IET Image Processing

Research Article

FuSENet: fused squeeze-and-excitation network for spectral-spatial hyperspectral image classification

ISSN 1751-9659
Received on 21st November 2019
Revised 11th February 2020
Accepted on 24th March 2020
doi: 10.1049/iet-ipr.2019.1462
www.ietdl.org

Swalpa Kumar Roy[1] ✉, *Shiv Ram Dubey*[2], *Subhrasankar Chatterjee*[3], *Bidyut Baran Chaudhuri*[4]
[1]*Department of Computer Science & Engineering, Jalpaiguri Government of Engineering College, West Bengal-735102, India*
[2]*Computer Vision Group, Indian Institute of Information Technology, Sri City, Chittoor, Andhra Pradesh-517646, India*
[3]*Department of Computer Science & Engineering, Indian Institute of Technology, Kharagpur, West Bengal-721302, India*
[4]*Computer Vision and Pattern Recognition Unit, Indian Statistical Institute, Kolkata-700108, India*
✉ *E-mail: swalpa@cse.jgec.ac.in*

Abstract: Deep learning-based approaches have become very prominent in recent years due to its outstanding performance as compared to the hand-extracted feature-based methods. Convolutional neural network (CNN) is a type of deep learning architecture to deal with the image/video data. Residual network and squeeze and excitation network (SENet) are among recent developments in CNN for image classification. However, the performance of SENet depends on the squeeze operation done by global pooling, which sometimes may lead to poor performance. In this study, the authors propose a bilinear fusion mechanism over different types of squeeze operation such as global pooling and max pooling. The excitation operation is performed using the fused output of squeeze operation. They used to model the proposed fused SENet with the residual unit and name it as FuSENet. Here the classification experiments are performed over benchmark hyperspectral image datasets. The experimental results confirm the superiority of the proposed FuSENet method with respect to the state-of-the-art methods. The source code of the complete system is made publicly available at https://github.com/swalpa/FuSENet.

1 Introduction

Deep learning-based techniques have become the recent trend of research due to their great performance in practice. Convolutional neural network (CNN) has emerged as a very popular architecture to solve the image recognition problem [1]. The first revolutionary work in this area was AlexNet CNN model [1] which won the ImageNet large-scale object recognition challenge [2] in 2012. Since then various variants of CNNs have been proposed for different problems such as residual network (ResNet) [3], squeeze and excitation network (SENet) [4] for image classification; regions with CNN (R-CNN) [5], Fast-R-CNN [6], Faster-R-CNN [7] for object detection; Mask-R-CNN [8] for image segmentation; local bit-plane decoded Alexnet descriptor [9] for biomedical image retrieval; dual CNN [10] for depth estimation; HybridSN [11], genetic neural network [12] for hyperspectral image (HSI) classification; RCCNet [13] for colon cancer classification etc. The recent works over CNN are image classification [14], medical image analysis [15], deep hashing [16], HSI classification [17, 18], face anti-spoofing [19, 20], texture classification [21] etc.

The residual-based CNN, called ResNet [3], is a widely used network due to the skip connection. The skip connection of ResNet facilitates the better optimisation of the network using the gradient descent method as it provides the superhighway for gradient flow during back-propagation. Several variants of ResNet have been investigated such as ResNeXt [22], DenseNet [23] etc. Moreover, ResNet is also used for the experimentation over loss functions [24] and optimisation methods [25]. The SENet [4] is one of the very recent breakthroughs in a deep learning community. The SENet tries to enhance the inter-channel relationship between different channels of CNN activation. Basically, first, it squeezes the volume using global pooling across spatial dimension, then an excitation factor is generated using a small neural network over squeezed data, and finally, the channels of input activation volume are excited with this excitation factor. This type of network might be better suited for a HSI classification problem as it contains many channels at different wavelengths. Thus, to reduce the effect of redundancy, the automatic prioritisation of intermediate channels is needed, which is provided by the excitation scores of such a network. However, a major problem with SENet is associated with only one type of pooling for squeezing, which may miss the relevant information. We tackle this problem by fusing the excitation scores, computed using different squeeze networks.

The HSI contains the information in several spectral bands of imaging [26]. The HSI has a wide range of real-world applications such as earth observations and land cover classification such as greenery detection, environment analysis, crop analysis, and many more [27, 28]. The CNNs have also shown very promising performance for HSI classification task [29–31]. Some researchers have also explored the fusion in CNN such as the dual-path network (DPNet) [32], convolutional feature fusion network [33], deep feature fusion network [34], 3D–2D CNN fusion [11], spatial–spectral squeeze-and-excitation residual network [35], and spectral–spatial squeeze-and-excitation residual bag-of-features learning [36] for HSI classification. These methods incorporate the feature fusion at the feature level. However, we aim to incorporate the fusion within a layered residual block.

In this study, we propose a fusion of SENet (FuSENet) for HSI classification. The original SENet [4] contains only one squeeze and excitation block. However, the proposed FuSENet uses fusion of two bilinear SENets with different squeezing strategies. The proposed method captures the channel relationship in multiple ways to make the excitation factor more relevant. The main contribution is as follows:

(1) A FuSENet model is proposed by fusing the excitation scores from multiple squeezing channels.
(2) Two squeezing channels are considered to generate the excitation scores through global pooling and max pooling, respectively.
(3) The fusion of multiple squeeze and excitation scores are better suited for HSI data as it has multiple channels.
(4) The proposed fused scores prioritises the important channels towards better training and convergence.
(5) An extensive HSI classification experiment is performed to show the improved performance using the proposed FuSENet model.

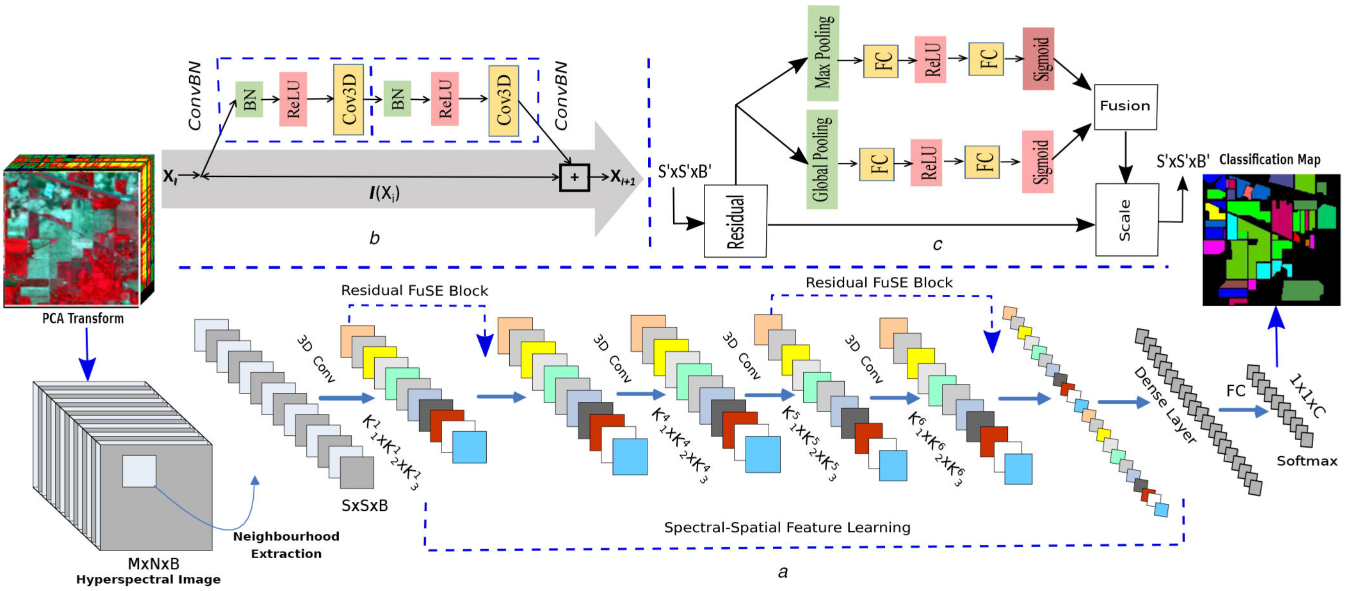

Fig. 1 *Proposed* `FuSENet` *end-to-end learning model for HSI classification. The 3D blocks of dimension $S \times S \times B$ are extracted and used for training and testing purpose. The kernel settings are as follows: $K_1^{1/4/5/6} = 15$, $K_2^{1/4/5/6} = 15$, and $K_3^{1/4/5/6} = 5$, respectively, and all the layers use 32 numbers of kernels*
(a) The spectral-spatial learning block, ***(b)*** Residual learning block, ***(c)*** The squeeze-and-excitation fusion block using global average and max pooling

(6) The weight of the proposed `FuSENet` model is reduced because less no. of blocks are needed as compared to the SENet.

