



**HAL**  
open science

# Permuted Spectral and Permuted Spectral-Spatial CNN Models for PolSAR- Multispectral Data based Land Cover Classification

Gopal S. Phartiyal, Nicolas Brodu, Dharmendra Singh, Hussein Yahia, Khalid Daoudi

► **To cite this version:**

Gopal S. Phartiyal, Nicolas Brodu, Dharmendra Singh, Hussein Yahia, Khalid Daoudi. Permuted Spectral and Permuted Spectral-Spatial CNN Models for PolSAR- Multispectral Data based Land Cover Classification. International Journal of Remote Sensing, 2020, 42 (3), 10.1080/01431161.2020.1823041 . hal-02910884

**HAL Id: hal-02910884**

**<https://hal.science/hal-02910884>**

Submitted on 3 Aug 2020

**HAL** is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

# Permuted Spectral and Permuted Spectral-Spatial CNN Models for PolSAR-Multispectral Data based Land Cover Classification

Gopal Singh Phartiyal, Nicolas Brodu, Dharmendra Singh, Hussein Yahia, Khalid Daoudi

<sup>1,3</sup>*Department of Electronics and Communication Engineering, Indian Institute of Technology, Roorkee 247667, India*

<sup>2,4,5</sup>*GeoStat, INRIA, France*

*Abstract*—It is a challenge to develop methods which can process the PolSAR and multispectral (MS) data modalities together without losing information from either for remote sensing applications. This paper presents a study which attempts to introduce novel deep learning based remote sensing data processing frameworks that utilizes convolutional neural networks (CNNs) in both spatial and spectral domains to perform land cover (LC) classification with PolSAR-MS data. Also since earth observation remotely sensed data have usually larger spectral depth than normal camera image data, exploiting the spectral information in remote sensing (RS) data is crucial as well. In fact, convolutions in the sub-spectral space are intuitive and alternative to the process of feature selection. Recently, researchers have gained success in exploiting the spectral information of RS data, especially the hyperspectral data with CNNs. In this paper, exploitation of the spectral information in the PolSAR-MS data via a permuted localized spectral convolution along with localized spatial convolution is proposed. Further, the study in this paper also establishes the significance of performing permuted localized spectral convolutions over non-localized or localized spectral convolutions. Two models are proposed, namely a permuted local spectral convolutional network (*Perm-LS-CNN*) and a permuted local spectral-spatial convolutional network (*Perm-LSS-CNN*). These models are trained on ground truth class data points measured directly on the terrain. The evaluation of the generalization performance is done using ground truth knowledge on selected well known regions in the study areas. Comparison with other popular machine learning classifiers shows that the *Perm-LSS-CNN* model provides better classification results in terms of both accuracy and generalization.

*Index Terms*— CNNs, PolSAR-MS, spectral-spatial convolution, generalization, Perm-LSS-CNN.

## 1. Introduction

Today, both synthetic aperture radar (SAR) and multispectral (MS) remote sensing data are easily available with good spatial and temporal resolutions. The two data modalities are crucial for earth studies and complement each other in many applications such as land cover classification, crop monitoring, or change detection. In general, multispectral remote sensing data provides spectral reflectance information and PolSAR remote sensing data provides scattering and polarizing behaviour information about the earth surface objects. These earth surface objects are more commonly termed as land covers by the remote sensing community. The basic idea behind utilizing the two data modalities together is to acquire better spatial, spectral, and temporal information on the land covers. Although, higher spectral information can be achieved through hyperspectral remote sensing but at a cost of spatial and temporal information. Also, since it is an optical remote sensing technique, it suffers from the classical challenges (clouds). For applications such as land use land cover classification or land cover monitoring, all three aspects are crucial and a multi sensor approach is more beneficial. Using PolSAR and MS data modalities together is shown to be a convenient and successful strategy. Methodologies listed in many review articles such as in Pohl and Van Genderen (1998), or more recently in J. Zhang (2010) or Joshi et al. (2016) have been used by the earth observation data processing community in processing multispectral and PolSAR data together. Authors in Joshi et al. (2016) provide an extensive review on the application of PolSAR and MS data fusion for remote sensing applications. The reviewed methodologies include traditional, machine learning, and knowledge based decision trees. Authors in Joshi et al. (2016) also remarked that the studies in multi-sensor data fusion based land cover classification mostly use MS and PolSAR data with machine learning methods. Most popular machine learning-data fusion techniques include artificial neural networks (ANNs or NNs) and support vector machines (SVMs). In fact, studies (J. Zhang 2010; Joshi et al. 2016; Lary et al. 2016; and P. Feng et al. 2019) have also recognized the significance of machine learning in various multi-sensor remote sensing applications. From the aforementioned review, two points are evident; first, utilization of PolSAR and MS data together is a successful strategy and second, machine learning techniques are successful in utilizing the two data modalities together.

However, it is challenging for (NNs) with shallow depth (number of layers  $\leq 2$ ) to extract high level abstract/complex features from high dimensional remote sensing data without overfitting or losing generalization. Generalization refers to the ability of how well an algorithm performs on ‘out of the seen samples’ data points. If an algorithm is performing well outside training data, then the algorithm is showing good generalization performance and vice versa. Researchers in many studies (Zhu et al. 2017; W. Li et al. 2016; and L. Zhang, Zhang, and Du 2016) are exploring new advanced machine learning approaches for

remote sensing applications that are able to extract complex features from high dimensional remote sensing data. In the direction, deep neural networks with deep learning are the high-potential and powerful tools for extracting high level information from remote sensing data (Zhong, Hu, and Zhou 2019; and Huang, Zhao, and Song 2018). Authors in Zhu et al. (2017) and L. Zhang, Zhang, and Du (2016) have reviewed studies which used and demonstrated the significance of deep neural network frameworks for remote sensing applications. Convolutional deep neural networks (CNNs) are recently the most widely used deep network models in remote sensing data based land cover classification (Zhu et al. 2017). CNNs have the ability extract complex features directly from data (Yann LeCun, Yoshua Bengio, and Geoffrey Hinton 2015). Another advantage of focusing on developing CNNs for PolSAR-MS data processing is that CNNs can directly utilize multi-sensor high dimensional remote sensing data with efficient performance for remote sensing applications. Unlike the conventional data fusion methods, this reduces the need of fusing the PolSAR-MS data modalities together before using it further.

L. Zhang, Zhang, and Du (2016) presents a comparison of CNN based models with some popular remote sensing data classifiers for processing hyperspectral remote sensing data. The CNN model reported in L. Zhang, Zhang, and Du (2016) is designed to effectively capture global spectral features along with local spatial features. Similar studies are reported in Chen et al. (2016), Mei et al. (2016), and J. Feng et al. (2019). The CNN models in these studies employed one, two, and three dimensional convolutions to extract spectral and spatial information from unimodal remote sensing data. Moreover, CNNs have also been used with multisensor, for example PolSAR-MS data in remote sensing applications. Kussul et al. (2017) used one and two dimensional CNNs on Landsat8-Sentinell1 data for land cover and crop type classification and further compared the results with random forest (RF) classifier. Their model reports better classification performance by capturing local contextual (spectral and/or spatial) features whereas the RF classifier captured only the global dominant features. The study in Kussul et al. (2017) indicates potential of CNNs with PolSAR-MS data for remote sensing applications and motivates for similar studies..

However, one important aspect comes to consideration while processing PolSAR and MS data together using CNNs. The aspect is the efficient exploitation of the spectral information present in the multisensor data. Exploitation of spectral information of hyperspectral remote sensing data has been done in past, either exclusively with one dimensional CNNs or, along with the spatial information using two, or three dimensional CNNs. Many studies (Chen et al. 2014; Hu et al. 2015; Mei et al. 2016; He et al. 2016; and Yu, Jia, and Xu 2017) and more have used one dimensional CNNs as spectral feature extractors for hyperspectral remote sensing data classification. In contrast, extracting spectral information along with spatial information from hyperspectral remote sensing data using CNNs is a recently popular approach. The approach has however improved the results in land cover monitoring and classification applications.

Because hyperspectral remote sensing data have large spectral dimensions, extracting spectral information is computationally expensive and compounding with extraction of spatial information makes it harder. Therefore, some studies (Makantasis et al. 2015; Yue et al. 2015; Aptoula, Ozdemir, and Yanikoglu 2016; and L. Zhang, Zhang, and Du 2016) used principal component analysis (PCA) for feature dimensionality reduction and then used two dimensional CNNs for classification applications. While, alternate to this, some studies (H. Zhang et al. 2017; Yue, Mao, and Li 2016; and Zhao and Du 2016) used one and two dimensional CNNs separately to extract the spectral and spatial information and combined the separately extracted features later. Further, some studies used three dimensional CNNs to extract local spectral and spatial information simultaneously. Chen et al. (2016) and Ying Li, Zhang, and Shen (2017) used three dimensional CNNs as spectral-spatial feature extractors and classifiers for hyperspectral remote sensing data.

In summary, enough evidence suggests that; one, two, and three dimensional CNNs have been successfully used to extract spectral and/or spatial information from hyperspectral remote sensing data. However, the significance of spectral and/or spatial CNNs as land cover classifiers with high dimensional multisensor data is still interesting to investigate. Therefore, the study presented in this paper focusses on two aspects.

- To critically analyse the one dimensional (1D), two dimensional (2D), and three dimensional (3D) CNN capabilities to perform land cover classification using multi-sensor (PolSAR-MS) data.
- Development of novel classification frameworks based on 1D (spectral) or 3D (spectral-spatial) CNNs; in order to improve the generalization ability of the classifier during land cover classification with PolSAR-MS satellite data.

1D, 2D, and 3D CNN frameworks for PolSAR-MS data based land cover classification are discussed in section 4. The idea of extracting the spectral information efficiently from PolSAR-MS data using CNNs is achieved by performing a permuted localized spectral convolution alone, or along with localized spatial convolution on PolSAR-MS data. Spectral permutation before localized spectral convolution is proposed to increase the generalization ability of the CNN classifiers. Based on this idea, two models are proposed, namely permuted local spectral CNN (*Perm-LS-CNN*) and permuted local spectral-spatial CNN (*Perm-LSS-CNN*). Both the models are discussed in section 5.

The remaining of the paper is organized as follows; Section 2 provides a brief background on PolSAR features and CNNs utilized during the study. Section 3 provides information on the experimental

setup i.e. study areas, datasets, ground truth, and other resources considered and used during the study. Section 4 discusses the popular 1D, 2D, and the 3D CNN models used for remote sensing data processing so far and further presents and discusses 1D, 2D, and 3D CNN model configurations used in this study for PolSAR-MS data processing. Section 5 presents the novel CNN based models developed during the study for PolSAR-MS data processing for land cover classification. Results from the proposed and compared methods are discussed in section 6. Finally, section 7 concludes the paper.

## 2. Theoretical Background

### 2.1. Polarization Signatures

Polarization signature (PS) is a three dimensional graphical representation of backscattering behaviour of a target, or in the case of current study, a land cover. In PS representation, x-axis and y-axis represent ellipticity angle and orientation angle respectively and z-axis represent received backscattered power coefficient (sigma naught). Orientation or tilt angle ( $\psi$ ) varies from  $-90^0$  to  $90^0$  and ellipticity angle ( $\chi$ ) varies from  $-45^0$  to  $+45^0$ . The signatures are computed using equation (1).

$$\sigma(\chi_i\psi_i\chi_j\psi_j) = \frac{\pi\pi}{k} \begin{pmatrix} 1 \\ \cos\chi_i \cos\psi_i \\ \cos\chi_i \sin\psi_i \\ \sin\chi_i \end{pmatrix} (\mathbf{K}) \begin{pmatrix} 1 \\ \cos\chi_j \cos\psi_j \\ \cos\chi_j \sin\psi_j \\ \sin\chi_j \end{pmatrix} \quad (1)$$

Where  $\sigma$  represents backscattering coefficient, suffix  $i$  and  $j$  represent transmit and received combinations respectively.  $\mathbf{K}$  is the Kennaugh matrix, which provides received power, given the polarization characteristics of the receiving antenna (Antenna Laboratory, 1952). It is also known as the Stokes scattering matrix or the Stokes scattering operator. Computation of  $\mathbf{K}$  parameters is done similar to (Harold 2007).  $k$  is the propagation constant. Co-polarized signature is obtained by transmit and received combination of  $\psi_i = \psi_j$  and  $\chi_i = \chi_j$ , and cross polarized signature of  $\psi_i = 90 + \psi_j$ , and  $\chi_i = -\chi_j$ . Ellipticity angle defines the polarization behavior (linear, circular or elliptical polarization), while orientation or tilt angle defines polarization states, for example, horizontal or vertical polarization (Jong-Sen Lee, Mitchell R. Grunes, and Wolfgang-Martin Boerner; 1997). For the sake of visualization, co-polarized PS of an arbitrary land cover is shown in Figure 1. H and V basis have been considered and the co-polarized signatures are considered in the current study.

