International Journal of Informatics and Communication Technology (IJ-ICT)
Vol. 14, No. 2, August 2025, pp. 393∼404
ISSN: 2252-8776, DOI: 10.11591/ijict.v14i2.pp393-404

Categorizing hyperspectral imagery using convolutional neural networks for land cover analysis

Assia Nouna[1,2], Soumaya Nouna[2], Mohamed Mansouri[2], Achchab Boujamaa[2]
[1]Laboratory LAMSAD, ENSA Berrechid, Department of Mathematics and Informatics, Hassan First University of Settat, Berrechid, Morocco
[2]Laboratory LAMSAD, ENSA Berrechid, Hassan First University of Settat, Berrechid, Morocco

Article Info

Article history:

Received Aug 30, 2024
Revised Nov 28, 2024
Accepted Dec 15, 2024

Keywords:

Convolutional neural network
Deep learning
HSI classification
Hyperspectral images
K-nearest neighbors
Support vector machine

ABSTRACT

Categorizing hyperspectral imagery (HSI) is crucial in various remote sensing applications, including environmental monitoring, agriculture, and urban planning. Recently, numerous approaches have emerged, with convolutional neural network (CNN)-based algorithms demonstrating remarkable performance in HSI classification due to their ability to learn complex spatial-spectral features. However, these algorithms often require significant computational resources and storage capacity, which can be limiting in practical applications. In this study, we propose a novel CNN architecture tailored for HSI classification within the spectral domain, focusing on optimizing computational efficiency without compromising accuracy. The architecture leverages advanced spectral feature extraction techniques to enhance classification performance. Experimental evaluations on multiple benchmark hyperspectral datasets reveal that the proposed approach not only improves classification accuracy but also achieves a superior balance between performance and computational demand compared to traditional methods like K-nearest neighbors (KNN) and other deep learning-based techniques. Our results demonstrate the potential of the proposed CNN model in advancing the field of HSI classification, offering a viable solution for real-world applications with constrained computational resources.

Corresponding Author:

Assia Nouna
Laboratory LAMSAD, ENSA Berrechid, Department of Mathematics and Informatics
Hassan First University of Settat
Berrechid, Morocco
Email: a.nouna@uhp.ac.ma

1. INTRODUCTION

Remote sensors capture hyperspectral imagery (HSI) [1], which contain several hundred as many channels of observation at highly spectral resolution. The richness of the spectral information present in HSI allows for the development of numerous traditional classification approaches, including K-nearest neighbors (KNN), logistic regression, and minimum distance [2]. In recent times, there have been proposed some improved methods for extracting features and advanced classifiers, for instance, the spectral and spatial classification [3] and Fisher's local discriminant analysis, which have shown to be more effective.

The support vector machine (SVM) [4] is widely regarded as a reliable and effective approach to the tasks of hyperspectral classification, mainly when dealing with limited training samples. SVM functions by seeking an optimal decision hyperplane that can separate two-class data by utilizing a high-dimensional feature space included in the kernel. Several adaptations to the SVM for the classification of HSIs have been introduced in the current literature to enhance classification performance [5], [6].

The classification of remote sensing data has been explored using neural networks, like the multilayer perceptron (MLP) [7] and the radial basis function neural networks. Ratle *et al.* [8], a semisupervised neural network framework was proposed for HSI classification on a large scale. However, in remote sensing classification tasks, SVM has been shown to outperform classical neural networks in classification accuracy and computational cost. Despite this, [9] considers that deep neural network (DNN) architecture as a powerful classification model that can compete with SVM in terms of classification performance.

Deep learning methods have shown great potential in various domains. Convolutional neural networks (CNNs) [9] are particularly effective for processing visual-related tasks within deep learning. CNNs are a class of multilayer models that take inspiration from biological neural networksthat can be end-to-end trained, starting with the raw pixels of the image and ending with the classifier. The CNN concept was initially presented in [10], further developed by [11], and refinement and simplification in subsequent work [12].

