

Revisiting Deep Hyperspectral Feature Extraction Networks via Gradient Centralized Convolution

Swalpa Kumar Roy, *Student Member, IEEE*, Purbayan Kar, Danfeng Hong, *Senior Member, IEEE*, Xin Wu, *Member, IEEE*, Antonio Plaza, *Fellow, IEEE*, and Jocelyn Chanussot *Fellow, IEEE*

Abstract—The hyperspectral images are composed of a variety of textures across the different bands which increase the spectral similarity and makes it difficult to predict the pixel-wise labels without inducing additional complexity at the feature level. To extract robust and discriminative features from the different regions of land-cover, the hyperspectral research community is still seeking such type of convolutions which can efficiently deal with fine-grained texture information during the feature extraction phase, which often overlook this aspect by vanilla convolution. To overcome the above shortcoming, this paper proposes a generalized gradient centralized 3D convolution (G2C-Conv3D) operation, which is a weighted combination between the vanilla and gradient centralized 3D convolutions (GC-Conv3D) to extract both the *intensity level* semantic information and *gradient level* information. Which can be easily plugged into the existing HSI feature extraction networks to boost the performance of accurate prediction for land-cover types. To validate the feasibility of the proposed G2C-Conv3D, we have considered the existing CNN3D, MS3DNet, ContextNet and SSRN feature extraction models and as well as CAE3D, VAE3D, and SAE3D autoencoder (AE) networks, respectively. All these networks are embedded with G2C-Conv3D convolution to implement both generalized gradient centralized feature extraction networks (G2C-FE) and generalized gradient centralized autoencoder networks (G2C-AE) for fine-grained spectral-spatial feature learning. In addition, G2C-Conv2D is also considered with few networks. The extensive experiments are conducted on four most widely used hyperspectral datasets i.e., IP, KSC, UH, and UP, respectively, and compared with nine methods. The results demonstrate that the proposed G2C-Conv3D can effectively enhanced the feature learning ability of the existing networks and both the qualitative and quantitative results show the superiority and effectiveness of the proposed G2C-Conv3D. The source codes will be publicly available at <https://github.com/danfenghong/G2C-Conv3D-HSI>.

Index Terms—Convolutional Neural Networks (CNNs), Gradi-

ent Centralized 3D Convolution (GC-Conv3D), Generalized Gradient Centralized 3D Convolution (G2C-Conv3D), Hyperspectral images (HSI), ResNets.

I. INTRODUCTION

WITH the rapid development of remote sensing satellite sensors, the land-cover classification with various earth observation (EO) datasets, e.g. hyperspectral images (HSI) [1], synthetic aperture radar (SAR) [2], light detection and ranging (LiDAR) [3] are becoming the hot research topic among the remote sensing communities [4]. Hyperspectral images contain rich spectral information which is encoded in the form of spectral bands and can be used to distinguish between the different land-cover classes of interest but especially having slightly more complex spectral behaviours [5]. On the other hand, the distinction between the spectrally similar texture classes can be made easily by capturing more subtle discrepancies from the contiguous representation of spectral bands associated with each pixel. However, due to the high spectral variability causes by the noise artifacts [6], [7] introduced adversarial, the discriminative feature extraction becomes difficult from such data.

In the early days of HSI classification, the hand-crafted descriptors were widely used to extract the feature using the supervised or unsupervised manner and learn them for classification [8]–[12] but failed to explore the spectral-spatial feature effectively. Among the conventional approaches, local binary pattern (LBP) is one of the simplest and computationally efficient texture operators use to extract illumination invariant spectral features and classify them using extreme learning machine (ELM) [13]. Wei *et al.*, proposed dual-channel 1D CNN framework to extract both spectral feature from raw HSI and spatial discriminative feature from LBP transform image for classification [14]. Anwer *et al.* introduced a two-stream deep architecture for feature fusion between the standard RGB image and texture coded LBP mapped image which are extracted through the VGG network for remote sensing scene classification [15]. Morphological operators are the good choice to extract spectral-spatial morphological profiles (MPs) mutually from raw HSI and then classification can be performed using supervised algorithms [16]. In a similar way, to achieve promising classification performance the extracted MPs can be classified using support vector machine [17]. Dalla Mura *et al.* introduced a new variants of MPs i.e., attribute profiles (APs) [18] to retrieve more enhance spectral-spatial features. To reduce the trainable parameters similar to SVM, random forest (RF) and logistic regression [19]

This work was supported by the MIAI@Grenoble Alpes (ANR-19-P3IA-0003) and the AXA Research Fund, by the Spanish Ministerio de Ciencia e Innovación under Project PID2019-110315RB-I00 (APRISA). This work was supported by the China Postdoctoral Science Foundation Funded Project No. 2021M690385, National Natural Science Foundation of China under Grant 62101045 (Corresponding author: Xin Wu).

S.K. Roy and P. Kar are with the Department of Computer Science and Engineering at Jalpaiguri Government Engineering College, 735102, India (e-mail: swalpa@cse.jgec.ac.in; pk2208@cse.jgec.ac.in).

D. Hong is with the Key Laboratory of Digital Earth Science, Aerospace Information Research Institute, Chinese Academy of Sciences, Beijing 100094, China (e-mail: hongdf@aircas.ac.cn).

X. Wu is with the School of Information and Electronics, Beijing Institute of Technology, 100081 Beijing, China, and Beijing Key Laboratory of Fractional Signals and Systems, 100081 Beijing, China (e-mail: 040251522wuxin@163.com).

A. Plaza is with the Hyperspectral Computing Laboratory, Department of Technology of Computers and Communications, Escuela Politécnica, University of Extremadura, 10003 Cáceres, Spain (e-mail: aplaza@unex.es).

J. Chanussot is with the Univ. Grenoble Alpes, INRIA, CNRS, Grenoble INP, LJK, 38000 Grenoble, France (e-mail: jocelyn@hi.is).

classifier also shown great success in the field land-cover classification task [20]. Sun *et al.* [21] proposed a powerful target detection method by well making use of binary trees that enable the significant separation between the target and the background. The proposed method can effectively highlight the target while suppressing the background without being constrained by traditional model assumptions to achieve the superior performance in target detection.

Recently, deep learning (DL) has already made significant progress in HSI classification task [22]. Inspired by the success of DL techniques, Chen *et al.* is the first to explore 2D stacked autoencoder [19] for the HSI classification task. Zhao *et al.* [23] introduce multi-scale sparse convolution with stacked sparse autoencoder (SSAE) to increase the discriminative power of latent codebook and then the latent feature is used for classification. Li *et al.* introduced deep belief networks (DBN) for unsupervised feature extraction from HSI data [24]. Autoencoder (AE) is an unsupervised model and known to be an extractor of spectral features from unlabelled data [25], but unlike AE the CNN is able to extract spatial features as well and combining both of this feature can achieve significant improvements [26]. To extract the spatial semantic features from HSI the 2D convolution was the most widely used operation [27]. Among the successful attempts, Xu *et al.* proposed local binary convolution (LBPCNN) [28], is formed using a set of randomly generated convolutional weights which are sparse, as well as binary. Luan *et al.* introduced Gabor convolution [29] which are computationally easy and used to preserve local invariance characteristic and also boost the resistance to the local spatial changes. Yu *et al.* proposed the central difference convolution which shows excellent feature representation ability for the intrinsic face images [30]. Two other related widely used convolution for image classification task are dilated convolution [31] and deformable convolution [32]. Makantasis *et al.* used a trainable CNN2D framework for extraction of HSI feature and classify each test pixel using a supervised manner [33]. Lee *et al.* proposed an end-to-end convolutional neural network that can effectively handle the redundant spectral bands during feature learning stages [34]. Roy *et al.* proposed morphological CNN to learn shape invariant feature transform for robust HSI classification [35]. Challa *et al.* introduced watershed classifier (TripletWS) as the last classification layer by replacing the widely used softmax classifier which is extended the watershed operator from mathematical morphology [36].

Spatial information alone could not achieve satisfactory performance in such challenging land-cover types. The 3D convolution is proven to be effective for integrating the spectral and spatial features, which played an important role in HSI classification [37], [38]. Hamida *et al.* introduced CNN3D network for joint spectral-spatial feature extraction [39] and the performance can be further enhanced by introducing multi-scale strategy in feature learning process [40]. However, it is difficult to train a CNN3D network with a less number of training samples, to elevate this aspect, Fang *et al.* introduced a lightweight semi-supervised collaborative learning framework using the joint spectral and spatial feature for HSI prediction by reducing the models parameters [41]. Inspired by the

ResNet architecture [42], Paoletti *et al.* introduces pyramid based residual learning for HSI classification problem [43] and Ma *et al.* proposed two branches spectral and spatial network to learn joint residual feature representation for HSI classification [44]. To increase the discriminative feature learning, multimodal CNN models received great interest in terms of HSI and LiDAR feature fusion [45]–[47]. Roy *et al.* tried to explore both 3D and 2D features in a sequential fashion [48] and also studied the effects of learning adaptive kernel in ResNet [49] in the context of HSI classification. Recently, Zhu *et al.* introduced the spectral and spatial attention mechanism to recalibrate the feature learning process of exiting SSRN [50]. Hong *et al.* proposed graph convolution network (GCN) to better represent the relations between the spectral and spatial feature for HSI classification [51]. In addition to the above models generative adversarial networks have made good progress in HSI classification task [52], [53].

To increase the classification performance of the existing models but without introducing additional complexity for example add few more layers to the network would be an interesting research area. But searching for an optimized network that can enhance the classification performance requires years of research. Therefore, designing an efficient model is a time consuming process. In this paper, we introduced an efficient convolution operation that can extract the *gradient level* details called gradient centralized 3D convolution, which is basically overlooked by the vanilla convolution. In addition, finally a weighted combination between vanilla and gradient centralized 3D convolution is proposed to form *generalized gradient centralized 3D convolution* by extracting both intensity level semantic details and gradient level information from raw hyperspectral data which also ensures fine-grained feature extraction. The generalized gradient centralized 3D convolution (G2C-Conv3D) operation can be achieved through two step process i.e., sampling and aggregation, respectively. This can be easily embedded with any existing CNNs to reduce the models' floating-point operations (FLOPs) and also ensure even better classification accuracy. The main contributions of this work can be summarized as follows.

- The gradient centralized 3D convolution (GC-Conv3D) helps to extract the *gradient level* invariant information during training from raw HSI data which are often overlooked by vanilla convolution.
- We introduce generalized gradient centralized 3D convolution (G2C-Conv3D) convolution to tackle the long-standing and most challenging problems of vanilla convolution for spectral-spatial feature learning by weighted combination of both vanilla and GC-Conv3D convolution operation. To best of our knowledge, this is the first reported G2C-Conv3D convolution for HSI classification in general.
- The G2C-Conv3D reduces FLOPs, easy to implement and can be plugged into existing networks to boot the HSI classification performance.
- The proposed G2C-Conv3D can easily be adapted for other classification tasks for volumetric data while maintaining the same hyperparameter settings. The main

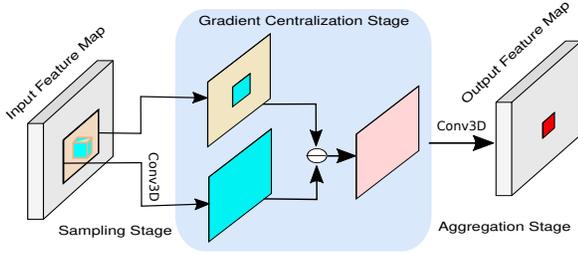


Fig. 1. Generalized representation of the proposed gradient centralized 3D convolution.

advantage of the G2C-Conv3D is its robustness, and exhibited remarkably stable performance when applied to existing networks.

The rest of the paper is organized as follows. The motivation behind the proposed method is described in Section II. Section III introduces the proposed G2C-Conv3D convolution operation. The experimental results with detailed discussion are derived in Section IV. Finally, conclusions are given in Section V.

II. MOTIVATION

Convolution operations are the backbone architecture of many deep neural networks (DNNs). In computer vision, many networks have commonly used convolutions operation are fundamentally explored in the image domain for extracting the basic visual intrinsic patterns which are more robust and discriminative for backend image classification tasks. Moreover, vanilla convolution always extracts intensity level semantic features which are not sufficient for discriminative classification, and to boost the performance the gradient level information are equally important for accurate prediction of HSI pixels which is often overlooked by vanilla convolution.

Vanilla 2D Convolution: The 2D convolution operation $*$ in l^{th} layer can be defined in image domain as follows:

$$\mathcal{O}_{m,l}^{(i,j)} = \sum_{d=1}^D \sum_{u,v=-k}^k W_{m,d}^{(u,v)} * \mathcal{X}_{(l-1),d}^{(i-u,j-v)} \quad (1)$$

where (i, j) denotes the current position on both the D -dimensional l^{th} layer input \mathcal{X}_D^{l-1} and output m^{th} feature maps $\mathcal{O}_D^{m,l}$ and the sampling points within the local receptive field region of size $(k \times k)$ can be denoted by R . Eqn. (1) mainly combines two steps: *sampling* and *aggregation*, respectively. The sampling of local receptive field in a region R over the input feature map \mathcal{X}_D with dilation = 1 and kernel of size 3, the enumerates local receptive points will be $R = \{(-1,-1), (-1,0), (-1,1), (0,-1), (0,0), (0,1), (1,-1), (1,0), \text{ and } (1,1)\}$. The sampled values are aggregated via D -dimensional m^{th} learnable weights W_D^m . It can be observed from Eqn. (1) that the key characteristic of the convolution is its translation invariance property due to the same learnable weights W_D^m is applied throughout the whole input feature maps \mathcal{X}_D .

III. PROPOSED CLASSIFICATION FRAMEWORK

This article aims to solve the most substantial problems of the former vanilla convolution in the HSI classification

task. Inspired by the characteristics of the local binary pattern (LBP) [54], we first introduced the gradient centralized operation over the vanilla 2D convolution and then proposed to extend the gradient centralized convolution into 3D to process the volumetric hyperspectral data. Then to extract the fine-grained information from the limited and imbalanced HSI samples a weighted combination between both sources of information i.e. *intensity level* semantic features and *gradient level* of details are considered where vanilla convolution extracts the *intensity level* semantic features and gradient centralized convolution captures the *gradient level* of details, respectively. Finally, the compact form of gradient centralized convolution can be plugged into several exiting networks to extract more robust and fine-grained texture features encoded in different spectral bands for efficient classification of HSI data which are generally overlooked by the existing standard vanilla CNNs.

A. Gradient Centralized Convolution:

To enhance both the representation and generalization capability like-wise LBP [28], [54], introducing gradient information to the convolution operation, is able to extract detailed *gradient level* information called gradient centralized convolution (GC-Conv2D). On the other hand like 2D vanilla convolution, the gradient centralized convolution operation also combines two common steps, i.e., *sampling* and *aggregation*, respectively. The sampling step is the same as in vanilla convolution (Eqn. (1)) while in gradient centralized convolution, the aggregation is done slightly different way where the center oriented gradient information of the sampled values are summarized and the Eqn. (1) becomes

$$\begin{aligned} \mathcal{O}_{m,l}^{(i,j)} &= \sum_{d=1}^D \sum_{(u,v)=-k}^k W_{m,d}^{(u,v)} * [\mathcal{X}_{(l-1),d}^{(i-u,j-v)} - \mathcal{X}_{(l-1),d}^{(0,0)}] \\ &= \sum_{d=1}^D \sum_{(u,v)=-k}^k W_{m,d}^{(u,v)} * \nabla \mathcal{X} \end{aligned} \quad (2)$$

where $(\mathcal{X}_{(l-1),d}^{(i-u,j-v)} - \mathcal{X}_{(l-1),d}^{(0,0)})$ captures the rate of changes of intensity w.r.t the center pixel, $\mathcal{X}_{(l-1),d}^{(0,0)}$ for a window of size $(k \times k)$ which can be referred as local gradients, $\nabla \mathcal{X}$ and it can also be noted that the gradient value with respect to the center position $(0, 0)$ itself always equals to zero.

B. Gradient Centralized 3D Convolution

Similar to the gradient centralized 2D convolution, to calculate the l^{th} layer m^{th} output feature maps $\mathcal{O}_{m,l}^{(i,j,r)}$ at position (i, j, r) gradient centralized 3D convolution introduces an extra convolutional parameter z to the m^{th} learnable weights matrix $W_{m,d}^{(u,v,z)}$ and the value of the weight parameter at position (u, v, z) and d is the depth of the kernel to perform the channel-wise convolution operation in z direction for the

intermediate $(l-1)^{th}$ layer input feature maps $\mathcal{X}_{(l-1),d}^{(i-u,i-v,r-z)}$ shown in Eqn. (3).

$$\begin{aligned} O_{m,l}^{(i,j,r)} &= \sum_d^{[1,D]} \sum_{(u,v,z)}^{[-k,k]} W_{m,d}^{(u,v,z)} \otimes [\mathcal{X}_{(l-1),d}^{(i-u,i-v,r-z)} - \mathcal{X}_{(l-1),d}^{(0,0,0)}] \\ &= \sum_d^{[1,D]} \sum_{(u,v,z)}^{[-k,k]} W_{m,d}^{(u,v,z)} \otimes \nabla \mathcal{X} \end{aligned} \quad (3)$$

where \otimes denotes gradient centralized 3D convolution (GC-Conv3D) operation. It can be noted that to generate a volumetric feature map as output the depth of the weight matrix (or kernel) is always lesser than the depth of the input feature maps which ensures the convolution operation in-depth direction. Please note the range of variables is shown at the top of the summation for compact representation of Eqn. (3).

