A Fast and Compact 3D CNN for Hyperspectral Image Classification

Muhammad Ahmad, Adil Mehmood Khan, Manuel Mazzara, Salvatore Distefano, Mohsin Ali, Muhammad Shahzad Sarfraz

***Abstract*—Hyperspectral Images (HSI) are used in a large number of real-world applications. HSI classification (HSIC) is a challenging task due to high inter-class similarity, high intra-class variability, overlapping, and nested regions. The 2D Convolutional Neural Network (CNN) is a viable classification approach since HSIC depends on both Spectral-Spatial information. The 3D CNN is a good alternative for improving accuracy of HSIC, but it can be computationally-intensive due to the volume and spectral dimensions of HSI. Furthermore, these models may fail to extract quality feature maps and underperform over the regions having similar textures. This work proposes a 3D CNN model that utilizes both spatial-spectral feature maps to improve the performance of HSIC. For this purpose, the HSI cube is first divided into small overlapping 3D patches, which are processed to generate 3D feature maps using a 3D kernel function over multiple contiguous bands of the spectral information in a computationally efficient way. In brief, our end-to-end trained model requires fewer parameters to significantly reduce the convergence time while providing better accuracy than existing models. The results are further compared with several state-of-the-art 2D/3D CNN models, demonstrating remarkable performance both in terms of accuracy and computational time.**

***Index Terms*—3D Convolutional Neural Network (CNN); Kernel Function; Classification; Hyperspectral Images (HSI).**

I. Introduction

HYPERSPECTRAL sensors collect information (*reflectance*) in several hundreds of contiguous bands with a very high spectral resolution enabling us to classify objects based on their spectral signatures. However, these images are in relatively low spatial resolution due to the sensor limitations, SNR, and complexity constraints which significantly affect the performance for several real-world applications [1]. The traditional classifiers, for instance, KNN [2], SVM [3], Maximum Likelihood [4], Logistic Regression [2] and Extreme Learning Machine (ELM) [5] only employ spectral information. These classifiers do not perform well due to spectral redundancy and high correlation among the spectral bands. Furthermore, these classifiers fail to preserve the important spatial variability of Hyperspectral data which also results in their low performance.

The simplest way to improve the classification performance is to design a classifier that should incorporate both spectral and spatial information. Spatial information is considered as additional discriminatory information associated with the size, shape, and structure of the object which, if provided correctly, brings more competitive results. Spatial-spectral classifiers can generally be classified into two groups. The first category explores spatial and spectral information separately. The spatial information is extracted in advance using entropy [6], morphological operations [7], [8], low rank representation [9], attribute profiles [10] and fuzziness [11]. Later this information is combined with spectral information to perform pixel-level classification.

The second category fuses spatial-spectral information to get joint features [12], for instance, 3D wavelet, scattering wavelet and Gabor filter [13], [14] are generated at different frequencies and scales to extract the joint spatial-spectral features for classification. HSI are in 3D cubes thus the former category results in several 3D features, i.e., spatial-spectra feature cubes comprising key information, thus preserving joint spatial-spectral correlations that enables the extracted features to produce better results. However, the classical feature extraction methods are based on shallow learning and handcrafted features which largely depend on domain knowledge [15]. Accordingly, Deep models have been used to automatically learn low to high-level features from raw HSI data which have attained incredible success for HSIC.

The last few years witnessed an intensive improvement in CNN for HSIC where the spatial features are tailored by a 2D CNN model [16], [17], [18]. These spatial features are usually extracted separately that, to some extent, void the reason to jointly exploit the spatial-spectral information for HISC. A hybrid spectral CNN for HSIC has been proposed in [15], in which the authors proposed a 3D CNN followed by a spatial 2D CNN model. The 3D convolutional layers facilitate the spectral-spatial feature representation whereas 2D convolutional layers are used to learn abstract level information. The hybrid model produces better results as compared to the conventional 3D models but still lacks at extracting the abstract level spatial information. Recently, Paoletti et. al., [19], [20] proposed two deep pyramidal residual networks for HSI feature extraction and classification. The former work only considered spectral information for HSIC whereas the latter considered both the spectral-spatial capsule network for feature learning and classification. Chen et. al., proposed a 3D CNN model for feature extraction and classification [21]. Similarly, Zhong et. al., [22] proposed a spatial-spectral residual network for HSIC in which the residual blocks used identity mapping to connect 3D convolutional layers. Mou et. al., [23] proposed an unsupervised HSIC to further explore the residual CNNs.