Recently, CNNs have achieved better results some of the conventional approaches, including the human performers [13], in a variety of tasks related to vision, such as classification of images [14], [15], detection of objects, the labeling of scenes, classification of house numbers, and recognition of faces. Additionally, CNNs have also applied to speech recognition [16] and proved to be effective models for understanding visual image content. In a study by [17], researchers utilized CNNs for HSI classification [18], [19], specifically using stacked autoencoders (SAEs) to extract distinctive features. These findings confirm that CNNs are an efficient class of models for solving visual-related problems and can produce state-of-the-art results in visual image classification. It has been shown that CNNs are superior in classification performance compared to traditional SVM and DNNs on tasks related to images [20], [21]. There is a lack of literature on the application of CNNs with multiple layers for HSI classification [22], as CNNs have primarily been used in visual-related problems [23]-[25].

The paper demonstrates that hyperspectral data can be accurately classified using CNNs with a suitable layer architecture. Through experiments, it was observed that traditional CNN models, like the LeNet-5 based on two convolutional layers, are unsuitable. Instead, a 20-layer CNN architecture with supervised HSI classification weights was proposed, which was proven to be a more effective and straightforward approach. Our proposed method has been proven to outperform the traditional KNN and conventional deep learning architectures through various experiments. To our knowledge, this is -one of the first times a multi-layer CNN has been utilized for the HSI classification, resulting in outstanding results. This paper is structured into different sections. Section 2 provides a concise overview of CNNs, where the traditional CNN structure and -their formation process are discussed. In section 3, we conduct experiments to compare the efficiency of our approach to K-nearest neighbor (KNN) and DNN using commonly used datasets. The paper concludes by summarizing the obtained results in section 4.

2. DESCRIPTION OF METHODOLOGY

This study presents a novel CNN architecture specifically designed for HSI classification within the spectral domain. The methodology consists of three primary phases: (i) An introduction to CNNs, emphasizing their suitability for processing spectral data and their advantages over traditional classification methods like KNN; (ii) A detailed exploration of HSI classification techniques using CNNs, highlighting the integration of spectral features; and (iii) A comprehensive description of the proposed CNN model, including its architecture, training strategy, forward propagation, and backpropagation mechanisms. The training phase utilizes a carefully curated set of hyperspectral datasets, applying rigorous preprocessing and evaluation protocols to ensure high accuracy and robust performance. Experimental results demonstrate the proposed method's superior classification accuracy and efficiency compared to conventional approaches, establishing its potential for advanced remote sensing applications. This section provides a step-by-step account of the experimental setup, implementation details, and validation metrics to ensure replicability and transparency of the research findings.

2.1. Convolution neural network

CNNs produce feed-forward neural networks composed of diverse layers of convolution, layers of maximum pooling, and layers of fully connected. A CNN can benefit from local spatial correlation by implementing a specific pattern of the local connections among the neurons in the adjacent layers. The convoluted layers are alternated by layers of maximum pooling, reproducing the complex and simple nature of the cells of the mammalian visual cortex. A CNN is composed of a pair or several pairs of maximum pooling and convolution layers and results in a fully connected neural network. The convolutional network architecture can be found in Figure 1.

In other words, inside the DNN, every hidden activation x_i can be calculated simply by taking the entire input X and multiplying it by the W weights inside this layer. But inside CNN, every hidden activating is calculated using the multiplication of the short input by the W weights. Then W weights are then distributed over all the input areas. The neurons which are on the identical layer are given identical weights. Weight sharing is a fundament of CNNs, as it reduces the overall number of training components and results in better model performance and more efficient training. Usually, the convolutional layer is succeeded by a maximum pooling layer.

A CNN detects features across the input data by replicating weights. However, when an input image is shifted, the feature detecting neuron also shifts. To address this, we use pooling to render characteristics invariant to location. This is achieved by summarizing multiple neuron outputs in convolutional layers with the pooling function, typically maximal. The max pooling function retrieves a maximum data value of an input, partitioning the input data into the non-overlapping window and outputting the maximal values within every sub-region. This enables the complexity of the calculation for the higher layers to be reduced and the invariance of the translation to be ensured. Finally, to enable the classification, a CNN's calculation chain ends with a fully connected network integrating the information on all the locations of all the feature maps in the lower layer.

