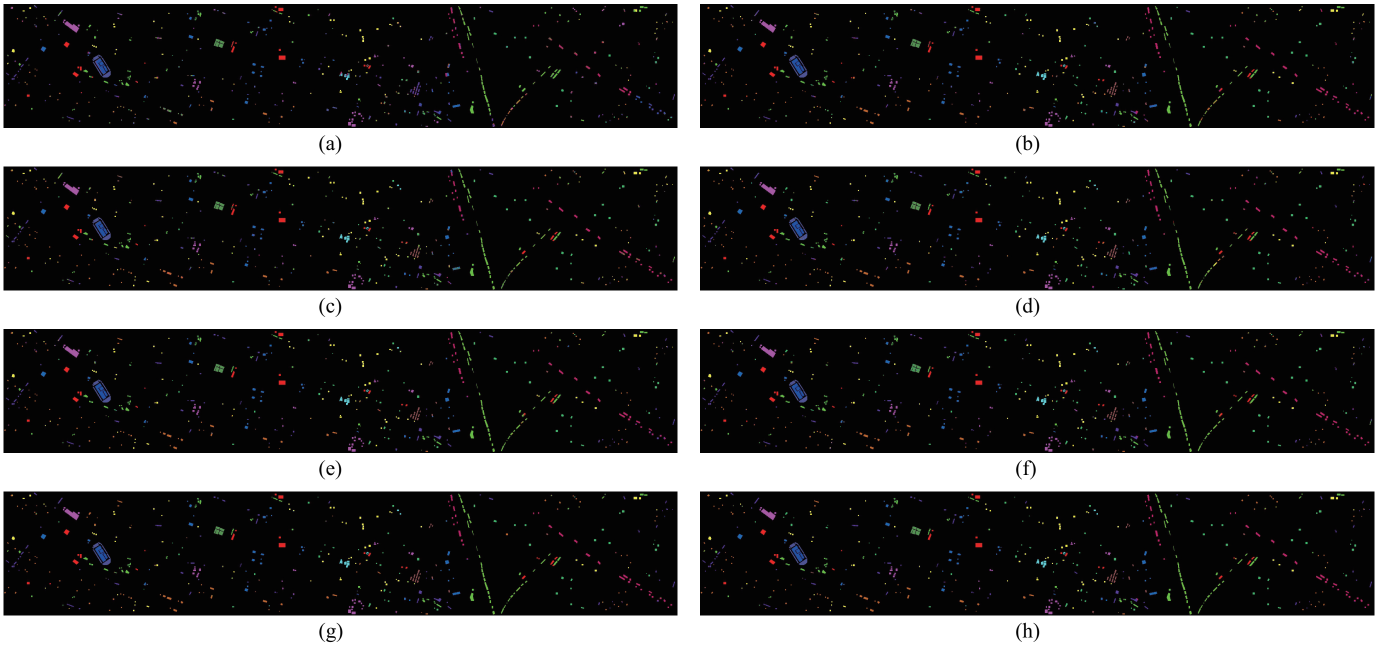

(a) (b) (c) (d) (e) (f) (g) (h)

Fig. 10. Classification maps for Houston. (a) 3DCNN [16], (b) M3D-DCNN, (c) 3DCNN [15], (d) SSRN, (e) HybridSN, (f) MCNN, (g) SPRN, (h) MS^{2}FENet.

compared to the other methods.

5) Training and Test Time: The execution time of training and testing for the various networks under consideration is shown in Table VI. We see that MS^{2}FENet falls roughly in the middle—not as fast as HybridSN, yet not as slow as SSRN.

B. MFC Partitioing

As described in Sec. II-B and illustrated in Fig. 2, the first step in the MFC layer is the partitioning of the input feature map along the channel dimension. While Fig. 2 depicts four partitions (and all experimental results up to now use that number of partitions), a different number of partitions could feasibly be used, and, indeed, the choice of partitioning affects the classification performance of the proposed MS^{2}FENet. Fig. 15 examines MS^{2}FENet performance as the number of MFC partitions varies between one (i.e., no partitioning) and five.