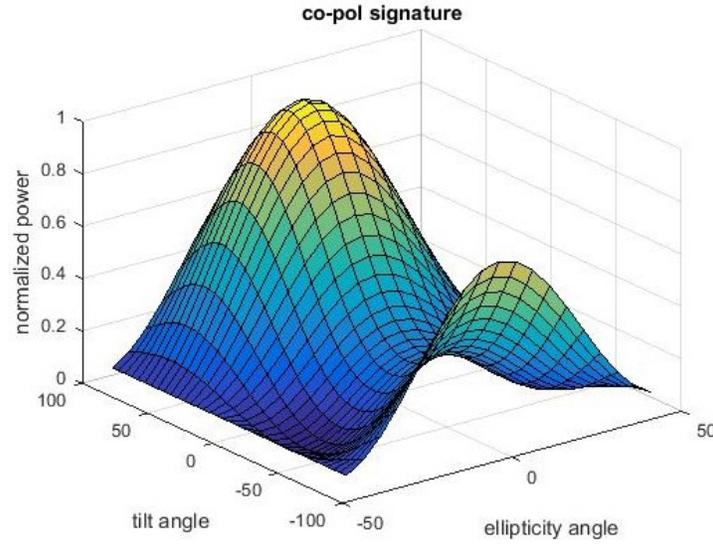


Figure 1 Co-polarized polarization signature of an arbitrary land cover

## 2.2. Convolutional Neural Networks

CNNs are a popular class of deep feedforward artificial neural networks proposed in various studies (Ian Goodfellow, Bengio, and Courville 2017b; Ciresan, Meier, and Schmidhuber 2012; Fukushima and Miyake 1982; LeCun et al. 1998; and W. Zhang et al. 1991). Building blocks of a conventional CNN are; a convolutional layer, a pooling layer, a dropout layer, a fully connected layer, and a loss layer. A generic arrangement of the layers is discussed in Ian Goodfellow, Bengio, and Courville (2017a) and Albelwi and Mahmood (2017).

Imagine a CNN with  $M$  layers, the output state vector of the  $m^{th}$  layer is denoted by  $\mathbf{O}^m$ , where  $m \in \{1, 2, \dots, M\}$  and,  $\mathbf{O}^0$  represents the input data. The input layer is then connected to a convolutional layer. In the convolutional layer, a 1D, 2D or 3D convolution is performed with kernels. For the sake of understanding, a 2D convolution is explained here. The matrix  $\mathbf{W}_1$  encodes the weights of the links connecting the first convolutional layer to the input layer. A vector  $\mathbf{b}_1$  which denotes a bias term is also added. Subsequently, a non-linear pointwise activation operation  $g(\cdot)$  is performed. Finally, a pooling layer is used which helps in selection of dominant features. The convolution plus pooling in the  $m^{th}$  layer is represented in equation (2).

$$\mathbf{O}^m = pool(g(\mathbf{O}^{m-1} \otimes \mathbf{W}_m + \mathbf{b}_m)) \quad (2)$$

Where,  $\otimes$  indicates convolution operation and  $pool$  denotes a spatial aggregation, as defined in Boureau et al. (2010). Several convolutional and pooling layers can be stacked to form a hierarchical

feature extraction architecture. The resultant features are then combined to form a one-dimensional feature vector to be taken as input by the fully connected layer. The fully connected layer acts as a typical multi perceptron layer. The last layer is usually a *linear* (for regression scenarios) or *softmax* (for classification scenarios) layer, with the number of neurons matching the number of output variables or classes desired. This layer also specifies how training penalizes the deviation between the predicted and true labels and is normally the final layer.

The weights,  $\{\mathbf{W}_1, \mathbf{W}_2, \dots, \mathbf{W}_M\}$  and the biases,  $\{b_1, b_2, \dots, b_M\}$  of the CNN compose the model parameters, which are iteratively and jointly optimized via overall accuracy maximization over the training set. Similar networks can be realized for 1D or 3D CNNs accordingly. However, the architectures of 1D, 2D, and 3D CNNs for remote sensing data processing are as follows.

- For 1D CNNs, the remote sensing data input is a vector (usually a vector of spectral features).  $\mathbf{W}_m$  is a matrix and  $b_m$  is vector. In general, 1D CNNs are used as feature extractors in remote sensing data, especially hyperspectral data processing.
- For 2D CNNs, the convolution is performed on the two spatial dimensions (e.g. images). The input may contain extra dimensions (such as spectral information), but these are not used in the convolution.  $\mathbf{W}_m$  is a 3D matrix and  $b_m$  is vector. It is important to note that in 2D CNNs, local convolutions are only performed in the spatial dimensions and not in the spectral dimension. 2D CNNs are used as spatial feature extractors and classifiers.
- In 3D CNNs, convolutions are performed in the spatial dimensions and the temporal dimension simultaneously. However, in the absence of temporal information and presence of high spectral information, spectral dimension replaces the temporal dimension and local spectral convolutions are possible.  $\mathbf{W}_m$  is a 4D matrix and  $b_m$  is vector. Therefore, 3D CNNs are used as spatial temporal or spatial spectral feature extractors and classifiers in remote sensing data processing.

The hyper parameters to be set during the CNN architecture design include; the number and shape of convolution filters to be used in the convolutional layer, stride and dilation values, and the pooling window shape. Regularization methods, preventing the CNN from overfitting, are also used. Among many, Dropout is the most popular regularization method (Srivastava et al. 2014). Dropout is a technique where randomly selected neurons are ignored during training. This means that their contribution to the activation of downstream neurons is temporally removed on the forward pass and any weight updates are not applied

to the neuron on the backward pass. Other methods include Drop Connect, stochastic pooling, early stopping, and weight decay.

The current study utilizes and investigates the ability of 1D, 2D, and 3D CNNs as land cover classifiers on PolSAR-MS data. Also, the current study proposes new frameworks based on these CNNs.

### 3. Experiment Setup

#### 3.1. Study area

Two areas are considered for the analysis, the development and the testing of the proposed methodology. The first study area includes Roorkee city at the foothills of the great Himalayas of the northern India and its neighbouring region. The area extends from 29.951 degrees North and 77.865 degrees East to 29.803 degrees North and 78.068 degrees East. The area covers about 361 km<sup>2</sup> of landscape. Another study area includes Haridwar city of the northern India and neighbouring region. The area extends from 29.973 degrees North and 77.995 degrees East to 29.823 degrees North and 78.168 degrees East. The area covers about 270 km<sup>2</sup> of landscape. Both study areas have mixed class type landscape, which is beneficial for the evaluation of classifiers performance in mixed class scenarios.

#### 3.2. Dataset

Multisensor dataset is used in the study i.e. multispectral and PolSAR data acquired over the study areas. Multispectral data used is the Landsat-8 multispectral data acquired on March 17, 2015. PolSAR data used is the ALOS PALSAR-2 L band fully polarimetric single look complex PolSAR data acquired on March 13, 2015.

Preprocessing is required with both the multispectral and the PolSAR raw data. The landsat-8 multispectral data suffers from atmospheric perturbations whereas the PolSAR data suffers from speckle noise. Preprocessing is done separately on the multispectral and the PolSAR data. PALSAR-2 single look complex PolSAR data is calibrated, multilooked, speckle filtered, and terrain corrected to obtain backscattering (sigma naught) coefficients in a procedure similar to P Mishra, Singh, and Yamaguchi (2011). Gamma MAP polarimetric speckle filter with a window size of  $3 \times 3$  is used for speckle filtering similar to Pooja Mishra, Garg, and Singh (2017). Four features namely  $HH$ ,  $HV$ ,  $VH$ , and  $VV$  are obtained. Since,  $HV$  and  $VH$  are similar due to the reciprocity theorem, only  $HH$ ,  $HV$ , and  $VV$  are used further. A false colour composite using the selected features is displayed in Figure 2(a) for visualization purpose. Derived PolSAR features are also used in this study. Polarization signatures (PSs) are computed from the

single look complex PolSAR data with procedure similar to Phartiyal et al. (2017). Further, polarization signatures correlation features (PSCFs) are computed from the PSs using procedure similar to as explained in Phartiyal, Kumar, and Singh (2020). PSCFs provide the degree of correlation between canonical/standard target PSs and observed/pixel PSs. Based on the analysis of Phartiyal, Kumar, and Singh (2020), four PSCFs namely “*corr\_co\_Di*”, “*corr\_co\_FP*”, “*corr\_co\_HD*”, and “*corr\_co\_VD*” are used as derived PolSAR features in the current study. Overall, three basic (*HH*, *HV*, and *VV*) and four derived (*corr\_co\_Di*, *corr\_co\_FP*, *corr\_co\_HD*, and *corr\_co\_VD*) PolSAR features are used in the current study. Landsat 8 operational land imager (OLI) multispectral data is atmospherically corrected first. Out of the 11 bands, 6 bands namely *blue*, *green*, *red*, *near infrared*, *short wave infrared 1*, and *short wave infrared 2* are selected for the study. These selected bands are pan sharpened as explained in Gillespie, Kahle, and Walker (1987) and then resampled to the PALSAR-2 PolSAR data spatial resolution of 6 meters using the “bicubic spline” interpolation method (Rajput, Ghosh, and Kumar 2014). A false colour composite using the selected multispectral bands is displayed in Figure 2(b) for visualization purpose. The spatial resolutions of the PolSAR and MS datasets are used judiciously via CNNs later in the study towards improvement of the land cover classification. The re-sampled, pre-processed multispectral bands are then co-registered with the selected pre-processed PolSAR features and the co-registered 13 bands (six from Landsat-8 and seven from PALSAR-2 data) are termed as PolSAR-MS data further in the study and are used as input data for further analysis and study. The flowchart shown in Figure 3 depicts the experiment setup.

### **3.3. Ground Truth**

Five land cover classes are considered namely bare soil (BS), tall vegetation (TV), short vegetation (SV), urban, and water. Ground truth class data points are measured directly on the terrain based on visual inspection. Approximately 1200 data points are collected for both the study areas. Details of the ground truth collected for both study areas are shown in Table 1.

The experiment setup i.e. the study areas, the datasets, or the ground truth data points, is used for, first, a critical analysis of the performance of conventional CNN frameworks and, second, the development of novel CNN classification frameworks for land cover classification.