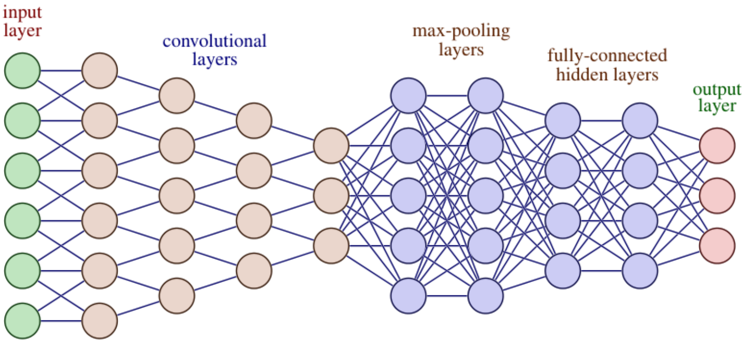


Figure 1. U-Net architecture

2.2. HSI classification based on the CNN

CNN HSI classification is a technique used to classify HSIs using CNN. HSIs are multidimensional data that contain information about the reflectance or emission of light at different wavelengths of the electromagnetic spectrum. These images have many applications in areas such as environmental monitoring, mining exploration and urban planning.

The CNN model is trained to classify each pixel in the HSI into one of several predefined classes. The model takes the HSI as input and applies convolutional filters to extract features from the spectral data. These features are then passed through several layers of the network, including pooling layers, fully connected layers, and activation functions, to reduce the dimensionality and classify the pixels into their respective classes.

The main advantage of using CNN-Based HSI Classification is that it can automatically learn relevant features from the input data, eliminating the need for manual feature extraction. This approach allows for improved accuracy in classification, as the model can identify subtle differences in the spectral characteristics of different classes. Additionally, CNN-Based HSI Classification can handle high-dimensional data, which makes it well-suited for processing HSIs.

Overall, the HSI classification based on CNN is a powerful and efficient technique for the analysis of HSIs and has many practical applications in various fields, including remote sensing, agriculture, and geology.

2.3. The proposed HSI classification based on CNN

2.3.1. The architecture of proposed CNN

The CNN depends on the way in which convolutional and maximum pooling layers are constructed and the way in which networks are formed. So, the procedure adopted for HSI classification in the proposed work can be illustrated by using Figure 2.