C. Generalized Gradient Centralized 3D Convolution

In case of HSI classification task, richer spectral-spatial information are encoded in large number of narrow and contiguous spectral bands. So, it is important to distinguish between the two visually similar but different land cover class descriptions by increasing the inter-class dissimilarity and as well as increases the intra-class similarity. To achieve this both the sources of information i.e., *intensity level* semantic features as well as the *gradient level* of details are crucial and hence a better and feasible model can be formed by utilizing the weighted (θ) combination between the vanilla convolution and the gradient centralized operation to extract the fine-grained details. Finally, the compact representation of gradient centralized 3D convolution operation shown in Eqn. (4), where the parameter $\theta \in [0, 1]$ highlights the percentages of contribution between the *intensity level* and *gradient level* details extracted using the convolution operation shown in Eqn. (4). The generalized representation of the proposed gradient centralized 3D convolution is shown in Fig. 1. The higher preference to gradient centralized convolution is given by setting the larger value to θ in Eqn. (4). To provide highest preference the θ value 0.70 is set to gradient centralized 3D convolution and similarly 0.30 set for vanilla convolution, experimentally. If θ value is selected as 1 then Eqn. (4) becomes conventional gradient centralized 3D convolution. The weighted combination of both sources of information ensure the fine-grained feature extraction for HSI classification task and which also provides more robust and generalization capacity. In the rest of paper, we will use this generalized gradient centralized 3D convolution (G2C-Conv3D) to show the performance improvement for the existing models.

D. Generalized Gradient Centralized Spectral residual block

To extract the informative spectral features from the raw HSI input cubes, $\mathbf{X} \in \mathcal{R}^{S \times S \times B}$, in general the spectral residual blocks is be used but it can be noted that the vanilla 3D convolution fails to distinguish different types of texture composed in multiple spectral bands and overlook the fine-grained information. To overcome this aspect and to

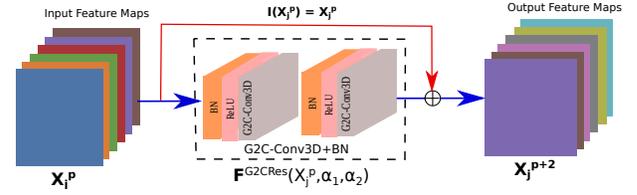


Fig. 2. Formation of Residual block with generalized gradient centralized 3D convolution.

extract more robust and discriminative fined-grained spectral feature representation using the layered specific learnable kernels during the back-propagation of the network. We have replaced all the vanilla 3D convolutions with generalized gradient centralized 3D convolution (G2C-Conv3D) and named it generalized gradient centralized spectral residual block (G2CSResNet). In G2CSResNet block, the j^{th} input feature cubes, $\mathbf{X}_j^p \in \mathcal{R}^{S \times S \times B}$ from p^{th} layer is sequentially passed through two consecutive convolution operations using the filter banks $H_j = \{H_j^{p+i} | 1 \leq i \leq 2\}$ having kernel of 3D shape $k_1^p \times k_2^p \times k_3^p$ in the $(p+1)^{th}$ and $(p+2)^{th}$ layers, which are parameterized with ω_1 and ω_2 and mathematically represented by the feed-forward residual function $F^{G2CRes}(X_j^p; \alpha_1, \alpha_2)$. Then, the j^{th} output feature maps X_j^{p+1} and X_j^{p+2} after the 3D convolutions operation in the successive $(p+1)^{th}$ and $(p+2)^{th}$ layers are directly added with an identity mapping $\mathcal{I}(X_j^p) = X_j^p$ as a skip connection, and can be defined as

$$X_j^{p+2} = \Phi(\mathcal{I}(X_j^p) \oplus F^{G2CRes}(X_j^p; \alpha_1, \alpha_2)) \quad (5a)$$

$$F^{G2CRes}(X_j^p; \alpha_1, \alpha_2) = \text{BN}((X_j^{p+1}) \otimes H_j^{p+2} \oplus b_j^{p+2}) \quad (5b)$$

$$X_j^{p+1} = \text{BN}(\Phi(X_j^p \otimes H_j^{p+1} \oplus b_j^{p+1})) \quad (5c)$$

where \otimes denotes the proposed G2C-Conv3D operation, Φ is ReLU activation function [55], $\text{BN}(X_j^q) = \frac{X_j^q - E[X_j^q]}{\sqrt{\text{Var}[X_j^q]}}$ is the batch normalization [56], $E(\cdot)$ and $\text{Var}(\cdot)$ represent mean and variance of the input tensor, $\alpha_1 = \{H_j^{p+1}, H_j^{p+2}\}$, and $\alpha_2 = \{b_j^{p+1}, b_j^{p+2}\}$, respectively. The spatial shapes of the output 3D feature maps are kept unchanged by a padding strategy where the values of the border area of the output is copied to the padding areas of the next feature maps. $H_j = \{H_j^{p+i} | 1 \leq i \leq 2\}$ and $b_j = \{b_j^{p+i} | 1 \leq i \leq 2\}$ denote the j^{th} weight matrix and bias vector associated with the two consecutive $(p+1)^{th}$ and $(p+2)^{th}$ G2C-Conv3D+BN¹ layers, respectively.

E. Generalized Gradient Centralized Spatial Residual Block

Similar to the G2CSResNet block, the robust and invariant fine-grained spatial feature representation can be extracted through the proposed generalized gradient centralized spatial residual block (G2CSpResNet) for further improvement when the vanilla 3D convolution are replaced with the proposed G2C-Conv3D operation. This can be achieved by the learnable filter banks namely H_j^{q+1} and H_j^{q+2} within the two consecutive $(q+1)^{th}$ and $(q+2)^{th}$ G2C-Conv3D layers and the shape of trainable kernels are $k_1^q \times k_2^q \times k_3^q$ where k_3^q is same as the depth of the 3D input feature

¹We can combine 3D convolution and BN layers as G2C-Conv3D+BN for naming simplicity.

$$\mathcal{O}_{m,l}^{(i,j,r)} = \theta \cdot \underbrace{\sum_d^{[1,D]} \sum_{(u,v,z)}^{[-k,k]} W_{m,d}^{(u,v,z)} \otimes [\mathcal{X}_{(l-1),d}^{(i-u,i-v,r-z)} - \mathcal{X}_{(l-1),d}^{(0,0,0)}]}_{\text{Gradient Centralized 3D Convolution}} + (1-\theta) \cdot \underbrace{\sum_d^{[1,D]} \sum_{(u,v,z)}^{[-k,k]} W_{m,d}^{(u,v,z)} \otimes [\mathcal{X}_{(l-1),d}^{(i-u,i-v,r-z)}]}_{\text{Vanilla 3D Convolution}} \quad (4)$$

maps. The spatial size of output 3D feature maps X^{q+1} and X^{q+2} in the $(q+1)^{th}$ and $(q+2)^{th}$ layers are kept unchanged to by a small neighboring padding strategy of size 3×3 . The j^{th} output feature maps X_j^{q+1} and X_j^{q+2} after the G2C-Conv3D convolutions operation in the successive $(q+1)^{th}$ and $(q+2)^{th}$ layers are then directly added with a skip connection to learn an identity mapping $\mathcal{I}(X_j^q) = X_j^q$ as shown in Fig. 2, which are parameterized with θ_1 and θ_2 and can be represented by a feed-forward gradient centralized residual function $F^{G2CRes}(X_j^q; \beta_1, \beta_2)$ and mathematically becomes

$$X_j^{q+2} = \Phi(\mathcal{I}(X_j^q) \oplus F^{G2CRes}(X_j^q; \beta_1, \beta_2)) \quad (6a)$$

$$F^{G2CRes}(X_j^q; \beta_1, \beta_2) = \text{BN}((X_j^{q+1}) \otimes H_j^{l+2} \oplus b_j^{q+2}) \quad (6b)$$

$$X_j^{q+1} = \text{BN}(\Phi(X_j^q \otimes H_j^{q+1} \oplus b_j^{q+1})) \quad (6c)$$

where $\beta_1 = \{H_j^{q+1}, H_j^{q+2}\}$, $\beta_2 = \{b_j^{q+1}, b_j^{q+2}\}$, and $H_j = \{H^{q+i} | 1 \leq i \leq 2\}$ and $b_j = \{b_j^{q+i} | 1 \leq i \leq 2\}$ denote the weight matrix and bias vector associated with j^{th} input of the two consecutive G2C-Conv3D+BN layers $(q+1)^{th}$ and $(q+2)^{th}$, respectively².

F. Revisiting Deep Classification Models

Now a days there have been increasing efforts to use model-based deep learning framework as mainstream research for HSI classification task rather than improving the existing. This is the first attempt where we introduced the G2C-Conv3D to ensure the utilization of fine-grained textural details for imbalanced HSI classification which are basically overlooked by the vanilla CNN3D models. The performance of the existing DNN models can be improved significantly by replacing all the vanilla 3D convolution layers with G2C-Conv3D. To show the effectiveness of the proposed G2C-Conv3D convolution, we have considered the following models i.e., SSRN [37], Conv3D [39], [57], MS3DNet [40], and ContextNet [34] and embeded them with generalized gradient centralized 3D convolution to modelled into G2C-SSRN, G2C-Conv3DNet, G2C-MS3DNet, and G2C-ContextNet, respectively.

Generalized Gradient Centralized Spectral-Spatial Residual Network: Like spectral features, the spatial neighboring pixels provide some meaningful structural relationship, for example, buildings have geometric shapes whereas the fractal-like appearance can be seen for forest regions, by considering this information, performance of the models improve significantly. The most common way for classifying hyperspectral data is to consider the joint spectral-spatial feature using some specific kernels and extracted them during the training phase. The simplest and widely used for joint spectral-spatial features, the SSRN [58] is one of the successful network

TABLE I
LAYER-WISE DETAILS OF THE PROPOSED G2C-SSRN ARCHITECTURE (G2C-CONV3D: GENERALIZED GRADIENT CENTRALIZED 3D CONVOLUTION, BN: BATCH NORM, RELU: ACTIVATION FUNCTION, AVGPPOOL-3D: AVERAGE POOL AND B: BANDS.)

Layers	Output Shape	Kernel and Stride
Input Layer	Input Shape=(9, 9, 200)	
[G2C-Conv3D BN ReLU]	$\begin{bmatrix} 24 \times 9 \times 9 \times 97 \\ 24 \times 9 \times 9 \times 97 \\ 24 \times 9 \times 9 \times 97 \end{bmatrix}$	Kernel = (1, 1, 7) Stride = (1, 1, 2)
G2CResNet Block	$\begin{bmatrix} 24 \times 9 \times 9 \times 97 \\ 24 \times 9 \times 9 \times 97 \\ 24 \times 9 \times 9 \times 97 \end{bmatrix} \times 2$	Kernel = (1, 1, 7) Padding = (0, 0, 3) Stride = (1, 1, 1)
[G2C-Conv3D BN ReLU]	$\begin{bmatrix} 128 \times 9 \times 9 \times 1 \\ 128 \times 9 \times 9 \times 1 \\ 128 \times 9 \times 9 \times 1 \end{bmatrix}$	Kernel = (1, 1, $\lceil \frac{B-6}{2} \rceil$) Stride = (1, 1, 1)
[G2C-Conv3D BN ReLU]	$\begin{bmatrix} 24 \times 7 \times 7 \times 1 \\ 24 \times 7 \times 7 \times 1 \\ 24 \times 7 \times 7 \times 1 \end{bmatrix}$	Kernel = (3, 3, 128) Stride = (1, 1, 1)
G2CSpResNet Block	$\begin{bmatrix} 24 \times 7 \times 7 \times 1 \\ 24 \times 7 \times 7 \times 1 \\ 24 \times 7 \times 7 \times 1 \end{bmatrix} \times 2$	Kernel = (3, 3, 1) Padding = (1, 1, 0) Stride = (1, 1, 1)
AvgPool-3D	$24 \times 1 \times 1$	Kernel = (5, 5, 1)
Dense	dropout = 0.5	
FC	class = 16	

architecture for HSI classification, which includes a spectral feature learning and a spatial feature learning phase, an AvgPool-3D layer, and a fully connected (FC) layer. A shortcut connection between every other residual block for preserving the information loss and besides, this provides better optimization in end-to-end joint representation learning. The SSRN achieves good overall performance but the performance is not satisfactory when a limited number of samples are considered for training, due to lacks of the fine-grained detailed information which is basically overlooked by the vanilla 3D convolution operation used in SSRN. To overcome this shortcomings, we have introduced the generalized gradient centralized spectral-spatial residual network (G2C-SSRN) for fine-grained HSI classification shown in Fig. 3 where the underlying architecture keep remain same but all the Conv3D+BN are replaced with G2C-Conv3D+BN to ensure the fine-grained feature extraction using both the ResBlocs i.e., G2CResNet and G2CSpResNet, respectively (Please see SubSec. III-D and III-E for more details). The spectral feature is extracted using G2CResNet block to focus on detecting the correlations among the different spectral bands. Similarly, the spatial feature extraction phase is applied to utilize the spatial neighboring information around the centred pixel [59]. To achieve this G2CSpResNet block aims to force the model to focus on detecting and highlighting the correlations in the spatial domain. Due to the fine-grained behavior of the G2C-Conv3D operation, the G2C-SSRN network improves the model generalization power and gradually boots the overall classification performance of the HSI task. Fig. 3 shows the complete overview of G2C-SSRN network architecture.

²Remaining notations are same with Subsection A.

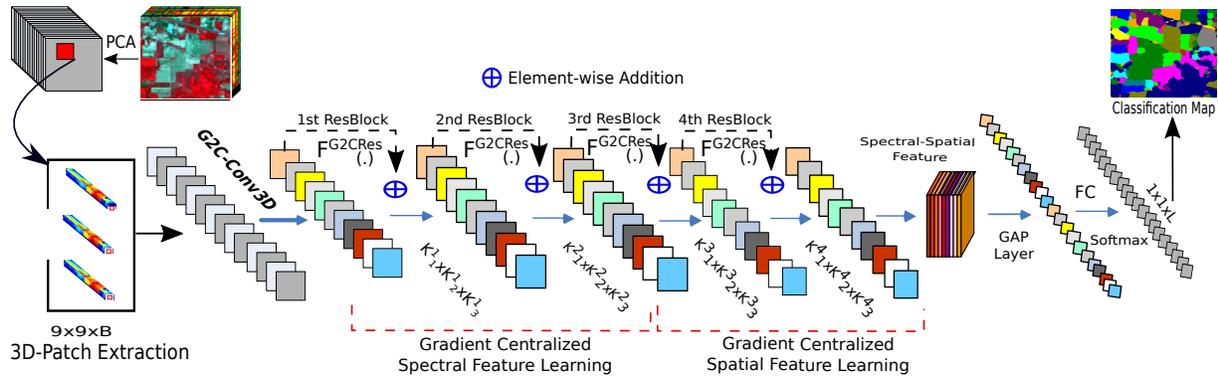


Fig. 3. Graphical representation of the generalized gradient centralized spectral spatial residual network (G2C-SSRN) for HSI classification.

To explain the architecture of G2C-SSRN, we consider the 3D input patch of size $7 \times 7 \times 200$ to the network extracted from raw IP dataset. To extract the low-level spectral features in the beginning G2C-Conv3D+BN+ReLU, 24 kernels of shape $(1 \times 1 \times 7)$ with a stride of $(1, 1, 2)$ is applied to generate output feature of shape $(24 \times 9 \times 9 \times 97)$. Then the spectral feature is extracted with two consecutive G2CResNet blocks using 24 kernels of same shape as previous layer but with an stride of $(1, 1, 1)$ to produce output feature maps $(24 \times 9 \times 9 \times 97)$. In the middle again 128 and 24 kernels of shape $(1 \times 1 \times \lceil \frac{B-C}{2} \rceil)$ and $(3 \times 3 \times 128)$, respectively are used in two consecutive G2C-Conv3D+BN+ReLU layers. Similarly, spatial feature is extracted with two consecutive G2CSpResNet blocks using 24 kernels of shape $(3 \times 3 \times 1)$ with an stride of $(1, 1, 1)$ to have an output feature maps of size $(24 \times 7 \times 7 \times 1)$ and finally, an AvgPool-3D is employed using the kernel of size $(5 \times 5 \times 1)$ followed by a FC layer for calculating the class probabilities. The layer-wise details of the G2C-SSRN network architecture shown in Table I. **Generalized Gradient**

This approach also utilizes maximum use of the information presented in the volumetric data and without introducing additional complexity. Unlike the CNN3D approaches, here we introduce a novel and efficient generalized gradient centralized 3D convolution network (G2C-Conv3DNet) architecture for HSI classification that, replaces all the vanilla 3D convolutions with G2C-Conv3D and helps the model to processes the fine-grained details from both the spatial and as well as spectral dimension simultaneously. The new G2C-Conv3DNet network learns better representations from the limited and imbalanced available samples using the fewer trainable parameters. In G2C-Conv3DNet, the 3D input data cube $9 \times 9 \times B$ is passed through two consecutive G2C-Conv3D+BN+ReLU blocks having 32 and 16 gradient centralized kernels of shape $(5 \times 5 \times 24)$ and $(5 \times 5 \times 16)$, respectively to utilize spatial neighboring information around a center pixel. Finally, the max-pool3D is performed with the kernel of shape $(2, 2, 1)$ before the feature flattening, which is also followed by additional two FC layers for predicting the classification scores. An overview of the G2C-Conv3DNet architecture for HSI classification is shown in Fig. 4.

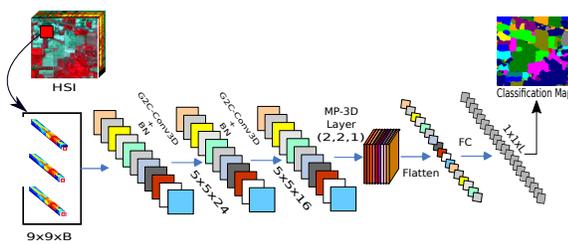


Fig. 4. Graphical representation of the generalized gradient centralized 3D convolution network (G2C-Conv3DNet) for HSI classification.

Centralized 3D Convolutional Network: This is a general CNN model for HSI classification task [39], which uses two consecutive vanilla 3D convolutions followed by batch normalization (Conv3D+BN), a max pooling layer and finally, two fully connected (FC) layers at the end. Due to its ability to learn low-level spectral-spatial feature representation, CNN3D is widely used in the remote sensing community [57]. The advantage of such CNN3D framework is that it can take care of both the spatio-spectral information and merged in a non separable way at the early stages of the training process.