This paper is organised as follows: Section 2 presents the proposed `FuSENet` model; Section 3 carries the experiments and results analysis; and Section 4 concludes the paper.

2 Proposed `FuSENet` model

We propose a `FuSENet` model in this study for HSI classification. Motivated from the nature of HSI data (i.e. many no. of channels in input data), we make use the SENet [4] in the proposed method. Basically, the proposed `FuSENet` model fuses the excitation scores computed through different squeeze channels. The proposed `FuSENet` model is described in Fig. 1. Let us consider a 3D HSI, $\boldsymbol{X}_{\text{org}} \in \mathscr{R}^{M \times N \times D}$, where M, N, and D are the width, height, and spectral dimension, respectively. To remove the spectral redundancy, first, we employ the principal component analysis (PCA) and reduce the number of spectral bands from D to B. The output of the PCA is denoted by $\boldsymbol{X}_r \in \mathscr{R}^{M \times N \times B}$, where B is the reduced spectral bands after PCA. Furthermore, $\boldsymbol{X}_r$ is divided into several regions, defined as $\boldsymbol{x}_{i,j,k} \in \mathscr{R}^{S \times S \times B}$ centred at pixel (i, j) with S as the spatial dimension. The centre of each region $\boldsymbol{x}_{i,j} = [x_{i,j,1}, \ldots x_{i,j,B}]$ is labelled as $Y = (y_1, y_2, \ldots, y_C)$, where C is the number of classes. These regions along with the ground truth class labels of its centre are used for training and validation in classification framework.

We propose a `FuSENet` by considering the ResNet [3] as the base model. The batch normalisation (BN) [37] is used followed by a 3D convolutional layer within the residual blocks. The conventional residual blocks shown in Fig. 1*b* and can be formulated as

$$
\begin{aligned}
X_{\text{out}}^{l+2} &= \mathscr{I}(X_{\text{in}}^{l}) + \mathscr{F}(X_{\text{in}}^{l}; \theta, \Omega) \\
\mathscr{F}(X_{\text{in}}^{l}; \theta, \Omega) &= \phi(X_{\text{in}}^{l+1}) * W^{l+2} + b^{l+2} \\
X_{\text{in}}^{l+1} &= \phi(X_{\text{in}}^{l}) * W^{l+1} + b^{l+1}
\end{aligned}
\tag{1}
$$

where $X_{\text{in}}^{l} \in \mathscr{R}^{S \times S \times B}$ and $X_{\text{out}}^{l+2} \in \mathscr{R}^{S' \times S' \times B'}$ represent the input and output feature maps. The residual function is defined by $\mathscr{F}(\cdot)$, parameterised by θ and Ω to represent the convolutional parameter of two subsequent layers and $*$ and ϕ represent the convolution operation and activation function, respectively. Every residual block is followed by the proposed fused squeeze-and-excitation (FuSE) block (as depicted by residual FuSE block in Fig. 1*c*). The output of the FuSE block is used to re-calibrate the input channels of that block. Basically, the FuSE block uses global average pooling (GAP) and GMP for squeezing operation as given by

$$
\begin{aligned}
\text{SQ}_{\text{avg}}^{c} &= \frac{1}{S' \times S'} \sum_{i=1}^{S'} \sum_{j=1}^{S'} (f_{i,j,c}) \\
\text{SQ}_{\text{max}}^{c} &= \max_{i,j=1}^{S',S'} (f_{i,j,c})
\end{aligned}
\tag{2}
$$

where $f \in \mathscr{R}^{S' \times S' \times B'}$ is the input feature map to FuSE block, $f_{i,j,c}$ is the feature at (i,j) position in the cth channel and $\text{SQ}_{\text{avg}}^{c}$ and $\text{SQ}_{\text{max}}^{c}$ are the squeezed values for the cth channel using global average and max pooling, respectively. Basically, the squeeze operation extracts the channel-wise information. Moreover, the global pooling will retail the information in global context, whereas the max pooling will retain the information in local context. The excitation networks are used to prioritise the features extracted by the squeeze operation. The uses of multiple squeeze channels ensures that the final excitation scores should not be biased towards global or local information.

The excitation factors (EX_{avg} and EX_{max}) corresponding to SQ_{avg} and SQ_{max}, respectively, are computed as

$$
\text{EX}_{\text{avg}} = \sigma(W_{2,\text{avg}}, \delta(W_{1,\text{avg}}, \text{SQ}_{\text{avg}})) \tag{3}
$$

and

$$
\text{EX}_{\text{max}} = \sigma(W_{2,\text{max}}, \delta(W_{1,\text{max}}, SQ_{\text{max}})) \tag{4}
$$

where σ and δ refer to the sigmoid and ReLU activation functions, respectively [4], $\boldsymbol{W}_1 \in \mathscr{R}^{(B/r) \times B}$ and $\boldsymbol{W}_2 \in \mathscr{R}^{(B \times B)/r}$ are the weights of first and second dense layers of the FuSE block, and r is a constant factor by which at first the dimension of squeezed data is decreased and then increased before the sigmoid layer. In the proposed FuSE block, we fuse the excitation factors to compute the final scaling factor (s) as

$$
s_c = F(\text{EX}_{\text{avg}}^{c}, \text{EX}_{\text{max}}^{c}) \tag{5}
$$

where $\text{EX}_{\text{avg}}^{c}$ and $\text{EX}_{\text{max}}^{c}$ are the excitation factors for the cth channel corresponding to the GAP and GMP, respectively, s_c is the scaling factor after fusion, and $F \in \{\text{Sum}, \text{Prod}, \text{Max}\}$ is the fusion strategy. The Sum, Prod, and Max are the summation, product, and

Table 1 The class-wise varying training, validation, and test samples and the performance measured in terms of OA, Kappa, and AA metric over Indian Pines (IP)

		IP			
Class	Name	Training	Validation	Test	Accuracy
1	alfalfa	6	2	38	100
2	corn-no till	214	71	1143	98.47
3	corn-min till	124	41	665	98.53
4	corn	35	11	191	98.56
5	grass-pasture	72	24	387	99.95
6	grass-trees	109	36	585	99.60
7	grass-pasture-mowed	4	1	23	98.75
8	hay-windrowed	71	23	384	99.40
9	oats	3	1	16	100
10	soybean-no till	145	48	779	98.54
11	soybean-min till	368	122	1965	98.42
12	soybean-clean	88	29	476	98.94
13	wheat	30	10	165	98.66
14	woods	189	63	1013	98.98
15	buildings-grass-trees-drives	57	19	310	100
16	stone-steel-towers	13	4	76	98.36
	OA				99.01 ± 0.1
	Kappa				98.60 ± 0.1
	AA				98.64 ± 0.1
	total	1528	505	8216	

maximum fusion strategies. As shown in Fig. 1*c*, the final scaling factor is used to scale the FuSE block input that can re-formulate the residual block as

$$X_{\text{out}}^{l+2}(i,j,c) = \mathcal{I}(X_{\text{in}}^{l}(i,j,c)) + s_c \times \mathcal{F}_c(X_{\text{in}}^{l}; \theta, \Omega) \tag{6}$$

where $\mathcal{F}_c$ represents the *c*th channel of the residual output, $X_{\text{out}}^{l+2}(i,j,c)$ is the final output from the `FuSENet` block. Since, it is the channel-wise product of the scalar s_c and the feature map $\mathcal{F}_c(X_{\text{in}}^{l}; \theta, \Omega)$. Hence, the `FuSENet` block captures the importance of different channels and improves the inter-channel relationship. The *feature recalibration* using the proposed FuSE block prioritises the important channels using a higher scaling factor and improves the feature representation produced by the residual network (ResNet) [3]. A dense layer and a softmax layer are used on the flattened output of the final residual `FuSENet` block as shown in Fig. 1.