M. Ahmad and M. S. Sarfraz are with the Department of Computer Science, National University of Computer and Emerging Sciences, Islamabad, Chiniot-Faisalabad Campus, Chiniot 35400, Pakistan; E-mail: mahmad00@gmail.com

A. M. Khan and M. Mazzara are with Innopolis University, Innopolis 420500, Russia; E-mail: a.khan@innopolis.ru; m.mazzara@innopolis.ru

S. Distefano and M. Ahmad are with Dipartimento di Matematica e Informatica—MIFT, University of Messina, Messina 98121, Italy; E-mail: sdistefano@unime.it

M. Ali is with the Department of Computer Engineering, Khwaja Fareed University of Engineering and Information Technology, Rahim Yar Khan, 64200, Pakistan.

Our review of the literature revealed several shortcomings, including but not limited to;

1) Though CNNs have become a promising method for HSIC, their memory requirement and high computational complexity makes it challenging to accelerate their performance. This work investigates their application to HSIC targetting high accuracy but under controlled computational cost, in terms of time it takes for them to converge. To achieve this, our work progressively modifies a baseline model while preserving its accuracy and reducing its time complexity.
2) Preserving channel relationship information is a challenging problem. CNN models are usually trained on reshaped spectral bands or use single band (gray-scale) information (containing different properties), resulting in failure to extract the "fine structural/spatial information of HSI". Furthermore, the high inter-class similarity, intra-class variability, overlapping and nested regions of HSI data makes classification a challenging problem. To overcome the said issue, the proposed architecture first divides the HSI cube into small overlapping 3D patches. These patches are processed to generate 3D feature maps using 3D kernel function over multiple contiguous bands to preserve the joint spatial and spectral information for the feature learning process which exploits important discriminatory information for HSIC.
3) As a preprocessing step, incremental Principle Component Analysis (iPCA) is employed to reduce the redundancy among the bands to process the few important wavelengths out of the entire HSI cube. Finally, to increase the number of spatial-spectral feature maps, four 3D convolutional layers are deployed to ensure that the model is able to discriminate the spatial information within different spectral bands without any loss.

In a nutshell, our end-to-end trained model requires fewer parameters, which significantly reduces the time it takes for the model to converge without compromising its accuracy, which is better than the existing models, as evident by our experimental results. We performed a comparative study with several state-of-the-art 2D/3D CNN methods proposed in the literature, and the experimental/comparative results confirm the superiority of our approach.

The rest of the paper is structured as follows: Section II presents the proposed methodology. Section III describes the experimental Datasets, Results, and discussion. Finally Section IV concludes the paper with possible future research directions.

II. Proposed Methodology

Let us assume a HSI can be expressed as $X = [x_1, x_2, x_3, ..., x_L]^T \in R^{L\times(N\times M)}$ consisting of $N \times M$ samples associated with C classes per band with total L bands, in which each sample is represented as (x_i, y_j), where y_j is the class label of x_i sample. The HSI pixels exhibit high inter-class similarity, high intra-class variability, overlapping, and nested regions. To overcome the aforesaid issues, iPCA is applied to eliminate the redundant bands. iPCA reduces the number of bands (L to B, where $B \ll L$) while maintaining the spatial dimensions as shown in Figure 1[1].

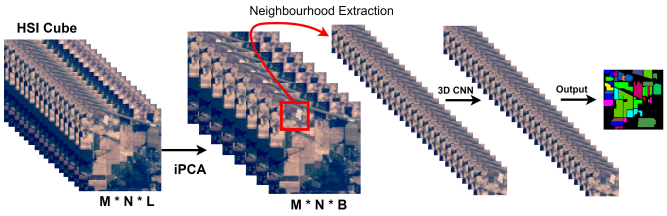


Fig. 1: Proposed 3D CNN Model for HSIC. 3D CNN Model details, i.e., the number of 3D Convectional and fully connected layers, can be found in Table I.