Generalized Gradient Centralized Multi-Scale Convolutional Network: Due to the high-resolution nature of hyper-spectral scene [40], textures of different land-cover classes are varying based on the shapes and scales presence. This approach takes advantage of both the spatial and spectral feature information presences in the raw volumetric data but missed to focus on the fine-grained detailed information for HSI classification which is overlooked by the vanilla 3D convolution operation. But also sacrifices the information loss up to some extent due to not having residual skip connections between the intermediate layers. Unlike MS3DNet, the G2C-MS3DNet can capable to re-calibrate both the fine-grained multi-scale spectral and as well as spatial features with the help of G2C-Conv3D convolution operation employed in a multi-scale fashion and learns even better representation from the limited and imbalanced available samples during training. In G2C-MS3DNet, the spatial G2C-Conv3D is performed using 16 kernels of size $(2 \times 2 \times 11)$ with stride of shape $(1, 1, 3)$ to focus mainly on spatial features. Then, two consecutive G2C-Conv3D layers use 16 kernels of sizes $(1 \times 1 \times 1)$, $(1 \times 1 \times 3)$, $(1 \times 1 \times 5)$ and $(1 \times 1 \times 11)$, respectively

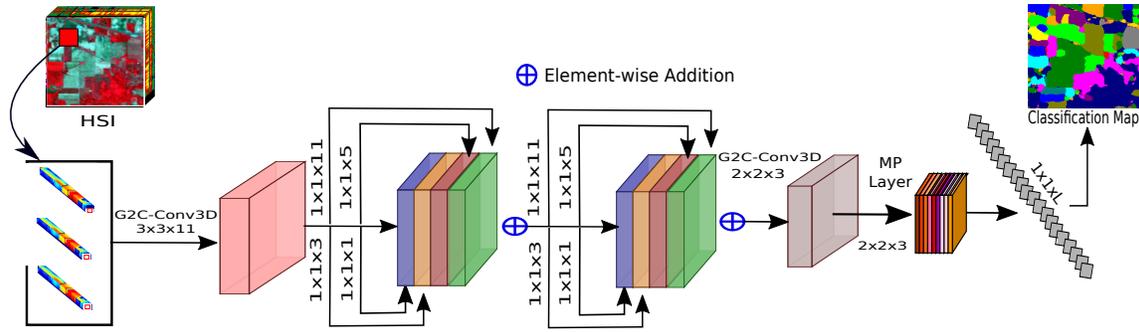


Fig. 5. Graphical representation of the generalized gradient centralized multi-scale 3D convolution network (G2C-MS3DNet) for HSI classification.

with stride of (1,1,1) to extract the multi-scale information and combine all the individual feature maps by an element-wise addition, to have an output feature maps of size 16 immediately after the ReLU non-linearity's, which is followed by a G2C-Conv3D operation using kernel of size (2×2×3) and a stride of (1,1,1). The features are flattened immediately after the max-pool3D operation and a FC layer is employed for final classification. The graphical representation of G2C-MS3DNet model for HSI classification is shown in Fig. 5.

Generalized Gradient Centralized Contextual Network:

The overall performance of HSI classification can be enhanced by introducing the multi-scale strategy into the convolutional neural networks and further emphasized by the residual learning paradigm [42] to control the information flow during back-propagation of the network [34]. This feature makes the model to learn more robust representation from raw HSI data cubes and spatial neighboring relationship as compared to the conventional CNNs but due to the lacks of fine-grained details the performance of the model gradually degrades for complex HSI scene e.g., University of Houston (UH). To overcome the aforementioned flaws, this paper introduced G2C-Conv3D convolution into the above model to form G2C-ContextNet which ensures the extraction of fine-grained details from the homogeneous land-cover texture classes and enhance the overall performance of HSI classification. To utilize the benefits of B spectral bands and inspired by the Inception paradigm [60], the initial layer of G2C-ContextNet extracts multi-scale fine-grained spatial-spectral feature with the help of spatially increasing kernels of sizes (1×1×B), (3×3×B) and (5×5×B), respectively. These individual feature maps are combined and pass through ReLU non-linearity as well as local response normalization (LRN) [61]. Then, a G2C-Conv3D layer kernelized with (1×1×B) is employed and followed by two consecutive residual blocks along with two more G2C-Conv3D convolution layer to generate the feature maps that directly project in the final separable space for classification immediately after the max-pool2D operation. This whole structure of the model for HSI classification is illustrated in Fig. 6.

G. Revisiting Autoencoders for HSI classification

An autoencoder [62], the simplest reconstruction neural network consists with an encoder and a decoder architecture.

The aims of encoder network $q_{\theta}(\delta|\mathcal{X})$ is to map the input $\mathcal{X} \in \mathcal{R}^{9 \times 9 \times B}$ into a hidden representation $\delta \in \mathcal{R}^d$, unlike autoencoder, the decoder network $p_{\phi}(\hat{\mathcal{X}}|\delta)$ reconstructs back $\hat{\mathcal{X}} \in \mathcal{R}^{9 \times 9 \times B}$ the original representation from δ by minimizing the discrepancy between the original and its reconstructed representation using the widely used mean squared error \mathcal{L} , given as $\mathcal{L}(\mathcal{X}, \hat{\mathcal{X}}) = \|\mathcal{X} - \hat{\mathcal{X}}\|_2$. Fig. 7 shows a three layered convolutional autoencoder (CAE) network having symmetric in shape where each convolutional layers consisting of a Conv3D, and batch normalization (BN) followed by a ReLU non-linearities. The dimension of hidden representation $\delta \in \mathcal{R}^d$ is smaller than \mathcal{X} , and the reduce dimension can be used for further classification task once the reconstruction is very similar to the original.

Similar to the autoencoder network, the variational autoencoder (VAE) network [63] tries to map into a distribution by generating a mean (μ) and variance (Σ) of a Gaussian distribution in each dimension instead of projecting whole \mathcal{X} into a fixed hidden representation δ as shown in Fig. 7. Accordingly z is sampled from the variational normal distribution $N(\mu, \Sigma)$ and the sampled z is projected through few hidden layers via up-sampling method followed by G2C-Conv3D operation and then to the reconstructed image space $\hat{\mathcal{X}}$ of the original \mathcal{X} . Due to not having proper global feature representations and the loss function of the VAE consists with a negative log-likelihood and a KL-regularizer [64], [65]. The individual loss l_i that depends on a single data point x_i and can be calculated as follows,

$$l_i(\theta, \phi) = -E_{\delta \sim q_{\theta}(\delta|x_i)}[\log p_{\phi}(x_i|\delta)] + KL(q_{\theta}(\delta|x_i)||p(\delta)) \quad (7)$$

The first term in Eqn. (7) represents the reconstruction loss, encountered by the expected negative log-likelihood of the i -th data point x_i . This loss term encourages $p_{\phi}(x_i|\delta)$ to learn the representation better. Similarly, the second term indicates the Kullback-Leibler (KL) divergence which can be obtained from the encoder's distribution $q_{\theta}(\delta|x_i)$ and prior $p(z)$. The KL-divergence ensures how much information is lost during the reconstruction process.

The stack autoencoder (SAE) [19], [66] is a neural network consist of several convolutional autoencoders where output of each latent representation is connected to the input of the successive autoencoder. After the successful training of the latent feature of the first autoencoder using sufficient input

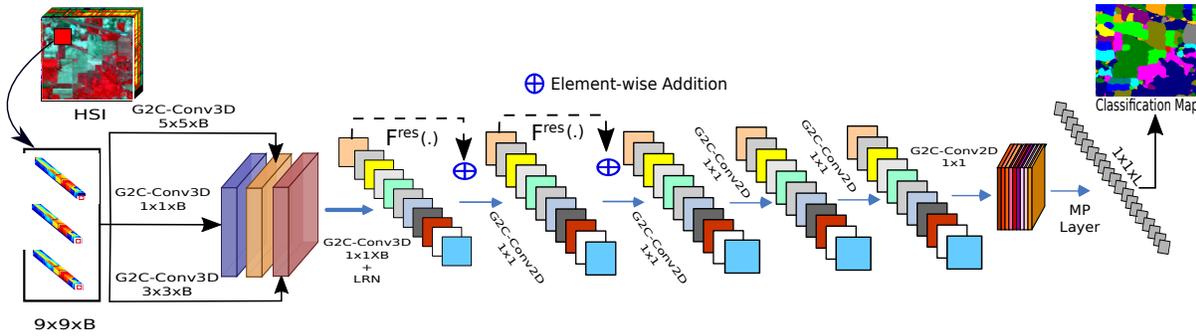


Fig. 6. Graphical representation of generalized gradient centralized contextual network (G2C-ContextNet) for HSI classification.

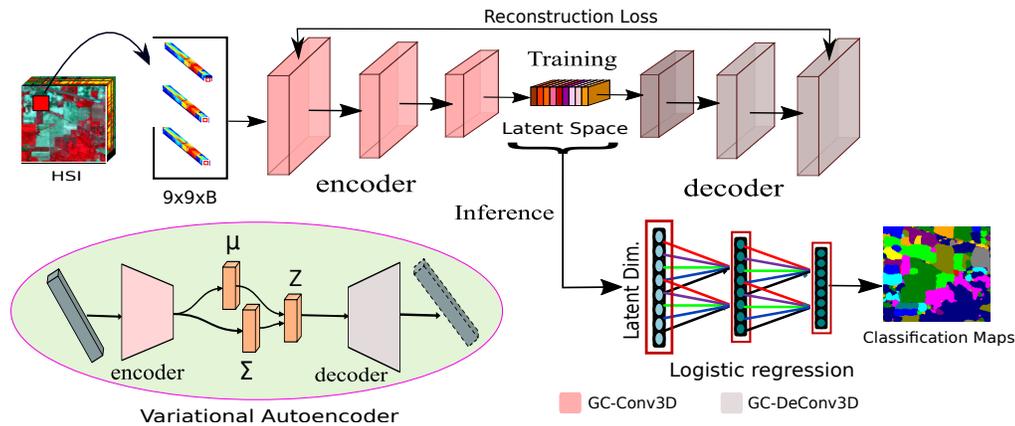


Fig. 7. Graphical representation of the reconstruction networks including variational autoencoder (VAE) which consists of generalized gradient centralized encoder (G2C-Encoder) and decoder (G2C-Decoder) and a classifier for final prediction.

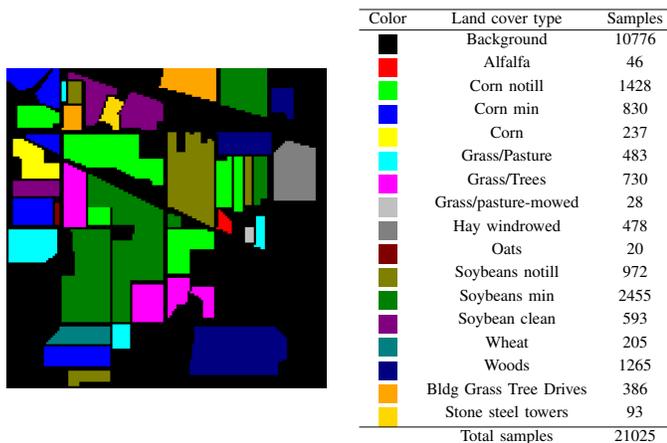


Fig. 8. The ground truth, type associated with the land-cover classes, and number of available samples in the Indian Pines (IP) dataset.

data then this learned latent feature is used to train for the successive layers of next AE and this will continued until the training is completed or the reconstructed quality of HSI is sufficiently good enough.

The above discussed AE and its variants are initially designed with vanilla convolution (i.e., 2D or 3D) and unable to achieved enough discriminative ability to represent the latent space effectively this is due to the weak feature representation produced by the vanilla convolution. To increase the discriminative power into the hidden dimension, and this can

be achieved by generalized gradient centralized convolutions which is plugged with the models CAE2D, CAE3D, VAE3D, and SAE3D to encode more robust fine-grained information extracted during convolution operation and we termed the models as G2C-CAE2D, G2C-CAE3D, G2C-VAE3D, and G2C-SAE2D, respectively.

IV. EXPERIMENTAL RESULT AND DISCUSSION

A. Experimental Data

To verify the performance of the proposed gradient centralized convolutional models, four well-known and benchmark datasets (i.e., IP, KSC, UP, and UH)³ were considered for the evaluation. The detailed description of the datasets are given as follows

Indian Pines (IP) data sets was gathered by AVIRIS [67] (Airborne Visible/Infrared Imaging Spectrometer) sensor over the Indian Pines test site in North-western Indiana in 1992. IP has images with 145×145 spatial dimension of 20m/pixel and 224 spectral bands where the wavelength range of 400 to 2500 nm, 24 bands have been removed out of which 4 null spectral bands and other 20 bands are corrupted by the atmospheric water absorption. The 16 mutually exclusive vegetation classes are available into the IP data set. However, about 50% (10249) pixels from a total of 21025 contain ground truth information

³<http://dase.grss-ieee.org/>

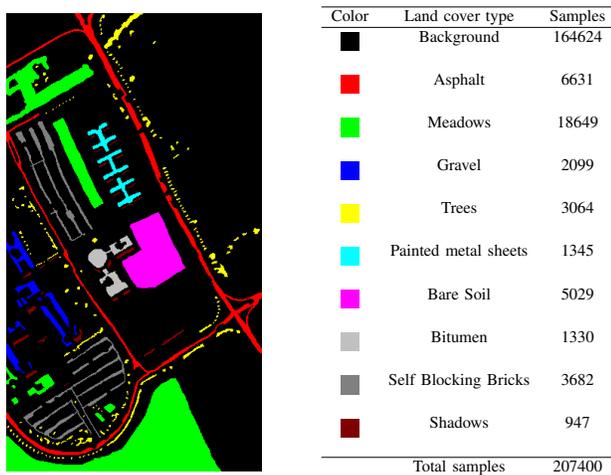


Fig. 9. The ground truth, type associated with the land-cover classes, and the number of available samples in the University of Pavia (UP) dataset.

from 16 different land-cover classes. The ground truth and class-specific samples of 16 land-cover for IP data are shown in Fig. 8.

The **University of Pavia (UP) dataset** acquired by the RO-SIS (Reflective Optics System Imaging Spectrometer) sensor during a flight campaign over the university campus at Pavia, Northern Italy in 2001 and it consists of 610×340 pixel spatially with 103 spectral bands in the wavelength range of 430 to 860 nm and having 1.3 mpp spatial resolution. The ground truth is designed to have 9 urban land-cover classes. Moreover, about 20% of the total 207400 pixels contain ground truth information. The class-specific samples of 9 different land-cover for UP data and ground truth are shown in Fig. 9.

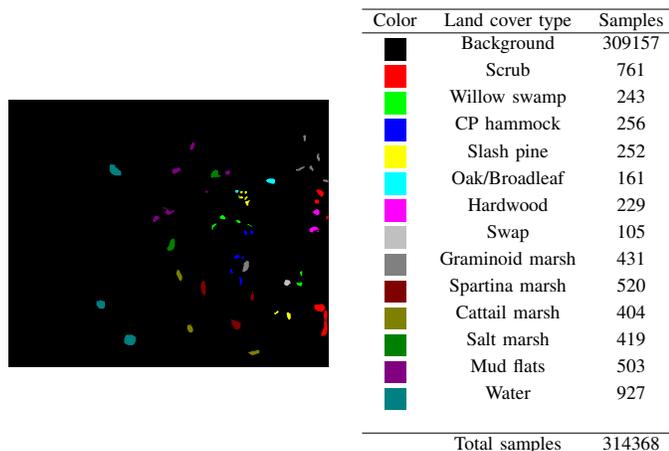


Fig. 10. The ground truth, type associated with the land-cover classes, and the number of available samples in the Kennedy Space Centre (KSC) dataset.

The Kennedy Space Center (KSC) dataset was gathered in 1996 by AVIRIS [67] with wavelengths ranging from 400 to 2500 nm. The images have a spatial dimension 512×614 pixels and 176 spectral bands after removal of some low signal-to-noise ratio (SNR) bands. The KSC dataset consists of in total 5202 samples of 13 upland and wetland classes. The ground truth of 13 land-cover and the class-specific samples for KSC data is shown in Fig. 10.

The IEEE Geoscience and Remote Sensing Society published the **University of Houston (UH) dataset** –collected by the Compact Airborne Spectrographic Imager (CASI)– in 2013 [68], [69], as part of its Data Fusion Contest. It is composed of 340×1905 pixels with 144 spectral bands. The spatial resolution of this dataset is 2.5 m with a wavelength ranging from 0.38 to 1.05 μm . Finally, the ground truth comprises 15 different land-cover classes, which are shown along with the samples in Fig. 11.

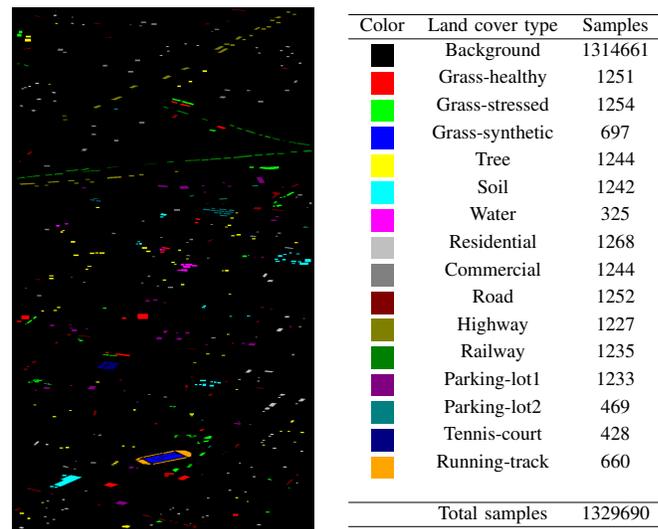


Fig. 11. The ground truth, type associated with the land-cover classes, and the number of available samples in the University of Houston (UH) dataset.