3 Experiments and discussion

A series of experiments are conducted to test the superiority of the proposed `FuSENet` model. The results are compared with state-of-the-art models such as support vector machines (SVMs) [38], 2D-CNN [39], 3D-CNN [40], M3D-CNN [41], two-CNN [42], SENet [4], DPNet [32], convolutional feature fusion network (ConvFeaFuNet) [33], and deep feature fusion network (DFeaFuNet) [34], respectively. The model is trained using the RMSProp optimiser with a learning rate of 0.001 for 1000 epochs over each HSI dataset. The categorical cross-entropy loss is minimised using back-propagation. BN and 50% of dropout are used to deal with over-fitting.

3.1 Hyperspectral datasets

The IP, UP and SA HSI datasets [http://dase.grss-ieee.org/] are used for experiments and analysis. The used IP dataset [43] contains the images with 200 spectral bands and having a spatial dimension of size 145 × 145 from 16 mutually exclusive vegetation classes. The used UP dataset consists of 610 × 340 pixel spatially with 103 spectral bands from nine urban land-cover classes. The used SA dataset comprises images of 512 × 217 spatial dimension and 200 spectral bands from 16 vegetation classes.

Once the proposed `FuSENet` is successfully trained under the above settings and can check the performance of the model over the test samples. To validate, we have adopted three relevant measurements for calculating the classification performance. Overall accuracy (OA) is determined by the sum of class-wise correctly classified pixels divided by the total number of presence test pixels; the class accuracy (CA) denotes the percentage of accurately classified samples in each category

$$\text{OA} = \frac{\sum_{i=1}^{n} p_{ii}}{N} \times 100\% \tag{7}$$

$$\text{CA} = \frac{p_{ii}}{\sum_{i=1}^{n} p_{ij}} \times 100\% \tag{8}$$

where the total number of classes and the total number of pixels in the dataset are represented by *n*, and *N*, respectively. p_{ii} represents the pixel, which are perfectly classified and p_{ij} the pixels actually belonging to the *i*th class of the HSI data and during classification it was assigned into the *j*th class. Average accuracy (AA) denotes the sum of the class-wise accuracy divided by the number of classes present in the dataset and can be calculated as

$$AA = \frac{\sum_{i=1}^{n} CA_i}{n} \times 100\% \tag{9}$$

and Kappa coefficient (kappa) is another metric of statistical measurement, which calculates mutual information between the ground truth map and predicted classification map and also shows a strong agreement, which can be defined as follows:

$$\text{kappa} = \frac{N\sum_{i=1}^{n} p_{ii} - \sum_{i=1}^{n} p_i \sum_{j=1}^{n} p_j}{N^2 - \sum_{i=1}^{n} p_i \sum_{j=1}^{n} p_j} \tag{10}$$

where the diagonal elements of the corresponding confusion matrix, the total sum of the *i*th row and total sum of the *j*th column are represented by p_{ii}, p_i, and p_j, respectively.

To well explore in both the spectral and spatial context and to perform the unbiased comparison, various small spectral–spatial 3D input patches are extracted for each HSI dataset, such as 30 spectral bands are extracted with a spatial window of sizes 15 × 15 and 13 × 13, for the IP dataset and similarly 15 spectral bands are extracted with a spatial window of sizes 15 × 15 and 13 × 13, for both UP and SA datasets, respectively.

To evaluate the effectiveness of the proposed `FuSENet`, the whole extracted 3D input patches are randomly selected into three sets viz., training, validating, and testing. During the training, 20%, and 10% available labelled 3D patches are selected from each class. In addition, unlabelled 5% available sample patches from each class were supplied for the model validation and the remaining samples for model testing. Moreover, to avoid the biases among different classes with imbalance samples, the experiments were repeated ten times and the evaluated performances are reported in terms of mean ± std value. The detailed description of the training, validation, and testing along with class-wise classification accuracies, i.e. OAs, AAs, and kappa coefficients are given in Tables 1–3, respectively.

In addition, Table 4 shows the comparison with the well-known HSI classification methods under varying training samples, i.e. 10 and 20%, respectively. It is observed from the table that the proposed `FuSENet` model reaches consistent classification performance in both the scenarios as compared to SVM, 2D-CNN, 3D-CNN, M3D-CNN, two-CNN, DPNet, ConvFeaFuNet, DFeaFuNet, SENet (GMP), and SENet (GMP), respectively. The OAs achieved by SENet (GAP) model is higher than other compared spectral- or spatial-based methods since the method uses 3D convolution within the residual learning block and is capable of extracting both the spectral–spatial feature representation jointly,

Table 2 The class-wise varying training, validation, and test samples, and the performance measured in term of OA, Kappa, and AA metric over University of Pavia (UP)

		UP			
Class	Name	Training	Validation	Test	Accuracy
1	asphalt	994	331	5306	99.89
2	meadows	2797	932	14920	99.93
3	gravel	314	104	1681	98.75
4	trees	459	153	2452	98.71
5	painted metal sheets	201	67	1077	98.69
6	bare soil	754	251	4024	99.76
7	bitumen	199	66	1065	99.94
8	self-blocking bricks	552	184	2946	98.31
9	shadows	142	47	758	100
	OA				99.42 ± 0.2
	Kappa				99.21 ± 0.3
	AA				99.33 ± 0.2
	total	6412	2135	34,229	

Table 3 The class-wise varying training, validation, and test samples and the performance measured in terms of OA, Kappa, and AA metric over Salinas Scene (SA)

		SA			
Class	Name	Training	Validation	Test	Accuracy
1	Brocoli_green_weeds_1	301	100	1608	100
2	Brocoli_green_weeds_2	558	186	2982	100
3	Fallow	296	98	1582	99.63
4	Fallow_rough_plow	209	69	1116	99.21
5	Fallow_smooth	401	133	2144	100
6	Stubble	593	197	3169	100
7	Celery	536	178	2865	99.61
8	Grapes_untrained	1690	563	9018	99.97
9	Soil_vinyard_develop	930	310	4963	99.97
10	Corn_senesced_green_weeds	491	163	2624	99.95
11	Lettuce_romaine_4wk	160	53	855	100
12	Lettuce_romaine_5wk	289	96	1542	99.84
13	Lettuce_romaine_6wk	137	45	734	98.88
14	Lettuce_romaine_7wk	160	53	857	99.96
15	Vinyard_untrained	1090	363	5815	99.59
16	Vinyard_vertical_trellis	271	90	1446	98.59
	OA				99.68 ± 0.2
	Kappa				99.74 ± 0.1
	AA				99.69 ± 0.1
	total	8112	2697	43,320	

which provides the discriminative information to accurately classify the target HSI data. The proposed FuSENet consistently outperform OAs as compared to the SENet (GAP) by an average improvement of +0.49, +1.12, and +0.12 for IP, UP, and SA, respectively. The convergence of training losses shown in Figs. 2*a–c* and the convergence of the accuracies shown in Figs. 2*d–f* for IP, UP, and SA using the proposed FuSENet framework and its different variations, i.e. SENet (GAP) and SENet (GMP), respectively. It can be observed from the figures that the proposed model smoothly converge as compared to its other variants in both scenarios.

To explore the robustness of the proposed FuSENet, Table 4 shows the classification performance of FuSENet in terms of OA, Kappa, and AA using varying training samples 10 and 20% over IP, UP, and SA datasets, respectively.