From Fig. 15, we see that, when the number of partitions is increased beyond a single partition, the classification accuracies improve significantly. This suggests that the multi-

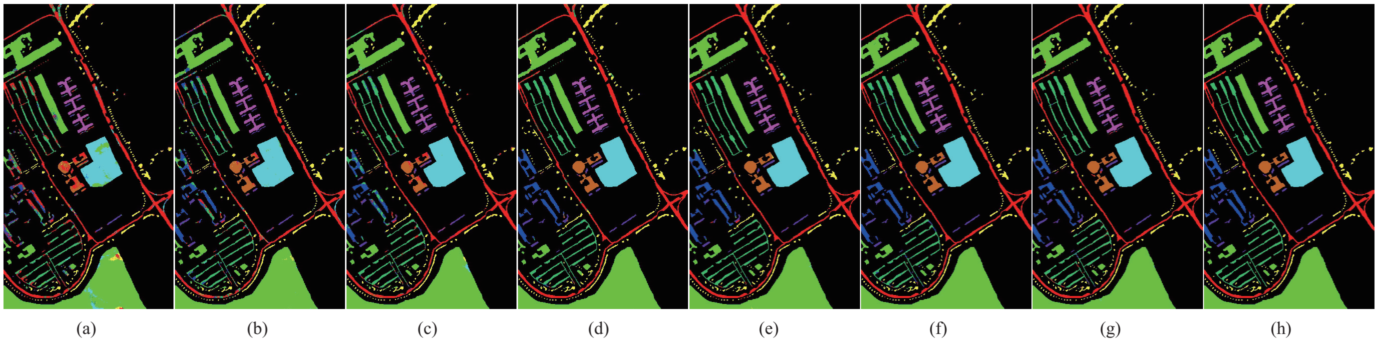


Fig. 11. Classification maps for Pavia University. (a) 3DCNN [16], (b) M3D-DCNN, (c) 3DCNN [15], (d) SSRN, (e) HybridSN, (f) MCNN, (g) SPRN, (h) MS^{2}FENet.

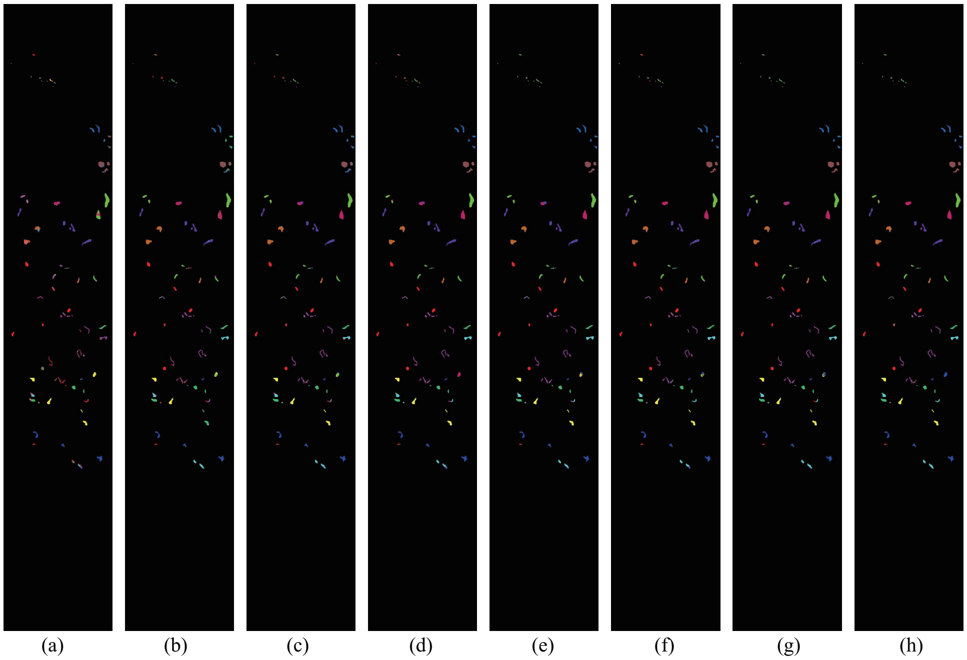


Fig. 12. Classification maps for Botswana. (a) 3DCNN [16], (b) M3D-DCNN, (c) 3DCNN [15], (d) SSRN, (e) HybridSN, (f) MCNN, (g) SPRN, (h) MS^{2}FENet.