TABLE I: Layer based Summary of our Proposed 3D CNN Model architecture shown in Figure 2 with Window Size set as 11×11.

Layer	Output Shape	# of Parameters
Input Layer	(11, 11, 20, 1)	0
Conv3D_1 (Conv3D)	(9, 9, 14, 8)	512
Conv3D_2 (Conv3D)	(7, 7, 10, 16)	5776
Conv3D_3 (Conv3D)	(5, 5, 8, 32)	13856
Conv3D_4 (Conv3D)	(3, 3, 6, 64)	55360
Flatten_1 (Flatten)	(3456)	0
Dense_1 (Dense)	(256)	884992
Dropout_1 (Dropout)	(256)	0
Dense_2 (Dense)	(128)	32896
Dropout_2 (Dropout)	(128)	0
Dense_3 (Dense)	(# of Classes)	774
In total, **994,166** trainable parameters are required		

In order to pass the HSI cube to the model, it must be divided into a small overlapping 3D spatial patches on which the ground labels are formed based on the central pixel, as shown in Figure 2. The process creates neighboring patches $P \in R^{S\times S\times B}$ centered at the spatial location (a, b) covering $(S \times S)$ spatial windows [15]. The total of n patches given by $(M–S + 1) \times (N–S + 1)$. Thus, these patches cover the width from $\frac{a-(S-1)}{2}$ to $\frac{a+(S-1)}{2}$ and height from $\frac{b-(S-1)}{2}$ to $\frac{b+(S-1)}{2}$.

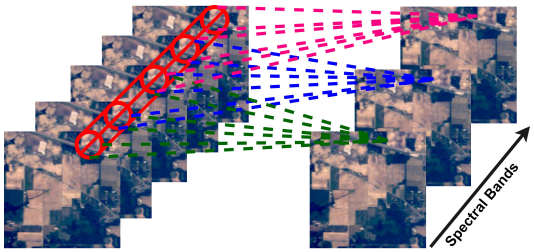


Fig. 2: 3D Convolution Operation

The input patches are first convolved with a 3D kernel function which computes the sum of the dot product between kernel function and input patch [24], [15]. Later these learned features are processed through an activation function that introduces the nonlinearity. The activation values at spatial position (x, y, z) in the i^{th} layer and j^{th} feature map is denoted as $v_{i,j}^{x,y,z}$. Thus, the final model can be created as follows:

[1]Demo code is available at GitHub

$$v_{i,j}^{x,y,z} = \mathcal{F}\Bigg(\sum_{\tau=1}^{d_{i-1}}\sum_{\lambda=-\nu}^{\nu}\sum_{\rho=-\gamma}^{\gamma}\sum_{\phi=-\delta}^{\delta} w_{i,j,\tau}^{\lambda,\rho,\phi} \times v_{(i-1),\tau}^{(x+\lambda),(y+\rho),(z+\phi)} + b_{i,j}\Bigg)$$

where $\mathcal{F}$ is an activation function, d_{i-1} be the number of 3D feature maps at $(i-1)^{th}$ layer and $w_{i,j}$ be the depth of the kernel, $b_{i,j}$ is the bias, $2\delta+1$, $2\gamma+1$ and $2\nu+1$ be the height, width and depth of the kernel.