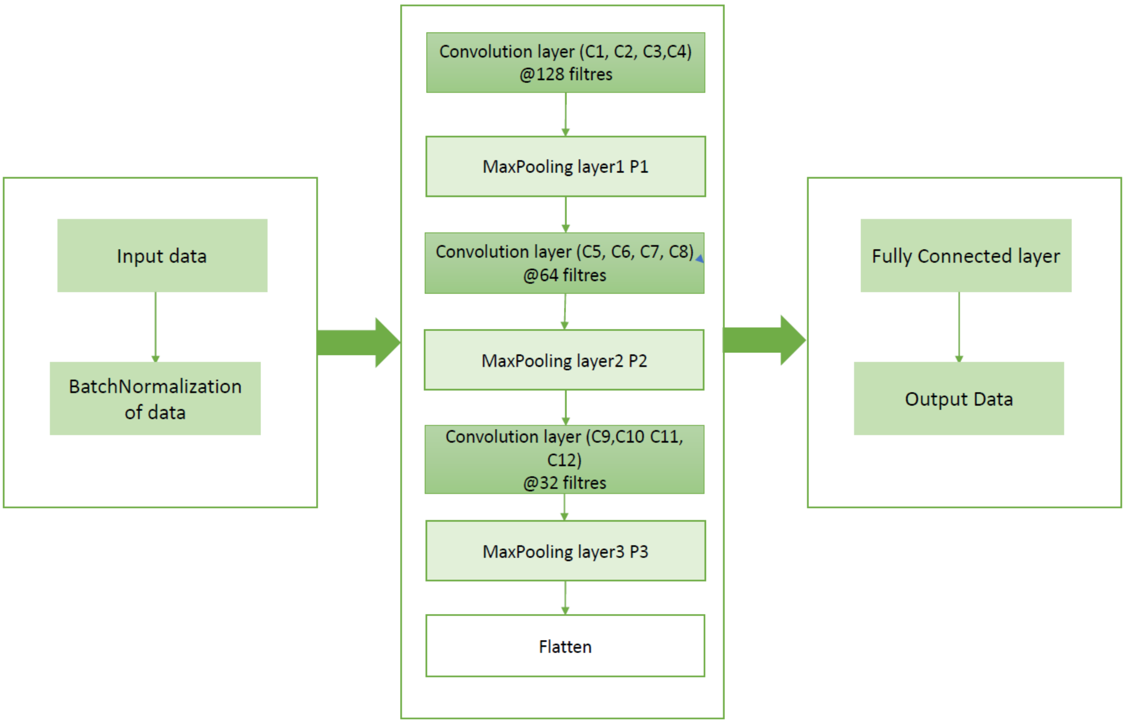

Input data
BatchNormalization of data
Convolution layer (C1, C2, C3,C4) @128 filtres
MaxPooling layer1 P1
Convolution layer (C5, C6, C7, C8) @64 filtres
MaxPooling layer2 P2
Convolution layer (C9,C10 C11, C12) @32 filtres
MaxPooling layer3 P3
Flatten
Fully Connected layer
Output Data

Figure 2. The proposed CNN classifier architecture

Within our architecture, the HSI data is first taken into account and normalized. Hyperspectral data are split into training and test data with merged features and are delivered as inputs to the CNN algorithm to achieve classification. The CNN architecture is composed primarily of tree groups (C_1, C_2, C_3) of convolution layers, each group C_1 contains four layers, so in total we have twelve layers $(c_1, c_2.....c_{12})$ and tree pooling layers (P_1, P_2, P_3). Finaly the filters used in the 3 groups are currently $K_1 = 128$, $K_2 = 64$, and $K_3 = 32$ respectively.

We considered the input training data size as $(X_1, Y_1, 1)$, when we applied the first step with the first group of convolutional layer C_1 with K_1 filter, the data are transformed into (X_1, Y_1, K_1) and (X_2, Y_2, K_2) where $X_2 = \frac{X_1}{2}$ and $Y_2 = \frac{Y_1}{2}$. After applying the second step with the second group of convolutional layer C_2 with K_2 filter, the data are transformed into (X_2, Y_2, K_2) and (X_3, Y_3, K_2) where $X_3 = \frac{X_2}{2}$ and $Y_3 = \frac{Y_3}{2}$. Lastly, the data become (X_3, Y_3, K_3) after applying the third step with the third group of convolutional layer C_3 and pooling layers P_3.

The finished output is classified with the SoftMax function. Moreover, the dropout and batch normalization (BN) layers are employed on the suggested network.

2.3.2. Training approaches

In this section, we outline the techniques we employed to train the space of parameters for the algorithm proposed for the CNN classifier. Firstly, every trained parameter with our CNN is randomly initialized to values from -1 to 1. The formation process comprises two main stages: forward propagation and Backpropagation. During a forward propagation stage, CNN computes a classification output of an input data based on the current set of parameters. Inside a subsequent Back propagation stage, trainable parameters are updated to minimize the difference between the real and desired classification outputs. The ultimate aim of this process is to reduce the discrepancy between the two outputs as much as possible.

A. Forward propagation:

The input layer of our CNN (with $L + 1$ layers and $L = 19$) consists of v_1 elements, while the output layer comprises v_{20} elements. Additionally, there are a number of hidden elements present in the convolutional layer C, pooling layer P, and fully connected layer F. These layers are commonly referred to as hidden layers.