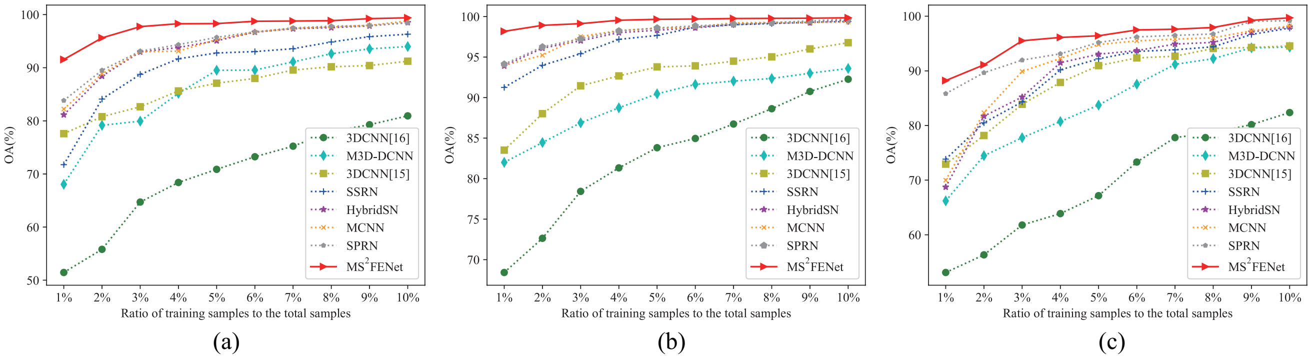
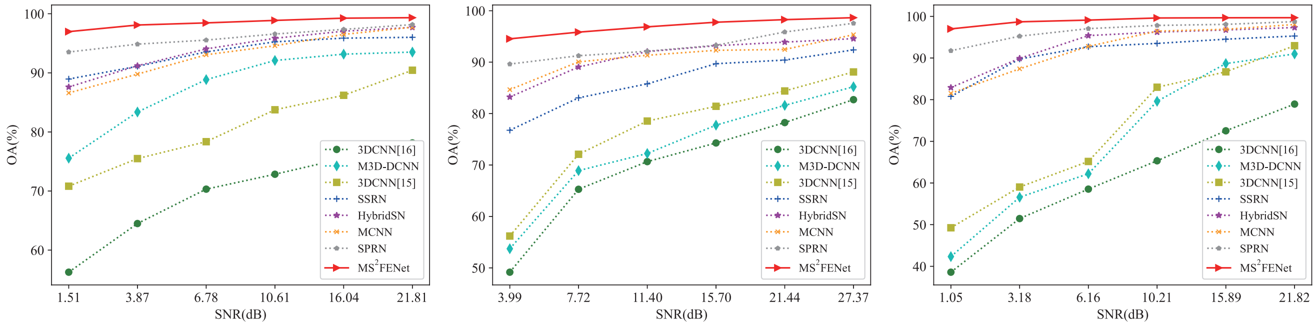


Fig. 13. Classification accuracy for varying training ratio. (a) Houston, (b) Pavia University, (c) Botswana.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/JSTARS.2022.3179446, IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing



(a) (b) (c)

Fig. 14. Classification performance for varying SNRs. (a) Houston, (b) Pavia University, (c) Botswana.

TABLE VI
EXECUTION TIME

Dataset	Method	Train (s)	Test (s)
Houston	3DCNN [16]	16.74	11.40
	M3D-DCNN	33.53	34.47
	3DCNN [15]	31.91	10.48
	SSRN	82.65	93.47
	HybridSN	8.99	0.70
	MCNN	57.39	4.67
	SPRN	39.10	39.72
	MS^{2}FENet	23.53	1.64
Pavia University	3DCNN [16]	49.18	3.36
	M3D-DCNN	91.72	9.21
	3DCNN [15]	92.00	2.77
	SSRN	156.97	21.71
	HybridSN	37.29	3.53
	MCNN	74.64	8.57
	SPRN	95.37	11.14
	MS^{2}FENet	83.19	6.45
Botswana	3DCNN [16]	3.62	6.42
	M3D-DCNN	7.61	19.38
	3DCNN [15]	6.65	5.96
	SSRN	19.84	62.62
	HybridSN	1.19	0.10
	MCNN	10.56	0.79
	SPRN	11.15	31.00
	MS^{2}FENet	3.86	0.20

branch structure and feature reuse within the MFC layers of MS^{2}FENet enlarge the convolutional receptive fields, leading to more effective extraction of multiscale features. Uniformly, peak performance is reached for four partitions. When the number of partitions is less than four, the multiscale spatial-spectral features extracted are likely insufficient and may reduce discrimination ability. On the other hand, greater than four partitions appears to lead to an overly complex network at risk for overfitting. Consequently, for the remainder of the experiments, four partitions are used.