The performance is reported in terms of the OA, AA and Kappa coefficient. The best-achieved results are highlighted in bold. It is observed from the experimental results that the proposed FuSENet outperforms other methods over each dataset in terms of each evaluation criterion. We also test the method simply using GAP, i.e. SENet (GAP) and GMP, i.e. SENet (GMP), respectively, where we have found that the performance of fusion is significantly improved as compared to SENet methods due to the fact that fusion yields a better weight calibration feature maps at the end. To validate the impact of different fusion techniques a comparison with other fusion strategies such as addition (Sum), multiplication (Multiply) and maximum (Max) are presented in Table 5. It is observed that the Max fusion is better suited for the proposed method and we prefer to use the maximum (Max) fusion strategy between the sigmoid output of GAP and GMP in the proposed FuSENet method (shown in Fig. 1*c*).

The feature representation of any CNN is always dependent upon the convolutional filter banks and the ability to produce the discriminative feature maps can be controlled by the number of kernels used in the filter banks. To show the impact of the number of kernels in the proposed network, we vary the number of kernels from 16 to 64 and the step size is taken with a power of 2 and shown in Fig. 3. It can be observed from Fig. 3 that the model achieved the highest classification performance with 64 kernels in each convolutional filter bank for IP, UP, and SA, datasets, respectively.

To well explore the impact of different spatial 3D input patches in the proposed FuSENet framework. Table 6 compares the performance of the proposed FuSENet framework in terms of OAs,

Table 4 The classification accuracies (%) in terms of OA, Kappa, and AA using the proposed FuSENet methods with varying training data 10 and 20%, respectively

Training samples	Methods	IP dataset			UP dataset			SA dataset		
		OA	Kappa	AA	OA	Kappa	AA	OA	Kappa	AA
10%	SVM	81.67 ± 0.65	78.76 ± 0.77	79.84 ± 3.37	90.58 ± 0.47	87.21 ± 0.70	92.99 ± 0.36	94.46 ± 0.12	93.13 ± 0.34	93.01 ± 0.60
	2D-CNN	80.27 ± 1.2	78.26 ± 2.1	68.32 ± 4.1	96.63 ± 0.2	95.53 ± 1.0	94.84 ± 1.4	96.34 ± 0.3	95.93 ± 0.9	94.36 ± 0.5
	3D-CNN	82.62 ± 0.1	79.25 ± 0.3	76.51 ± 0.1	96.34 ± 0.2	94.90 ± 1.2	97.03 ± 0.6	85.00 ± 0.1	83.20 ± 0.7	89.63 ± 0.2
	M3D-CNN	81.39 ± 2.6	81.20 ± 2.0	75.22 ± 0.7	95.95 ± 0.6	93.40 ± 0.4	97.52 ± 1.0	94.20 ± 0.8	93.61 ± 0.3	96.66 ± 0.5
	two-CNN	96.71 ± 0.1	96.10 ± 0.10	96.16 ± 0.12	97.71 ± 0.1	97.62 ± 0.1	97.45 ± 0.2	97.12 ± 0.30	96.98 ± 0.20	97.00 ± 0.20
	DPNet	96.04 ± 0.2	96.97 ± 0.07	96.93 ± 0.07	97.67 ± 0.1	97.58 ± 0.1	97.27 ± 0.2	97.97± 0.1	97.95 ± 0.1	98.11 ± 0.1
	ConvFeaFuNet	97.39 ± 0.4	97.16 ± 0.5	97.01 ± 0.5	97.72 ± 0.1	97.64 ± 0.2	97.20 ± 0.2	97.89 ± 0.1	97.83 ± 0.1	97.91 ± 0.1
	DFeaFuNet	97.45 ± 0.2	97.79 ± 0.2	97.61 ± 0.2	97.38 ± 0.3	96.93 ± 0.4	97.32 ± 0.3	98.74 ± 0.2	98.91 ± 0.1	98.88 ± 0.1
	SENet(GMP)	97.48 ± 0.3	97.84 ± 0.2	97.91 ± 0.3	97.56 ± 0.5	97.41 ± 0.4	97.47 ± 0.4	98.88 ± 0.1	98.93 ± 0.2	99.01 ± 0.1
	SENet(GAP)	97.62 ± 0.3	97.91 ± 0.2	97.88 ± 0.3	97.53 ± 0.6	97.48 ± 0.5	97.52 ± 0.5	99.11 ± 0.2	98.89 ± 0.2	99.06 ± 0.2
	FuSENet	**1.11 ± 0.2**	**1.25 ± 0.2**	**1.32 ± 0.2**	**1.65 ± 0.3**	**1.69 ± 0.3**	**1.68 ± 0.4**	**1.23 ± 0.1**	**1.97 ± 0.2**	**1.16 ± 0.1**
20%	SVM	86.24 ± 0.38	84.27 ± 0.45	83.15 ± 1.10	95.20 ± 0.13	93.63 ± 0.17	93.60 ± 0.14	94.15 ± 0.10	93.48 ± 0.11	97.23 ± 0.11
	2D-CNN	86.90 ± 1.3	85.01 ± 1.6	82.70 ± 1.0	96.02 ± 0.4	96.04 ± 0.3	95.10 ± 0.1	96.15 ± 0.6	95.71 ± 0.7	98.27 ± 0.2
	3D-CNN	89.23 ± 0.2	87.70 ± 0.3	87.87 ± 0.1	97.30 ± 0.3	96.22 ± 0.1	97.02 ± 0.1	94.54 ± 0.5	93.81 ± 0.3	96.79 ± 0.6
	M3D-CNN	93.67 ± 0.1	92.70 ± 0.3	93.60 ± 0.6	97.41 ± 0.2	96.05 ± 0.6	98.22 ± 0.1	94.92 ± 0.3	94.40 ± 0.1	97.28 ± 0.2
	Two-CNN	98.73 ± 0.2	98.71 ± 0.2	98.73 ± 0.2	98.72 ± 0.25	98.40 ± 0.17	98.45 ± 0.20	98.13 ± 0.43	98.01 ± 0.20	98.10 ± 0.20
	DPNet	98.84 ± 0.1	98.33 ± 0.1	98.42 ± 0.1	98.37 ± 0.1	98.32 ± 0.1	98.39 ± 0.2	98.27 ± 0.2	98.15 ± 0.1	98.21 ± 0.1
	ConvFeaFuNet	98.79 ± 0.3	98.46 ± 0.4	98.71 ± 0.3	98.51 ± 0.2	98.54 ± 0.2	98.57 ± 0.2	98.44 ± 0.1	98.48 ± 0.0	98.45 ± 0.0
	DFeaFuNet	98.75 ± 0.2	98.72 ± 0.2	98.49 ± 0.2	98.93 ± 0.3	98.91 ± 0.4	98.94 ± 0.3	98.98 ± 0.2	99.01 ± 0.1	98.96 ± 0.1
	SENet(GMP)	98.53 ± 0.6	98.27 ± 0.8	97.91 ± 1.5	99.05 ± 0.2	98.81 ± 0.2	98.86 ± 0.2	99.07 ± 0.3	99.19 ± 0.2	99.13 ± 0.2
	SENet(GAP)	98.76 ± 0.5	98.43 ± 0.7	98.20 ± 1.0	99.36 ± 0.1	99.20 ± 0.1	99.30 ± 0.1	99.50 ± 0.1	99.55 ± 0.1	99.40 ± 0.1
	FuSENet	**1.01 ± 0.1**	**1.60 ± 0.1**	**1.64 ± 0.1**	**1.42 ± 0.2**	**1.21 ± 0.3**	**1.33 ± 0.2**	**1.68 ± 0.2**	**1.74 ± 0.1**	**1.69 ± 0.1**