Supposing that a_i is the input of the i layer, so we calculate a_{i+1} as,

$$a_{i+1} = g_i(z_i) \tag{1}$$

where

$$z_i = W_i^T a_i + b_i \tag{2}$$

and the W_i^T weight matrix for the i layer operates for the entry data, while the b_i vector of additive bias is used for the same layer. The g activation function used in the C -layers and F-layer is the ReLu function. The $\max(z)$ function is applied in the P-layer. Since our CNN classifier is designed to classify multiple classes. Finaly the output of layer F used a SoftMax function. This model is defined as a SoftMax regression model.

$$y = \frac{1}{\sum_{h=1}^{v_{11}} e^{W_{L,h}^T a_L + b_{L,h}}} \begin{bmatrix} e^{W_{L,1}^T a_L + b_{L,1}} \\ e^{W_{L,2}^T a_L + b_{L,2}} \\ \vdots \\ e^{W_{L,v_{20}}^T a_L + b_{L,v_{20}}} \end{bmatrix} \tag{3}$$

In each iteration, the probability of all classes is represented by the final probability indicated in the output layer as the y output vector, which is obtained by multiplying the a input vector with $L + 1$ ($y = a_{L+1}$).

B. Back-propagation:

In the process of back-propagation, the trainable parameters undergo updates through the utilization of the gradient descent technique. This approach involves the minimization of a cost function, coupled with the computation of the partial derivative of said function concerning each trainable parameter.

Once a CNN classifier's architecture and associated trainable parameters are defined, we have the ability to construct it and reload any saved parameters for the purpose of classifying HSI data. The classification process is analogous to the forward propagation step, whereby we can determine the classification outcome.

3. EXPERIMENTAL EVALUTION AND ANALYSES

This study focused on evaluating the effectiveness of a novel CNN architecture for HSI classification, addressing a gap in previous research where computational efficiency and spectral feature extraction were often overlooked. The first section, study area and data collection, describes the use of the Pavia University dataset, predominantly consisting of vegetation regions, to assess the classification performance of the proposed model. The second section, results and analysis, demonstrates that the proposed CNN method significantly improves classification accuracy for 5 out of 9 classes, particularly for vegetation types, compared to traditional KNNs methods. These results suggest a strong correlation between spectral feature extraction and improved classification outcomes, without compromising computational efficiency. The final section, findings and comparisons, shows that the proposed method outperforms existing techniques in terms of accuracy and computational requirements, while addressing the limitations of focusing on a single dataset. Future studies could extend this approach to more diverse environments, confirming its robustness and exploring optimization strategies under resource-constrained conditions. Overall, this work highlights the potential for CNN-based HSI classification to revolutionize remote sensing applications by enhancing accuracy without increasing resource demands.

3.1. Study area and data collection

3.1.1. Hyperspectral imaging

Hyperspectral remote sensing is the extraction of information from objects or sceneries on the Earth's surface using light collected by airborne or spaceborne sensors. Small, commercial, high spatial, and spectral resolution instruments have been increasingly used in lab-scale applications (industries also including environmental management, agriculture, urban planning, and military applications) using hyperspectral sensing and its imaging modality, hyperspectral imaging.

Millions of spatial coregistered images corresponding to various spectral channels compensate hyperspectral data. A HSIs structure is as follows: each pixel is represented like vector of the B-dimensional characteristics along the spectral dimension, which is referred to as the substance spectrum within that pixel. This abundance of data in each spatial area improves an ability to discern between various physical materials. As a result, hyperspectral photography opens up new avenues for the classification of pictures, a critical step in a vast range of applications such as precision agriculture.

3.1.2. Data set of hyperspectral images

The studies used Pavia data sets, each presenting an urban region, a spatial resolution of 1.3m per pixel, and several bands of 103 bands. The following section presents this data set.

Pavia is a dataset acquired over the city of Pavia, Italy, utilizing the ROSIS sensor and a ground sample distance (GSD) of 1.3 m. In Figure 3, Pavia University (103 bands, 610 340px) and Pavia Center (102 bands, 1096 715px) are the two sections. In this art, we will use the first part of the data (the Pavia University) which contains 9 categories of interest labeled throughout half of the surface (see in Table 1). They are made up of various urban components (including bricks, asphalt, and metals), as well as water and plants. As it is one of the largest labeled HSI datasets and enables the evaluation of the usage of HSI for future applications, it has long been one of the key reference datasets.