C. Pointwise Convolution for Feature Fusion

Fig. 4 illustrates the feature-fusion block within the MFC layer of MS^{2}FENet. Therein, feature fusion is accomplished via 2D pointwise convolution with $M = CB$ kernels of size 1×1, necessitating reshaping of feature maps from 4D to 3D, and vice versa. Alternatively, a 3D pointwise convolution using

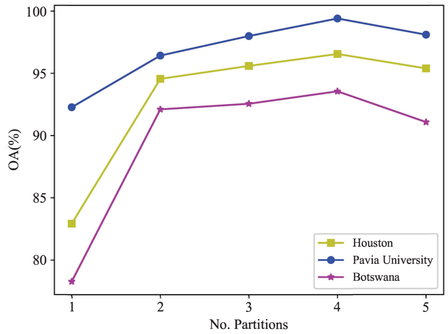


Fig. 15. MS^{2}FENet OA performance for varying numbers of MFC partitions.

C kernels of size $1\times1\times1$ could have been employed, obviating the need for feature-map reshaping. Table VII compares these two feature-fusion alternatives and leads to the conclusion that 2D pointwise convolution uniformly outperforms the 3D alternative.

D. Ablation Experiments

We conduct a battery of ablation experiments to verify various facets of the MS^{2}FENet design. Specifically, we look at the impact of using a multi-branch structure, depthwise 3D convolution, and a feature-fusion block within the MFC layer. As discussed in Sec. II-C, depthwise 3D convolution can reduce the number of network parameters as compared to conventional 3D convolution. The multi-branch structure within MFC can effectively extract multiscale information from feature maps, while the feature-fusion block combines information from across the multi-branch structure. We design three additional networks as shown in Table VIII to gauge the effects of these three main components within the proposed MS^{2}FENet. These three additional networks are:

1) Network A: To analyze the impact of the multi-branch structure on classification performance, input feature maps for MS^{2}FE module are not partitioned. Thus, for Network A, the multi-branch structure is replaced by a single-branch structure in which there is no feature-

TABLE VII
MS^{2}FENet OA Performance for 2D and 3D Pointwise Convolution for MFC Feature Fusion for Training Ratios of 1%, 5%, and 10%

Methods	Houston			Pavia University			Botswana		
	1%	5%	10%	1%	5%	10%	1%	5%	10%
3D Pointwise Convolution	87.91±0.943	97.39±0.432	99.08±0.094	97.14±0.208	99.50±0.024	99.69±0.048	76.34±2.416	94.82±0.755	99.60±0.215
2D Pointwise Convolution	**91.55±0.672**	**98.31±0.200**	**99.40±0.067**	**98.18±0.181**	**99.66±0.045**	**99.85±0.027**	**88.22±0.139**	**96.42±0.320**	**99.71±0.133**

TABLE VIII
Structure of Ablation-Experiment Networks

Methods	Depthwise 3D Conv	Multi-Branch	Feature-Fusion Block
Network A	✓		
Network B	✓	✓	
Network C		✓	✓
MS^{2}FENet	✓	✓	✓

fusion block. Thus, only depthwise 3D convolution is used to extract spatial-spectral features in the MFC layer.

2) Network B: To explore the impact of the feature-fusion block for multiscale feature extraction, we remove this block. Thus, the MFC layer simply concatenates features from different scales.

3) Network C: The effect of depthwise 3D convolution on the MFC layer is investigated by using instead conventional 3D convolution on each partition.

The resulting ablation networks are evaluated in terms of OA performance in Table IX and analyzed below. Additionally, computational complexity, in terms of network parameters as well as FLOPs, is tabulated in Table X for both Network C and MS^{2}FENet.

1) Influence of the Multi-Branch Structure: We see in Table IX that Network B consistently outperforms Network A, regardless of the dataset or training ratio. This suggests that the multi-branch structure within the MFC layer yields significant performance enhancement beyond a simple depthwise 3D convolution.