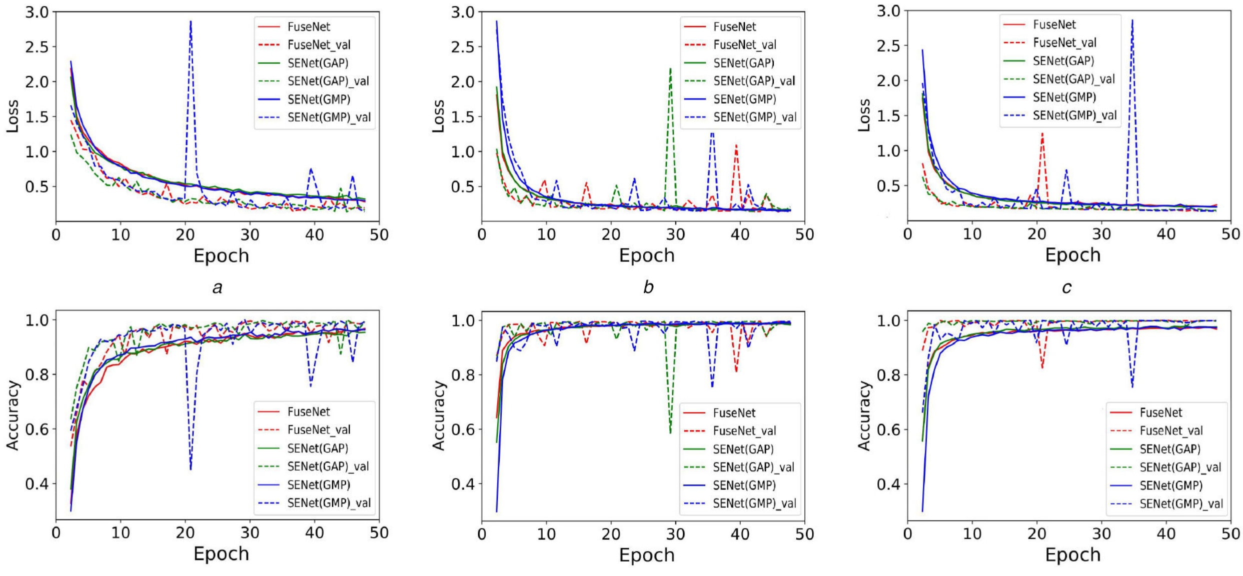

Fig. 2 *The convergence analysis*

(a–c) The convergence of loss versus epochs, ***(d–f)*** The accuracy versus epochs using the SENet (GAP), SENet (GMP) and FuSENet models over IP, UP, and SA datasets, respectively

Table 5 The performance of FuSENet using different fusion strategies such as addition (Sum), multiplication (Mul) and maximum (Max) over each dataset

Training samples	Methods	IP dataset			UP dataset			Salinas Scene dataset		
		OA	Kappa	AA	OA	Kappa	AA	OA	Kappa	AA
20%	Sum	94.74 ± 0.2	94.56 ± 0.2	95.65 ± 0.2	96.39 ± 0.2	97.94 ± 0.2	96.49 ± 0.3	98.07 ± 0.1	98.39 ± 0.1	98.11 ± 0.1
	Multiply	95.31 ± 0.2	96.77 ± 0.3	95.41 ± 0.3	97.33 ± 0.2	97.45 ± 0.1	96.87 ± 0.2	98.12 ± 0.1	98.73 ± 0.2	98.25 ± 0.1
	Max	**1.01 ± 0.1**	**1.60 ± 0.1**	**1.64 ± 0.1**	**199.42 ± 0.2**	**1.21 ± 0.3**	**1.33 ± 0.2**	**1.68 ± 0.2**	**1.74 ± 0.1**	**1.69 ± 0.1**

AAs, and kappa of the IN, UP, and SA datasets under different spatial input patches of sizes 13 × 13 and 15 × 15, respectively. The results are reported in Table 6, the proposed FuSENet framework shows sound classification performance for the IP dataset when the spatial window of size was taken as 15 × 15 and it was 13 × 13 for UP and SA datasets, in addition, 20% of available labelled samples are supplied during training. The average OAs improved by the FuSENet are +1.25, +0.35, and 0.29 between the

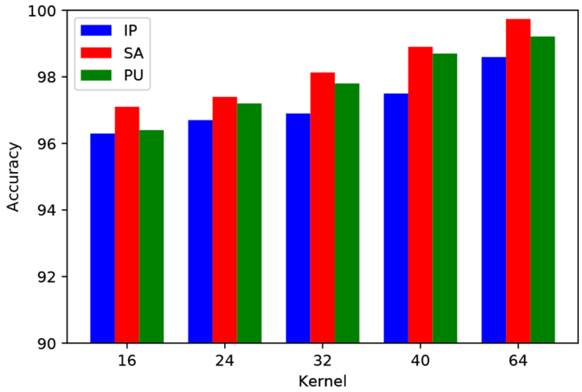

Fig. 3 *The influence of OAs(%) with varying numbers of kernels for IP, UP, and SA, respectively*

Table 6 The influence of varying training samples (i.e. 20, 10, 5%) with respect to the spatial window of size ($S \times S$) (i.e. 15×15 and 13×13) over the performance of the proposed `FuSENet` on IP, UP, and SA datasets

Training, %	Window size	IP dataset			UP dataset			SA dataset		
		OA	AA	Kappa	OA	AA	Kappa	OA	AA	Kappa
20		**1.01 ± 0.1**	**1.60 ± 0.1**	**1.64 ± 0.1**	99.42± 0.2	99.21 ± 0.3	99.33 ± 0.2	**1.68 ± 0.2**	**1.74 ± 0.1**	**1.69 ± 0.1**
10	15 × 15	98.11 ± 0.2	98.25 ± 0.2	98.32 ± 0.2	97.65 ± 0.3	97.69 ± 0.3	97.68 ± 0.4	99.23 ± 0.1	98.97 ± 0.2	99.16 ± 0.1
5		93.48 ± 2.2	93.11 ± 2.1	91.27 ± 2.3	**1.58 ± 0.1**	99.36 ± 0.1	**1.44 ± 0.1**	99.14 ± 0.2	99.56 ± 0.1	99.04 ± 0.2
20		**1.76 ± 0.4**	**1.45 ± 0.4**	**1.28 ± 0.4**	**99.77 ± 0.2**	**99.67 ± 0.2**	**1.69 ± 0.2**	**1.97 ± 0.0**	**1.97 ± 0.0**	**1.96 ± 0.0**
10	13 × 13	96.14 ± 1.3	96.17 ± 1.2	95.45 ± 1.6	99.67 ± 0.2	99.62 ± 0.2	99.56 ± 0.2	99.94 ± 0.0	99.92 ± 0.0	99.93 ± 0.0
5		93.48 ± 2.2	93.11 ± 2.1	91.27 ± 2.3	99.58 ± 0.1	99.36 ± 0.1	99.44 ± 0.1	99.14 ± 0.2	99.56 ± 0.1	99.04 ± 0.2

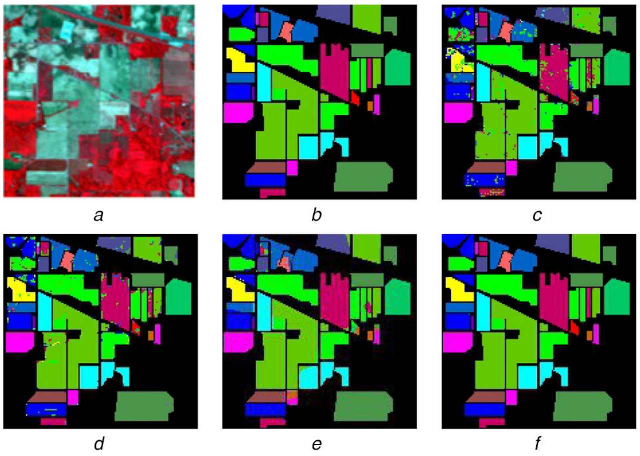

Fig. 4 *Classification map for IP*

(a) False colour image, ***(b)*** Ground truth, ***(c–f)*** Predicted classification maps for 2D-CNN, 3D-CNN, SENet, and `FuSENet`, respectively

two spatial windows of sizes 13×13 and 15×15 for IP, UP, and SA datasets, respectively.

The visualisation of the classification maps using different methods along with false colour images, and their respective ground truth maps over the three datasets, i.e. IP, UP, and SA are shown in Figs. 4–6, respectively. In addition, the class legends of IP, UP, and SA datasets are also shown in Figs. 7*a–c*. A quality classification map can be visualised using the proposed `FuSENet` method and which make sense from the quantitative comparison shown in Tables 1–3, respectively. The classification map generated through 3D-CNN is better as compared to SVM and 2D-CNN but still, there exist some artefacts within the class boundaries. This is mainly because 2D-CNN uses only spatial information to predict the target pixels. During the training proposed `FuSENet` is capable of learning more discriminative and powerful spectral–spatial feature representations consecutively by simply 'excite' the feature that helps in classification layer while suppressing the ineffective feature based on the patterns produced by the `FuSENet` over the feature maps. So, the proposed `FuSENet` produces smooth and more accurate classification maps over all the HSI datasets as compared to the other methods.