In short, the proposed 3D CNN convolutional kernels are as follows: $3D\_conv\_layer1 = 8 \times 3 \times 3 \times 7 \times 1$ where $K_1^1 = 3, K_2^1 = 3$ and $K_3^1 = 7$. $3D\_conv\_layer2 = 16 \times 3 \times 3 \times 5 \times 8$ where $K_1^2 = 3, K_2^2 = 3$ and $K_3^2 = 5$. $3D\_conv\_layer3 = 32 \times 3 \times 3 \times 3 \times 16$ where $K_1^3 = 3, K_2^3 = 3$ and $K_3^3 = 3$ and finally $3D\_conv\_layer4 = 64 \times 3 \times 3 \times 3 \times 16$ where $K_1^3 = 3, K_2^3 = 3$ and $K_3^3 = 3$. To increase the number of spatial-spectral feature maps, four 3D convolutional layers are deployed before the flatten layer to make sure the model is able to discriminate the spatial information within different spectral bands without any loss. Further details regarding the proposed model can be found in Table I. The total number of parameters (i.e., tune-able weights) of our proposed 3D CNN model is $994,166$. The weights are initially randomized and optimized using Adam optimizer back-propagation with a soft-max loss function. The weights are updated using a mini-batch of size 256 with 50 epochs without batch normalization and augmentation.

III. Experimental Datasets and Results

The Salinas dataset (SD) was acquired over Salinas Valley California using AVIRIS sensor. SD is of size $512\times217\times224$ with a 3.7 meter spatial resolution with 512×217 is spatial and 224 spectral dimensions. SD consists of vineyard fields, vegetables, and bare soils. SD consist of 16 classes. A few water absorption bands $108-112, 154-167$, and 224 are removed before analysis.

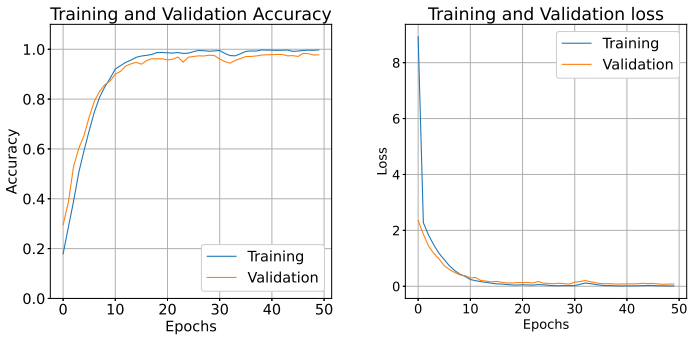
Indian Pines Dataset (IPD) is obtained over northwestern Indiana's test site by Airborne Visible / Infrared Imaging Spectrometer (AVIRIS) sensor. IPD is of size $145 \times 145 \times 224$ in the wavelength range $0.4-2.5\times10^{-6}$ meters where 145×145 is the spatial and 224 spectral dimensions. IPD consists of $1/3$ forest and $2/3$ agriculture area and other naturally evergreen vegetation. Some crops in the early stages of their growth are also present with approximately less than 5% of total coverage. Low-density housing, building, and small roads, two dual-lane highway and a railway line are also a part of IPD. The IPD ground truth comprised of 16 classes which are not mutually exclusive. The water absorption bands have been removed before the experiments thus the remaining 200 bands are used in this experiment.

Pavia University Dataset (PUD) gathered over Pavia in northern Italy using a Reflective Optics System Imaging Spectrometer (ROSIS) optical sensor. PUD consists of 610×610 spatial and 103 spectral bands with a spatial resolution of 1.3 meters. PUD ground truth classes are 9. Further details about the experimental datasets can be found at [25].

All the experiments were performed on an online platform known as Google Colab [26]. Google Colab is an online platform that requires a good speed of internet to run any environment. Google Colab provides an option to execute the codes on python 3 notebook with Graphical Processing Unit (GPU), 25 GB of Random Access Memory (RAM) and 358.27 GB of could storage for data computation. In all the experiments, the initial Test/Train set is divided into a $30/70\%$ ratio on which Training samples (70% of the entire population) are further divided into $50/50\%$ for the Training and Validation set.