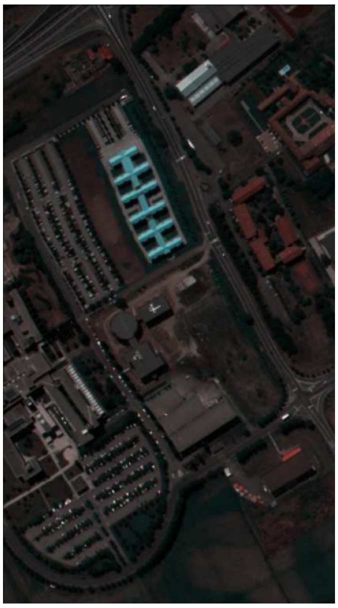


Figure 3. Pavia University composite image

Table 1. 9 classes of Pavia

Label	Class	Samples
1	Asphalt	6631
2	Meadows	18649
3	Gravel	2099
4	Trees	3064
5	Painted metal sheets	1345
6	Bare soil	5029
7	Bitumen	1330
8	Self-blocking bricks	3682
9	Shadows	947

In the original recorded image, there are 115 data channels (with a spectral range of 0.43 to 0.86 m). The studies were conducted using the remaining 103 bands after the 12 most noisy channels were deleted (see in Figure 4).

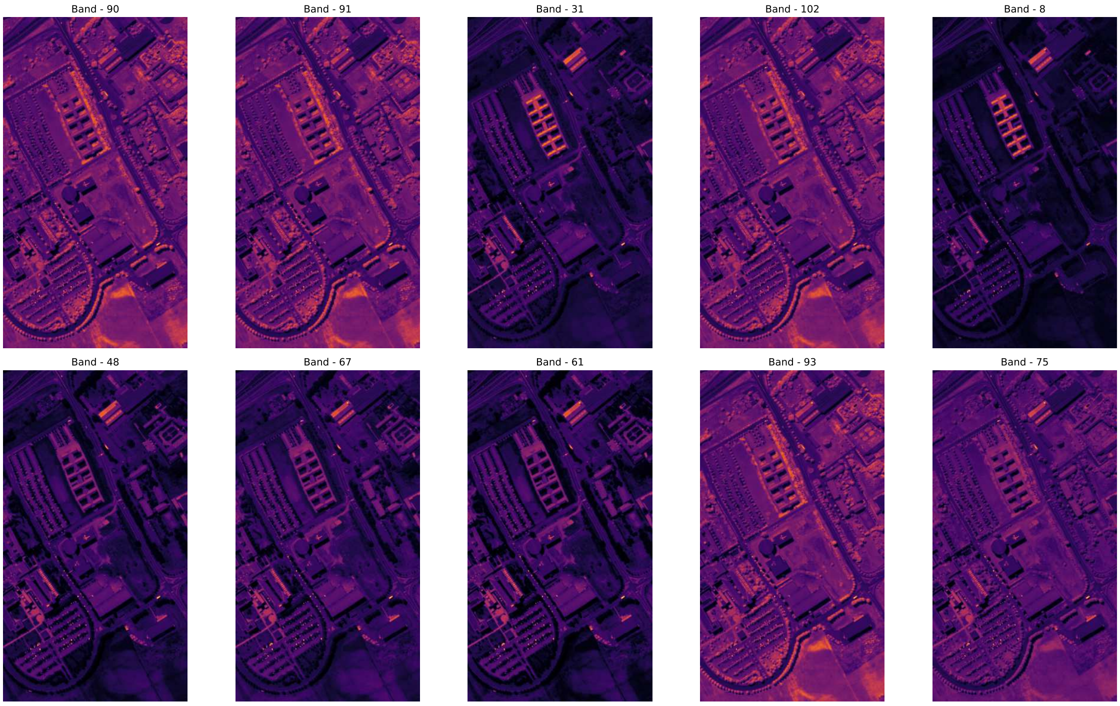


Figure 4. Bands of Pavia University

3.2. Result and analysis

3.2.1. Get the data ready for training and testing

Once the data has been integrated, it must be fed into the CNN network. Before this step, it is important to split the data into training and testing sets and to remove any background pixel samples. In the case of the Pavia University data, 1, 64, 624 pixels were removed for each band. This preparation process helps reduce the computing burden and more information can be found in Table 2.