B. Experimental Setup

In order to examine the effectiveness of the proposed gradient centralized 3D convolution (GC-Conv3D) operation, we have conducted the experiments with four classical machine learning methods and seven deep representation learning based models. The classical machine learning methods are available on [70]⁴ and which include: RF [71], SVM with radial basis function [72], gated recurrent unit (GRU) [73], and long short term memory (LSTM) [74]. Whereas ResNet [75], ContextNet [34], MS3DNet [40], ENL-FCN [76], DPyResNet [43], HResNetAM [77], LBPCNN [28], TripletWS [36] and SSRN [37], respectively, are considered for the deep representation learning models. The experiments are conducted using a 64-bit Ubuntu 18.04LTS operating system with NVIDIA Tesla V100 $\times 4$, each of having 32-GB of graphics processing unit. The whole framework is implemented in PyTorch 1.7.0 with CUDA 10.1 enabled. The deep CNNs are considered to enable the learning of spectral-spatial feature representation, and all the networks are trained by extracting 3D patches of size $9 \times 9 \times B$ after the PCA transformation where B is the spectral bands of Indian Pines (IP), Kennedy Space Center (KSC), University of Pavia (UP), and the University of Houston (UH) datasets are set to 200, 176, 103 and 144, respectively. The training is performed with a batch of size 32 and runs the

⁴https://github.com/mhaut/hyperspectral_deeplearning_review

whole system five times repeatedly, each run consists of 200 epochs for all the considered networks. The Cosine annealing scheduler is used for the update of the learning rate during training and the Adam [78] optimizer is used for training all the networks. In addition the experimental results tables are obtained using 10% of randomly selected training samples while the remaining 90% samples are used for the testing purpose over the four considered datasets.

C. Classification Results

Tables II-V report the average classification accuracies in terms of the OAs, AAs, and $\text{Kappa} \times 100$ coefficients for all the compared methods, including the class-specific accuracies for all the classes of the considered IP, KSC, UP and UH datasets and the randomly selected 10% of training samples. It can be noted that all of the reported results are taken on average over five individual runs. The highest achieved accuracies are represented in bold across all the methods considered for comparison.

The class imbalance is a long-standing problem for IP data set and hence pre-processing becomes challenging. Table II demonstrates that $G2C-SSRN$ achieves excellent performance in terms of average OA ($98.51 \pm 0.002\%$), AA ($96.24 \pm 0.025\%$) and Kappa ($98.41 \pm 0.002\%$), respectively. Whereas the second highest OA, AA and Kappa performance achieved by $SSRN$ ($98.28 \pm 0.004\%$), $ENL-FCN$ ($95.21 \pm 0.0028\%$), and $SSRN$ ($98.05 \pm 0.002\%$) reported in Table II. It can also be observed that even $ENL-FCN$ shows good AA ($95.21 \pm 0.028\%$) performance as compared to $ContextNet$ ($94.96 \pm 0.003\%$) but outperforms $ENL-FCN$ in terms of both OA ($96.98 \pm 0.006\%$) and Kappa ($96.55 \pm 0.007\%$) with a significant margin. The $ResNet$ model is performing better in all respect as compared to $MS3DNet$ even though 2D convolutional layers are used to design the $ResNet$ model and shows better or comparable OA but achieves worst AA and Kappa values than $LBPCNN$. Similarly, $ResNet$ outperforms $DPyResNet$ in terms of OA ($92.44 \pm 0.006\%$) and Kappa ($91.37 \pm 0.006\%$), respectively. According to the experiments performed using classical models, The SVM and $LSTM$ show similar OA and Kappa performance but $LSTM$ outperforms SVM in AA with a margin of 2.17%. It is worth to be pointed that RF shows worst performance among all the compared methods.

In a similar way, Table III shows the results of $G2C-SSRN$ and other comparing methods in the HSI classification task obtained for the KSC dataset. We observe that the OA ($99.37 \pm 0.003\%$), AA ($98.89 \pm 0.005\%$) and Kappa ($99.30 \pm 0.003\%$) values for the proposed $G2C-SSRN$ method are the highest among all the considered models. Whereas the second highest OA ($99.25 \pm 0.020\%$) and Kappa ($99.13 \pm 0.018\%$) achieve by the $ENL-FCN$ and outperforms $ContextNet$ significantly, which reveals the discriminative power of non-local attention blocks applied in the HSI classification task. In addition, the performance of $LBPCNN$ in terms of OA, AA, and Kappa values are better than $ContextNet$, $MS3DNet$ and $DPyResNet$, respectively. Whereas $MS3DNet$ and $DPyResNet$ seem to reach similar OA ($95.61 \pm 0.019\%$) and Kappa ($95.11 \pm 0.021\%$) performance but $DPyResNet$

($94.22 \pm 0.024\%$) shows better AA value as compared with $MS3DNet$ ($93.65 \pm 0.023\%$) may be because of the pyramid feature learning paradigm. Besides this, recurrent networks (GRU and $LSTM$) exhibit maximum OA performance gains of 4.66% especially when compared with RF but SVM achieving the lowest accuracy results among all the classical models.

To determine the generalization power of the proposed $G2C-SSRN$ network, the experiments are conducted even in the highly imbalanced and high-resolution UP scene and shown in Fig. 9. The $G2C-SSRN$ network achieves classification performance in terms of average OA ($99.83 \pm 0.001\%$), AA ($99.74 \pm 0.001\%$), Kappa ($99.75 \pm 0.001\%$) and outperforms all the compared methods in three quantitative measurements. The OA performance of $ENL-FCN$ ($99.76 \pm 0.002\%$) shows better as compared to $SSRN$ ($99.77 \pm 0.001\%$). The $ContextNet$ ($99.57 \pm 0.001\%$) achieves higher OA than $DPyResNet$ ($97.05 \pm 0.010\%$) due to the nature of 3D convolution used in the layers of $ContextNet$. Besides it, $ResNet$ and $DPyResNet$ show close improvements in performance but achieves significantly less OA, AA, and Kappa as compared to $LBPCNN$. Among all the classical models SVM ($94.19 \pm 0.002\%$) achieves sound overall classification performance. Similarly, recurrent networks (GRU and $LSTM$) seem to perform equally but reach lower OA results in comparison with the proposed $G2C-SSRN$ network. In this sense, the $G2C-SSRN$ takes advantage of the fine-grained texture details extracted from the raw UP dataset. Due to the better generalization ability, SVM ($94.19 \pm 0.002\%$) outperforms RF ($90.41 \pm 0.001\%$) in terms of OAs.

Finally, to explore the convergence stability of the model $G2C-SSRN$, a widely used complex UH scene is considered too for the experiment. The total number of land-cover specific samples along with ground truth are shown in Fig. 11 for the UH dataset. Table V shows the quantitative results for the UH dataset in terms of OA, AA, and Kappa, respectively, which also includes the number of training and test samples for the experiment. The proposed $G2C-SSRN$ model constantly outperforms all the compared methods in quantitative measurements OA ($99.29 \pm 0.001\%$), AA ($99.34 \pm 0.0\%$), and Kappa ($99.22 \pm 0.001\%$) and achieves OA (0.59%), AA (0.61%), and Kappa (0.62%) performance gain when compared with $SSRN$. Residual like networks (i.e., $ResNet$, $ContextNet$ and $MS3DNet$) seem to show similar performance improvements but the overall performance of $DPyResNet$ ($76.88 \pm 0.134\%$) is not satisfactory. $LBPCNN$ outperforms $ResNet$, $ContextNet$, $MS3DNet$ and $DPyResNet$ in terms of OA, AA and Kappa values. Among all the classical models, GRU ($85.94 \pm 0.10\%$) outperforms the overall performance of SVM ($81.24 \pm 0.0\%$) and $LSTM$ ($82.36 \pm 0.55\%$). Both $DPyResNet$ and RF show similar performance improvement.

There is another factor, spatial input 3D patch size, which can directly influence the overall classification performance of the networks. Table VI shows the OAs using $ResNet$, $DPyResNet$, $SSRN$, $ENL-FCN$, $MS3DNet$, $HResNetAM$, $LBPCNN$, and $G2C-SSRN$, respectively based on the spatial window of size 7×7 , 9×9 , and 11×11 for all the considered four datasets. The best achieved OAs are highlighted in bold for each model. It is observed from the result table that the best OAs achieved

TABLE II
OA, AA, AND κ VALUES ON IP DATASET USING RANDOMLY SELECTED 10% OF TRAINING SAMPLES.

Class	Train	Test	Classical Models				Deep Neural Networks							
			RF	SVM	GRU	LSTM	ResNet	ContextNet	MS3DNet	ENL-FCN	DPyResNet	SSRN	LBPCCN	GC-SSRN
1	4	42	28.46 ± 0.061	51.22 ± 0.190	69.92 ± 0.141	69.11 ± 0.090	98.66 ± 0.018	88.78 ± 0.080	66.67 ± 0.471	97.56 ± 0.000	94.59 ± 0.076	57.78 ± 0.433	92.23 ± 0.054	85.92 ± 0.199
2	142	1286	56.63 ± 0.024	81.22 ± 0.037	76.96 ± 0.013	74.22 ± 0.016	87.85 ± 0.029	98.19 ± 0.005	75.94 ± 0.080	93.15 ± 0.000	93.83 ± 0.040	98.37 ± 0.012	94.57 ± 0.013	98.44 ± 0.009
3	83	747	48.42 ± 0.013	65.82 ± 0.013	67.20 ± 0.041	71.49 ± 0.030	92.71 ± 0.007	97.37 ± 0.028	81.39 ± 0.007	97.59 ± 0.000	89.30 ± 0.003	97.47 ± 0.010	90.12 ± 0.004	97.66 ± 0.014
4	23	214	33.49 ± 0.025	57.75 ± 0.041	61.82 ± 0.039	60.72 ± 0.041	95.43 ± 0.046	97.04 ± 0.021	88.63 ± 0.063	91.55 ± 0.000	93.51 ± 0.055	99.12 ± 0.0099	94.26 ± 0.045	98.94 ± 0.008
5	48	435	85.21 ± 0.025	90.04 ± 0.014	85.36 ± 0.022	87.51 ± 0.015	98.23 ± 0.015	97.78 ± 0.015	95.61 ± 0.054	97.47 ± 0.000	99.26 ± 0.004	97.79 ± 0.013	97.14 ± 0.005	95.67 ± 0.028
6	73	657	92.64 ± 0.027	96.25 ± 0.006	94.22 ± 0.008	94.77 ± 0.015	97.98 ± 0.011	98.60 ± 0.008	97.78 ± 0.026	99.24 ± 0.000	98.52 ± 0.007	98.50 ± 0.010	99.01 ± 0.008	98.44 ± 0.009
7	2	26	2.67 ± 0.038	73.33 ± 0.019	50.67 ± 0.068	85.33 ± 0.094	92.98 ± 0.099	90.35 ± 0.098	100.00 ± 0.000	100.00 ± 0.000	83.08 ± 0.178	66.67 ± 0.471	85.27 ± 0.124	87.38 ± 0.178
8	47	431	97.67 ± 0.015	97.98 ± 0.009	97.83 ± 0.001	97.83 ± 0.009	95.06 ± 0.014	97.76 ± 0.026	89.51 ± 0.091	97.44 ± 0.000	97.63 ± 0.022	96.45 ± 0.029	97.89 ± 0.043	98.07 ± 0.015
9	2	18	9.26 ± 0.094	50.00 ± 0.045	37.04 ± 0.146	53.70 ± 0.139	90.83 ± 0.283	86.90 ± 0.102	66.67 ± 0.471	72.22 ± 0.000	66.66 ± 0.471	56.25 ± 0.418	70.24 ± 0.149	93.33 ± 0.094
10	97	875	60.91 ± 0.047	73.87 ± 0.018	76.00 ± 0.013	73.68 ± 0.025	96.05 ± 0.013	96.08 ± 0.018	81.47 ± 0.070	94.74 ± 0.000	93.77 ± 0.029	98.33 ± 0.009	93.12 ± 0.014	97.53 ± 0.011
11	245	2210	87.88 ± 0.019	82.90 ± 0.012	80.31 ± 0.027	84.93 ± 0.024	93.32 ± 0.041	97.35 ± 0.004	76.69 ± 0.096	95.61 ± 0.000	89.78 ± 0.040	99.08 ± 0.005	90.76 ± 0.059	97.78 ± 0.017
12	59	534	41.26 ± 0.030	74.91 ± 0.043	78.65 ± 0.014	73.35 ± 0.052	86.65 ± 0.077	94.00 ± 0.012	88.65 ± 0.036	97.00 ± 0.000	83.43 ± 0.107	98.46 ± 0.009	85.12 ± 0.119	86.76 ± 0.022
13	20	185	90.09 ± 0.040	96.94 ± 0.021	96.94 ± 0.014	98.74 ± 0.005	82.16 ± 0.076	95.01 ± 0.03	99.78 ± 0.003	97.83 ± 0.000	98.19 ± 0.021	100.00 ± 0.000	99.27 ± 0.017	99.16 ± 0.005
14	126	1139	95.46 ± 0.014	93.82 ± 0.010	94.50 ± 0.012	96.22 ± 0.004	95.39 ± 0.016	98.49 ± 0.014	90.06 ± 0.087	99.12 ± 0.000	96.00 ± 0.021	98.63 ± 0.010	97.13 ± 0.031	99.19 ± 0.003
15	38	348	41.11 ± 0.029	60.42 ± 0.044	65.71 ± 0.019	60.04 ± 0.029	90.96 ± 0.127	94.10 ± 0.031	88.21 ± 0.044	92.80 ± 0.000	91.22 ± 0.040	99.24 ± 0.005	92.22 ± 0.051	98.18 ± 0.011
16	9	84	79.37 ± 0.030	91.27 ± 0.054	82.54 ± 0.037	90.87 ± 0.022	94.73 ± 0.038	93.57 ± 0.046	88.53 ± 0.021	100.00 ± 0.000	70.90 ± 0.388	95.63 ± 0.062	73.17 ± 0.447	92.05 ± 0.092
OA	1018	9251	72.98 ± 0.006	82.00 ± 0.006	81.24 ± 0.003	82.13 ± 0.004	92.44 ± 0.006	96.98 ± 0.006	83.44 ± 0.060	96.15 ± 0.054	91.47 ± 0.029	98.28 ± 0.004	92.15 ± 0.013	98.51 ± 0.002
AA			59.41 ± 0.005	77.36 ± 0.019	75.98 ± 0.008	79.53 ± 0.005	91.19 ± 0.025	94.96 ± 0.003	86.91 ± 0.084	95.21 ± 0.028	94.14 ± 0.006	91.01 ± 0.080	95.27 ± 0.004	96.24 ± 0.025
$\kappa \times 100$			68.62 ± 0.007	79.41 ± 0.007	78.58 ± 0.004	79.54 ± 0.004	91.37 ± 0.006	96.55 ± 0.007	80.82 ± 0.070	95.60 ± 0.030	90.20 ± 0.034	98.05 ± 0.005	92.23 ± 0.023	98.41 ± 0.002

TABLE III
OA, AA, AND κ VALUES ON KSC DATASET USING RANDOMLY SELECTED 10% OF TRAINING SAMPLES.

Class	Train	Test	Classical Models				Deep Neural Networks							
			RF	SVM	GRU	LSTM	ResNet	ContextNet	MS3DNet	ENL-FCN	DPyResNet	SSRN	LBPCCN	GC-SSRN
1	76	685	94.79 ± 0.012	95.43 ± 0.023	96.98 ± 0.011	94.60 ± 0.004	94.73 ± 0.008	99.78 ± 0.001	96.42 ± 0.009	99.71 ± 0.000	99.06 ± 0.010	99.95 ± 0.001	99.02 ± 0.020	100.00 ± 0.000
2	24	219	83.71 ± 0.012	83.76 ± 0.012	82.04 ± 0.023	85.69 ± 0.012	66.45 ± 0.310	98.79 ± 0.014	95.88 ± 0.012	100.00 ± 0.000	89.72 ± 0.026	100.00 ± 0.000	89.89 ± 0.037	97.47 ± 0.035
3	25	231	86.09 ± 0.020	78.41 ± 0.218	89.13 ± 0.023	91.16 ± 0.040	65.08 ± 0.187	82.83 ± 0.047	80.12 ± 0.168	100.00 ± 0.000	81.84 ± 0.074	96.66 ± 0.005	82.87 ± 0.089	99.01 ± 0.013
4	25	227	71.22 ± 0.061	27.17 ± 0.173	56.98 ± 0.062	68.14 ± 0.049	73.62 ± 0.185	78.41 ± 0.165	90.06 ± 0.012	98.67 ± 0.000	89.83 ± 0.040	91.22 ± 0.047	91.73 ± 0.051	97.58 ± 0.018
5	16	145	47.59 ± 0.060	22.99 ± 0.170	68.74 ± 0.092	49.89 ± 0.107	60.74 ± 0.275	74.22 ± 0.097	85.86 ± 0.034	98.61 ± 0.000	88.34 ± 0.095	100.00 ± 0.000	90.37 ± 0.090	97.05 ± 0.023
6	22	207	48.22 ± 0.014	36.89 ± 0.078	63.27 ± 0.100	54.21 ± 0.062	66.58 ± 0.324	92.64 ± 0.050	85.61 ± 0.030	100.00 ± 0.000	88.54 ± 0.138	98.45 ± 0.022	90.27 ± 0.127	97.04 ± 0.041
7	10	95	79.43 ± 0.096	87.94 ± 0.027	90.78 ± 0.031	89.01 ± 0.013	66.20 ± 0.468	94.60 ± 0.037	90.75 ± 0.088	100.00 ± 0.000	100.00 ± 0.000	95.42 ± 0.050	99.65 ± 0.015	98.16 ± 0.025
8	43	388	78.61 ± 0.054	70.19 ± 0.073	90.03 ± 0.031	92.53 ± 0.011	92.53 ± 0.081	97.49 ± 0.009	98.99 ± 0.008	100.00 ± 0.000	94.81 ± 0.037	99.80 ± 0.003	95.27 ± 0.045	99.71 ± 0.002
9	52	468	89.46 ± 0.011	85.33 ± 0.021	96.01 ± 0.015	95.37 ± 0.028	90.82 ± 0.106	99.92 ± 0.001	97.44 ± 0.028	100.00 ± 0.000	99.06 ± 0.002	100.00 ± 0.000	99.07 ± 0.001	100.00 ± 0.000
10	40	364	88.43 ± 0.034	78.88 ± 0.069	91.09 ± 0.015	94.03 ± 0.007	87.59 ± 0.171	100.00 ± 0.000	98.78 ± 0.013	100.00 ± 0.000	99.46 ± 0.004	100.00 ± 0.000	99.54 ± 0.003	100.00 ± 0.000
11	41	378	95.58 ± 0.014	93.81 ± 0.008	96.02 ± 0.026	96.11 ± 0.008	98.96 ± 0.015	99.90 ± 0.001	98.67 ± 0.013	100.00 ± 0.000	99.90 ± 0.001	100.00 ± 0.000	99.90 ± 0.001	100.00 ± 0.000
12	50	453	82.63 ± 0.032	86.98 ± 0.009	89.99 ± 0.035	92.35 ± 0.035	84.26 ± 0.207	99.17 ± 0.006	99.06 ± 0.005	100.00 ± 0.000	94.42 ± 0.056	100.00 ± 0.000	95.21 ± 0.027	100.00 ± 0.000
13	92	835	99.60 ± 0.002	100.00 ± 0.000	99.92 ± 0.001	99.96 ± 0.001	100.00 ± 0.000	99.96 ± 0.001	100.00 ± 0.000	100.00 ± 0.000	99.96 ± 0.001	100.00 ± 0.000	99.99 ± 0.001	100.00 ± 0.000
OA	516	4695	86.17 ± 0.004	81.27 ± 0.008	90.13 ± 0.004	90.18 ± 0.005	79.73 ± 0.207	96.34 ± 0.014	95.61 ± 0.019	99.25 ± 0.020	95.61 ± 0.019	99.19 ± 0.004	96.27 ± 0.011	99.37 ± 0.003
AA			80.25 ± 0.004	72.90 ± 0.021	85.46 ± 0.010	84.85 ± 0.008	80.58 ± 0.169	93.65 ± 0.026	93.66 ± 0.023	98.77 ± 0.021	94.22 ± 0.024	98.80 ± 0.008	95.34 ± 0.023	98.89 ± 0.005
$\kappa \times 100$			84.59 ± 0.004	79.09 ± 0.009	88.99 ± 0.004	89.06 ± 0.006	78.06 ± 0.222	95.93 ± 0.016	95.11 ± 0.021	99.13 ± 0.018	95.11 ± 0.021	99.11 ± 0.004	96.22 ± 0.031	99.30 ± 0.003

TABLE IV
OA, AA, AND κ VALUES ON UP DATASET USING RANDOMLY SELECTED 10% OF TRAINING SAMPLES.