For fair comparisons, the learning rate for all the experiments is set to 0.001, $relu$ as an activation function is used for all layers except last on which $softmax$ is used, the patch sizes are set as $11\times11\times20$, $13\times13\times20$, $15\times15\times20$, $17\times17\times20$, $19\times19\times20$, $21\times21\times20$ and $25\times25\times20$, respectively with 20 most informative bands selected by iPCA method. For evaluation purposes, the Average Accuracy (AA), Overall Accuracy (OA), and Kappa (κ) coefficient have been computed from the confusion matrices. AA represents the average class-wise classification performance, OA is computed as the number of correctly classified examples out of the total test examples, and finally, κ is known as a statistical metric that considered the mutual information regarding a strong agreement among classification and ground-truth maps. Along with OA, AA, and κ metrics, several statistical tests are also being considered such as F1-Score, Precision, and Recall.

The convergence loss and accuracy of our proposed 3D CNN model for a 50 number of epochs are shown in Figure 3. From these figures, one can conclude that the proposed model is converged almost around 20 epochs.



(a) Accuracy (b) Loss

Fig. 3: Accuracy and Loss for Training and Validation sets on Indian Pines Dataset with 11×11 window patch correspond to the 50 number of Epochs.

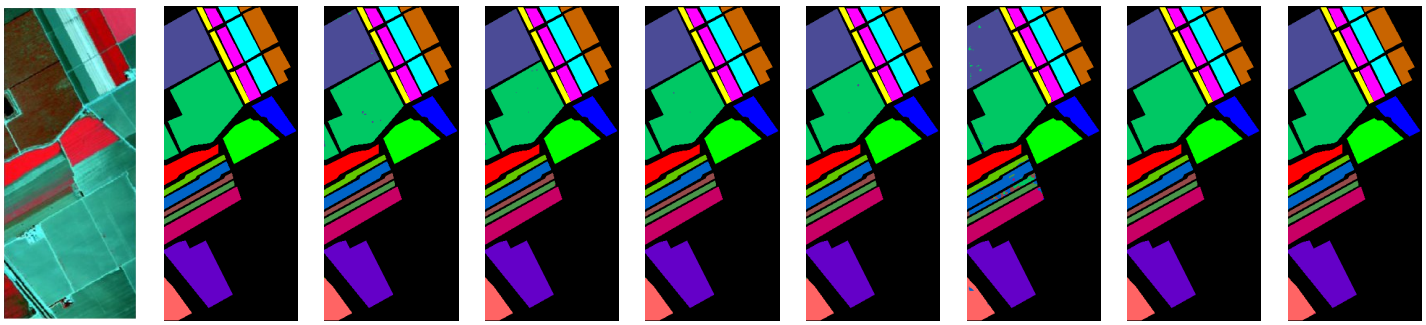
Whereas, the computational time of our proposed model is shown in Table II which reveals a fast convergence and computational efficiency of our proposed model. The computational time highly depends on the speed of the internet and available RAM.

TABLE II: Computational time in minutes for all the experimental datasets with several window sizes.

Dataset	Proposed with Several Window Sizes							2D CNN	3D-CNN	MS-3D-CNN
	11×11	13×13	15×15	17×17	19×19	21×21	25×25			
SL-A	0.22	0.23	0.56	0.28	0.98	0.37	0.45	—	—	—
SL	1.34	1.41	1.60	2.00	3.17	2.63	3.52	2.2	74.0	25.5
IP	0.33	0.33	0.61	0.78	0.62	0.58	0.76	1.9	15.2	14.1
PU	2.16	5.26	1.35	2.00	2.46	2.14	2.83	1.8	58.0	20.3

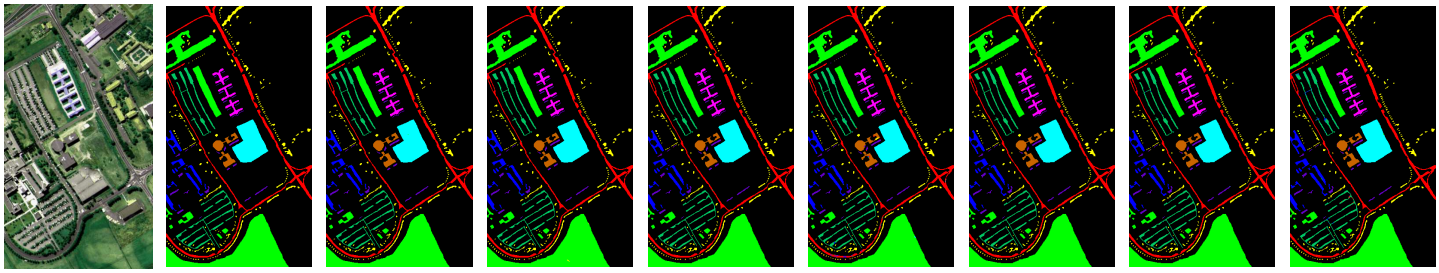
(a) GT (b) GT (c) 11×11 (d) 13×13 (e) 15×15 (f) 17×17 (g) 19×19 (h) 21×21 (i) 25×25