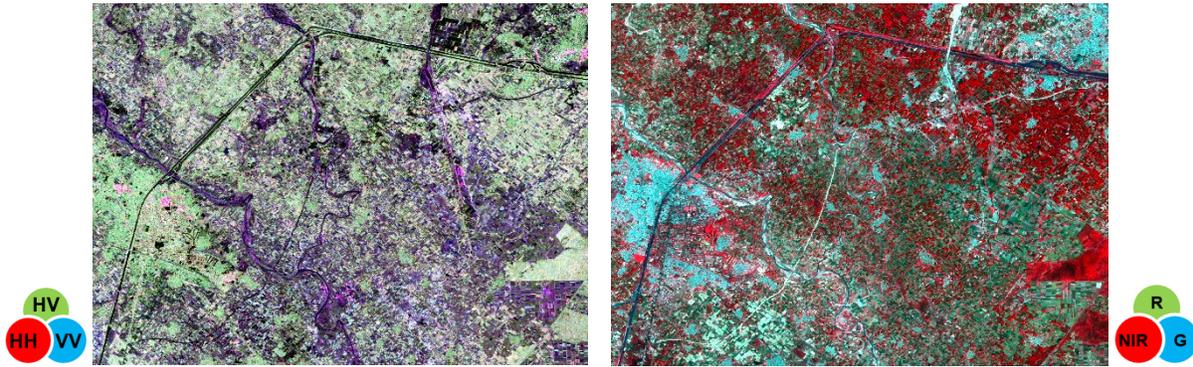


Figure 2 Datasets for Roorkee study area: (a). False colour composite of ALOS PALSAR-2 PolSAR data and, (b). False colour composite of Lansat-8 multispectral data

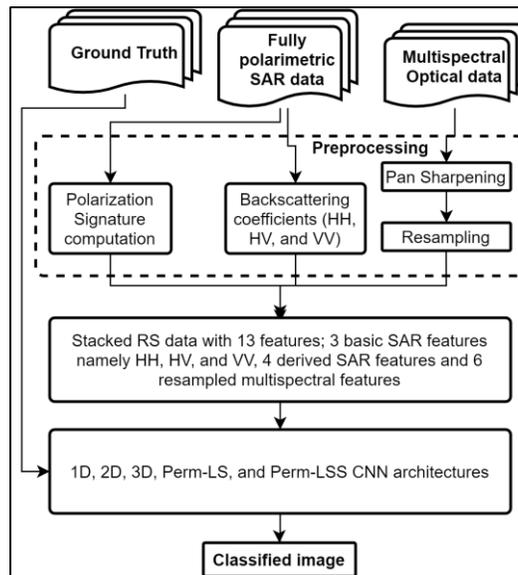


Figure 3 Flowchart of the CNNs based land cover classification methodology with PolSAR-MS data

Table 1 Summary of the ground truth collected for study. BS = Bare Soil, TV = Tall Vegetation, SV = Short Vegetation. All points are acquired directly on the field by visual inspection at selected GPS coordinates.

Land Cover	Roorkee		Haridwar	
	Training and validation	Testing	Training and validation	Testing
BS	200	40	200	40
TV	200	40	200	40
SV	200	40	200	40
Urban	200	40	200	40
Water	200	40	200	40
Total	1000	200	1000	200

## 4. 1D, 2D, and 3D CNNs for Land Cover Classification with PolSAR-MS Data

### 4.1. One dimensional CNNs for PolSAR-MS data.

One dimensional or 1D CNNs have been successfully used with unimodal remote sensing data in the recent years. Chen et al. (2014) used stacked auto encoders for classification with hyperspectral remote sensing data. They also used PCA for hyperspectral feature reduction before classification as the number of input features is very high (say 176 for KSC dataset). A range i.e. from 4 to 8 principle components are considered as features during classification. PolSAR-MS data used in the present study only have 13 spectral bands and do not require feature reduction. Hu et al. (2015) used different 1D CNNs for two popular hyperspectral remote sensing data (the Salimas Valley and the Indian Pines dataset) classification. Mei et al. (2016) also used 1D CNN for hyperspectral remote sensing data classification. All these studies report that 1D CNNs are good feature extractors for hyperspectral remote sensing data. Most of these studies used some form of feature reduction technique before feeding data to the 1D CNN in order to reduce the computational complexity of the model. The reduced number of bands are on the same order as the number of bands present in the PolSAR-MS data considered in this study, which justifies the direct use of 1D CNNs for PolSAR-MS remote sensing data.

Two variants of 1D CNNs are considered here. In the first 1D CNN model i.e. *1D-CNN-v1*, the kernel size of the filter in the first convolutional layer i.e. *C1* layer is equal to the spectral depth of PolSAR-MS data i.e. 13. One dimensional spectral convolution is shown in Figure 4(b). In the second 1D CNN model, i.e. *1D-CNN-v2*, the kernel size is less than the spectral depth of PolSAR-MS data. One dimensional local spectral convolution is visualized in Figure 4(c). Models similar to *1D-CNN-v2* has been successfully tested with hyperspectral data (Hu et al. 2015, and Chen et al. 2016). The present study focuses on the analysis of the performance of 1D CNNs; both *1D-CNN-v1* and *1D-CNN-v2* with PolSAR-MS data.

The selected hyper-parameter settings are listed in Table 2. Apart from kernel size, the rest of the parameters are similar for both models. Rectified linear unit or *ReLU* and *Softmax* activation functions are used in the intermediate and output layers respectively. A dropout strategy is considered for regularization and dropout fraction values of 0.2 for *C1* and 0.1 for *C2* and *FC* layer are set. Pooling is not considered because it hinders the full usability of the spectral variation. The loss function used in both models is the *categorical cross entropy* function. It optimizes the classification accuracy based on probabilities of each class during classification. The learning rate and the number of epochs are set to 0.01 and 200 respectively.

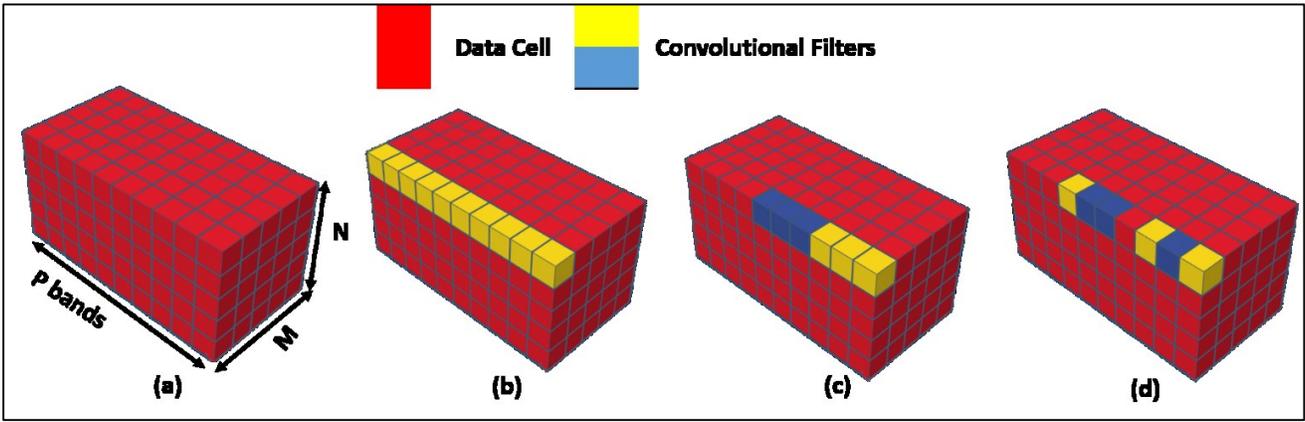


Figure 4 (a). Data cuboid with M by N pixels and P bands. (b). Convolution with  $1 \times 1 \times P$  convolutional filter where  $1 \times 1$  represents to the spatial extent and P represents total number of bands in the data. Each filter convolves with all the bands as all times. This strategy is employed in 1D CNNs during per pixel classification of remote sensing data. (c). Convolution with  $1 \times 1 \times 3$  convolutional filters where  $1 \times 1$  represents the spatial extent and 3 represents the number of adjacent bands to be convolved simultaneously. One filter can convolve with three adjacent bands only. For example, a first filter (in yellow) convolves with bands 1, 2, and 3 whereas a second filter (in blue) convolves with bands 4, 5, and 6. This strategy is also employed in 1D CNNs (Chen et al. 2016). (d). Convolution with  $1 \times 1 \times 3$  convolutional filters where  $1 \times 1$  represents to the spatial extent and 3 represents number of arbitrarily selected bands to be involved. For example, first filter (in yellow) convolves with bands 1, 3, and 7 whereas second filter (in blue) convolves with bands 2, 5, and 6. This strategy is employed in the proposed Perm-LS-CNN framework.

Table 2 *1D-CNN-v1*, *1D-CNN-v2*, *2D-CNN-v1* and *3D-CNN-v1* CNN model configurations, hyperparameter and training settings

Model	Input	Layer			
		Convolutional layer 1 (C1)	Convolutional layer 2 (C2)	Fully connected layer (FC)	Output layer
<i>1D-CNN-v1</i>	Input data size = $1 \times 13$ Normalization = Batch Normalization	Filters = 20 Kernel size = $1 \times 13$ Activation = ReLU Dropout fraction = 0.2 Pooling size = 0	Filters = 20 Kernel size = $1 \times 1$ Activation = ReLU Dropout fraction = 0.1 Pooling size = 0	Nodes = 16 Activation = ReLU Dropout fraction = 0.1	Labels = 5 Activation = Softmax
<i>1D-CNN-v2</i>	Input data size = $1 \times 13$ Normalization = Batch	Filters = 20 Kernel size = $1 \times 4$ Activation = ReLU Dropout fraction =	Filters = 20 Kernel size = $1 \times 4$ Activation =		

	Normalization	0.2	ReLU
		Pooling size = 0	Dropout fraction = 0.1
<i>2D-CNN-v1</i>	Input data size = $3 \times 3 \times 13$	Filters = 20	Pooling size = 0
	Normalization = Batch Normalization	Kernel size = $3 \times 3$	Filters = 20
		Activation = ReLU	Kernel size = $1 \times 1$
		Dropout fraction = 0.2	Activation = ReLU
		Pooling size = 0	Dropout fraction = 0.1
			Pooling size = 0
<i>3D-CNN-v1</i>	Input data size = $3 \times 3 \times 13$	Filters = 20	Filters = 20
	Normalization = Batch Normalization	Kernel size = $3 \times 3 \times 4$	Kernel size = $1 \times 1 \times 4$
		Activation = ReLU	Activation = ReLU
		Dropout fraction = 0.2	Dropout fraction = 0.1
		Pooling size = 0	Pooling size = 0
Training parameters	Loss function = Categorical cross entropy		
	Learning rate = 0.01		
	Number of epochs = 200		
	Optimizer = Adam (Kingma and Ba 2015)		
	10 fold cross validation strategy is employed during training		

---

#### 4.2. Two Dimensional CNN for PolSAR-MS data.

The use of spectral information without spatial information by 1D CNNs undermines the full potential of remote sensing data. Adding spatial information provides contextual aid to the classifiers, especially CNNs in the classification process. Two dimensional or 2D CNNs have also been successfully used with unimodal remote sensing data in the recent years. Sharma et al. (2017) used patch based 2D CNN model for land cover classification with multispectral data. They opted a  $5 \times 5 \times 8$  Landsat 8 image patch and a  $3 \times 3$  2D convolutional kernel. Geng et al. (2015) used 2D CNNs for land cover classification with PolSAR data. Many studies (Makantasis et al. 2015; Yue et al. 2015; Zhao et al. 2015; Zhao and Du 2016; and Yunsong Li, Xie, and Li 2017) used 2D CNNs for classification with hyperspectral remote sensing data. The need for dimension reduction in these studies is already discussed in the previous section. These studies support the utilization of 2D CNNs with PolSAR-MS data for land cover classification. Kussul et al. (2017) used 2D CNNs for crop type classification with PolSAR and multispectral data. They used Landsat-8 multispectral and Sentinel-1 C-Band PolSAR data. Although Kussul et al. (2017) have studied the PolSAR-MS data potential for crop type classification with 2D CNNs, the current study differs on: – first, using PolSAR-MS on a single time stamp with a focus on spectral and spatial features. Kussul et al.

(2017) used time series PolSAR-MS data with a focus on temporal and spectral features; and – second, Quad PolSAR data is considered in the present study in contrast to dual PolSAR used in Kussul et al. (2017). The increased channels add complementary information which in turn add an indirect weighting of the spatial features. Analysis of fully polarimetric PolSAR with multispectral data for classification is therefore beneficial.