Class	Train	Test	Classical Models				Deep Neural Networks							
			RF	SVM	GRU	LSTM	ResNet	ContextNet	MS3DNet	ENL-FCN	DPyResNet	SSRN	LBPCCN	GC-SSRN
1	663	5968	91.11 ± 0.007	94.30 ± 0.008	93.34 ± 0.003	95.47 ± 0.005	96.82 ± 0.023	99.56 ± 0.002	99.36 ± 0.001	99.98 ± 0.000	98.35 ± 0.017	99.85 ± 0.001	99.02 ± 0.014	99.93 ± 0.002
2	1864	16785	98.11 ± 0.003	97.65 ± 0.002	97.54 ± 0.002	96.91 ± 0.002	98.59 ± 0.008	99.85 ± 0.002	99.80 ± 0.000	100.00 ± 0.000	98.76 ± 0.008	99.98 ± 0.000	99.98 ± 0.000	99.91 ± 0.000
3	209	1890	67.71 ± 0.014	81.26 ± 0.018	77.08 ± 0.038	78.01 ± 0.011	90.01 ± 0.031	99.19 ± 0.004	99.02 ± 0.017	100.00 ± 0.000	99.68 ± 0.003	99.68 ± 0.003	99.68 ± 0.003	99.68 ± 0.003
4	306	2758	88.20 ± 0.006	94.63 ± 0.004	93.22 ± 0.023	94.92 ± 0.007	99.32 ± 0.003	99.80 ± 0.002	99.71 ± 0.001	98.94 ± 0.000	99.20 ± 0.005	99.92 ± 0.001	99.23 ± 0.004	99.84 ± 0.004
5	134	1211	98.93 ± 0.002	99.20 ± 0.002	99.42 ± 0.004	99.26 ± 0.003	99.81 ± 0.000	99.91 ± 0.001	99.94 ± 0.000	100.00 ± 0.000	99.72 ± 0.003	99.94 ± 0.000	99.89 ± 0.003	99.95 ± 0.000
6	502	4527	72.14 ± 0.022	90.58 ± 0.008	87.41 ± 0.016	87.85 ± 0.012	99.41 ± 0.002	99.75 ± 0.003	99.43 ± 0.003	99.87 ± 0.000	98.52 ± 0.006	99.95 ± 0.001	98.91 ± 0.005	99.95 ± 0.000
7	133	1197	75.69 ± 0.017	85.71 ± 0.011	85.38 ± 0.039	80.23 ± 0.023	96.90 ± 0.017	98.37 ± 0.022	99.18 ± 0.005	100.00 ± 0.000	97.37 ± 0.004	100.00 ± 0.000	98.57 ± 0.003	100.00 ± 0.000
8	368	3314	89.64 ± 0.013	88.20 ± 0.003	88.56 ± 0.024	88.49 ± 0.008	92.00 ± 0.044	98.48 ± 0.008	97.13 ± 0.005	99.69 ± 0.000	84.51 ± 0.071	98.28 ± 0.015	87.21 ± 0.042	98.49 ± 0.001
9	94	854	99.77 ± 0.002	99.84 ± 0.001	99.84 ± 0.001	99.88 ± 0.001	98.88 ± 0.012	99.26 ± 0.005	99.74 ± 0.002	100.00 ± 0.000	99.60 ± 0.001	99.39 ± 0.003	99.47 ± 0.001	100.00 ± 0.000
OA	4273	38503	90.41 ± 0.001	94.19 ± 0.002	93.34 ± 0.002	93.45 ± 0.001	97.38 ± 0.007	99.57 ± 0.001	99.35 ± 0.001	99.76 ± 0.002	97.05 ± 0.010	99.77 ± 0.001	98.24 ± 0.021	99.83 ± 0.001
AA			86.81 ± 0.002	92.38 ± 0.003	91.31 ± 0.008	91.23 ± 0.001	96.86 ± 0.005	99.25 ± 0.002	99.15 ± 0.002	99.70 ± 0.002	96.69 ± 0.006	99.66 ± 0.001	97.23 ± 0.004	99.74 ± 0.001
$\kappa \times 100$			87.10 ± 0.002	92.29 ± 0.002	91.15 ± 0.003	91.30 ± 0.001	96.52 ± 0.009	99.43 ± 0.001	99.13 ± 0.002	99.72 ± 0.001	96.08 ± 0.013	99.69 ± 0.001	97.05 ± 0.025	99.75 ± 0.001

TABLE V
OA, AA, AND κ VALUES ON UH DATASET USING RANDOMLY SELECTED 10% OF TRAINING SAMPLES.

Class	Train	Test
-------	-------	------

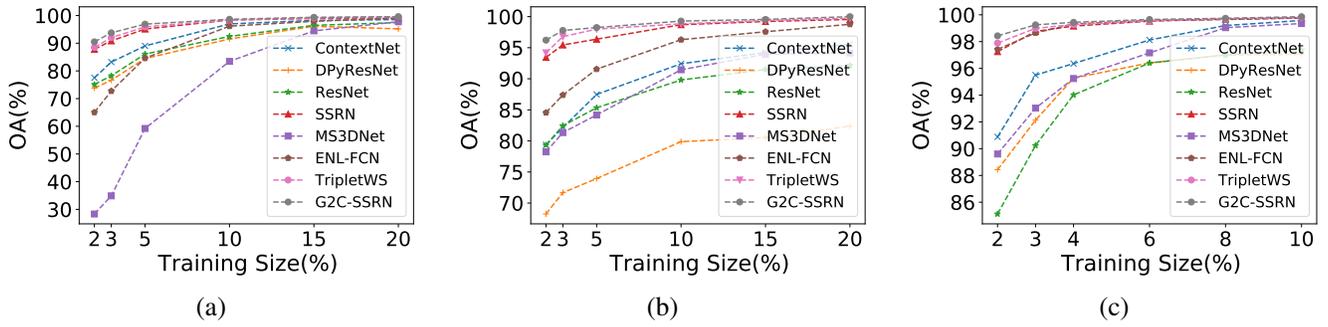


Fig. 12. OA achieved by different methods with varying training sample sizes which are randomly taken from (a) IP (b) UH and (c) UP datasets.

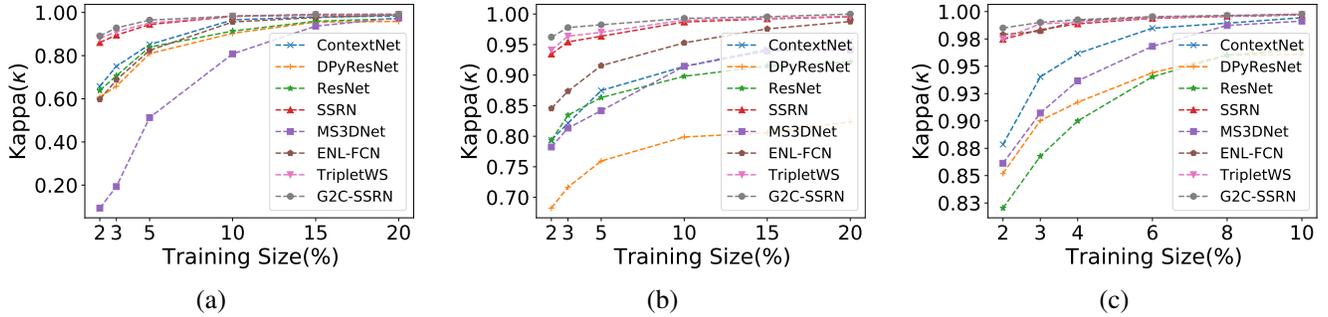


Fig. 13. coefficient achieved by different methods with varying training sample sizes which are randomly taken from (a) IP (b) UH and (c) UP datasets.

TABLE VI
OVERALL ACCURACY (%) OF RESNET, DPyRESNET, SSRN, ENL-FCN, MS3DNET, HRESNETAM, LBPCNN AND THE GC-SSRN NETWORK USING DIFFERENT SPATIAL WINDOW SIZES AND 10% OF TRAINING SAMPLES.

Datasets	Spatial Window	Models							
		ResNet	DPyResNet	SSRN	ENL-FCN	MS3DNet	HResNetAM	LBPCNN	GC-SSRN
IP	7×7	87.47 ± 0.008	91.31 ± 0.006	98.10 ± 0.005	69.25 ± 0.001	77.33 ± 0.015	97.77 ± 0.002	90.17 ± 0.045	98.24 ± 0.001
KSC		76.03 ± 0.052	82.12 ± 0.033	98.97 ± 0.004	27.40 ± 0.001	95.95 ± 0.007	98.69 ± 0.003	95.24 ± 0.023	99.04 ± 0.002
UP		96.33 ± 0.001	96.30 ± 0.006	99.54 ± 0.002	90.32 ± 0.001	98.26 ± 0.001	99.43 ± 0.002	96.57 ± 0.041	99.53 ± 0.001
UH		85.77 ± 0.011	76.05 ± 0.154	98.35 ± 0.002	92.54 ± 0.021	87.30 ± 0.024	98.67 ± 0.001	89.99 ± 0.013	99.11 ± 0.002
IP	9×9	92.44 ± 0.006	91.47 ± 0.029	98.28 ± 0.004	96.15 ± 0.054	83.44 ± 0.060	98.24 ± 0.005	92.15 ± 0.013	98.51 ± 0.002
KSC		95.61 ± 0.019	95.61 ± 0.019	99.19 ± 0.004	99.25 ± 0.020	95.61 ± 0.019	98.89 ± 0.002	96.22 ± 0.031	99.37 ± 0.003
UP		97.38 ± 0.007	97.05 ± 0.010	99.77 ± 0.001	99.76 ± 0.002	99.35 ± 0.001	99.65 ± 0.001	97.05 ± 0.025	99.83 ± 0.001
UH		86.81 ± 0.037	76.88 ± 0.134	98.70 ± 0.000	93.29 ± 0.012	88.45 ± 0.036	98.85 ± 0.003	90.23 ± 0.022	99.29 ± 0.001
IP	11×11	92.67 ± 0.008	92.66 ± 0.003	98.18 ± 0.003	56.48 ± 0.001	68.90 ± 0.150	98.19 ± 0.003	91.23 ± 0.024	98.32 ± 0.002
KSC		93.36 ± 0.023	95.47 ± 0.007	98.83 ± 0.006	53.44 ± 0.001	94.27 ± 0.002	98.75 ± 0.005	96.19 ± 0.011	99.41 ± 0.003
UP		97.83 ± 0.006	97.21 ± 0.016	99.82 ± 0.001	93.92 ± 0.001	99.45 ± 0.001	99.81 ± 0.002	96.89 ± 0.031	99.89 ± 0.001
UH		87.51 ± 0.021	78.54 ± 0.107	98.95 ± 0.001	93.77 ± 0.034	90.24 ± 0.011	99.11 ± 0.002	90.14 ± 0.033	99.36 ± 0.001

TABLE VII
ACCURACY ANALYSIS IN TERMS OF AVERAGE OAS, AAS AND KAPPA'S FOR THE SHALLOW 2D AND 3D CNN MODELS WITH VANILLA AND GENERALIZED GRADIENT CENTRALIZED CONVOLUTIONS.

Dataset	Matrices	CNN models embedded with vanilla convolution				CNN models embedded with gradient centralized convolution			
		CNN2D	CNN3D	ContextNet	MS3DNet	G2C-Conv2DNet	G2C-Conv3DNet	G2C-ContextNet	G2C-MS3DNet
IP	OA	83.14 ± 1.002	86.57 ± 0.004	96.98 ± 0.006	83.44 ± 0.060	86.27 ± 0.952	89.23 ± 0.002	98.14 ± 0.007	86.01 ± 0.040
	AA	75.28 ± 2.001	85.27 ± 0.005	94.96 ± 0.003	86.91 ± 0.084	79.55 ± 1.572	87.64 ± 0.004	96.66 ± 0.002	88.57 ± 0.052
	Kappa	80.59 ± 1.180	85.38 ± 0.004	96.55 ± 0.007	80.82 ± 0.070	83.14 ± 1.270	87.55 ± 0.002	98.43 ± 0.007	83.55 ± 0.050
UP	OA	85.42 ± 2.001	91.13 ± 0.027	99.57 ± 0.001	99.35 ± 0.001	89.14 ± 1.127	93.61 ± 0.054	99.76 ± 0.002	99.71 ± 0.002
	AA	89.69 ± 1.650	92.44 ± 0.033	99.35 ± 0.002	99.15 ± 0.002	91.27 ± 2.003	93.21 ± 0.011	99.61 ± 0.003	99.57 ± 0.001
	Kappa	80.80 ± 3.005	90.65 ± 0.024	99.43 ± 0.001	99.13 ± 0.002	83.33 ± 1.002	92.14 ± 0.033	99.72 ± 0.001	99.47 ± 0.002
KSC	OA	88.78 ± 1.250	94.10 ± 0.004	96.34 ± 0.014	95.61 ± 0.019	90.66 ± 0.333	96.85 ± 0.003	98.25 ± 0.016	97.18 ± 0.011
	AA	89.02 ± 3.154	95.51 ± 0.002	93.65 ± 0.026	93.66 ± 0.023	91.88 ± 2.014	97.15 ± 0.004	95.64 ± 0.033	95.95 ± 0.033
	Kappa	89.57 ± 2.002	95.24 ± 0.004	95.93 ± 0.016	95.11 ± 0.021	91.47 ± 1.307	97.22 ± 0.003	97.11 ± 0.034	97.24 ± 0.031
UH	OA	80.15 ± 1.114	84.83 ± 0.044	89.42 ± 0.024	88.45 ± 0.036	83.28 ± 1.005	87.18 ± 0.034	92.13 ± 0.014	91.23 ± 0.013
	AA	74.37 ± 2.152	86.35 ± 0.017	93.67 ± 0.017	91.77 ± 0.024	76.23 ± 1.117	89.25 ± 0.044	95.57 ± 0.024	92.47 ± 0.014
	Kappa	81.59 ± 3.002	84.36 ± 0.032	89.38 ± 0.021	88.28 ± 0.032	84.08 ± 2.215	87.33 ± 0.023	92.11 ± 0.016	90.11 ± 0.033

methods for a smaller number of training samples. The SSRN achieves better OA and Kappa with varying training sample sets as compared with ENL-FCN for IP and UH datasets. The poor performances are given by MS3DNet for IP, DPyResNet

for UH and finally, ResNet for UP data sets in terms of OA and Kappa accuracies, respectively. It can be noted that both the G2C-SSRN and SSRN seem to perform equally on all three data sets, when a larger training set (e.g., more than 10%

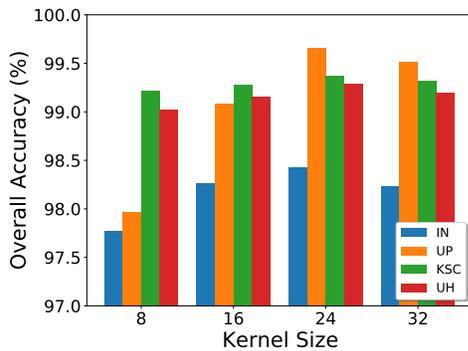


Fig. 14. Impacts of the number of kernels with 10% of training samples randomly taken from IP, KSC and UP, respectively.

IP, UH and 6% over UP dataset) is considered. As compared to SSRN, TripletWS shows better Kappa values for limited training samples. Similarly, the robustness of the G2C-SSRN is clearly visualized when compared the difference of achieved performance between G2C-SSRN and SSRN using a sufficiently smaller training set (e.g., less than 5%). The reason behind the models use G2C-Conv3D operation for the fine-grained spectral-spatial feature extraction those are basically overlooked by vanilla Conv3D operation. In addition, Fig. 14 shows the impact of the different number of kernels used to train the G2C-SSRN on IP, KSC, UH, and KSC datasets, respectively. It can be seen that 24 is the optimal number of kernels for G2C-SSRN model and provides better OAs for all four datasets. Hence, we fixed 24 kernels for all four datasets. The changes of overall accuracies over the different number of kernels are almost the same for KSC and high for UP datasets.