Fig. 4: **Indian Pines Dataset** Ground Truths for each spatial dimensions processed through our proposed model.



(a) SA (b) GT (c) 11×11 (d) 13×13 (e) 15×15 (f) 17×17 (g) 19×19 (h) 21×21 (i) 25×25

Fig. 5: **Salinas Dataset** Ground Truths for each spatial dimensions processed through our proposed model.



(a) GT (b) GT (c) 11×11 (d) 13×13 (e) 15×15 (f) 17×17 (g) 19×19 (h) 21×21 (i) 25×25

Fig. 6: **Pavia University Dataset** Ground Truths for each spatial dimensions processed through our proposed model.

(a) GT (b) GT (c) 11×11 (d) 13×13 (e) 15×15 (f) 17×17 (g) 19×19 (h) 21×21 (i) 25×25

Fig. 7: **Salinas-A Dataset** Ground Truths for each spatial dimensions processed through our proposed model.

TABLE III: Impact of window size on our proposed model

Window	PU			IP			SA			SL-A		
	OA	AA	κ	OA	AA	κ	OA	AA	κ	OA	AA	κ
11×11	99.94	99.89	99.92	88.65	83.52	87.11	99.80	99.91	99.78	100	100	100
13×13	99.81	99.65	99.75	95.38	94.14	94.72	99.93	99.94	99.93	100	100	100
15×15	99.85	99.62	99.80	93.69	93.09	92.79	99.99	99.99	99.99	100	100	100
17×17	99.05	98.49	98.75	91.80	91.74	90.62	99.95	99.97	99.95	99.93	99.93	99.92
19×19	99.93	99.78	99.91	93.13	93.42	92.15	98.04	94.02	97.81	100	100	100
21×21	99.78	99.43	99.72	94.34	91.31	93.52	99.99	99.99	99.99	100	100	100
25×25	98.79	97.67	98.39	97.75	96.17	97.44	99.96	99.93	99.95	100	100	100

The accuracy analysis i.e., OA, AA, and κ based on the impact of spatial dimensions processed by the proposed model is presented in Table III. While looking into the Table III, one can conclude that the window size of 11×11 is enough for Pavia University, Salinas, and Salinas-A dataset whereas the window size of 13×13 and 25×25 both works almost the same. Furthermore, the classification maps (geographical locations for each class) according to the different number of window sizes (spatial dimensions) are shown in Figures 4-7. In regards to comparison, the proposed model is compared with several state-of-the-art methods published in recent years. From experimental results listed in Table IV one can conclude that the proposed model has competitive results and to some extent better in regards to the other methods. The comparative methods includes Multi-scale-3D-CNN [27], 3D/2D-CNN [24], [28]. All the comparative methods are being trained according to the settings mentioned in their respective papers. Experiments listed in Table IV shows the proposed method improves the results significantly then the state-of-the-art methods with even fewer training samples, number of

convolutional layers, number of filters, number of epochs, and above all, in less computational time.

TABLE IV: Comparative evaluations with State-of-the-art methods while considering 11×11 Spatial dimensions with even less number of training samples (i.e., $60/40\%$ (train/test) and $70/30\%$ (train/validation)).