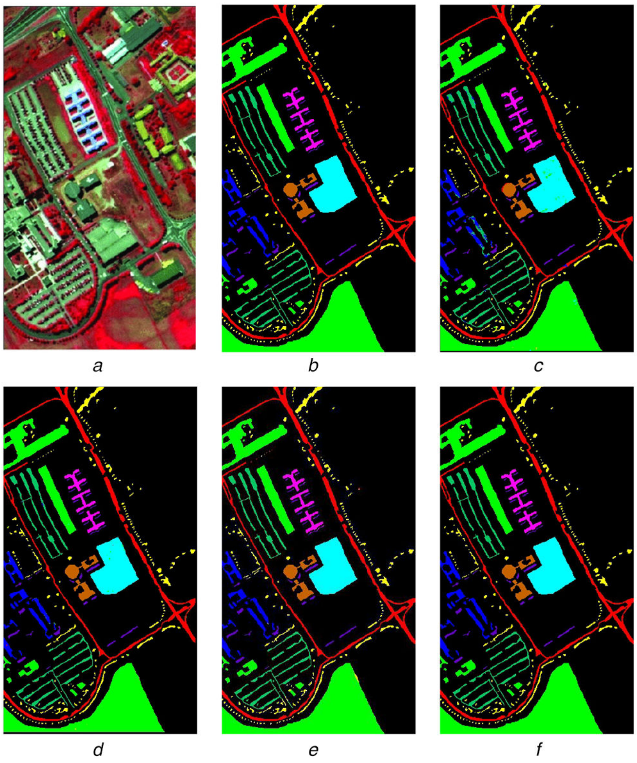

Fig. 5 *Classification map for Pavia University*

(a) False colour image, ***(b)*** Ground truth, ***(c-f)*** Predicted classification maps for 2D-CNN, 3D-CNN, SENet, and `FuSENet`, respectively

To further analyse the classification performance shown in Table 4 of the proposed `FuSENet` model, we have conducted the one way statistical analysis of variance (ANOVA) test [44]. This ANOVA experiment is performed to explore the reason behind the improved classification performance achieved by `FuSENet` as compared to SVM, 3D-CNN, two-CNN, DFeaFuNet, and SENet (GAP), respectively. The null hypothesis H_0 can determine difference among group means is not significant for the test. In an experiment, if the p-value is lesser than the pre-selected significant

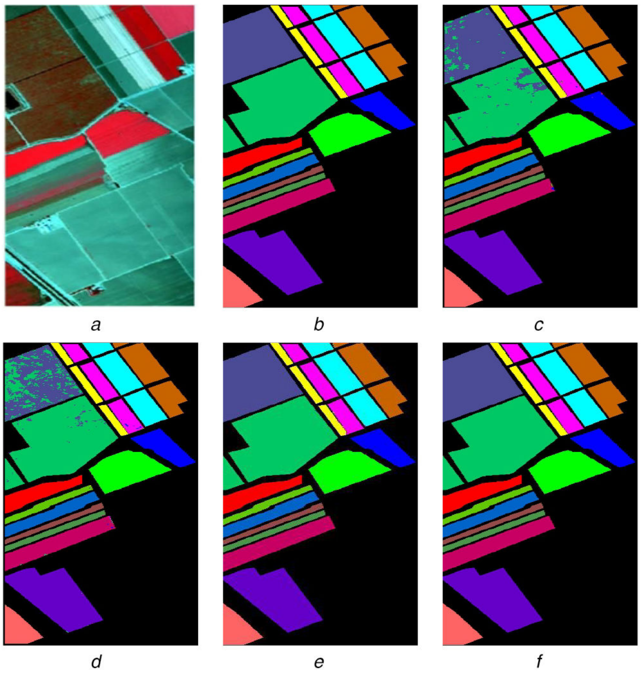

Fig. 6 *Classification map for SA*
(a) False colour image, ***(b)*** Ground truth, ***(c–f)*** Predicted classification maps for 2D-CNN, 3D-CNN, SENet, and `FuSENet`, respectively

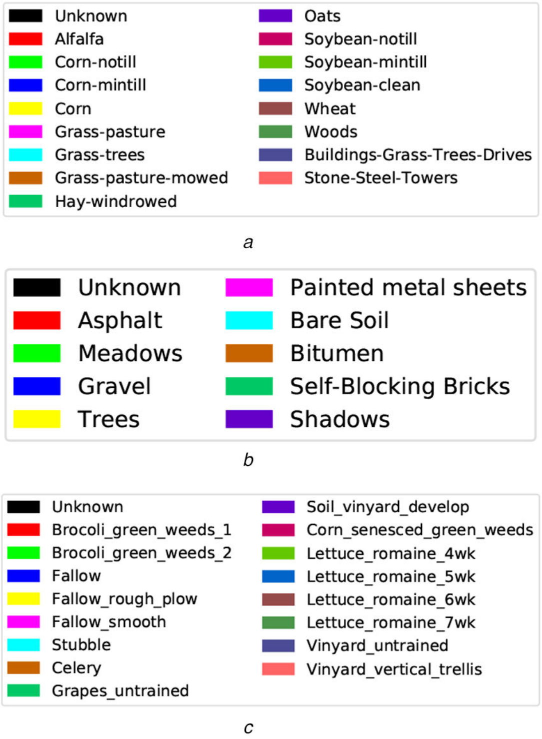

Fig. 7 *The class legend for*
(a) IP, ***(b)*** UP, ***(c)*** SA datasets, respectively, where black legend shows background class

Table 7 One way statistical ANOVA test where the level of significance is selected as $\alpha = 0.05$ for IP, UP, and SA, respectively

	IP dataset				UP dataset				SA dataset			
	Sum_Sq	df	*F*	Prob(*p*)>*F*	Sum_Sq	df	*F*	Prob $(p) > F$	Sum_Sq	df	*F*	Prob$(p) > F$
groups	1570.418	5.0	5130.690	1.61×10^{-35}	212.725	5.0	514.927	1.37×10^{-23}	756.928	5.0	6490.738	9.60×10^{-37}
residual	1.469	24.0			1.982	24.0			0.559	24.0		
total	1571.88	29.0			214.708	29.0			757.48	29		

level, which implies that at least one group mean is significantly different from the others and we can simply reject hypothesis H_0. The significance level is kept for the one way ANOVA test as $\alpha = 0.05$ and the test results for three different datasets are shown in Table 7. In addition, the box plot corresponding to aforementioned ANOVA test for three different datasets is also shown in Figs. 8*a* and *b*, which clearly indicates that the mean performance of `FuSENet` is significantly better than the methods

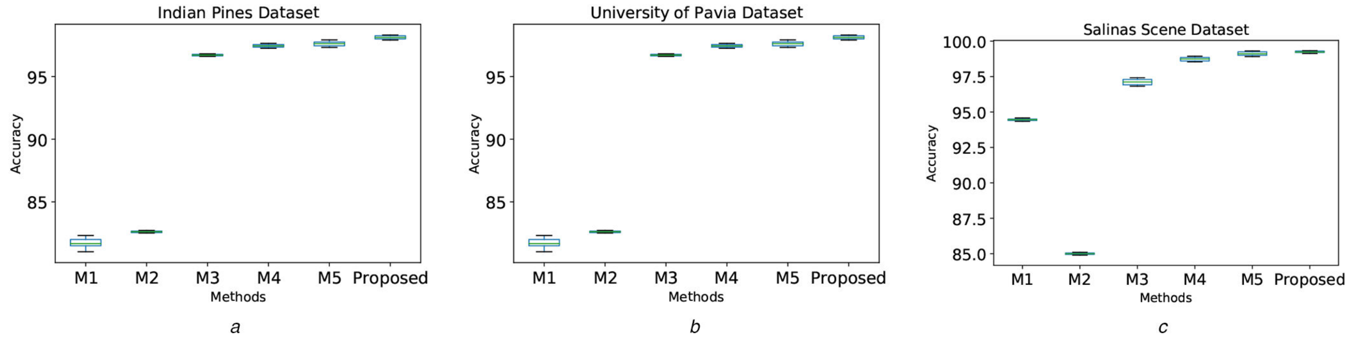

Fig. 8 *The significance test*
(a–c) The box plot (methods versus accuracy) of one way statistical ANOVA test for ‘M1’: SVM, ‘M2’: 3D-CNN, ‘M3’: Two-CNN, ‘M4’: DFeaFuNet, ‘M5’: SENet(GAP) and proposed `FuSENet` models over IP, UP and SA datasets, respectively