A 2D CNN model namely *2D-CNN-v1* is proposed here. *2D-CNN-v1* configuration and hyperparameter settings considered are listed in Table 2. The idea of a local spatial convolution is shown in Figure 5(b). The configuration for the intermediate and output layers and learning parameters are the same as in section 4.1 and are summarized in Table 2.

### ***4.3. Three Dimensional CNN for PolSAR-MS data.***

The *2D-CNN-v1* model discussed in section 4.2 is successful because of its spatial contextual awareness. The localized spatial convolutions provide local connectivity, which controls the impact of neighbouring pixels on the pixel to classify. Parameter sharing between local filters also makes the learning faster and less cumbersome, in contrast to fully connected neural networks. In remote sensing data applications, spectral information is as important, if not more, as spatial information. Local connectivity in spectral dimension also provides the same benefits. Advantages of local spectral convolutions and local spatial convolutions together can be exploited using 3D CNNs. The 3D CNNs have been successfully used with hyperspectral (unimodal) remote sensing data in the recent years. Chen et al. (2016), H. Zhang et al. (2017), and J. Feng et al. (2019) used 3D CNNs for spectral spatial information extraction for fusion and classification applications.

The proposed 3D CNN model *3D-CNN-v1* architecture and hyperparameters considered are listed in Table 2. The *CI* layer of *2D-CNN-v1* in section 4.2 uses the complete spectral depth, i.e. 13 bands, during convolutions whereas *3D-CNN-v1* uses only a subset during convolutions. The idea of local spectral and local spatial convolution is shown in Figure 5(c). This provides an improvement in classification accuracy and robustness compared to other 3D CNN models (Chen et al. 2016; and J. Feng et al. 2019). Configuration and learning parameters are the same as in section 4.1 and are summarized in Table 2.

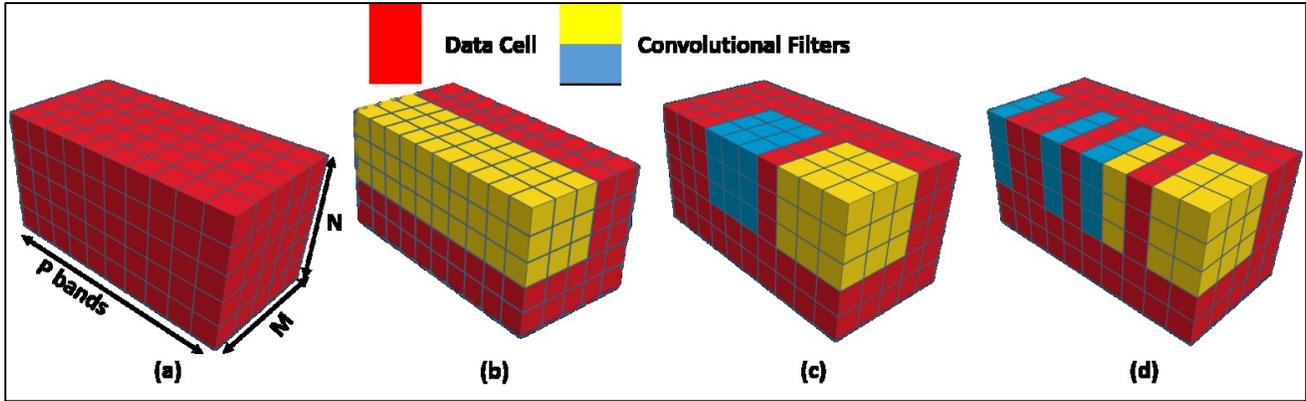


Figure 5: (a). Data cuboid with M by N pixels and P bands. (b). Convolution with  $3 \times 3 \times P$  convolutional filter where  $3 \times 3$  represents the spatial extent and P represents the total number of bands in the data. Each filter convolves with all the bands. This strategy is employed in 2D CNNs. (c). Convolution with  $3 \times 3 \times 3$  convolutional filters where  $3 \times 3$  represents to the spatial extent and 3 represents number of adjacent bands to be convolved simultaneously. One filter can convolve with three adjacent bands only. For example, a first filter (in yellow) convolves with bands 1, 2, and 3 whereas a second filter (in blue) convolves with bands 5, 6, and 7. This strategy is employed in 3D CNNs (Chen et al. 2016). (d). Convolution with  $3 \times 3 \times 3$  convolutional filters where  $3 \times 3$  represents the spatial extent and 3 represents the number of arbitrarily selected bands to be involved. For example, a first filter (in yellow) convolves with bands 1, 2, and 4 whereas a second filter (in blue) convolves with bands 5, 7, and 10. This strategy is employed in the proposed Perm-LSS-CNN framework.

## 5. Proposed Permuted Local Spectral CNN (*Perm-LSS-CNN*) Permuted Local Spectral-Spatial CNN (*Perm-LSS-CNN*)

In *ID-CNN-v2*, a local convolution is performed over the spectral dimension. The convolution filter size in the spectral dimension i.e. 4 is less than the total number of input bands i.e. 13. This local spectral convolution improves the classification performance as discussed in Chen et al. (2016). Figure 4(c) depicts the pictorial representation of the local spectral convolution employed in *ID-CNN-v2*. However, an issue arises in the process of local spectral convolution with high dimensional remote sensing data such as hyperspectral remote sensing data or multisensor data similar to the PolSAR-MS data used in this study. While performing local spectral convolution, various combinations of spectral bands are not convoluted together and therefore, under-utilizing the full potential of the spectral information present in the data. For example, it is evident from Figure 4(c), that the second spectral band and the tenth spectral band cannot not

get convoluted together using *ID-CNN-v2* and hence the use of spectral information is limited. This issue needs more attention while processing multisensor data with CNNs. Similarly, for the PolSAR-MS data used in this study, some polarimetric channels might not convolute together with some multispectral bands. For example, out of the 13 bands, only 4 consecutive bands get convoluted together with a convolutional kernel window of 4. Moreover, the stacking of PolSAR and MS data in a common feature set induces arbitrary ordering and irrelevant band neighbourhood relationships: MS neighbour bands may be related by some wavelength ordering relation, but the boundary between PolSAR and MS bands is irrelevant. Therefore, using local filters in these conditions should be well generalized.

To address the issue, the present study proposes an ensemble approach which attempts to improve the utilization of spectral information. To present a proof of concept, this paper introduces a permuted local spectral CNN model namely *Perm-LS-CNN* which is based on the *ID-CNN-v2* discussed in section 4.1. *Perm-LS-CNN* performs local spectral convolutions on permuted sets of the PolSAR-MS data. The permuted sets of the PolSAR-MS data provide increased combinations of PolSAR-MS bands that are missing from using only the original PolSAR-MS band ordering. The idea of local spectral convolution with permuted PolSAR-MS bands can be realized in Figure 4(d). The proposed *Perm-LS-CNN* model configurations are provided in Table 3. A permutation layer before the first convolution layer is added to obtain permuted sets of the original PolSAR-MS bands. Although the optimal number of permuted sets required for full utilization of the spectral information is not derived in this study, 20 permuted sets are used as a proof of concept. Other configuration and hyperparameters are similar to *ID-CNN-v2* and are listed in Table 3.

Further, based on the *Perm-LS-CNN*, another novel classification framework for land cover classification using PolSAR-MS data is introduced here. This framework extends the idea of *Perm-LS-CNN* to spatial context. The proposed permuted local spectral-spatial CNN or the *Perm-LSS-CNN* utilizes the spatial feature extraction capability of CNNs, along with the idea of permuted local spectral convolution proposed here. The added spatial context helps CNNs to extract more complex and abstract features, which further improves the classification results. The idea of permuted spectral-spatial convolution is presented in Figure 5(d). The proposed *Perm-LSS-CNN* model configurations are provided in Table 3. Similar to *Perm-LS-CNN*, 20 permuted sets are created in the *Perm-LSS-CNN* model. All hyperparameters are set as for *3D-CNN-v1* and are listed in Table 3.

The advantage and significance of the permuted local spectral-spatial convolutions used in *Perm-LSS-CNN* over the local spatial convolutions used in *2D-CNN-v1* or local spectral-spatial convolutions used in *3D-CNN-v1* is evaluated based on a class separability measure proposed in Cumming and Van

Zyl (1989). The separability measure is computed using equation (3).  $SI_{ij}$  is the measure of separability between class  $i$  and  $j$ .  $\mu_i$  and  $\mu_j$  are mean values for class  $i$  and  $j$  respectively, and  $\sigma_i$  and  $\sigma_j$  are standard deviation for class  $i$  and  $j$  respectively.

$$SI_{ij} = \frac{|\mu_i - \mu_j|}{|\sigma_i - \sigma_j|} \quad (3)$$

Table 3 *Perm-LS-CNN* and *Perm-LSS-CNN* model configurations, hyperparameters and training settings

Model	Layer					
	Input	Permutation layer (P1)	Convolutional layer 1 (C1)	Convolutional layer 2 (C2)	Fully connected layer (FC)	Output layer
<i>Perm-LS-CNN</i>	Input image size = $1 \times 13$ Normalization = Batch Normalization	Permuted sets = 20 Kernel size = $1 \times 13$ Activation = Linear	Filters = 20 Kernel size = $1 \times 4$ Activation = ReLU Dropout fraction = 0.2 Pooling size = 0	Filters = 20 Kernel size = $1 \times 4$ Activation = ReLU Dropout fraction = 0.1 Pooling size = 0	Nodes = 16 Activation = ReLU Dropout fraction = 0.1	Labels = 5 Activation = Softmax
<i>Perm-LSS-CNN</i>	Input image size = $3 \times 3 \times 13$ Normalization = Batch Normalization	Permuted sets = 20 Kernel size = $1 \times 1 \times 13$ Activation = Linear	Filters = 20 Kernel size = $3 \times 3 \times 4$ Activation = ReLU Dropout fraction = 0.2 Pooling size = 0	Filters = 20 Kernel size = $1 \times 1 \times 4$ Activation = ReLU Dropout fraction = 0.1 Pooling size = 0	Dropout fraction = 0.1	
Training parameters	Loss function = Categorical cross entropy Learning rate = 0.01 Number of epochs = 200 Optimizer = Adam (Kingma and Ba 2015) 10 fold cross validation strategy is employed during training					

In particular,  $0.8 < SI_{ij} < 1.5$  indicates a good feature,  $SI_{ij} > 2.0$  indicates the best feature, and  $SI_{ij} < 0.8$  indicates the worst feature, for separation of the two classes,  $i$  and  $j$ . Remote sensing studies have used this measure for feature selection purposes (Wu et al. 2011, Mishra, Garg, and Singh 2017).

The class separabilities are measured on the ground truth class samples of Roorkee area and at the first convolutional layer i.e. *C1*, of the *2D-CNN-v1*, *3D-CNN-v1* and the proposed *Perm-LSS-CNN* models for Roorkee PolSAR-MS data. For each model and class pair, the highest SI value achieved using equation

3 is given in Table 4. For example, in the BS-water class pair case, the highest SI value of; 1.82 for *2D-CNN-v1*, 1.99 for *3D-CNN-v1*, and 2.87 for the *Perm-LSS-CNN* are obtained. In other example, in the tall vegetation (denoted as ‘TV)-short vegetation (denoted as ‘SV) class pair case, the highest SI value of; 1.46 for *2D-CNN-v1*, 1.56 for *3D-CNN-v1*, and 2.77 for the *Perm-LSS-CNN* are obtained.

Overall, it is observed from Table 4 that the *Perm-LSS-CNN* model transforms the input data space to a feature space where the classes are more significantly separable than what is achieved by *2D-CNN-v1* or *3D-CNN-v1*.

The above experiment establishes the advantage of permuted local convolution in the spectral dimension with spatial context over the convolution strategies used in *2D-CNN-v1* or *3D-CNN-v1*.

Table 4 separability index values for all the possible class pairs for the 2D CNN-v1, 3D CNN-v1, and ‘Perm-LSS-CNN filters.