Dataset	MS-3D-CNN			3D-CNN			2D-CNN			**Proposed**		
	OA	AA	Kappa	OA	AA	Kappa	OA	AA	Kappa	OA	AA	Kappa
PU	95.95	97.52	93.40	96.34	97.03	94.90	96.63	94.84	95.53	**98.40**	**97.89**	**97.89**
IP	81.39	75.22	81.20	82.62	76.51	79.25	80.27	68.32	75.26	**97.75**	**94.54**	**97.44**
SA	94.20	96.66	93.61	85.00	89.63	83.20	96.34	94.36	95.93	**98.06**	**98.80**	**97.85**

IV. Conclusion

Hyperspectral Image Classification (HSIC) is a challenging task due to high inter-class similarity, high intra-class variability, overlapping, and nested regions. Though 2D Convolutional Neural Networks (CNNs) have emerged as a viable approach for HSIC, 3D CNNs are a better alternative because accurate HSIC depends on both Spectral-Spatial information. However, 3D CNN can be highly computational complex due to their volume and spectral dimensions. Therefore, this paper proposed a compact and fast 3D CNN model, which overcame the above-mentioned challenges. Our model provided state-of-the-art experimental results in a computationally efficient fashion on four Hyperspectral benchmark datasets. We resolved the problem of inter-class similarity and high intra-class variability using 3D convolution-based spatial-spectral information. To summarize, our end-to-end trained 3D CNN has fewer parameters, better recognition accuracy, and fast convergence time than existing 2D/3D CNN models. The experimental results reveal that the proposed method outperformed the state of the art methods on various public benchmarks while being less complex than the conventional 3D CNN models.

References

[1] S. Mei, X. Yuan, J. Ji, Y. Zhang, S. Wan, and Q. Du, "Hyperspectral image spatial super-resolution via 3d full convolutional neural network," *Remote Sensing*, vol. 9, p. 1139, 11 2017.

[2] M. Ahmad, A. Khan, A. M. Khan, M. Mazzara, S. Distefano, A. Sohaib, and O. Nibouche, "Spatial prior fuzziness pool-based interactive classification of hyperspectral images," *Remote Sensing*, vol. 11, no. 9, May. 2019. [Online]. Available: https://www.mdpi.com/2072-4292/11/9/1136

[3] Y. Wang, W. Yu, and Z. Fang, "Multiple kernel-based svm classification of hyperspectral images by combining spectral, spatial, and semantic information," *Remote Sensing*, vol. 12, no. 1, p. 120, 2020.

[4] A. Alcolea, M. E. Paoletti, J. M. Haut, J. Resano, and A. Plaza, "Inference in supervised spectral classifiers for on-board hyperspectral imaging: An overview," *Remote Sensing*, vol. 12, no. 3, p. 534, 2020.

[5] M. Ahmad, S. Shabbir, D. Oliva, M. Mazzara, and S. Distefano, "Spatial-prior generalized fuzziness extreme learning machine autoencoder-based active learning for hyperspectral image classification," *Optik-International Journal for Light and Electron Optics*, 2020. [Online]. Available: https://www.sciencedirect.com/science/article/abs/pii/S0030402619316109

[6] D. Tuia, M. Volpi, M. D. Mura, A. Rakotomamonjy, and R. Flamary, "Automatic feature learning for spatio-spectral image classification with sparse svm," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 52, pp. 6062–6074, 2014.

[7] P. Ghamisi, M. Dalla Mura, and J. A. Benediktsson, "A survey on spectral–spatial classification techniques based on attribute profiles," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 53, no. 5, pp. 2335–2353, 2015.

[8] J. A. Benediktsson, J. A. Palmason, and J. R. Sveinsson, "Classification of hyperspectral data from urban areas based on extended morphological profiles," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 43, no. 3, pp. 480–491, 2005.

[9] S. Jia, X. Zhang, and Q. Li, "Spectral–spatial hyperspectral image classification using $\ell_{1/2}$ regularized low-rank representation and sparse representation-based graph cuts," *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, vol. 8, no. 6, pp. 2473–2484, 2015.

[10] M. Dalla Mura, A. Villa, J. A. Benediktsson, J. Chanussot, and L. Bruzzone, "Classification of hyperspectral images by using extended morphological attribute profiles and independent component analysis," *IEEE Geoscience and Remote Sensing Letters*, vol. 8, no. 3, pp. 542–546, 2011.