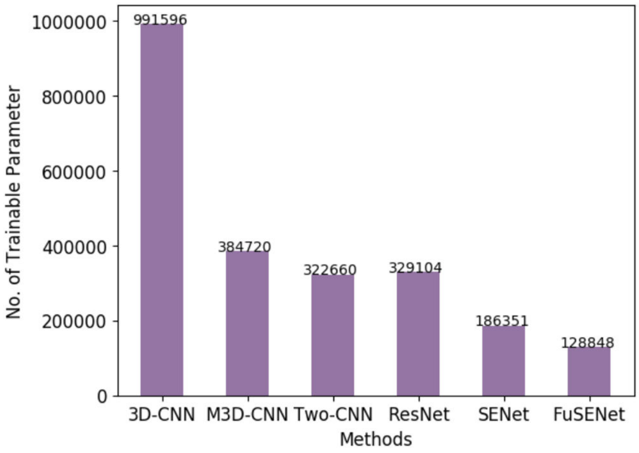

Fig. 9 *The number of learnable model parameters for the methods 3D-CNN, M3D-CNN, two-CNN, ResNet, SENet, and `FuSENet`, respectively*

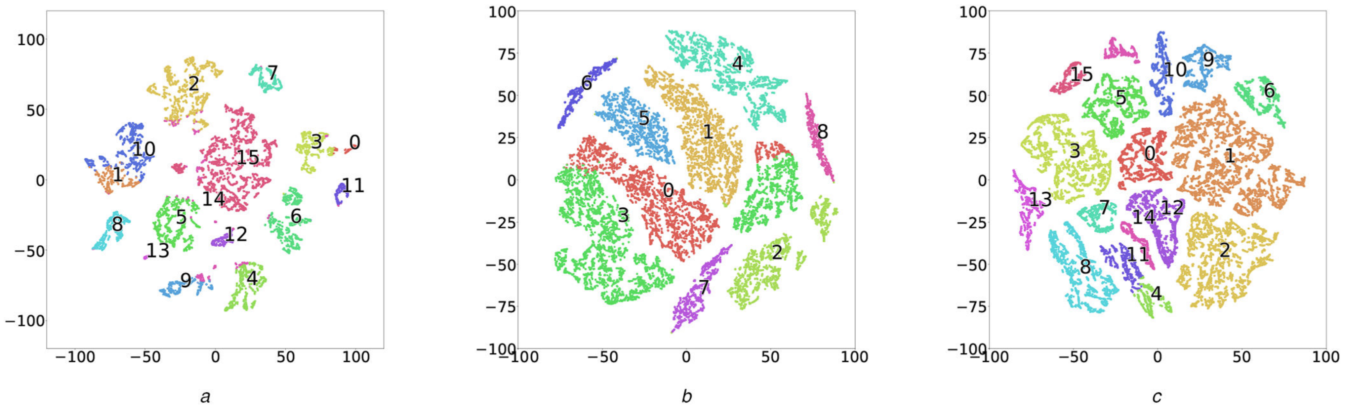

Fig. 10 *Two-dimension spectral–spatial feature visualisation of the proposed `FuSENet` via t-SNE where samples are represented through points and classes are shown in different colours for*
(a) IP, ***(b)*** UP, ***(c)*** SA datasets, respectively

such as SVM, 3D-CNN, two-CNN, DFeaFuNet, and SENet(GAP), respectively.

To train a deep network, it always requires expensive hardware such as graphical processing units (GPUs) and in the existing models, millions and millions of parameters need to be re-weighted during training. Hence, the number of parameters play an important role while training. Fig. 9 shows the comparative distribution of learnable weigh parameters with the state-of-the-art methods, i.e. 3D-CNN [40], M3D-CNN [41], two-CNN [42], ResNet [3], SENet [4], `FuSENet`, respectively. The proposed model contains less number of weight parameters as compared to others as observed from Fig. 9 and it is possible to train in a general configured machine with a minimum of 2 GB of GPU. To increase feature generalisation ability, it is important to extract the joint spectral–spatial features simultaneously. Since the proposed `FuSENet` uses the 3D residual learning block to extract joint spectral–spatial features and which ended with a high-dimensional abstract representation of the feature and difficult to visualise within the high range. To visualise the discriminative power of the proposed feature representation, T-distributed stochastic neighbor embedding (t-SNE) [45] helps to transform the dimensionality of the learned features into 2D space and became much easier to plot. Figs. 10*a–c* show the t-SNE visualisation of learned features in 2D for three HSI datasets, i.e. IP, UP, and SA, respectively. It can be clearly visualised from the figures that due to the use of large training epochs the learned spectral–spatial features from the same class clustered together and features from different classes are becoming much easier to separate.

4 Conclusion

In this study, a CNN model named `FuSENet` is proposed in the SENet framework. To design the `FuSENet`, we use two squeeze and excitation connection bi-linearly based on GAP and GMP, respectively. To better utilise both the characteristics are then fused into the *sigmoid* output of block SENet(GAP) and SENet(GMP), respectively, and computed the final scaling factor for each channel of input at any given layer. To enhance feature learning efficiency and avoid gradient vanishing problem, the proposed `FuSENet` method is combined with a 3D residual learning network and tested for a hyperspectral image classification problem over three benchmark datasets. The results are compared with state-of-the-art methods. The proposed `FuSENet` method has shown extremely good performance with a limited amount of training data. It is also discovered that the Max fusion is better suited to the proposed `FuSENet` method.

5 References

[1] Krizhevsky, A., Sutskever, I., Hinton, G.E.: 'Imagenet classification with deep convolutional neural networks'. Advances in Neural Information Processing Systems, 2012, pp. 1097–1105

[2] Russakovsky, O., Deng, J., Su, H., *et al.*: 'Imagenet large scale visual recognition challenge', *Int. J. Comput. Vis.*, 2015, **115**, (3), pp. 211–252

[3] He, K., Zhang, X., Ren, S., *et al.*: 'Deep residual learning for image recognition'. Proc. IEEE Conference on Computer Vision and Pattern Recognition, 2016, pp. 770–778

[4] Hu, J., Shen, L., Sun, G.: 'Squeeze-and-excitation networks'. Proc. IEEE Conf. on Computer Vision and Pattern Recognition, 2018, pp. 7132–7141

[5] Girshick, R., Donahue, J., Darrell, T., *et al.*: 'Rich feature hierarchies for accurate object detection and semantic segmentation'. Proc. IEEE Conf. on Computer Vision and Pattern Recognition, 2014, pp. 580–587

[6] Girshick, R.: 'Fast R-CNN'. Proc. IEEE Int. Conf. on Computer Vision, 2015, pp. 1440–1448

[7] Ren, S., He, K., Girshick, R., *et al.*: 'Faster R-CNN: towards real-time object detection with region proposal networks'. Advances in Neural Information Processing Systems, 2015, pp. 91–99

[8] He, K., Gkioxari, G., Dollár, P., *et al.*: 'Mask R-CNN'. 2017 IEEE Int. Conf. on Computer Vision (ICCV), 2017, pp. 2980–2988

[9] Dubey, S.R., Roy, S.K., Chakraborty, S., *et al.*: 'Local bit-plane decoded convolutional neural network features for biomedical image retrieval', *Neural Comput. Appl.*, 2019, pp. 1–13

[10] Repala, V.K., Dubey, S.R.: 'Dual CNN models for unsupervised monocular depth estimation', arXiv preprint arXiv:180406324, 2018