Class pairs	Highest Separability Index (SI) value		
	<i>2D CNN-v1</i>	<i>3D-CNN-v1</i>	<i>Perm-LSS-CNN</i>
BS vs Water	1.82	1.99	2.87
BS vs Urban	2.09	2.35	2.90
BS vs TV	2.98	3.82	3.75
BS vs SV	1.29	1.69	1.87
Water vs Urban	3.05	3.04	3.57
Water vs TV	2.59	3.75	4.38
Water vs SV	1.43	1.29	1.64
Urban vs TV	2.81	3.64	5.47
Urban vs SV	1.70	2.10	2.27
TV vs SV	1.47	1.56	1.77

## 6. Results and Discussions

The models *1D-CNN-v1*, *1D-CNN-v2*, *2D-CNN-v1*, *3D-CNN-v1*, *Perm-LS-CNN*, and *Perm-LSS-CNN* are trained and cross validated on ground truth class data points provided in Table 1. The training parameters for *1D-CNN-v2*, *2D-CNN-v1*, and *3D-CNN-v1* are provided in Table 2, and the training parameters for *Perm-LS-CNN*, and *Perm-LSS-CNN* are provided in Table 3 Classification results with these models are shown in Figure 6 and Table 5 for the Roorkee PolSAR-MS data set, and in Figure 7 and Table 6 for the Haridwar PolSAR-MS data set. Figure 6(a) and Figure 7(a) show Google Earth images for the Roorkee and the Haridwar study areas respectively for reference purposes. Heydari and Mountrakis (2019) reports a detailed comparison of deep neural networks with SVMs for land cover classification with remote sensing data. They deplore the lack of case studies where a quantitative and qualitative comparison of deep neural networks with SVMs with multi or high spectral remote sensing data. Therefore, two non-CNN classification methods namely radial basis kernel-support vector machine (RBF-SVM), and random forest

(RF) are also considered here for comparison. The RBF-SVM method parameters settings are; (a). Regularization parameter is set to 12.7 and, (b). Kernel influence parameter is set to 0.056568. The random forest method parameters settings are; (a). The number of features to consider when looking for the best split is set to 4, and, (b). The number of trees in the forest is set to 41.

From Table 5 and Table 6, it is evident that the proposed *Perm-LSS-CNN* provides the best performance for both the datasets with a 10-fold cross validation overall accuracy of 97.8 and quadratic kappa of 0.972 for the Roorkee data set, overall accuracy of 97.9 and quadratic kappa of 0.974 for the Haridwar data set. Individual class precision (a.k.a. user) and recall (a.k.a. producer) accuracies are also provided in Tables 5 and 6 for the Roorkee and the Haridwar data sets. The difference between the precision and the recall averaged over all land cover classes (termed as AD in table 5 and 6) is computed to evaluate the generalization ability and consistency of the methods. AD is minimum with *Perm-LSS-CNN* for Roorkee data and second minimum for the Haridwar data. It is also observed that the AD is consistent for both data sets, i.e. around 3 whereas the AD for other methods varies significantly with the two data sets. This indicates the good generalization ability of the *Perm-LSS-CNN*. Further, Table 5 and Table 6 also show that even if the overall accuracy of the proposed *Perm-LSS-CNN* is best, the other classifiers have shown comparable overall accuracies as well.

A detailed analysis of the classification performance of all the methods considered here is however required to support these tabular results. A visual analysis of the classified images from *2D-CNN-v1*, *3D-CNN-v1*, *Perm-LSS-CNN*, *RF*, and *RBF-SVM* is thus performed. Five ‘region of interest’ areas are selected for visual inspection from the Roorkee study area. The selected areas are marked as A, B, C, D, and E in Figure 6(a). They contain the points that are taken as ground truth. Table 7 shows the classification results on these selected areas obtained from the considered five methods.

In Table 7, Figure (a) show a river and man-made riverbed markers. In Figure (b), (c), (d), (e) and (f), the man-made riverbed structures are correctly classified by each method. But, in Figure (b), and (c), water flow in the river is not consistent which is undesirable, whereas, in Figure (d), (e) and (f), water flow is consistent. Figure (g) shows an agriculture field with a building located in the upper right corner. In Figure (h), and (i), the building is classified as SV by *RF* and *RBF-SVM*, whereas it is correctly classified as urban by *2D-CNN-v1*, *3D-CNN-v1*, and the *Perm-LSS-CNN* in Figure (j), (k) and (l). Figure (m) shows a dried up river, a canal, and a bridge over the canal. The river is partially classified as Short Vegetation in Figure (n), and (o) whereas, it is correctly classified as Bare Soil in Figure (p), (q) and (r). The bridge is well off the ground and oriented perpendicular to the orbit of the SAR sensor, hence exhibits double bounce scattering (only from one side). It is consistently classified as a urban structure in Figure (p) and (r). The same is wrongly classified in Figure (n), (o) and (q). Observing the above two examples suggest that

*RF* and *RBF-SVM* methods are unable to capture subtle changes in land covers. Also, *3D-CNN-v1* classified the bridge as Bare Soil, missing the double bounce nature inherited in the PolSAR features. Figure (s) show a peculiar TV pattern within a golf ground. The structural and spectral identity are better preserved by *Perm-LSS-CNN* (see Figure (x)) as compared to other classifiers. In Figure (t), and (u), most of the ground areas (Bare Soil or Short Vegetation) is wrongly classified as water. In Figure (v), the golf ground is approximately well classified, although the shape is a partially deteriorated. Figure (y) shows a tree line. Again, the structural and spectral identity is most significantly preserved by *Perm-LSS-CNN* as shown in in Figure (dl)).

Overall observation of Table 5, 6, and 7 suggest that the proposed *Perm-LSS-CNN* method yields better classification performance both in terms of classification accuracy and generalization ability by capturing the structural and spectral identity of land covers. It is also observed that classification accuracies alone cannot be used as reliable indicators for comparing classification methods because of the overfitting nature of supervised classification methods. For example, *RF* and *RBF-SVM* provides good classification accuracies (see Table 5 and Table 6) but lacks generalization performance (see Table 7). The generalization accuracy on well-known ground truth is the real test. The reason for the inferior generalization ability of the *2D-CNN-v1* is that the 2D convolutional filter is convolving with all the spectral features together, which reduces the possibility for generation of local spectrally significant features (see Figure 5(b)). The proposed *Perm-LSS-CNN* generates such combinations of spectral and spatial properties and hence shows improved generalization performance.

In summary, the analysis of the results in Table 5, 6, and 7 indicate the improved classification performance of the proposed *Perm-LSS-CNN* over other methods considered for comparison. The discussion also indicates an increase in the generalization ability of the *Perm-LSS-CNN* because of the ensemble learning scenario.

Further, training curves and training time of the CNN based methods used here are provided in appendix section of the paper. Table 8 provides summary of the training time taken by the considered classifiers for the Roorkee dataset. Table 8 indicates that the proposed *Perm-LSS-CNN* classifier takes the longest training time. It is because the *Perm-LSS-CNN* is trained with 20 ensembles of permuted spectral features. In fact, training the *Perm-LSS-CNN* with 20 permuted sets of input spectral features increased the execution time 3 to 4 folds. Moreover, the training time is not very crucial in the land cover classification application. Figure 8 provides the training curves for *1D-CNN-v2*, *2D-CNN-v1*, *3D-CNN-v1*, and *Perm-LSS-CNN*. The convergence of the validation curve of the *Perm-LSS-CNN* in Figure 8(d) indicates to the better generalization performance of the method.

## Conclusion

This paper presents a critical analysis of the potential and performance of the one dimensional, two dimensional, and three dimensional convolutional neural networks, where dimensionality refers to that of the convolved features: spectral, spatial or both. These CNNs are applied to land cover classification with multisensor (PolSAR plus multispectral) remote sensing data. Two novel CNN based models are introduced, where local filters in either spectral only (*Perm-LS-CNN*) or spatial-spectral domains (*Perm-LSS-CNN*) are applied after a permutation layer. The permutation layer generates an ensemble of band combinations, which are drawn upon by the CNN in order to generate powerful features for land cover classification. The results demonstrate the improved generalization ability of these models compared to both alternative CNN architectures and other machine learning models (SVM and random forests). *Perm-LSS-CNN* also fully uses both PolSAR and multispectral information, improving results compared to other popular methods for PolSAR-MS data based land cover classification. A separability measure is used for the model significance evaluation. Separability analysis states that the *Perm-LSS-CNN* transforms the input data space into a feature space where separability among classes increases. This increase in separability is achieved due to the inclusion of permuted local spectral convolution in addition to the local spatial convolution which is not available in the 1D-CNN-v1, 2D-CNN-v1, or 3D-CNN-v1 CNN models. A fine analysis on ground truth data provides an extra qualitative assessment which complements the quantitative classification results. These demonstrate that the proposed *Perm-LSS-CNN* model is able to generalize much better while keeping high classification accuracies.

## Acknowledgement

Authors would like to thank Department of Science and technology (DST), India, IFCAM (India-France), and “*Institut national de recherche en informatique et en automatique*” (INRIA), France for providing the funds. Authors would also like to thank JAXA, Japan for providing ALOS PALSAR 2 microwave PolSAR data for the research. Authors are thankful to RICET-IIT Roorkee, India for supporting the research

## Disclosure Statement

No potential conflict of interest was reported by the authors.

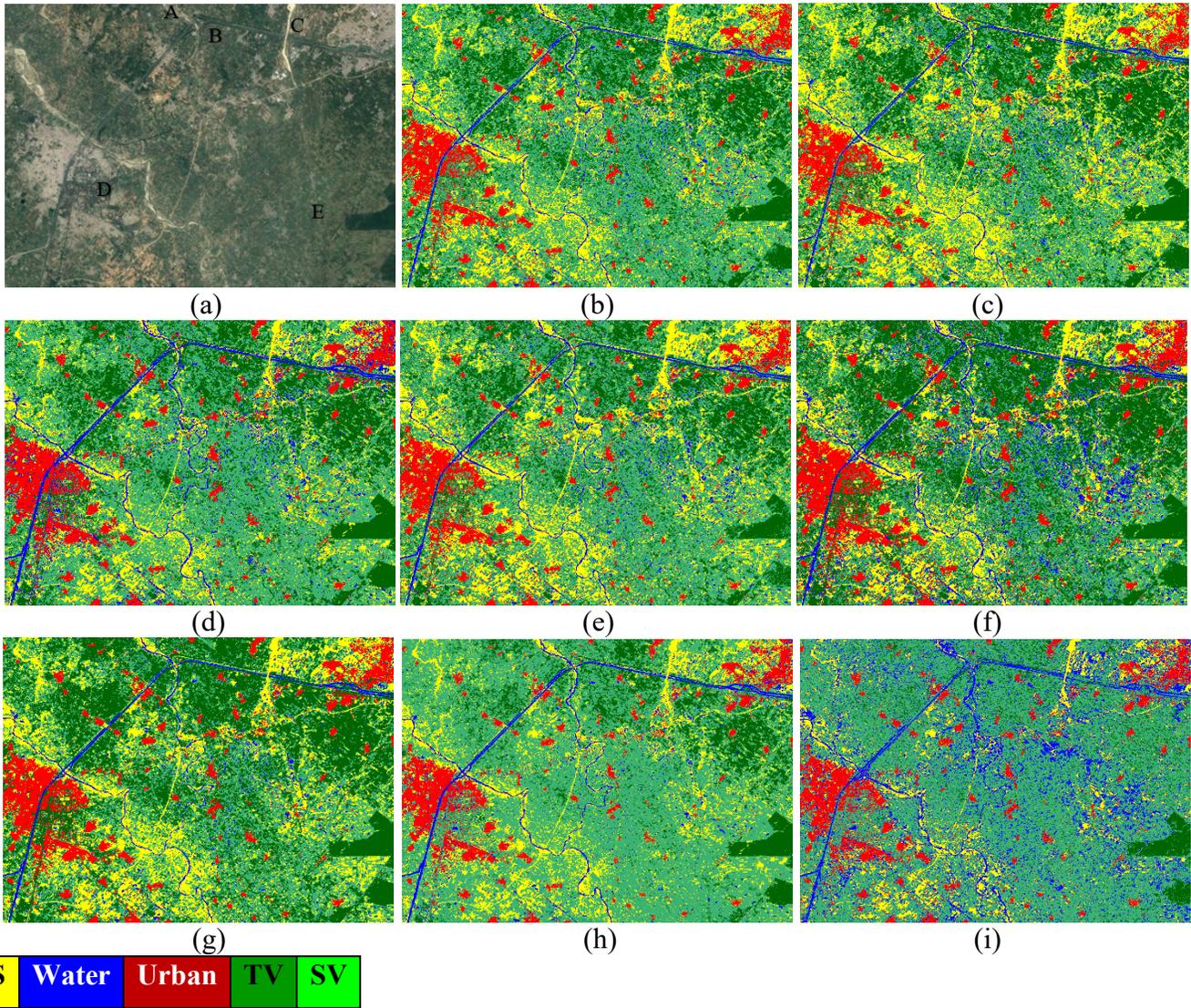


Figure 6 (a) Google Earth image of the Roorkee study area, India. Classified images are shown for different classifiers: (b) 1D-CNN-v1, (c) 1D-CNN-v2, (d) Perm-LS-CNN, (e) 2D-CNN-v1, (f) 3D-CNN-v1, (g) Perm-LSS-CNN, (h) RBF-SVM and, (i) RF.