[11] M. Ahmad, M. Mazzara, R. A. Raza, S. Distefano, M. Asif, M. S. Sarfraz, A. M. Khan, and A. Sohaib, "Multiclass non-randomized spectral–spatial active learning for hyperspectral image classification," *Applied Sciences*, vol. 10, no. 14, July. 2020. [Online]. Available: https://www.mdpi.com/2076-3417/10/14/4739

[12] Y. Zhong, A. Ma, and L. Zhang, "An adaptive memetic fuzzy clustering algorithm with spatial information for remote sensing imagery," *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, vol. 7, no. 4, pp. 1235–1248, 2014.

[13] L. Shen and S. Jia, "Three-dimensional gabor wavelets for pixel-based hyperspectral imagery classification," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 49, no. 12, pp. 5039–5046, 2011.

[14] Y. Qian, M. Ye, and J. Zhou, "Hyperspectral image classification based on structured sparse logistic regression and three-dimensional wavelet texture features," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 51, no. 4, pp. 2276–2291, 2013.

[15] S. Roy, G. Krishna, S. R. Dubey, and B. Chaudhuri, "Hybridsn: Exploring 3-d-2-d cnn feature hierarchy for hyperspectral image classification," *IEEE Geoscience and Remote Sensing Letters*, vol. 17, pp. 277–281, 06 2019.

[16] Y. Li and L. He, "An improved hybrid CNN for hyperspectral image classification," in *Eleventh International Conference on Graphics and Image Processing (ICGIP 2019)*, Z. Pan and X. Wang, Eds., vol. 11373, International Society for Optics and Photonics. SPIE, 2020, pp. 485 – 490. [Online]. Available: https://doi.org/10.1117/12.2557384

[17] B. Fang, Y. Bai, and Y. Li, "Combining spectral unmixing and 3d/2d dense networks with early-exiting strategy for hyperspectral image classification," *Remote Sensing*, vol. 12, p. 779, 02 2020.

[18] L. Huang and Y. Chen, "Dual-path siamese cnn for hyperspectral image classification with limited training samples," *IEEE Geoscience and Remote Sensing Letters*, pp. 1–5, 2020.

[19] M. E. Paoletti, J. M. Haut, R. Fernandez-Beltran, J. Plaza, A. J. Plaza, and F. Pla, "Deep pyramidal residual networks for spectral–spatial hyperspectral image classification," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 57, no. 2, pp. 740–754, 2019.

[20] M. E. Paoletti, J. M. Haut, R. Fernandez-Beltran, J. Plaza, A. Plaza, J. Li, and F. Pla, "Capsule networks for hyperspectral image classification," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 57, no. 4, pp. 2145–2160, 2019.

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

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

[23] L. Mou, P. Ghamisi, and X. X. Zhu, "Unsupervised spectral–spatial feature learning via deep residual conv–deconv network for hyperspectral image classification," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 56, no. 1, pp. 391–406, 2018.

[24] Y. Li, H. Zhang, and Q. Shen, "Spectral–spatial classification of hyperspectral imagery with 3d convolutional neural network," *Remote Sensing*, vol. 9, p. 67, 01 2017.

[25] *Hyperspectral Datasets Description*, 2020 (accessed 2020-01-12), http://www.ehu.eus/ccwintco/index.php/Hyperspectral_Remote_Sensing_Scenes.

[26] T. Carneiro, R. V. M. Da Nóbrega, T. Nepomuceno, G.-B. Bian, V. H. C. De Albuquerque, and P. P. Reboucas Filho, "Performance analysis of google colaboratory as a tool for accelerating deep learning applications," *IEEE Access*, vol. 6, pp. 61 677–61 685, 2018.

[27] M. He, B. Li, and H. Chen, "Multi-scale 3d deep convolutional neural network for hyperspectral image classification," in *2017 IEEE International Conference on Image Processing (ICIP)*, 2017, pp. 3904–3908.

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