2) Influence of the Feature-Fusion Block: In comparing MS^{2}FENet to Network B, we see that addition of the feature-fusion block brings typically several percentage points of improvement in classification performance beyond the multi-branch structure. This suggests that the fusion of features across the feature-map channels effectuated by the 2D pointwise convolution, along with the resulting fine-grained multiscale features, is effective.

3) Influence of Depthwise 3D Convolution: MS^{2}FENet uses depthwise 3D convolution instead of conventional 3D convolution as used in Network C. Consulting Tables IX and X, we see that the use of depthwise 3D convolution reduces the number of parameters and FLOPs without reduction of classification performance, effectively saving significant computational costs.

V. Conclusions

In this paper, multiscale spatial-spectral features were extracted from HSIs to improve supervised classification performance. To this end, we proposed MS^{2}FENet, a fine-grained multiscale feature-extraction network at the heart of which lay MFC layers that extracted multiscale spatial-spectral features via feature reuse and feature fusion. In MFC layers, a multi-branch structure expanded the receptive fields of depthwise 3D convolution in a manner more granular than that of other multi-branch architectures common in the literature. This was accomplished through the partitioning of input feature maps, applying hierarchical connections across the partitions, cross-channel feature fusion via pointwise convolution, and depthwise 3D convolution for feature extraction. Ablation experiments proved the ability and effectiveness of MS^{2}FENet to extract multiscale spatial-spectral features from HSIs, while a battery of experimental results comparing to other state-of-the-art networks demonstrated that the proposed MS^{2}FENet achieves outstanding classification performance while being robust to limited training data as well as added noise. In future work, we plan to explore lightweight networks with multiscale feature extraction for more efficient and accurate classification.

References

[1] L. Mou, P. Ghamisi, and X. X. Zhu, "Deep recurrent neural networks for hyperspectral image classification," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 55, no. 7, pp. 3639–3655, July 2017.

[2] M. E. Paoletti, J. M. Haut, J. Plaza, and A. Plaza, "Scalable recurrent neural network for hyperspectral image classification," *The Journal of Supercomputing*, vol. 76, pp. 8866–8882, 2020.

[3] A. Plaza, J. A. Benediktsson, J. W. Boardman, J. Brazile, L. Bruzzone, G. Camps-Valls, J. Chanussot, M. Fauvel, P. Gamba, A. Gualtieri, M. Marconcini, J. C. Tilton, and G. Trianni, "Recent advances in techniques for hyperspectral image processing," *Remote Sensing of Environment*, vol. 113, no. SUPPL. 1, pp. S110–S122, September 2009.

[4] J. A. Benediktsson and P. Ghamisi, *Spectral-Spatial Classification of Hyperspectral Remote Sensing Images*. Artech House, 2015.

[5] J. Jiang, J. Ma, C. Chen, Z. Wang, Z. Cai, and L. Wang, "SuperPCA: A superpixelwise PCA approach for unsupervised feature extraction of hyperspectral imagery," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 56, no. 8, pp. 4581–4593, August 2018.

[6] X. Zhang, X. Jiang, J. Jiang, Y. Zhang, X. Liu, and Z. Cai, "Spectral–spatial and superpixelwise PCA for unsupervised feature extraction of hyperspectral imagery," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 60, 2022.

[7] C. Shi and C.-M. Pun, "Multiscale superpixel-based hyperspectral image classification using recurrent neural networks with stacked autoencoders," *IEEE Transactions on Multimedia*, vol. 22, no. 2, pp. 487–501, February 2020.

[8] Y. Zhang, X. Wang, X. Jiang, and Y. Zhou, "Marginalized graph self-representation for unsupervised hyperspectral band selection," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 60, 2022.

[9] D. Hong, L. Gao, J. Yao, B. Zhang, A. Plaza, and J. Chanussot, "Graph convolutional networks for hyperspectral image classification," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 59, no. 7, pp. 5966–5978, July 2021.

[10] D. Hong, Z. Han, J. Yao, L. Gao, B. Zhang, A. Plaza, and J. Chanussot, "SpectralFormer: Rethinking hyperspectral image classification with transformers," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 60, 2022.