E. Impacts on Shallow Networks

It is hard to visualize the significant performance improvement for the considered G2C-SSRN network with two many layers and evaluating completely in the residual paradigm. In this sense, it is always been recommended to choose those shallow CNN models whose performance is significantly lesser than the state-of-the-art methods. It can be noted that if these models are able to achieve significant performance gain after removing the vanilla 3D convolutions and replaced with the proposed G2C-Conv3D convolution then the inference can be assumed universally true. To show the impacts of the proposed G2C-Conv3D convolution operation another sets of experiment have been conducted using the shallow CNN3D, ContextNet, and MS3DNet models and convert those into gradient centralized networks (i.e., G2C-CNN3DNet, G2C-ContextNet, and G2C-MS3DNet) after replacing all the 3D convolution layers with the proposed G2C-Conv3D convolution operations. We kept the experimental setting as discussed in SubSec. IV-B

Table VII shows the average OA, AA, and Kappa values obtained using the four HSI (i.e., IP, KSC, UP and UH) datasets for the considered vanilla networks and generalized gradient centralized networks (G2C-networks). It can be seen that the G2C-networks (i.e., G2C-CNN3DNet, G2C-ContextNet, and G2C-MS3DNet) outperform all the vanilla networks by

a significant margin over the four datasets. In addition, to the above models, we also include CNN2D which comprises two G2C-Conv2D convolution layers followed by a max-pool2D and two FC layers for the comparative analysis of classification results. Like CNN2D, the G2C-CNN2DNet model reaches the highest performance gain in terms of OA (3.72%) for UP dataset whereas both AA (4.27%) and Kappa (2.55%) simultaneously achieve for IP dataset. Similarly, when compared with CNN3D, the G2C-CNN3DNet model achieves highest classification performance gain in terms of OA (2.75%) for KSC dataset and both AA (2.90%) and Kappa (2.97%) achieve for UH dataset. In the same way, G2C-ContextNet outperforms the vanilla ContextNet model in three quantitative measurements for all the data sets. The improved performance can be visualized in terms of OA (2.71%), and Kappa (2.73%), respectively for the UH dataset, whereas the AA (1.99%) gain achieved for the KSC dataset. Finally, the G2C-MS3DNet shows 2.78% of OA, 2.29% of AA, and 2.73% of Kappa, respectively, obtained for UH, KSC and IP, datasets. As shown in Table VII, the highest achieved OAs, AAs, and Kappa's for each method are highlighted in bold.

The generalized gradient centralized networks (i.e., G2C-CNN2DNet, G2C-CNN3DNet, G2C-ContextNet, and G2C-MS3DNet) achieved noticeable performance gain over the vanilla networks (i.e., CNN2D, CNN3D, ContextNet, and MS3DNet) for the four considered datasets. This improvement reveals the benefit of both G2C-Conv2D and G2C-Conv3D convolution operation over the vanilla 2D/3D convolution operations. Besides it, to classify the very similar land-cover textures, in addition to the spectral-spatial features, the discriminative fine-grained information are equally important for HSI classification task. Its clear that the vanilla 2D/3D convolutions are missed to utilized these fine-grained information and leads to lower values for the quantitative measurements i.e., OAs, AAs and Kappas. In addition, the performance of existing models can be further improved using G2C-Conv3D operation, moreover, the shallow networks are highly benefited towards the improving of overall accuracy in HSI classification context. In contrast, our proposed G2C-Conv3D has the same latency as the vanilla convolutions (detailed shown in Table IX).

F. Impacts on Autoencoder Networks

To further study the robustness of generalized gradient centralized 3D convolutions we have considered the following reconstruction networks i.e., G2C-CAE3D, G2C-VAE3D and G2C-SAE3D for the experiments. Table VIII reports the average OA, AA and Kappa values obtained using 10% of randomly selected training samples taken from four HSI (i.e., IP, KSC, UP and UH) datasets for the considered vanilla autoencoder networks and generalized gradient centralized autoencoder networks (G2C-AE). It can be observed from Table VIII G2C-AE networks outperform all the vanilla AE networks by a significant margin over the four datasets. In addition, to the 3D AE models we have also considered G2C-CAE2D for the quantitative analysis of classification

TABLE VIII
ACCURACY ANALYSIS IN TERMS OF AVERAGE OAS, AAS AND KAPPA'S FOR THE 2D AND 3D RECONSTRUCTION MODELS WITH VANILLA AND GRADIENT CENTRALIZED CONVOLUTIONS.

Dataset	Matrices	Reconstruction Networks with Vanilla Convolution				Reconstruction Networks with Gradient Centralized Convolution			
		CAE2D	CAE3D	VAE3D	SAE3D	G2C-CAE2D	G2C-CAE3D	G2C-VAE3D	G2C-SAE3D
IP	OA	78.24 ± 0.23	80.15 ± 0.34	82.25 ± 0.57	81.45 ± 0.11	80.15 ± 0.28	83.45 ± 0.22	84.11 ± 0.41	84.14 ± 0.19
	AA	90.56 ± 0.15	83.37 ± 0.17	85.64 ± 0.28	86.84 ± 0.34	91.24 ± 0.18	85.28 ± 0.37	86.19 ± 0.38	87.91 ± 0.33
	Kappa	79.37 ± 0.44	80.88 ± 0.56	84.57 ± 0.39	83.22 ± 0.28	82.37 ± 0.54	84.21 ± 0.45	87.28 ± 0.46	86.28 ± 0.41
UP	OA	84.33 ± 0.14	86.74 ± 0.37	85.19 ± 0.12	87.27 ± 0.33	87.22 ± 0.15	88.77 ± 0.33	87.38 ± 0.24	90.38 ± 0.11
	AA	86.15 ± 0.26	87.12 ± 0.11	86.38 ± 0.35	89.13 ± 0.48	88.72 ± 0.23	89.15 ± 0.21	89.33 ± 0.57	89.54 ± 0.28
	Kappa	85.34 ± 0.57	85.37 ± 0.22	88.19 ± 0.24	88.24 ± 0.57	88.19 ± 0.45	88.54 ± 0.22	89.20 ± 0.44	90.14 ± 0.45
KSC	OA	85.19 ± 0.47	88.22 ± 0.38	90.28 ± 0.59	89.47 ± 0.28	87.57 ± 0.10	91.17 ± 0.22	92.37 ± 0.41	92.84 ± 0.38
	AA	86.22 ± 0.37	89.28 ± 0.14	91.14 ± 0.87	91.15 ± 0.33	88.14 ± 0.42	91.27 ± 0.42	93.66 ± 0.44	93.28 ± 0.43
	Kappa	85.17 ± 0.44	89.54 ± 0.22	90.89 ± 0.35	90.24 ± 0.17	88.23 ± 0.33	92.31 ± 0.34	92.19 ± 0.45	92.89 ± 0.55
UH	OA	60.39 ± 0.17	63.15 ± 0.27	64.19 ± 0.28	64.54 ± 0.66	61.47 ± 0.22	65.74 ± 0.43	67.27 ± 0.49	67.08 ± 0.67
	AA	58.27 ± 0.22	61.19 ± 0.44	62.27 ± 0.49	62.23 ± 0.33	60.32 ± 0.28	64.29 ± 0.57	66.70 ± 0.27	65.14 ± 0.27
	Kappa	62.37 ± 0.38	64.28 ± 0.35	63.38 ± 0.25	65.34 ± 0.17	63.26 ± 0.34	66.67 ± 0.70	67.04 ± 0.30	66.74 ± 0.36
Params		185.8k	555.5k	558.2k	1.1M	184.9k	553.2k	558.1k	1.1M

TABLE IX
COMPARISONS OF COMPUTATIONAL COSTS (FLOPS) AND MEMORY USES (MB) OVER THE FOUR DATASETS.

	CNN3D	ResNet	ContextNet	MS3DNet	ENL-FCN	DPyResNet	SSRN	G2C-Conv3DNet	G2C-MS3DNet	G2C-ContextNet	G2C-SSRN
Params	210.2K	21.9M	554.1K	269K	113.9K	22.3M	364.1K	209.3K	268.3K	553.4K	363.8K
GPU Memory (MB)	2.42	83.78	2.89	2.94	504.74	85.63	16.08	2.42	2.94	2.89	16.08
FLOPs ($\times 10^6$)	74.9	84.1	63.2	89.9	4683.9	85.1	150.1	74.2	89.1	62.1	149.2
Datasets	Computation Time (ms/sample)										
IP	12	15	14	2	22	17	91	11	2	14	89
KSC	1.3	1.6	2.9	0.5	3.7	1.5	2.5	1.2	0.5	2.8	2.4
UP	110	156	260	33	152	134	259	109	31	259	256
UH	5.9	6.4	12.9	3.5	12.5	7.4	11.8	5.7	3.5	12.8	11.6

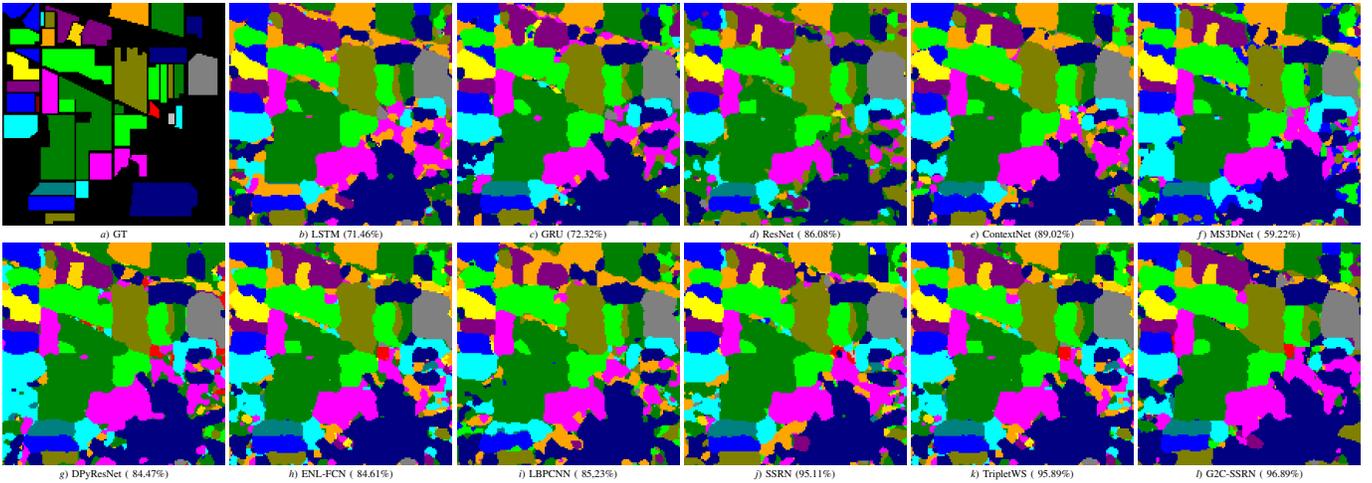


Fig. 15. Ground truth (a) and the classification maps predicted for IP dataset using: (b) LSTM, (c) GRU, (d) ResNet, (e) ContextNet, (f) MS3DNet, (g) DPyResNet, (h) ENL-FCN, (i) LBPCNN, (j) SSRN, (k) TripleTWS, and (l) G2C-SSRN models, using randomly selected 5% of training samples.

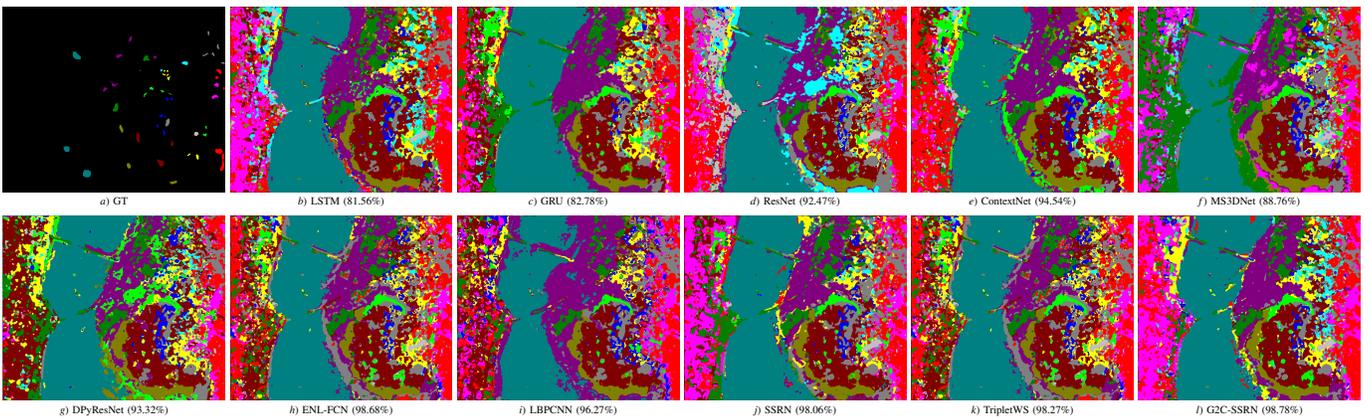


Fig. 16. Ground truth (a) and The classification maps predicted for KSC dataset using: (b) LSTM, (c) GRU, (d) ResNet, (e) ContextNet, (f) MS3DNet, (g) DPyResNet, (h) ENL-FCN, (i) LBPCNN, (j) SSRN, (k) TripleTWS, and (l) G2C-SSRN models, using randomly selected 5% of training samples.

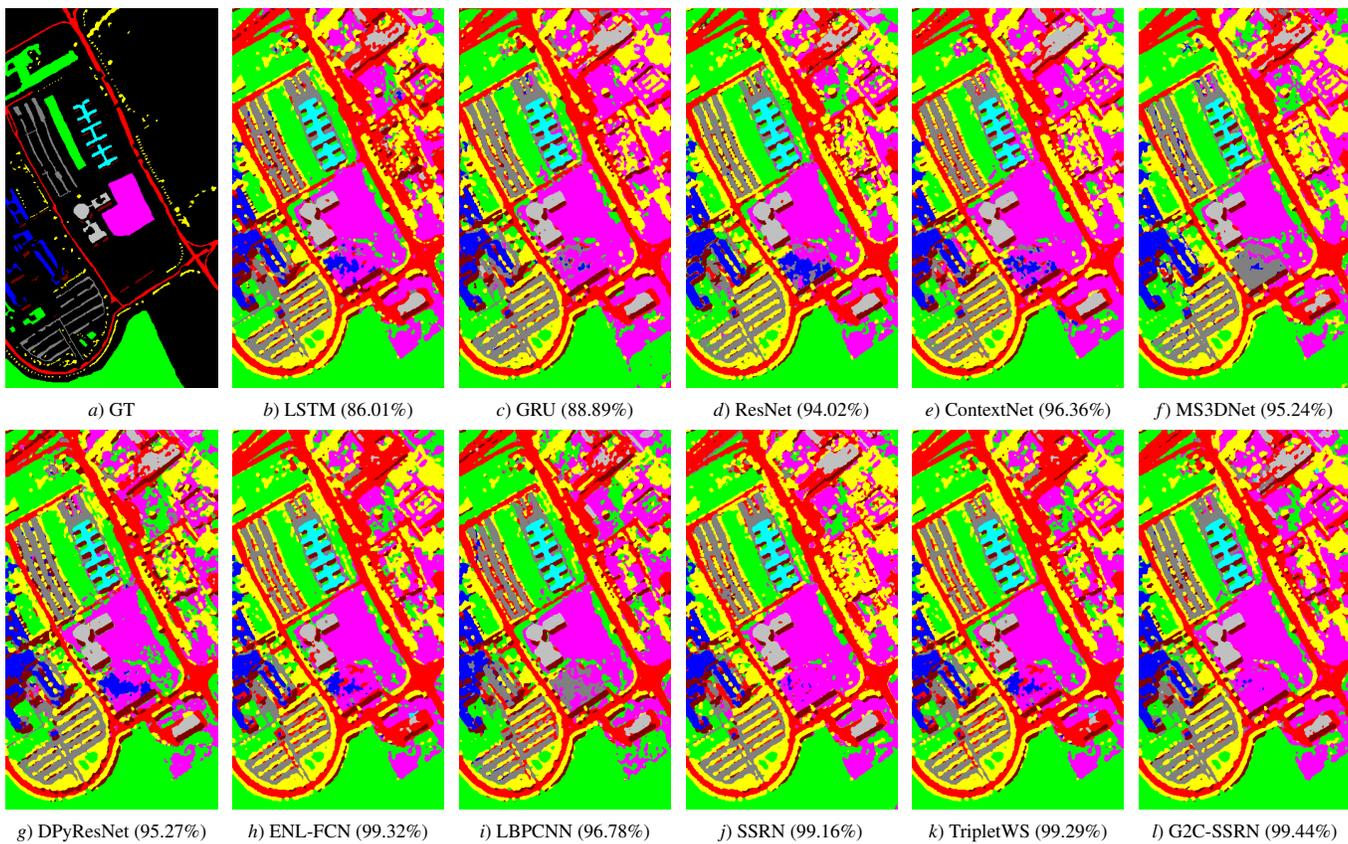


Fig. 17. Ground truth (a) and the classification maps predicted for UP dataset using: (b) LSTM, (c) GRU, (d) ResNet, (e) ContextNet, (f) MS3DNet, (g) DPyResNet, (h) ENL-FCN, (i) LBPCNN, (j) SSRN, (k) TripletWS, and (l) G2C-SSRN models, using randomly selected 5% of training samples.

results. The G2C-CAE2D achieves highest performance gain in terms of OA (2.38%) and Kappa (3.06%) respectively for KSC and AA (0.68%) for IP datasets over the model CAE2D. As compared with CAE3D, the G2C-CAE3D reaches classification performance gain in terms of OA (2.95%), AA (1.99%) and Kappa (2.77%) all for KSC dataset. Similarly, the gain in terms of OA (2.09%), AA (2.52%), and Kappa (1.3%) can be observed for KSC data using G2C-VAE3D over VAE3D. Finally, G2C-SAE3D shows the gain of 3.37%, 2.13%, and 2.65% quantitative results in terms of OA, AA and Kappa obtained for the KSC dataset when compared with SAE3D. The highest achieved OAs, AAs, and Kappa's for G2C-AE models are highlighted in bold in Table VIII. It's clear from the results table that the generalized gradient centralized autoencoder networks achieves significant performance gain over vanilla autoencoder networks and it's due to the encoding of fine-grained feature representation in the latent space during training. In classification the fine-grained detailed features help to accurately classify the pixels those having similar land-cover textures but different in category.