Table 5 Classification performance of various classifiers considered for PolSAR-MS Roorkee dataset. \*AD is the difference between precision and recall, averaged over all the land cover classes considered.

Method	Overall Accuracy (%)	Quadratic Kappa	Precision (%) (a.k.a. user accuracy)					Recall (%) (a.k.a. producer accuracy)					AD*(%)
			BS	TV	SV	Urban	Water	BS	TV	SV	Urban	Water	
<i>1D-CNN-v1</i>	95	0.938	94.5	100	85.2	100	97.5	93.4	92.4	97	97.5	96.3	4.9
<i>1D-CNN-v2</i>	95.2	0.939	96.4	100	84.5	100	97.1	93	91.8	96.6	98.1	98.4	5.3
<i>Perm-LS-CNN</i>	95.7	0.946	97	100	86.5	100	97.1	94.4	92.1	94.2	98.1	98.4	4.3
<i>2D-CNN-v1</i>	95.3	0.941	96.4	100	83.9	100	98.4	94.1	92.7	98	96.9	96.3	5.8
<i>3D-CNN-v1</i>	96.3	0.933	98.8	100	84.5	100	99.6	95.6	91	100	98.7	98.4	6.1

<i>Perm-LSS-CNN</i>	97.8	0.972	99.7	99.6	91.1	100	99.3	94.3	98.3	99.6	98.7	99	3.3
<i>RBF-SVM</i>	94.6	0.93	94.6	99.3	90.2	96.2	94	92.6	94	93.5	98	96.7	3.2
<i>RF</i>	91.7	0.932	86.8	96.8	88.2	96.8	93.1	92.1	97.1	65	88.6	90.6	7.9

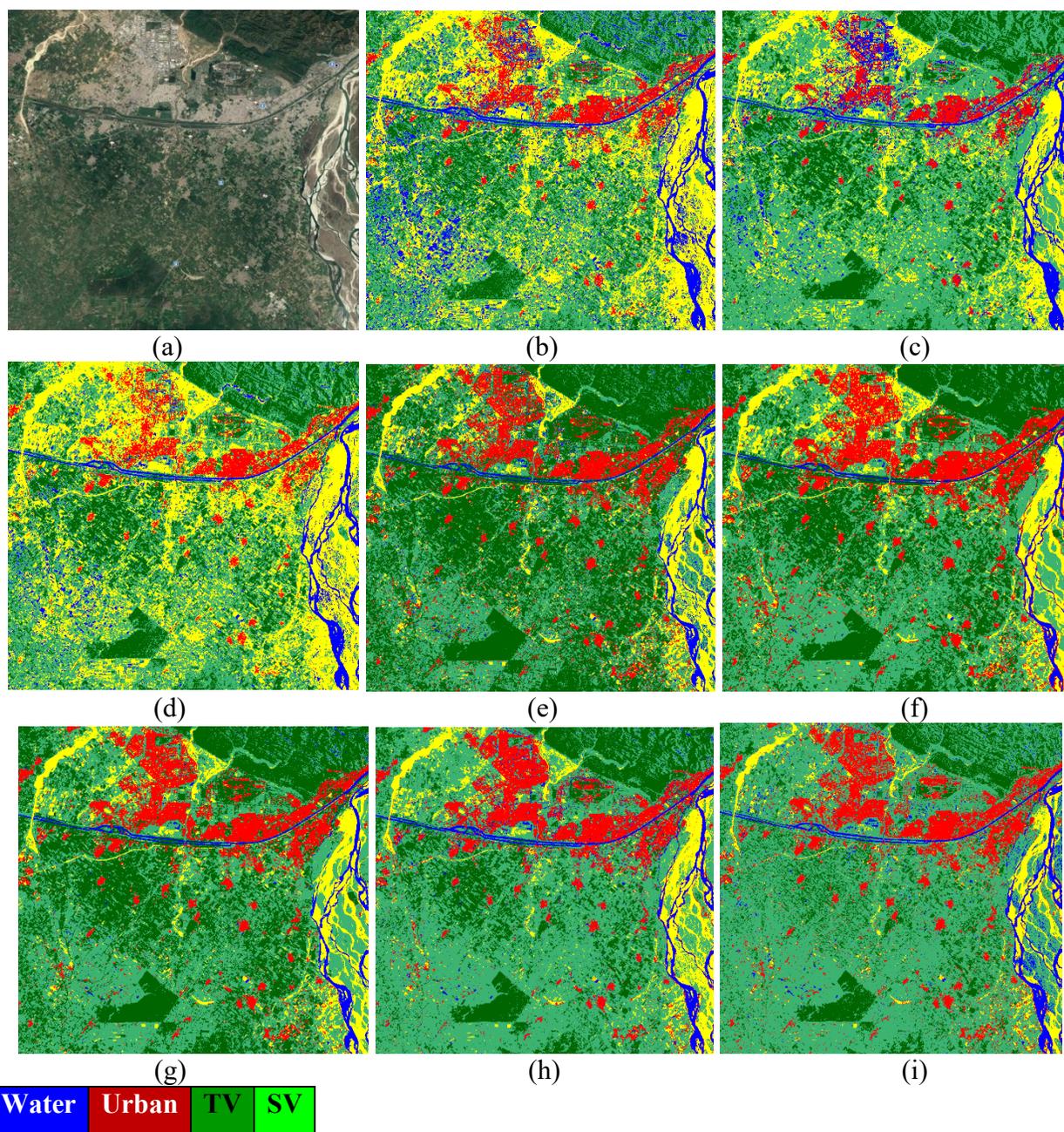


Figure 7 (a) Google Earth image of the Haridwar study area, India. Classified images are shown for different classifiers: (b) 1D-CNN-v1, (c) 1D-CNN-v2, (d) Perm-LS-CNN, (e) 2D-CNN-v1, (f) 3D-CNN-v1, (g) Perm-LSS-CNN, (h) RBF-SVM and, (i) RF.

Table 6 Classification performance of various classifiers considered for PolSAR-MS Haridwar dataset.

\*AD is the difference between precision and recall, averaged over all the land cover classes considered.

Method	Overall Accuracy (%)	Quadratic Kappa	Precision (%) (a.k.a. user accuracy)					Recall (%) (a.k.a producer accuracy)					AD*(%)
			BS	TV	SV	Urban	Water	BS	TV	SV	Urban	Water	
<i>1D-CNN-v1</i>	94.1	0.925	100	100	70.2	96.6	99.3	97.1	87.1	98.9	95	99.3	9.2
<i>1D-CNN-v2</i>	95	0.937	100	100	73.2	99.1	99.3	99.2	87.8	100	95.9	99.3	8.6
<i>Perm-LS-CNN</i>	96.1	0.95	100	100	77.8	100	100	100	89.2	100	97.5	100	7.1
<i>2D-CNN-v1</i>	94.7	0.933	100	100	70.9	99.6	100	97.8	89.2	100	92.9	99.3	9.9
<i>3D-CNN-v1</i>	96.5	0.955	100	100	80.1	99.1	100	98.5	90.3	99	100	100	6.2
<i>Perm-LSS-CNN</i>	97.9	0.974	100	100	89.3	99.1	100	99.2	96	99.1	96.7	100	3.4
<i>RBF-SVM</i>	96.1	0.95	98.5	100	86.2	95.8	97.2	97.8	92.7	95.7	96.6	100	4.3
<i>RF</i>	94.4	0.928	97	98.6	86.2	91.6	95.1	97.7	95.9	84.9	91.6	100	1.9

## References

- Albelwi, Saleh, and Ausif Mahmood. 2017. ‘A Framework for Designing the Architectures of Deep Convolutional Neural Networks’. *Entropy* 19 (6). doi:10.3390/e19060242.
- Antenna Laboratory, Department of Electrical Engineering. 1952. ‘Effects of Type of Polarization on Echo Characteristics’. Columbus, Ohio. doi:10.1002/nav.3800080206.
- Aptoula, Erchan, Murat Can Ozdemir, and Berrin Yanikoglu. 2016. ‘Deep Learning with Attribute Profiles for Hyperspectral Image Classification’. *IEEE Geoscience and Remote Sensing Letters* 13 (12). IEEE: 1970–74. doi:10.1109/LGRS.2016.2619354.
- Boureau, Y-Lan, Jean Ponce, Jean Ponce@ens Fr, and Yann Lecun. 2010. ‘A Theoretical Analysis of Feature Pooling in Visual Recognition’. In , 111–18. Haifa, Israel: ICML. <https://www.di.ens.fr/willow/pdfs/icml2010b.pdf>.
- Chen, Yushi, Hanlu Jiang, Chunyang Li, Xiuping Jia, and Pedram Ghamisi. 2016. ‘Deep Feature Extraction and Classification of Hyperspectral Images Based on Convolutional Neural Networks’. *IEEE Transactions on Geoscience and Remote Sensing* 54 (10). IEEE: 6232–51. doi:10.1109/TGRS.2016.2584107.
- Chen, Yushi, Zhouhan Lin, Xing Zhao, Gang Wang, and Yanfeng Gu. 2014. ‘Deep Learning-Based Classification of Hyperspectral Data’. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing* 7 (6). IEEE: 2094–2107. doi:10.1109/JSTARS.2014.2329330.
- Ciresan, D., Ueli Meier, and Jürgen Schmidhuber. 2012. ‘Multi-Column Deep Neural Networks for Image