TABLE IX
OA Performance for Ablation Experiments with Training Ratios of 1%, 3%, and 5%

Methods	Houston			Pavia University			Botswana		
	1%	3%	5%	1%	3%	5%	1%	3%	5%
Network A	71.74±2.541	87.35±1.254	90.67±0.871	92.53±1.433	95.26±0.501	99.13±0.224	73.08±2.075	83.76±0.770	88.86±1.284
Network B	85.90±1.589	95.16±0.545	96.84±0.491	96.47±0.231	98.19±0.753	99.53±0.046	80.90±1.994	92.38±1.132	95.74±0.928
Network C	90.58±0.511	97.21±0.383	98.29±0.157	97.82±0.298	98.70±0.717	99.64±0.044	83.03±1.503	93.53±0.324	96.32±0.573
MS^{2}FENet	**91.55±0.672**	**97.74±0.220**	**98.31±0.200**	**98.18±0.181**	**99.13±0.027**	**99.66±0.045**	**88.22±0.139**	**95.51±0.091**	**96.42±0.320**

TABLE X
Parameters and FLOPs for Network C and MS^{2}FENet
($1\text{M} = 1 \times 10^6$)

Methods	Houston		Pavia University		Botswana	
	Parameters	FLOPs	Parameters	FLOPs	Parameters	FLOPs
Network C	0.520M	57.970M	0.210M	41.714M	0.279M	26.185M
MS^{2}FENet	0.370M	22.518M	0.061M	12.243M	0.129M	9.256M

[11] D. Hong, L. Gao, N. Yokoya, J. Yao, J. Chanussot, Q. Du, and B. Zhang, “More diverse means better: Multimodal deep learning meets remote-sensing imagery classification,” *IEEE Transactions on Geoscience and Remote Sensing*, vol. 59, no. 5, pp. 4340–4354, May 2021.

[12] J. Yang, Y.-Q. Zhao, and J. C.-W. Chan, “Learning and transferring deep joint spectral-spatial features for hyperspectral classification,” *IEEE Transactions on Geoscience and Remote Sensing*, vol. 55, no. 8, pp. 4729–4742, August 2017.

[13] P. V. Arun, K. M. Buddhiraju, and A. Porwal, “Capsulenet-based spatial-spectral classifier for hyperspectral images,” *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, vol. 12, no. 6, pp. 1849–1865, June 2019.

[14] 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 Transactions on Geoscience and Remote Sensing*, vol. 54, no. 10, pp. 6232–6251, October 2016.

[15] Y. Li, H. Zhang, and Q. Shen, “Spectral-spatial classification of hyperspectral imagery with 3D convolutional neural network,” *Remote Sensing*, vol. 9, 2017.

[16] A. Ben Hamida, A. Benoit, P. Lambert, and C. Ben Amar, “3-D deep learning approach for remote sensing image classification,” *IEEE Transactions on Geoscience and Remote Sensing*, vol. 56, no. 8, pp. 4420–4434, August 2018.

[17] Z. Zhong, J. Li, Z. Luo, and M. Chapman, “Spectral-spatial residual network for hyperspectral image classification: A 3-D deep learning framework,” *IEEE Transactions on Geoscience and Remote Sensing*, vol. 56, no. 2, pp. 847–858, February 2018.

[18] S. K. Roy, G. Krishna, S. R. Dubey, and B. B. Chaudhuri, “HybridSN: Exploring 3-D-2-D CNN feature hierarchy for hyperspectral image classification,” *IEEE Geoscience and Remote Sensing Letters*, vol. 17, no. 2, pp. 277–281, February 2020.

[19] J. Zheng, Y. Feng, C. Bai, and J. Zhang, “Hyperspectral image classification using mixed convolutions and covariance pooling,” *IEEE Transactions on Geoscience and Remote Sensing*, vol. 59, no. 1, pp. 522–534, January 2021.

[20] S. Ghaderizadeh, D. Abbasi-Moghadam, A. Sharifi, N. Zhao, and A. Tariq, “Hyperspectral image classification using a hybrid 3D-2D convolutional neural networks,” *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, vol. 14, pp. 7570–7588, 2021.

[21] Z. Ge, G. Cao, X. Li, and P. Fu, “Hyperspectral image classification method based on 2D–3D CNN and multibranch feature fusion,” *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, vol. 13, pp. 5776–5788, 2020.