G. Analysis of Computational Cost and Memory Usage

Table-IX reports the number of weight parameters, GPU memory requirements, and the computational complexity of different models including generalized gradient centralized networks (i.e., G2C-CNN3DNet, G2C-ContextNet, G2C-MS3DNet, and G2C-SSRN) in terms of floating-point

operations (FLOPs). It is observed that ENL-FCN contains the least number of parameters while ResNet and DPyResNet use the maximum number of weights. It is worth to be mentioned that the proposed generalized gradient centralized networks use approximately the least number of parameters as compared to the vanilla networks. As shown in Table-IX, the computational cost can be represented by FLOPs ($\times 10^6$) where the proposed G2C-networks (i.e., G2C-CNN3DNet, G2C-ContextNet, G2C-MS3DNet, and G2C-SSRN) required 74.2, 62.1, 89.1, and 149.2 FLOPs operation, similarly, for the vanilla networks (i.e., CNN3D, ContextNet, MS3DNet, and SSRN) need 74.9, 63.2, 89.9, and 150.1, respectively. It is noticed that the proposed G2C-networks reduce the number of FLOPs by 0.7, 1.1, 0.8, and 0.9 when compared with CNN3D, MS3DNet, ContextNet, and SSRN. G2S-MS3DNet achieves maximum FLOPs reduction. In addition, Table IX, shows the per sample average training time over IP, KSC, UP and UH datasets using vanilla networks as well as G2C-networks. The G2C-Networks take minimum average training time of 1.2, 0.5, 2.8, and 2.4 ms per sample for KSC dataset. Both sets of models i.e, vanilla and G2C-networks require the same amount of GPU memory requirements. The number of trainable parameters for G2C-AEs are shown in last row of the Table VIII.

H. Visual Comparison on Predicted Maps

Another way to establish the power of pixel-wise classification methods by comparing clarities in the obtained classifica-

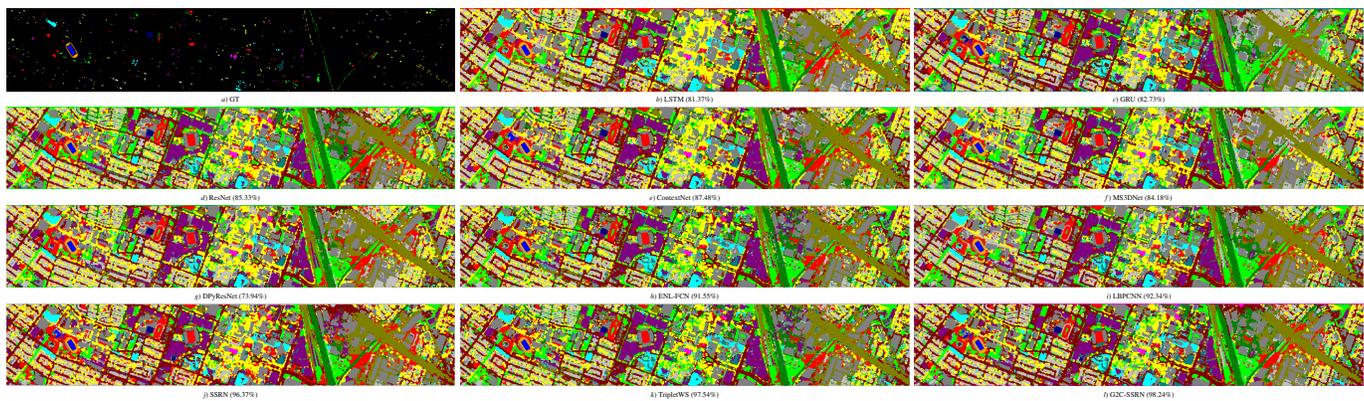


Fig. 18. Ground truth (a) and classification maps predicted for the UH dataset using: (b) LSTM, (c) GRU, (d) ResNet, (e) ContextNet, (f) MS3DNet, (g) DPYResNet, (h) ENL-FCN, (i) LBPCNN, (j) SSRN, (k) TripletWS, and (l) G2C-SSRN models, using randomly selected 5% of training samples.

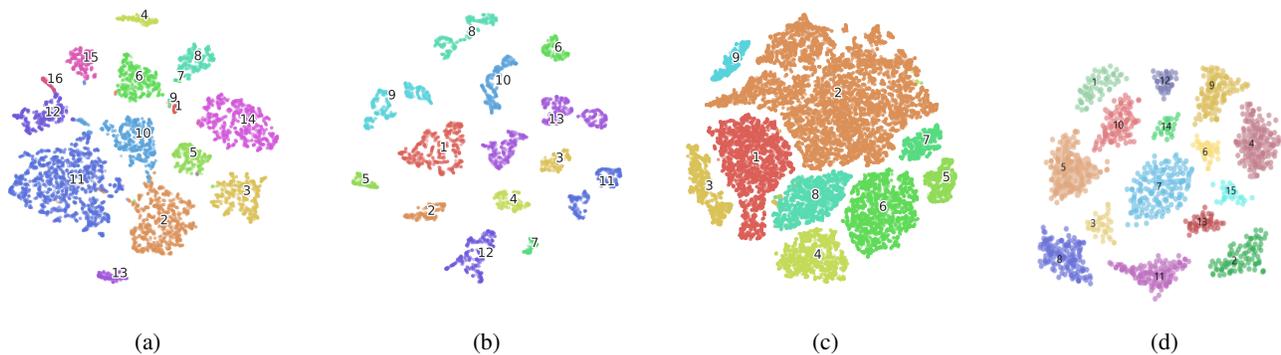


Fig. 19. Visualization of the G2C-SSRN 2-D spectral-spatial features for the test samples in different datasets via t-SNE. The points represent the features of test samples and their class labels are shown with different colors: (a) IP, (b) KSC, (c) UP, and (d) UH. (Best viewed in color)

tion maps. Figs. 15-18 show the obtained classification maps for the pixel-wise classification methods, i.e., MLP, RNN, LSTM, GRU, ResNet, ContextNet, MS3DNet, DPYResNet, ENL-FCN, LBPCNN, SSRN, TripletWS and G2C-SSRN using 5% of training samples randomly taken from IP, KSC, UP and UH datasets, respectively. It is observed that the predicted maps generated by classical algorithms generally contain “salt and paper” noise which appears due to the miss-classification of certain percentages of land-cover pixels surrounded by a spectrum-specially mixed within the local neighboring region. The salt and paper noise basically creates problems for the edges of two different land-cover even though the spectral feature can visually differentiate the different regions. On the contrary, this shortcoming can be overcome by learning spectral-spatial features during training and improve the prediction between the inner-areas of the different land-covers but problems still in the outputs due to not utilization of fine-grained details which are overlooked by the vanilla convolutions. The proposed G2C-Conv3D convolution extracts the fine-grained detailed features by locally preserving the morphological structures of the shape pattern and these features help G2C-SSRN to obtained good feature maps with fewer noise artifacts, more smooth boundary regions and can successfully remove miss-classified land-cover pixels from the inner land-cover regions as compared to other considered methods. It is also observed that the generated classification maps for SSRN, TripletWS and ENL-FCN are visually better

than ResNet, ContextNet, LBPCNN and MS3DNet in the sense that contains noise artifacts in some land-cover classes while the predicted maps of the G2C-SSRN are more accurate, smoother, and with better delineation of edges due to the more robust fine-grained representation of spectral-spatial features.

The representation of the learned feature for our trained model can be better visualized by projecting the extracted features into 2D space using the t-SNE method [79]. Figs. 19(a)-(d) show the t-SNE plots for IP, KSC, UP, and UH datasets. It can be seen that features from similar land-cover classes are clustered into a single group, similarly, the samples from different land-cover classes are easily separable from one another. So, we can make an inference that both proposed G2C-Conv2D and G2C-Conv3D can alleviate the existing models and focus to learn the abstract representation of fine-grained spectral-spatial details.

V. CONCLUSION

In this article, we first proposed a gradient centralized 3D convolution (GC-Conv3D) to extract gradient level details efficiently from 3D images during convolution operation which is often overlooked by vanilla convolution. One important advantage of GC-Conv3D is that it is capable to improve the model performance by extracting gradient level information. Furthermore, to boost the generalization capability of existing CNNs a weighted combination between the vanilla and gradient centralized 3D con-

volution is proposed to form generalized gradient centralized 3D convolution (G2C-Conv3D) for fine-grained feature extraction. Which is used to carefully design the generalized gradient centralized feature extraction networks (G2C-FE) i.e., G2C-SSRN, G2C-Conv2DNet, G2C-Conv3DNet, G2C-MS3DNet, and G2C-ContextNet for extracting of more robust and discriminative fine-grained feature representation than the vanilla 3D convolution in hyperspectral image classification task. In addition, the generalized gradient centralized autoencoder (G2C-AE) networks G2C-CAE2D, G2C-CAE3D, G2C-VAE3D, and G2C-SAE3D, respectively are also considered for the experimental study. In the experiments, the power of the G2C-Conv3D convolution is evaluated in the above discussed frameworks on four commonly used HSI datasets i.e., IP, KSC, UH, and UP. The results consistently demonstrate the advantages of G2C-Conv3D operation and the robustness of all the evaluated models, especially when a smaller number of samples are considered for training the networks. The G2C-SSRN model achieves good overall classification performance and outperforms the nine compared methods by a significant margin. Similarly, G2C-FE models also outperform its equivalent 2D and 3D vanilla networks in terms of OA, AA, and Kappa values. Most importantly, the proposed G2C-Conv3D convolution operation is simple to implement and it can easily be plugged into existing CNNs to boost the classification performance. We also believe the novel G2C-Conv3D convolution will introduce a new mileage in the hyperspectral research in the coming days.

ACKNOWLEDGMENT

Authors would like to thank Ganesan Narayanasamy who is leading IBM OpenPOWER/POWER enablement and ecosystem worldwide for his support to get the IBM AC922 system's access.

REFERENCES

- [1] K. Zheng, L. Gao, W. Liao, D. Hong, B. Zhang, X. Cui, and J. Chanussot, "Coupled convolutional neural network with adaptive response function learning for unsupervised hyperspectral super resolution," *IEEE Trans. Geosci. Remote Sens.*, vol. 59, no. 3, pp. 2487–2502, 2020.
- [2] S. Saha, F. Bovolo, and L. Bruzzone, "Building change detection in vhr sar images via unsupervised deep transcoding," *IEEE Trans. Geosci. Remote Sens.*, 2020.
- [3] M. Á. García-Sopo, A. Cuartero, P. G. Rodríguez, and A. Plaza, "Hyperspectral and lidar data integration and classification," in *Proc. IGARSS*. IEEE, 2015, pp. 57–60.
- [4] D. Hong, L. Gao, N. Yokoya, J. Yao, J. Chanussot, D. Qian, and B. Zhang, "More diverse means better: Multimodal deep learning meets remote-sensing imagery classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 59, no. 5, pp. 4340–4354, 2021.
- [5] D. Hong, W. He, N. Yokoya, J. Yao, L. Gao, L. Zhang, J. Chanussot, and X. X. Zhu, "Interpretable hyperspectral artificial intelligence: When non-convex modeling meets hyperspectral remote sensing," *IEEE Geosci. Remote Sens. Mag.*, vol. 9, no. 2, pp. 52–87, 2021.
- [6] J. M. Bioucas-Dias, A. Plaza, N. Dobigeon, M. Parente, Q. Du, P. Gader, and J. Chanussot, "Hyperspectral unmixing overview: Geometrical, statistical, and sparse regression-based approaches," *IEEE J. Sel. Top. Appl. Earth Obs. Remote Sens.*, vol. 5, no. 2, pp. 354–379, 2012.
- [7] D. Hong, N. Yokoya, J. Chanussot, and X. Zhu, "An augmented linear mixing model to address spectral variability for hyperspectral unmixing," *IEEE Trans. Image Process.*, vol. 28, no. 4, pp. 1923–1938, 2019.
- [8] P. Ghamisi, N. Yokoya, J. Li, W. Liao, S. Liu, J. Plaza, B. Rasti, and A. Plaza, "Advances in hyperspectral image and signal processing: A comprehensive overview of the state of the art," *IEEE Geosci. Remote Sens. Mag.*, vol. 5, no. 4, pp. 37–78, 2017.
- [9] Y. Li, Q. Li, Y. Liu, and W. Xie, "A spatial-spectral sift for hyperspectral image matching and classification," *Pattern Recognit. Lett.*, vol. XX, p. PP. 2018.
- [10] Y. Shao, N. Sang, C. Gao, and L. Ma, "Spatial and class structure regularized sparse representation graph for semi-supervised hyperspectral image classification," *Pattern Recognit.*, vol. 81, pp. 81–94, 2018.
- [11] B. Tu, N. Li, L. Fang, X. Yang, and J. Wu, "Hyperspectral image classification with a class-dependent spatial-spectral mixed metric," *Pattern Recognit. Lett.*, vol. 123, pp. 16–22, 2019.
- [12] H. Yu, L. Gao, W. Liao, B. Zhang, L. Zhuang, M. Song, and J. Chanussot, "Global spatial and local spectral similarity-based manifold learning group sparse representation for hyperspectral imagery classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 58, no. 5, pp. 3043–3056, 2019.
- [13] G.-B. Huang, Q.-Y. Zhu, and C.-K. Siew, "Extreme learning machine: theory and applications," *Neurocomputing*, vol. 70, no. 1-3, pp. 489–501, 2006.
- [14] X. Wei, X. Yu, B. Liu, and L. Zhi, "Convolutional neural networks and local binary patterns for hyperspectral image classification," *Eur. J. Remote Sens.*, vol. 52, no. 1, pp. 448–462, 2019.
- [15] R. M. Anwer, F. S. Khan, J. van de Weijer, M. Molinier, and J. Laaksonen, "Binary patterns encoded convolutional neural networks for texture recognition and remote sensing scene classification," *ISPRS J. Photogramm. Remote Sens.*, vol. 138, pp. 74–85, 2018.
- [16] J. A. Benediktsson, J. A. Palmason, and J. R. Sveinsson, "Classification of hyperspectral data from urban areas based on extended morphological profiles," *IEEE Trans. Geosci. Remote Sens.*, vol. 43, no. 3, pp. 480–491, 2005.
- [17] M. Fauvel, J. A. Benediktsson, J. Chanussot, and J. R. Sveinsson, "Spectral and spatial classification of hyperspectral data using svms and morphological profiles," *IEEE Trans. Geosci. Remote Sens.*, vol. 46, no. 11, pp. 3804–3814, 2008.
- [18] M. Dalla Mura, J. A. Benediktsson, B. Waske, and L. Bruzzone, "Morphological attribute profiles for the analysis of very high resolution images," *IEEE Trans. Geosci. Remote Sens.*, vol. 48, no. 10, pp. 3747–3762, 2010.
- [19] Y. Chen, Z. Lin, X. Zhao, G. Wang, and Y. Gu, "Deep learning-based classification of hyperspectral data," *IEEE J. Sel. Top. Appl. Earth Obs. Remote Sens.*, vol. 7, no. 6, pp. 2094–2107, 2014.
- [20] M. Pal, "Random forest classifier for remote sensing classification," *Int. J. Remote Sens.*, vol. 26, no. 1, pp. 217–222, 2005.
- [21] X. Sun, Y. Qu, L. Gao, X. Sun, H. Qi, B. Zhang, and T. Shen, "Target detection through tree-structured encoding for hyperspectral images," *IEEE Trans. Geosci. Remote Sens.*, vol. 59, no. 5, pp. 4233–4249, 2021.
- [22] B. Rasti, D. Hong, R. Hang, P. Ghamisi, X. Kang, J. Chanussot, and J. A. Benediktsson, "Feature extraction for hyperspectral imagery: The evolution from shallow to deep: Overview and toolbox," *IEEE Geosci. Remote Sens. Mag.*, vol. 8, no. 4, pp. 60–88, 2020.
- [23] C. Zhao, X. Wan, G. Zhao, B. Cui, W. Liu, and B. Qi, "Spectral-spatial classification of hyperspectral imagery based on stacked sparse autoencoder and random forest," *Eur. J. Remote Sens.*, vol. 50, no. 1, pp. 47–63, 2017.
- [24] T. Li, J. Zhang, and Y. Zhang, "Classification of hyperspectral image based on deep belief networks," in *Proc. ICIP*. IEEE, 2014, pp. 5132–5136.
- [25] C. Xing, L. Ma, and X. Yang, "Stacked denoise autoencoder based feature extraction and classification for hyperspectral images," *J. Sens.*, vol. 2016, 2016.
- [26] 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, 2017.
- [27] S. Yu, S. Jia, and C. Xu, "Convolutional neural networks for hyperspectral image classification," *Neurocomputing*, vol. 219, pp. 88–98, 2017.
- [28] F. Juefei-Xu, V. Naresh Boddeti, and M. Savvides, "Local binary convolutional neural networks," in *Proc. CVPR*, 2017, pp. 19–28.
- [29] S. Luan, C. Chen, B. Zhang, J. Han, and J. Liu, "Gabor convolutional networks," *IEEE Trans. Image Process.*, vol. 27, no. 9, pp. 4357–4366, 2018.
- [30] Z. Yu, C. Zhao, Z. Wang, Y. Qin, Z. Su, X. Li, F. Zhou, and G. Zhao, "Searching central difference convolutional networks for face anti-spoofing," in *Proc. CVPR*, 2020, pp. 5295–5305.
- [31] F. Yu and V. Koltun, "Multi-scale context aggregation by dilated convolutions," *arXiv preprint arXiv:1511.07122*, 2015.