- Classification'. In *IEEE Conference on Computer Vision and Pattern Recognition, CVPR*, 3642–49. Rhode Island, USA: IEEE. doi:10.1109/CVPR.2012.6248110.
- Cumming, Ian G., and Jakob J. Van Zyl. 1989. 'Feature Utility in Polarimetric Radar Image Classification'. In *Geoscience and Remote Sensing Symposium, 1989. IGARSS'89. 12th Canadian Symposium on Remote Sensing., 1989 International*, 1841–46. Vancouver, Canada: IEEE. doi:9910031209.
- Feng, Jie, Jiantong Chen, Liguang Liu, Xianghai Cao, Xiangrong Zhang, Licheng Jiao, and Tao Yu. 2019. 'CNN-Based Multilayer Spatial-Spectral Feature Fusion and Sample Augmentation with Local and Nonlocal Constraints for Hyperspectral Image Classification'. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing* 12 (4). IEEE: 1299–1313. doi:10.1109/JSTARS.2019.2900705.
- Feng, Puyu, Bin Wang, De Li Liu, and Qiang Yu. 2019. 'Machine Learning-Based Integration of Remotely-Sensed Drought Factors Can Improve the Estimation of Agricultural Drought in South-Eastern Australia'. *Agricultural Systems* 173 (July): 303–16. doi:10.1016/j.agsy.2019.03.015.
- Fukushima, Kunihiko, and Sei Miyake. 1982. 'Neocognitron: A Self-Organizing Neural Network Model for a Mechanism of Visual Pattern Recognition'. In *Competition and Cooperation in Neural Nets*, 267–85. Berlin, Heidelberg.: Springer.
- Geng, Jie, Jianchao Fan, Hongyu Wang, Xiaorui Ma, Baoming Li, and Fuliang Chen. 2015. 'High-Resolution SAR Image Classification via Deep Convolutional Autoencoders'. *IEEE Geoscience and Remote Sensing Letters* 12 (11). IEEE: 2351–55. doi:10.1109/LGRS.2015.2478256.
- Gillespie, Alan R, Anne B Kahle, and Richard E Walker. 1987. 'Color Enhancement of Highly Correlated Images. II. Channel Ratio and "Chromaticity" Transformation Techniques'. *Remote Sensing of Environment* 22 (3). Elsevier: 343–65. doi:10.1016/0034-4257(87)90088-5.
- Harold, Mott. 2007. 'The Kennaugh Matrix'. In *Remote Sensing with Polarimetric Radar*, 1st ed., 295–98. New Jersey: John Wiley & Sons. <http://www.wiley.com/go/permission>.
- He, Mingyi, Xiaohui Li, Yifan Zhang, Jing Zhang, and Weigang Wang. 2016. 'Hyperspectral Image Classification Based on Deep Stacking Network'. *International Geoscience and Remote Sensing Symposium (IGARSS)* 2016-Novem. IEEE: 3286–89. doi:10.1109/IGARSS.2016.7729850.
- Heydari, Shahriar S., and Giorgos Mountrakis. 2019. 'Meta-Analysis of Deep Neural Networks in Remote Sensing: A Comparative Study of Mono-Temporal Classification to Support Vector Machines'. *ISPRS Journal of Photogrammetry and Remote Sensing* 152 (June): 192–210.

doi:10.1016/j.isprsjprs.2019.04.016.

Hu, Wei, Yangyu Huang, Li Wei, Fan Zhang, and Hengchao Li. 2015. 'Deep Convolutional Neural Networks for Hyperspectral Image Classification'. *Journal of Sensors* 2015: 1–12.

doi:10.1155/2015/258619.

Huang, Bo, Bei Zhao, and Yimeng Song. 2018. 'Urban Land-Use Mapping Using a Deep Convolutional Neural Network with High Spatial Resolution Multispectral Remote Sensing Imagery'. *Remote Sensing of Environment* 214 (September): 73–86. doi:10.1016/j.rse.2018.04.050.

Ian Goodfellow, Yoshua Bengio, and Aaron Courville. 2017a. 'Convolutional Neural Networks'. In *Deep Learning*, 1st ed., 13:330–72. Cambridge: Cambridge, MIT Press. doi:10.1038/nmeth.3707.. 2017b. *Deep Learning. Nature Methods*. Vol. 13. doi:10.1038/nmeth.3707.

Jong-Sen Lee, Mitchell R. Grunes, and Wolfgang-Martin Boerner; 1997. 'Polarimetric Property Preservation in SAR Speckle Filtering'. In *Proc. SPIE 3120, Wideband Interferometric Sensing and Imaging Polarimetry*, edited by Harold Mott, 1–7. San Diego: SPIE. doi:10.1117/12.300624.

Joshi, Neha, Matthias Baumann, Andrea Ehammer, Rasmus Fensholt, Kenneth Grogan, Patrick Hostert, Martin Rudbeck Jepsen, et al. 2016. 'A Review of the Application of Optical and Radar Remote Sensing Data Fusion to Land Use Mapping and Monitoring'. *Remote Sensing* 8 (1): 1–23. doi:10.3390/rs8010070.

Kingma, Diederik P., and Jimmy Lei Ba. 2015. 'Adam: A Method for Stochastic Optimization'. In *3rd International Conference on Learning Representations, ICLR 2015 - Conference Track Proceedings*. International Conference on Learning Representations, ICLR.

Kussul, Nataliia, Mykola Lavreniuk, Sergii Skakun, and Andrii Shelestov. 2017. 'Deep Learning Classification of Land Cover and Crop Types Using Remote Sensing Data'. *IEEE Geoscience and Remote Sensing Letters* 14 (5): 778–82. doi:10.1109/LGRS.2017.2681128.

Lary, David J., Amir H. Alavi, Amir H. Gandomi, and Annette L. Walker. 2016. 'Machine Learning in Geosciences and Remote Sensing'. *Geoscience Frontiers* 7 (1). Elsevier Ltd: 3–10. doi:10.1016/j.gsf.2015.07.003.

LeCun, Yann, Léon Bottou, Yoshua Bengio, and Patrick Haffner. 1998. 'Gradient-Based Learning Applied to Document Recognition'. *Proceedings of the IEEE* 86 (11): 2278–2323. doi:10.1109/5.726791.

Li, Weijia, Haohuan Fu, Le Yu, Peng Gong, Duole Feng, Congcong Li, and Nicholas Clinton. 2016. 'Stacked Autoencoder-Based Deep Learning for Remote-Sensing Image Classification: A Case Study

- of African Land-Cover Mapping'. *International Journal of Remote Sensing* 37 (23). Taylor and Francis Ltd.: 5632–46. doi:10.1080/01431161.2016.1246775.
- Li, Ying, Haokui Zhang, and Qiang Shen. 2017. 'Spectral-Spatial Classification of Hyperspectral Imagery with 3D Convolutional Neural Network'. *Remote Sensing* 9 (1). doi:10.3390/rs9010067.
- Li, Yunsong, Weiyang Xie, and Huaqing Li. 2017. 'Hyperspectral Image Reconstruction by Deep Convolutional Neural Network for Classification'. *Pattern Recognition* 63 (March). Elsevier Ltd: 371–83. doi:10.1016/j.patcog.2016.10.019.
- Makantasis, Konstantinos, Konstantinos Karantzas, Anastasios Doulamis, and Nikolaos Doulamis. 2015. 'Deep Supervised Learning for Hyperspectral Data Classification through Convolutional Neural Networks'. *International Geoscience and Remote Sensing Symposium (IGARSS) 2015-Novem*. IEEE: 4959–62. doi:10.1109/IGARSS.2015.7326945.
- Mei, Shaohui, Jingyu Ji, Qianqian Bi, Junhui Hou, Qian Du, and Wei Li. 2016. 'Integrating Spectral and Spatial Information into Deep Convolutional Neural Networks for Hyperspectral Classification'. *International Geoscience and Remote Sensing Symposium (IGARSS) 2016-Novem (61201324)*. IEEE: 5067–70. doi:10.1109/IGARSS.2016.7730321.
- Mishra, P, Dharmendra Singh, and Y Yamaguchi. 2011. 'Land Cover Classification of Palsar Images by Knowledge Based Decision Tree Classifier and Supervised Classifiers Based on Sar Observables'. *Progress In Electromagnetics Research B* 30 (30): 47–70. doi:10.2528/PIERB11011405.
- Mishra, Pooja, Akanksha Garg, and Dharmendra Singh. 2017. 'Critical Analysis of Model-Based Incoherent Polarimetric Decomposition Methods and Investigation of Deorientation Effect'. *IEEE Transactions on Geoscience and Remote Sensing* 55 (9): 4868–77. doi:10.1109/TGRS.2017.2652060.
- Phartiyal, Gopal Singh, Kundan Kumar, D Singh, and K P Singh. 2017. 'Optimal Use of Polarimetric Signature on PALSAR -2 Data for Land Cover Classification'. In *Geoscience and Remote Sensing Symposium, IGRASS, 2017 IEEE International*, 4558–61. Fort Worth, Texas: IEEE.
- Phartiyal, Gopal Singh, Kundan Kumar, and Dharmendra Singh. 2020. 'An Improved Land Cover Classification Using Polarization Signatures for PALSAR 2 Data'. *Advances in Space Research* 65 (11). Elsevier Ltd: 2622–35. doi:10.1016/j.asr.2020.02.028.
- Pohl, C., and J. L. Van Genderen. 1998. 'Review Article Multisensor Image Fusion in Remote Sensing: Concepts, Methods and Applications'. *International Journal of Remote Sensing*. Taylor & Francis Group. doi:10.1080/014311698215748.

- Rajput, Upendra Kumar, Sanjay Kumar Ghosh, and Anil Kumar. 2014. 'Multisensor Fusion of Satellite Images for Urban Information Extraction Using Pseudo-Wigner Distribution'. *Journal of Applied Remote Sensing* 8 (1): 083668. doi:10.1117/1.jrs.8.083668.
- Sharma, Atharva, Xiuwen Liu, Xiaojun Yang, and Di Shi. 2017. 'A Patch-Based Convolutional Neural Network for Remote Sensing Image Classification'. *Neural Networks* 95 (November). Elsevier Ltd: 19–28. doi:10.1016/j.neunet.2017.07.017.
- Srivastava, Nitish, Geoffrey Hinton, Alex Krizhevsky, Ilya Sutskever, and Ruslan Salakhutdinov. 2014. 'Dropout: A Simple Way to Prevent Neural Networks from Overfitting'. *Journal of Machine Learning Research* 15: 1929–58. <http://jmlr.org/papers/volume15/srivastava14a/srivastava14a.pdf>.
- Wu, Fan, Chao Wang, Hong Zhang, Bo Zhang, and Yixian Tang. 2011. 'Rice Crop Monitoring in South China With RADARSAT-2 Quad-Polarization SAR Data' 8 (2): 196–200.
- Yann LeCun, Yoshua Bengio, and Geoffrey Hinton. 2015. 'Deep Learning'. *Nature*. doi:10.1038/nature14539.
- Yu, Shiqi, Sen Jia, and Chunyan Xu. 2017. 'Convolutional Neural Networks for Hyperspectral Image Classification'. *Neurocomputing* 219 (January). Elsevier B.V.: 88–98. doi:10.1016/j.neucom.2016.09.010.
- Yue, Jun, Shanjun Mao, and Mei Li. 2016. 'A Deep Learning Framework for Hyperspectral Image Classification Using Spatial Pyramid Pooling'. *Remote Sensing Letters* 7 (9). Taylor & Francis: 875–84. doi:10.1080/2150704X.2016.1193793.
- Yue, Jun, Wenzhi Zhao, Shanjun Mao, and Hui Liu. 2015. 'Spectral-Spatial Classification of Hyperspectral Images Using Deep Convolutional Neural Networks'. *Remote Sensing Letters* 6 (6). Taylor & Francis: 468–77. doi:10.1080/2150704X.2015.1047045.
- Zhang, Haokui, Ying Li, Yuzhu Zhang, and Qiang Shen. 2017. 'Spectral-Spatial Classification of Hyperspectral Imagery Using a Dual-Channel Convolutional Neural Network'. *Remote Sensing Letters* 8 (5). Taylor and Francis Ltd.: 438–47. doi:10.1080/2150704X.2017.1280200.
- Zhang, Jixian. 2010. 'Multi-Source Remote Sensing Data Fusion: Status and Trends'. *International Journal of Image and Data Fusion*. Taylor and Francis Ltd. doi:10.1080/19479830903561035.
- Zhang, L., L. Zhang, and Bo Du. 2016. 'Deep Learning for Remote Sensing Data: A Technical Tutorial in the State of the Art'. *IEEE Geoscience and Remote Sensing Magazine*. doi:10.1155/2016/7954154.
- Zhang, Wei, Akira Hasegawa, Kazuyoshi Itoh, and Yoshiki Ichioka. 1991. 'Image Processing of Human

Corneal Endothelium Based on a Learning Network’. *Applied Optics* 30 (29). Optical Society of America: 4211–17.