[11] Roy, S.K., Krishna, G., Dubey, S.R., *et al.*: 'Hybridsn: exploring 3-D-2-D CNN feature hierarchy for hyperspectral image classification', *IEEE Geosci. Remote Sens. Lett.*, 2020, **17**, (2), pp. 277–281

[12] Akbari, D.: 'Improved neural network classification of hyperspectral imagery using weighted genetic algorithm and hierarchical segmentation', *IET Image Process.*, 2019, **13**, (12), pp. 2169–2175

[13] Basha, S.S., Ghosh, S., Babu, K.K., *et al.*: 'RCCNet: an efficient convolutional neural network for histological routine colon cancer nuclei classification'. 2018 15th Int. Conf. on Control, Automation, Robotics and Vision (ICARCV), 2018, pp. 1222–1227

[14] Moradi, R., Berangi, R., Minaei, B.: 'Orthomaps: an efficient convolutional neural network with orthogonal feature maps for tiny image classification', *IET Image Process.*, 2019, **13**, (12), pp. 2067–2076

[15] Jin, Y., Jiang, X.B., Wei, Z.k., *et al.*: 'Chest x-ray image denoising method based on deep convolution neural network', *IET Image Process.*, 2019, **13**, (11), pp. 1970–1978

[16] Liao, J., Li, B., Yang, D., *et al.*: 'Deep supervised hashing network with integrated regularisation', *IET Image Process.*, 2019, **13**, (12), pp. 2143–2151

[17] Ahmad, M., Khan, A.M., Hussain, R.: 'Graph-based spatial–spectral feature learning for hyperspectral image classification', *IET Image Process.*, 2017, **11**, (12), pp. 1310–1316

[18] Shamsolmoali, P., Zareapoor, M., Yang, J.: 'Convolutional neural network in network (CNNIN): hyperspectral image classification and dimensionality reduction', *IET Image Process.*, 2018, **13**, (2), pp. 246–253

[19] Khammari, M.: 'Robust face anti-spoofing using CNN with LBP and WLD', *IET Image Process.*, 2019, **13**, (11), pp. 1880–1884

[20] Nagpal, C., Dubey, S.R.: 'A performance evaluation of convolutional neural networks for face anti spoofing'. 2019 Int. Joint Conf. on Neural Networks (IJCNN), 2019, pp. 1–8

[21] Roy, S.K., Dubey, S.R., Chanda, B., *et al.*: 'Texfusionnet: an ensemble of deep cnn feature for texture classification'. Proc. 3rd Int. Conf. on Computer Vision and Image Processing, 2020, pp. 271–283

[22] Xie, S., Girshick, R., Dollár, P., *et al.*: 'Aggregated residual transformations for deep neural networks'. Proc. IEEE Conf. on Computer Vision and Pattern Recognition, 2017, pp. 1492–1500

[23] Huang, G., Liu, Z., Van Der Maaten, L., *et al.*: 'Densely connected convolutional networks'. Proc. IEEE Conf. on Computer Vision and Pattern Recognition, 2017, pp. 4700–4708

[24] Roy, S.K., Manna, S., Dubey, S.R., *et al.*: 'LISHT: non-parametric linearly scaled hyperbolic tangent activation function for neural networks', arXiv preprint arXiv:190105894, 2019

[25] Dubey, S.R., Chakraborty, S., Roy, S.K., *et al.*: 'DIFFGRAD: an optimization method for convolutional neural networks', arXiv preprint arXiv:190911015, 2019

[26] Tu, B., Zhang, X., Kang, X., *et al.*: 'Hyperspectral image classification via fusing correlation coefficient and joint sparse representation', *IEEE Geosci. Remote Sens. Lett.*, 2018, **15**, (3), pp. 340–344

[27] Ji, H., Gao, Z., Mei, T., *et al.*: 'Vehicle detection in remote sensing images leveraging on simultaneous super-resolution', *IEEE Geosci. Remote Sens. Lett.*, 2020, **17**, pp. 676–680(in press)

[28] Chen, J., Wan, L., Zhu, J., *et al.*: 'Multi-scale spatial and channel-wise attention for improving object detection in remote sensing imagery', *IEEE Geosci. Remote Sens. Lett.*, 2020, **17**, pp. 681–685(In press)

[29] Li, W., Chen, C., Zhang, M., *et al.*: 'Data augmentation for hyperspectral image classification with deep CNN', *IEEE Geosci. Remote Sens. Lett.*, 2019, **16**, pp. 593–597

[30] Fang, L., Liu, Z., Song, W.: 'Deep hashing neural networks for hyperspectral image feature extraction', *IEEE Geosci. Remote Sens. Lett.*, 2019, **16**, pp. 1412–1416

[31] Li, S., Song, W., Fang, L., *et al.*: 'Deep learning for hyperspectral image classification: an overview', *IEEE Trans. Geosci. Remote Sens.*, 2019, **57**, pp. 6690–6709

[32] Kang, X., Zhuo, B., Duan, P.: 'Dual-path network-based hyperspectral image classification', *IEEE Geosci. Remote Sens. Lett.*, 2019, **16**, pp. 447–451

[33] Yu, Y., Gong, Z., Wang, C., *et al.*: 'An unsupervised convolutional feature fusion network for deep representation of remote sensing images', *IEEE Geosci. Remote Sens. Lett.*, 2018, **15**, (1), pp. 23–27

[34] Song, W., Li, S., Fang, L., *et al.*: 'Hyperspectral image classification with deep feature fusion network', *IEEE Trans. Geosci. Remote Sens.*, 2018, **56**, (6), pp. 3173–3184

[35] Wang, L., Peng, J., Sun, W.: 'Spatial–spectral squeeze-and-excitation residual network for hyperspectral image classification', *Remote Sens.*, 2019, **11**, (7), p. 884

[36] Roy, S.K., Chatterjee, S., Bhattacharyya, S., *et al.*: 'Lightweight spectral-spatial squeeze-and-excitation residual bag-of-features learning for hyperspectral classification', *IEEE Trans. Geosci. Remote Sens.*, 2020, (in press)

[37] Ioffe, S., Szegedy, C.: 'Batch normalization: accelerating deep network training by reducing internal covariate shift'. Int. Conf. on Machine Learning, 2015, pp. 448–456

[38] Melgani, F., Bruzzone, L.: 'Classification of hyperspectral remote sensing images with support vector machines', *IEEE Trans. Geosci. Remote Sens.*, 2004, **42**, (8), pp. 1778–1790

[39] Makantasis, K., Karantzalos, K., Doulamis, A., *et al.*: 'Deep supervised learning for hyperspectral data classification through convolutional neural networks'. IEEE Int. Geoscience and Remote Sensing Symp. (IGARSS), 2015, pp. 4959–4962

[40] Ben-Hamida, A., Benoit, A., Lambert, P., *et al.*: '3-D deep learning approach for remote sensing image classification', *IEEE Trans. Geosci. Remote Sens.*, 2018, **56**, (8), pp. 4420–4434

[41] He, M., Li, B., Chen, H.: 'Multi-scale 3d deep convolutional neural network for hyperspectral image classification'. IEEE Int. Conf. on Image Processing (ICIP), 2017, pp. 3904–3908

[42] Yang, J., Zhao, Y.Q., Chan, J.C.W.: 'Learning and transferring deep joint spectral–spatial features for hyperspectral classification', *IEEE Trans. Geosci. Remote Sens.*, 2017, **55**, (8), pp. 4729–4742

[43] Green, R.O., Eastwood, M.L., Sarture, C.M., *et al.*: 'Imaging spectroscopy and the airborne visible/infrared imaging spectrometer (AVIRIS)', *Remote Sens. Environ.*, 1998, **65**, (3), pp. 227–248

[44] Anscombe, F.: 'The validity of comparative experiments', *J. R. Stat. Soc. A, Gen.*, 1948, **111**, (3), pp. 181–211

[45] Maaten, L.V.D., Hinton, G.: 'Visualizing data using t-sne', *J. Mach. Learn. Res.*, 2008, **9**, pp. 2579–2605