- [32] J. Dai, H. Qi, Y. Xiong, Y. Li, G. Zhang, H. Hu, and Y. Wei, "Deformable convolutional networks," in *Proc. CVPR*, 2017, pp. 764–773.
- [33] K. Makantasis, K. Karantzas, A. Doulamis, and N. Doulamis, "Deep supervised learning for hyperspectral data classification through convolutional neural networks," in *Proc. IGARSS*. IEEE, 2015, pp. 4959–4962.
- [34] 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.
- [35] S. K. Roy, R. Mondal, M. E. Paoletti, J. M. Haut, and A. Plaza, "Morphological convolutional neural networks for hyperspectral image classification," *IEEE J. Sel. Top. Appl. Earth Obs. Remote Sens.*, vol. 14, pp. 8689–8702, 2021.
- [36] A. Challa, S. Danda, B. Sagar, and L. Najman, "Triplet-watershed for hyperspectral image classification," *IEEE Trans. Geosci. Remote Sens.*, 2021.
- [37] 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, 2017.
- [38] B. Liu, X. Yu, P. Zhang, X. Tan, R. Wang, and L. Zhi, "Spectral-spatial classification of hyperspectral image using three-dimensional convolution network," *J. Appl. Remote Sens.*, vol. 12, no. 1, p. 016005, 2018.
- [39] A. B. Hamida, A. Benoit, P. Lambert, and C. B. Amar, "3-d deep learning approach for remote sensing image classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 56, no. 8, pp. 4420–4434, 2018.
- [40] M. He, B. Li, and H. Chen, "Multi-scale 3d deep convolutional neural network for hyperspectral image classification," in *Proc. ICIP*. IEEE, 2017, pp. 3904–3908.
- [41] B. Fang, Y. Li, H. Zhang, and J. C.-W. Chan, "Collaborative learning of lightweight convolutional neural network and deep clustering for hyperspectral image semi-supervised classification with limited training samples," *ISPRS J. Photogramm. Remote Sens.*, vol. 161, pp. 164–178, 2020.
- [42] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proc. CVPR*, 2016, pp. 770–778.
- [43] M. E. Paoletti, J. M. Haut, R. Fernandez-Beltran, J. Plaza, A. J. Plaza, and F. Pla, "Deep pyramidal residual networks for spectral-spatial hyperspectral image classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 57, no. 2, pp. 740–754, 2018.
- [44] X. Ma, A. Fu, J. Wang, H. Wang, and B. Yin, "Hyperspectral image classification based on deep deconvolution network with skip architecture," *IEEE Trans. Geosci. Remote Sens.*, vol. 56, no. 8, pp. 4781–4791, 2018.
- [45] D. Hong, N. Yokoya, G.-S. Xia, J. Chanussot, and X. X. Zhu, "X-modalnet: A semi-supervised deep cross-modal network for classification of remote sensing data," *ISPRS J. Photogramm. Remote Sens.*, vol. 167, pp. 12–23, 2020.
- [46] R. Hang, Z. Li, P. Ghamisi, D. Hong, G. Xia, and Q. Liu, "Classification of hyperspectral and lidar data using coupled cnns," *IEEE Trans. Geosci. Remote Sens.*, vol. 58, no. 7, pp. 4939–4950, 2020.
- [47] D. Hong, J. Hu, J. Yao, J. Chanussot, and X. X. Zhu, "Multimodal remote sensing benchmark datasets for land cover classification with a shared and specific feature learning model," *ISPRS J. Photogramm. Remote Sens.*, vol. 178, pp. 68–80, 2021.
- [48] 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 Geosci. Remote Sens. Lett.*, vol. 17, no. 2, pp. 277–281, 2020.
- [49] S. K. Roy, S. Manna, T. Song, and L. Bruzzone, "Attention-based adaptive spectral-spatial kernel resnet for hyperspectral image classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 59, no. 9, pp. 7831–7843, 2021.
- [50] M. Zhu, L. Jiao, F. Liu, S. Yang, and J. Wang, "Residual spectral-spatial attention network for hyperspectral image classification," *IEEE Trans. Geosci. Remote Sens.*, 2020.
- [51] D. Hong, L. Gao, J. Yao, B. Zhang, A. Plaza, and J. Chanussot, "Graph convolutional networks for hyperspectral image classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 59, no. 7, pp. 5966–5978, 2021.
- [52] L. Zhu, Y. Chen, P. Ghamisi, and J. A. Benediktsson, "Generative adversarial networks for hyperspectral image classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 56, no. 9, pp. 5046–5063, 2018.
- [53] S. K. Roy, J. M. Haut, M. E. Paoletti, S. R. Dubey, and A. Plaza, "Generative adversarial minority oversampling for spectral-spatial hyperspectral image classification," *IEEE Trans. Geosci. Remote Sens.*, 2021.
- [54] T. Ojala, M. Pietikainen, and T. Maenpaa, "Multiresolution gray-scale and rotation invariant texture classification with local binary patterns," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 24, no. 7, pp. 971–987, 2002.
- [55] V. Nair and G. E. Hinton, "Rectified linear units improve restricted boltzmann machines," in *Proc. ICML*, 2010, pp. 807–814.
- [56] S. Ioffe and C. Szegedy, "Batch normalization: Accelerating deep network training by reducing internal covariate shift," in *Proc. ICML*, 2015, pp. 448–456.
- [57] M. Paoletti, J. Haut, J. Plaza, and A. Plaza, "Deep learning classifiers for hyperspectral imaging: A review," *ISPRS J. Photogramm. Remote Sens.*, vol. 158, pp. 279–317, 2019.
- [58] 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.
- [59] D. Hong, N. Yokoya, J. Chanussot, J. Xu, and X. X. Zhu, "Joint and progressive subspace analysis (jpsa) with spatial-spectral manifold alignment for semi-supervised hyperspectral dimensionality reduction," *IEEE Trans. Cybern.*, vol. 51, no. 7, pp. 3602–3615, 2021.
- [60] Z. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich, "Going deeper with convolutions," in *Proc. CVPR*, 2015, pp. 1–9.
- [61] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," in *Proc. NIPS*, 2012, pp. 1097–1105.
- [62] G. E. Hinton and R. R. Salakhutdinov, "Reducing the dimensionality of data with neural networks," *Science*, vol. 313, no. 5786, pp. 504–507, 2006.
- [63] D. Im Im, S. Ahn, R. Memisevic, and Y. Bengio, "Denoising criterion for variational auto-encoding framework," in *Proc. AAAI*, vol. 31, no. 1, 2017.
- [64] S. Kullback and R. A. Leibler, "On information and sufficiency," *Ann. Math. Stat.*, vol. 22, no. 1, pp. 79–86, 1951.
- [65] I. Higgins, L. Matthey, A. Pal, C. Burgess, X. Glorot, M. Botvinick, S. Mohamed, and A. Lerchner, "beta-vae: Learning basic visual concepts with a constrained variational framework," 2016.
- [66] X. Sun, F. Zhou, J. Dong, F. Gao, Q. Mu, and X. Wang, "Encoding spectral and spatial context information for hyperspectral image classification," *IEEE Geosci. Remote Sens. Lett.*, vol. 14, no. 12, pp. 2250–2254, 2017.
- [67] R. O. Green, M. L. Eastwood, C. M. Sarture, T. G. Chrien, M. Aronsson, B. J. Chippendale, J. A. Faust, B. E. Pavri, C. J. Chovit, M. Solis *et al.*, "Imaging spectroscopy and the airborne visible/infrared imaging spectrometer (aviris)," *Remote Sens. Environ.*, vol. 65, no. 3, pp. 227–248, 1998.
- [68] X. Xu, J. Li, and A. Plaza, "Fusion of hyperspectral and lidar data using morphological component analysis," in *Proc. IGARSS*. IEEE, 2016, pp. 3575–3578.
- [69] D. Hong, X. Wu, P. Ghamisi, J. Chanussot, N. Yokoya, and X. X. Zhu, "Invariant attribute profiles: A spatial-frequency joint feature extractor for hyperspectral image classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 58, no. 6, pp. 3791–3808, 2020.
- [70] M. Paoletti, J. Haut, J. Plaza, and A. Plaza, "Deep learning classifiers for hyperspectral imaging: A review," *ISPRS J. Photogramm. Remote Sens.*, vol. 158, pp. 279–317, 2019.
- [71] J. Ham, Y. Chen, M. M. Crawford, and J. Ghosh, "Investigation of the random forest framework for classification of hyperspectral data," *IEEE Trans. Geosci. Remote Sens.*, vol. 43, no. 3, pp. 492–501, 2005.
- [72] F. Melgani and L. Bruzzone, "Classification of hyperspectral remote sensing images with support vector machines," *IEEE Trans. Geosci. Remote Sens.*, vol. 42, no. 8, pp. 1778–1790, 2004.
- [73] K. Cho, B. Van Merriënboer, D. Bahdanau, and Y. Bengio, "On the properties of neural machine translation: Encoder-decoder approaches," *arXiv preprint arXiv:1409.1259*, 2014.
- [74] M. E. Paoletti, J. M. Haut, J. Plaza, and A. Plaza, "Scalable recurrent neural network for hyperspectral image classification," *J. Supercomput.*, vol. 76, no. 11, pp. 8866–8882, 2020.
- [75] He, Kaiming and Zhang, Xiangyu and Ren, Shaoqing and Sun, Jian, "Identity mappings in deep residual networks," in *Proc. ECCV*. Springer, 2016, pp. 630–645.
- [76] Y. Shen, S. Zhu, C. Chen, Q. Du, L. Xiao, J. Chen, and D. Pan, "Efficient deep learning of nonlocal features for hyperspectral image classification," *IEEE Trans. Geosci. Remote Sens.*, 2020.
- [77] Z. Xue, X. Yu, B. Liu, X. Tan, and X. Wei, "Hresnetam: Hierarchical residual network with attention mechanism for hyperspectral image classification," *IEEE J. Sel. Top. Appl. Earth Obs. Remote Sens.*, vol. 14, pp. 3566–3580, 2021.

- [78] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," *arXiv preprint arXiv:1412.6980*, 2014.
- [79] L. Van der Maaten and G. Hinton, "Visualizing data using t-sne." *J. Mach. Learn. Res.*, vol. 9, no. 11, 2008.



Swalpa Kumar Roy (S'15) received both the bachelor's and the master's degree in Computer Science and Engineering from West Bengal University of Technology, Kolkata, India, in 2012, and Indian Institute of Engineering Science and Technology, Shibpur, Howrah, India, (IEST Shibpur) in 2015. He is currently pursuing the Ph.D. degree jointly with Computer Vision and Pattern Recognition Unit, Indian Statistical Institute, Kolkata, and Department of Computer Science and Engineering, University of Calcutta, Kolkata. He was a Project Linked Person

with the Optical Character Recognition (OCR) Laboratory, Computer Vision and Pattern Recognition Unit, Indian Statistical Institute, Kolkata, from July 2015 to March 2016. He is currently working as an Assistant Professor with the Department of Computer Science and Engineering, Jalpaiguri Government Engineering College, Jalpaiguri, West Bengal, India. He was nominated for Indian National Academy of Engineering (INAE) engineering teachers mentoring fellowship by INAE Fellows in 2021 and also a recipient of the Outstanding Paper Award in second workshop: Hyperspectral Sensing Meets Machine Learning and Pattern Analysis (HyperMLPA) at the Workshop on Hyperspectral Imaging and Signal Processing: Evolution in Remote Sensing (WHISPERS) in 2021. He has served as a reviewer for the IEEE Transactions on Geoscience and Remote Sensing and IEEE Geoscience and Remote Sensing Letters. His research interests include computer vision, deep learning and remote sensing.



Purbayan Kar is a final year student of Jalpaiguri Government Engineering College, Jalpaiguri, India, pursuing B.Tech under the Department of Computer Science and Engineering. He is currently working as a research intern in Sony Research India and previously he holds the same position in Origin Health Singapore. His research interests include deep learning and remote sensing. He was nominated for Indian National Academy of Engineering (INAE) engineering students mentoring fellowship by INAE Fellows in 2021.



Danfeng Hong (S'16–M'19–SM'21) received the M.Sc. degree (summa cum laude) in computer vision from the College of Information Engineering, Qingdao University, Qingdao, China, in 2015, the Dr. -Ing degree (summa cum laude) from the Signal Processing in Earth Observation (SIPEO), Technical University of Munich (TUM), Munich, Germany, in 2019.

Since 2015, he has been a Research Associate at the Remote Sensing Technology Institute (IMF), German Aerospace Center (DLR), Oberpfaffenhofen, Germany. He is currently a Research Scientist and leads a Spectral Vision Working Group at IMF, DLR. He is also an Adjunct Scientist at GIPSA-lab, Grenoble INP, CNRS, Univ. Grenoble Alpes, Grenoble, France. His research interests include signal / image processing and analysis, hyperspectral remote sensing, machine / deep learning, artificial intelligence, and their applications in Earth Vision.

Dr. Hong is an Editorial Board Member of Remote Sensing and a Topical Associate Editor of the IEEE Transactions on Geoscience and Remote Sensing (TGRS). He was a recipient of the Best Reviewer Award of the IEEE TGRS in 2020 and the Jose Bioucas Dias award for recognizing the outstanding paper at the Workshop on Hyperspectral Imaging and Signal Processing: Evolution in Remote Sensing (WHISPERS) in 2021. He is also a Leading Guest Editor of the International Journal of Applied Earth Observation and Geoinformation, the IEEE Journal of Selected Topics in Applied Earth Observations, and Remote Sensing.



Xin Wu (S'19–M'20) received the M.Sc. degree in Computer Science and Technology from the College of Information Engineering, Qingdao University, Qingdao, China, in 2014, the Ph.D. degree from the School of Information and Electronics, Beijing Institute of Technology (BIT), Beijing, China, in 2020.

In 2018, she was a visiting student at the Photogrammetry and Image Analysis department of the Remote Sensing Technology Institute (IMF), German Aerospace Center (DLR), Oberpfaffenhofen, Germany. She is currently a Postdoctoral researcher in the School of Information and Electronics, BIT, Beijing, China. Her research interests include signal / image processing, fractional Fourier transform, deep learning and their applications in biometrics and geospatial object detection.

She was a recipient of the Jose Bioucas Dias award for recognizing the outstanding paper at the Workshop on Hyperspectral Imaging and Signal Processing: Evolution in Remote Sensing (WHISPERS) in 2021.



Antonio Plaza (M'02–SM'05–F'15) is a Full Professor and the Head of the Hyperspectral Computing Laboratory at the Department of Technology of Computers and Communications, University of Extremadura, where he received the M.Sc. degree in 1999 and the PhD degree in 2002, both in Computer Engineering. His main research interests comprise hyperspectral data processing and parallel computing of remote sensing data. He has authored more than 600 publications in this field, including 342 JCR journal papers (249 in IEEE journals), 24 book

chapters, and 330 peer-reviewed conference proceeding papers. He has guest edited 10 special issues on hyperspectral remote sensing for different journals. Prof. Plaza is a Fellow of IEEE "for contributions to hyperspectral data processing and parallel computing of Earth observation data" and a member of Academia Europaea, The Academy of Europe. He is a recipient of the recognition of Best Reviewers of the IEEE Geoscience and Remote Sensing Letters (in 2009) and a recipient of the recognition of Best Reviewers of the IEEE Transactions on Geoscience and Remote Sensing (in 2010), for which he served as Associate Editor in 2007-2012. He is also an Associate Editor for IEEE Access (receiving the recognition of Outstanding Associate Editor for the journal in 2017), and was also member of the Editorial Board of the IEEE Geoscience and Remote Sensing Newsletter (2011-2012) and the IEEE Geoscience and Remote Sensing Magazine (2013). He was also a member of the steering committee of the IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing (JSTARS). He is a recipient of the Best Column Award of the IEEE Signal Processing Magazine in 2015, the 2013 Best Paper Award of the JSTARS journal, and the most highly cited paper (2005-2010) in the Journal of Parallel and Distributed Computing. He received best paper awards at the IEEE Workshop on Hyperspectral Image and Signal Processing: Evolution in Remote Sensing, the IEEE International Conference on Space Technology, and the IEEE Symposium on Signal Processing and Information Technology. He served as the Director of Education Activities for the IEEE Geoscience and Remote Sensing Society (GRSS) in 2011-2012, and as President of the Spanish Chapter of IEEE GRSS in 2012-2016. He is currently serving as Chair of the Publications Awards Committee of IEEE GRSS and as a Vice-Chair of the Fellow Evaluations Committee of IEEE GRSS. He has reviewed more than 500 manuscripts for over 50 different journals. He served as the Editor-in-Chief of the IEEE Transactions on Geoscience and Remote Sensing journal for five years (2013-2017) and is currently serving as the Editor-in-Chief of the IEEE Journal on Miniaturization for air and Space Systems. He has been included in the 2018, 2019 and 2020 Highly Cited Researchers List (Clarivate Analytics). Additional information: <http://www.umbc.edu/rssi/pl/people/aplaza>



Jocelyn Chanussot (M'04–SM'04–F'12) received the M.Sc. degree in electrical engineering from the Grenoble Institute of Technology (Grenoble INP), Grenoble, France, in 1995, and the Ph.D. degree from the Université de Savoie, Annecy, France, in 1998. Since 1999, he has been with Grenoble INP, where he is currently a Professor of signal and image processing. His research interests include image analysis, hyperspectral remote sensing, data fusion, machine learning and artificial intelligence. He has been a visiting scholar at Stanford University (USA),

KTH (Sweden) and NUS (Singapore). Since 2013, he is an Adjunct Professor of the University of Iceland. In 2015-2017, he was a visiting professor at the University of California, Los Angeles (UCLA). He holds the AXA chair in remote sensing and is an Adjunct professor at the Chinese Academy of Sciences, Aerospace Information research Institute, Beijing.

Dr. Chanussot is the founding President of IEEE Geoscience and Remote Sensing French chapter (2007-2010) which received the 2010 IEEE GRSS Chapter Excellence Award. He has received multiple outstanding paper awards. He was the Vice-President of the IEEE Geoscience and Remote Sensing Society, in charge of meetings and symposia (2017-2019). He was the General Chair of the first IEEE GRSS Workshop on Hyperspectral Image and Signal Processing, Evolution in Remote sensing (WHISPERS). He was the Chair (2009-2011) and Cochair of the GRS Data Fusion Technical Committee (2005-2008). He was a member of the Machine Learning for Signal Processing Technical Committee of the IEEE Signal Processing Society (2006-2008) and the Program Chair of the IEEE International Workshop on Machine Learning for Signal Processing (2009). He is an Associate Editor for the IEEE Transactions on Geoscience and Remote Sensing, the IEEE Transactions on Image Processing and the Proceedings of the IEEE. He was the Editor-in-Chief of the IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing (2011-2015). In 2014 he served as a Guest Editor for the IEEE Signal Processing Magazine. He is a Fellow of the IEEE, a member of the Institut Universitaire de France (2012-2017) and a Highly Cited Researcher (Clarivate Analytics/Thomson Reuters).