[22] C. Yu, R. Han, M. Song, C. Liu, and C.-I. Chang, “A simplified 2D-3D CNN architecture for hyperspectral image classification based on spatial–spectral fusion,” *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, vol. 13, pp. 2485–2501, 2020.

[23] M. He, B. Li, and H. Chen, “Multi-scale 3D deep convolutional neural network for hyperspectral image classification,” in *Proceedings of the IEEE International Conference on Image Processing*, Beijing, China, September 2017, pp. 3904–3908.

[24] L. Jiao, M. Liang, H. Chen, S. Yang, H. Liu, and X. Cao, “Deep fully convolutional network-based spatial distribution prediction for hyperspectral image classification,” *IEEE Transactions on Geoscience and Remote Sensing*, vol. 55, no. 10, pp. 5585–5599, October 2017.

[25] M. Liang, L. Jiao, S. Yang, F. Liu, B. Hou, and H. Chen, “Deep multiscale spectral-spatial feature fusion for hyperspectral images classification,” *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, vol. 11, no. 8, pp. 2911–2924, August 2018.

[26] P. Duan, X. Kang, S. Li, and P. Ghamisi, “Noise-robust hyperspectral image classification via multi-scale total variation,” *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, vol. 12, no. 6, pp. 1948–1962, June 2019.

[27] S. Wu, J. Zhang, and C. Zhong, “Multiscale spectral-spatial unified networks for hyperspectral image classification,” in *Proceedings of the IEEE International Geoscience and Remote Sensing Symposium*, Yokohama, Japan, July 2019, pp. 2706–2709.

[28] F. Feng, S. Wang, C. Wang, and J. Zhang, “Learning deep hierarchical spatial-spectral features for hyperspectral image classification based on residual 3D-2D CNN,” *Sensors*, vol. 19, 2019.

[29] M. Zhang, W. Li, Q. Du, L. Gao, and B. Zhang, “Feature extraction for classification of hyperspectral and LiDAR data using patch-to-patch CNN,” *IEEE Transactions on Cybernetics*, vol. 50, no. 1, pp. 100–111, January 2020.

[30] X. Li, M. Ding, and A. Pižurica, “Deep feature fusion via two-stream convolutional neural network for hyperspectral image classification,” *IEEE Transactions on Geoscience and Remote Sensing*, vol. 58, no. 4, pp. 2615–2629, April 2020.

[31] Z. Lu, B. Xu, L. Sun, T. Zhan, and S. Tang, “3-D channel and spatial attention-based multiscale spatial-spectral residual network for hyperspectral image classification,” *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, vol. 13, pp. 4311–4324, 2020.

[32] C. Zhong, J. Zhang, and Y. Zhang, “Multiscale feature extraction based on convolutional sparse decomposition for hyperspectral image classification,” *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, vol. 13, pp. 4960–4972, 2020.

[33] X. Zhao, R. Tao, W. Li, H.-C. Li, Q. Du, W. Liao, and W. Philips, “Joint classification of hyperspectral and LiDAR data using hierarchical random walk and deep CNN architecture,” *IEEE Transactions on Geoscience and Remote Sensing*, vol. 58, no. 10, pp. 7355–7370, October 2020.

[34] C. Shi, D. Liao, Y. Xiong, T. Zhang, and L. Wang, “Hyperspectral image classification based on dual-branch spectral multiscale attention network,” *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, vol. 14, pp. 10 450–10 467, 2021.

[35] H. Gong, Q. Li, C. Li, H. Dai, Z. He, W. Wang, H. Li, F. Han, A. Tuniyazi, and T. Mu, “Multiscale information fusion for hyperspectral image classification based on hybrid 2D-3D CNN,” *Remote Sensing*, vol. 13, 2021.

[36] S.-H. Gao, M.-M. Cheng, K. Zhao, X.-Y. Zhang, M.-H. Yang, and P. Torr, “Res2Net: A new multi-scale backbone architecture,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 43, no. 2, pp. 652–662, February 2019.

[37] L. Bai, Q. Liu, C. Li, C. Zhu, Z. Ye, and M. Xi, “A lightweight and multiscale network for remote sensing image scene classification,” *IEEE Geoscience and Remote Sensing Letters*, to appear.