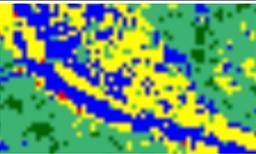
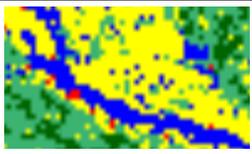
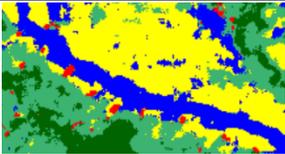
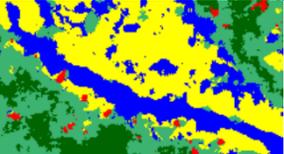
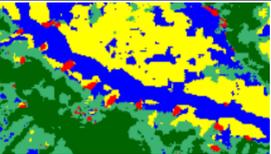
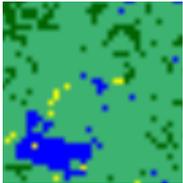
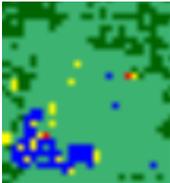
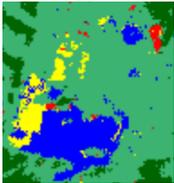
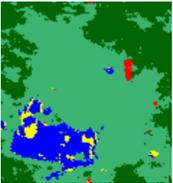
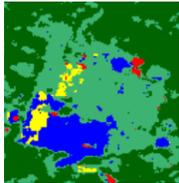
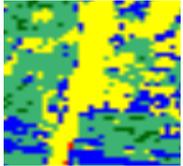
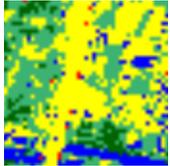
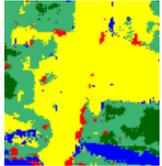
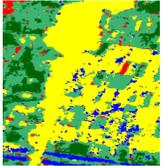
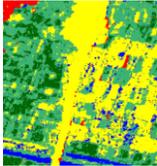
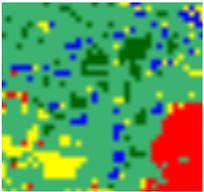
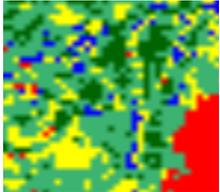
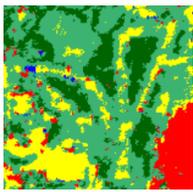
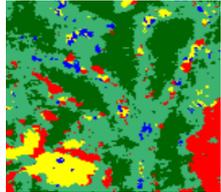
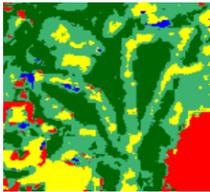
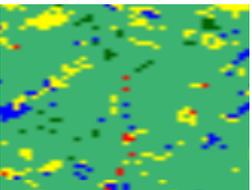
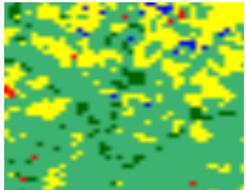
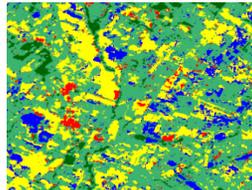
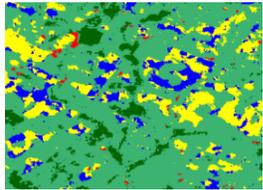
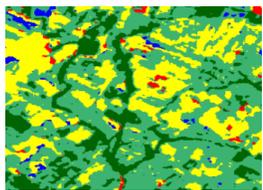
Zhao, Wenzhi, and Shihong Du. 2016. ‘Spectral-Spatial Feature Extraction for Hyperspectral Image Classification: A Dimension Reduction and Deep Learning Approach’. *IEEE Transactions on Geoscience and Remote Sensing* 54 (8). IEEE: 4544–54. doi:10.1109/TGRS.2016.2543748.

Zhao, Wenzhi, Zhou Guo, Jun Yue, Xiuyuan Zhang, and Liqun Luo. 2015. ‘On Combining Multiscale Deep Learning Features for the Classification of Hyperspectral Remote Sensing Imagery’. *International Journal of Remote Sensing* 36 (13): 3368–79. doi:10.1080/2150704X.2015.1062157.

Zhong, Liheng, Lina Hu, and Hang Zhou. 2019. ‘Deep Learning Based Multi-Temporal Crop Classification’. *Remote Sensing of Environment* 221 (February): 430–43. doi:10.1016/j.rse.2018.11.032.

Zhu, Xiao Xiang, Devis Tuia, Lichao Mou, Gui-Song Xia, Liangpei Zhang, Feng Xu, and Friedrich Fraundorfer. 2017. ‘Deep Learning in Remote Sensing: A Review’. *IEEE Geoscience and Remote Sensing Magazine* 5 (4): 8–36. doi:10.1109/MGRS.2017.2762307.

Table 7 Results from different classification techniques applied on the Roorkee data set. The areas include data points collected as ground truth.

Areas	Google Earth image	<i>RF</i>	<i>RBFSVM</i>	<i>2D-CNN-v1</i>	<i>3D-CNN-v1</i>	<i>Perm-LSS-CNN</i>
A (29.944039, 77.939183)						
	(a)	(b)	(c)	(d)	(e)	(f)
B (29.930272, 77.964298)						
	(g)	(h)	(i)	(j)	(k)	(l)
C (29.934405, 78.004034)						
	(m)	(n)	(o)	(p)	(q)	(r)
D (29.857237, 77.905425)						
	(s)	(t)	(u)	(v)	(w)	(x)
E (29.846738, 78.020642)						
	(y)	(z)	(a1)	(b1)	(c1)	(d1)

## Appendix

### *Performance of Classification Methods: Single Sensor vs Multi Sensor*

The performance of various classifiers is evaluated with single sensor and multisensor datasets to indicate to the impact of multisensor data on land cover classification results. Table 8 provides the classification performance of *1D-CNN-v2*, *2D-CNN-v1*, *3D-CNN-v1*, and *Perm-LSS-CNN* classifier with datasets namely PolSAR (PALSAR-2) only, multispectral (Landsat-8) only, and PolSAR-MS combined, table 8 indicates that the validation performance of the *Perm-LSS-CNN* classifier is similar to other classifiers even if the training performance is higher. This corresponds to the overfitting nature of *Perm-LSS-CNN* with low dimensional data due to the “spectral ensemble training” approach opted. However, as the dimensionality of the data increases, the “deep learning” and “spectral ensemble training” kicks in and increase the validation performance of the *Perm-LSS-CNN*. Further, the generalization performance of the *Perm-LSS-CNN* is also improved.

Table 8 Summary of classification performance and training time of the *1D-CNN-v2*, *2D-CNN-v1*, *3D-CNN-v1*, and *Perm-LSS-CNN* classifier with single sensor (PALSAR-2 or Landsat-8) and multisensor (PALSAR-2 and Landsat-8) datasets.

<b>Data</b>	<b>Model</b>	<b>Training Accuracy (%)</b>	<b>Validation Accuracy (%)</b>	<b>Training time (s)</b>
<b>PALSAR-2 only</b>	<i>1D-CNN-v2</i>	78	74	36.68
	<i>2D-CNN-v1</i>	86	80	38.60
	<i>3D-CNN-v1</i>	87	82	41.61
	<i>Perm-LSS-CNN</i>	91	82	160.27
	<i>RBF SVM</i>	75	69	10.50
	<i>RF</i>	77	71	15.27
<b>Landsat-8 only</b>	<i>1D-CNN-v2</i>	90	76	34.09
	<i>2D-CNN-v1</i>	91	81	35.39
	<i>3D-CNN-v1</i>	93	82	38.81
	<i>Perm-LSS-CNN</i>	96	86	149.11
	<i>RBF SVM</i>	92	85	10.56
	<i>RF</i>	88	82	15.32
<b>PALSAR-2 plus Landsat-8</b>	<i>1D-CNN-v2</i>	95	87	44.89
	<i>2D-CNN-v1</i>	96	87	40.80
<b>Landsat-8</b>	<i>3D-CNN-v1</i>	96.3	90	51.40
	<i>Perm-LSS-CNN</i>	98	97	165.2

<i>RF SVM</i>	94	86	10.88
<i>RF</i>	91	84	15.91

### Training Curves for CNN models

Figure 8 shows the training and validation curves for different methods

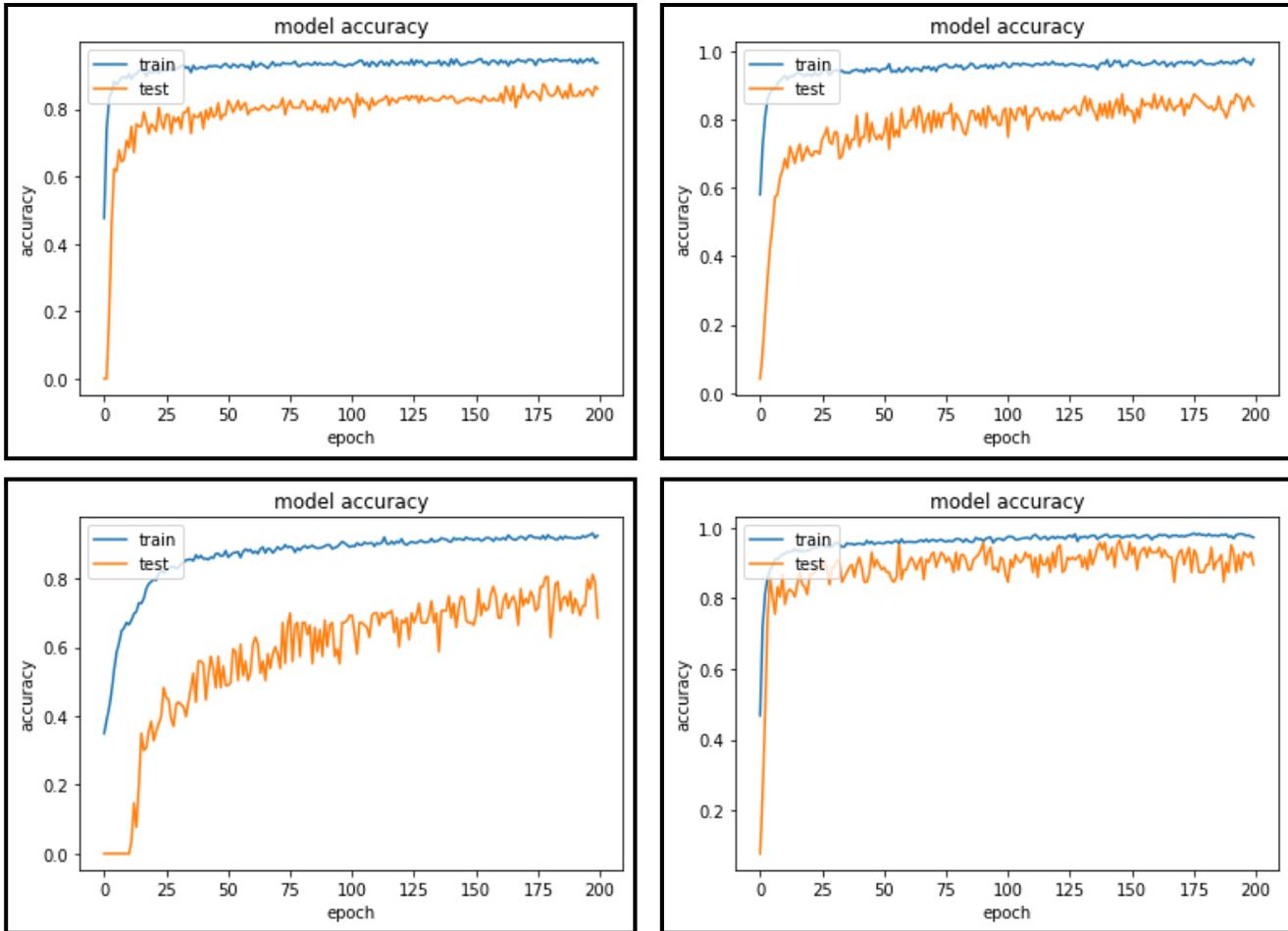


Figure 8 Training curves of models for Roorkee PolSAR-MS (PALSAR-2 plus Landsat-8) data: (a). 1D-CNN-v2, (b). 2D-CNN\_v1, (c). 3D-CNN\_v1 and, (d) The Perm-LSS-CNN model. The blue colour curve is the training curve and the orange colour curve is the validation curve.