[38] M. D. Farrell and R. M. Mersereau, “On the impact of PCA dimension reduction for hyperspectral detection of difficult targets,” *IEEE Geoscience and Remote Sensing Letters*, vol. 2, no. 2, pp. 192–195, April 2005.

[39] S. Ioffe and C. Szegedy, “Batch normalization: Accelerating deep network training by reducing internal covariate shift,” in *Proceedings of the International Conference on Machine Learning*, Lille, France, July 2015, pp. 448–456.

[40] K. He, X. Zhang, S. Ren, and J. Sun, “Delving deep into rectifiers: Surpassing human-level performance on ImageNet classification,” in *Proceedings of the International Conference on Computer Vision*, Santiago, Chile, December 2015, pp. 1026–1034.

[41] P. Gamba, “A collection of data for urban area characterization,” in *Proceedings of the IEEE International Geoscience and Remote Sensing Symposium*, vol. 1, Anchorage, Alaska, September 2004, pp. 69–72.

[42] X. Zhang, S. Shang, X. Tang, J. Feng, and L. Jiao, “Spectral partitioning residual network with spatial attention mechanism for hyperspectral image classification,” *IEEE Transactions on Geoscience and Remote Sensing*, to appear.

Zhen Ye received the B.S., M.S., and Ph.D. degrees in information and communication engineering from Northwestern Polytechnical University, Xi’an, China, in 2007, 2010, and 2015, respectively. She spent one year as an exchange student of Mississippi State University, Mississippi State, MS, USA.

She is currently an Associate Professor in the School of Electronics and Control Engineering, Chang’an University, Xi’an, China. Her research interests include hyperspectral image analysis, pattern recognition, and machine learning.

Cuiling Li received the B.S. degree from Linyi University, Linyi, China, in 2019. She is currently working toward the M.S. degree in control science and engineering with the School of Electronics and Control Engineering, Chang’an University, Xi’an, China.

Her research interests include machine learning and hyperspectral image classification.

Qingxin Liu received the B.S. degree from Linyi University, Linyi, China, in 2019. He is currently working toward the M.S. degree in transportation engineering with the School of Electronics and Control Engineering, Chang’an University, Xi’an, China.

His research interests include machine learning and remote-sensing scene classification.

Lin Bai (Member, IEEE) received the B.S. degree in electronic information science and technology from Northwest University, Xi’an, China, in 2003, and the M.S. degree in electronic science and technology from Xidian University, Xi’an, China, in 2006. He received his Ph.D. degree in signal and information processing from Northwestern Polytechnical University, Xi’an, China, in 2010.

He is currently an Associate Professor with the School of Electronic and Control Engineering, Chang’an University, Xi’an, China. His research interests include machine learning and remote-sensing image processing.

James E. Fowler (Fellow, IEEE) received the B.S. degree in computer and information science engineering and the M.S. and Ph.D. degrees in electrical engineering from The Ohio State University, Columbus, OH, USA, in 1990, 1992, and 1996, respectively.

In 1997, he held a postdoctoral assignment at the Université de Nice-Sophia Antipolis, France, and, in 2004, he was a Visiting Professor at Télécom ParisTech, Paris, France. He is currently a William L. Giles Distinguished Professor in the Department of Electrical and Computer Engineering at Mississippi State University, Starkville, MS, where he holds a Billie J. Ball Endowed Professorship.

Dr. Fowler was the Editor-in-Chief of *IEEE Signal Processing Letters* from 2017 to 2019. He was previously a Senior Area Editor for *IEEE Transactions on Image Processing* and Associate Editor for *IEEE Transactions on Computational Imaging*, *IEEE Transactions on Image Processing*, *IEEE Transactions on Multimedia*, and *IEEE Signal Processing Letters*. He is currently an Associate Editor for the *EURASIP Journal on Image and Video Processing*. He is currently the Chair of the Computational Imaging Technical Committee of the IEEE Signal Processing Society and is a former chair of the Society’s Image, Video, and Multidimensional Signal Processing Technical Committee. He was a General co-Chair of the 2014 IEEE International Conference on Image Processing, Paris, France, and was the Speech, Image, and Video Processing track chair of the 2021 Asilomar Conference on Signals, Systems, and Computers. He is currently the publicity chair of the Data Compression Conference.