Table 2. The number of training and test samples in the Pavia dataset

Label	Class	Samples	Training	Test
1	Asphalt	6631	4641	1990
2	Meadows	18649	13054	5604
3	Gravel	2099	1469	630
4	Trees	3064	2144	920
5	Painted metal sheets	1345	941	404
6	Bare soil	5029	3521	1508
7	Bitumen	1330	932	398
8	Self-blocking bricks	3682	2578	1104
9	Shadows	947	663	284
	Total	42776	29943	12833

3.2.2. Description of program

The program is written in Python using the Keras API, which is a high-level neural networks API that can run on top of TensorFlow. The Sequential class is imported from the Keras library and is used to initialize a sequential model. The model consists of multiple layers, including Conv1D, BN, MaxPooling1D, Dropout, and Dense (see in Table 3).

Table 3. Model layers

Layer (type)	Output shape	Param #
inputLayer (Conv1D)	(None, 101, 128)	512
Batch-normalization (Batch-normalization)	(None, 101, 128)	512
Layer1 (Conv1D)	(None, 99, 128)	49,280
Layer2 (Conv1D)	(None, 97, 128)	49,280
Layer3 (Conv1D)	(None, 95, 128)	49,280
Layer4 (Conv1D)	(None, 93, 128)	49,280
MaxPooling_Layer1 (MaxPooling1D)	(None, 46, 128)	0
Dropout1 (Dropout)	(None, 46, 128)	0
Layer5 (Conv1D)	(None, 44, 64)	24,640
Layer6 (Conv1D)	(None, 42, 64)	12,352
Layer7 (Conv1D)	(None, 40, 64)	12,352
Layer8 (Conv1D)	(None, 38, 64)	12,352
MaxPooling_Layer2 (MaxPooling1D)	(None, 19, 64)	0
Dropout2 (Dropout)	(None, 19, 64)	0
Layer9 (Conv1D)	(None, 17, 32)	6,176
Layer10 (Conv1D)	(None, 15, 32)	3,104
Layer11 (Conv1D)	(None, 13, 32)	3,104
Layer12 (Conv1D)	(None, 11, 32)	3,104
MaxPooling_Layer3 (MaxPooling1D)	(None, 5, 32)	0
Dropout3 (Dropout)	(None, 5, 32)	0
Flatten (Flatten)	(None, 160)	0
DenseLayer (Dense)	(None, 25)	4,025
OutputLayer (Dense)	(None, 10)	260

Conv1D is a one-dimensional convolutional layer that applies a convolution operation on the input data with a specified number of filters and kernel size. The activation function used for the convolutional layers is ReLU. BN is used to normalize the input data by adjusting and scaling the activations of the previous layers. MaxPooling1D is used to reduce the spatial size of the data by taking the maximum value within a specified pool size. Dropout is used to prevent over fitting by randomly dropping out a certain percentage of the input units. Flatten is used to convert the multidimensional input data into a one-dimensional array. Dense is a fully connected layer that applies a linear operation on the input data. The output layer uses the soft-max activation function to output the probability distribution of the classes. The model is then summarized using the model.summary() function, which outputs the layers, shapes, and parameters of the model.

3.2.3. Findings and comparisons

This study investigated the effectiveness of a novel CNN architecture for HSI classification, focusing on improving accuracy and computational efficiency. While earlier studies have successfully applied CNNs to HSI classification, they often focus on spatial feature extraction or dimensionality reduction without addressing the balance between high classification accuracy and resource efficiency. This gap becomes critical when applying these methods to large-scale datasets or edge computing environments, where computational resources are limited.

Once the data was input into the CNN architecture (Figure 2), the model followed a standard training procedure with 100 epochs, a batch size of 256, categorical cross entropy loss, and an Adam optimizer. Table 3 outlines the specifications of each CNN layer, providing a detailed breakdown of the architecture. The Pavia University dataset was used for evaluating the classification accuracy, and the results were compared against the traditional KNN classifier. Tables 4 and 5 summarize the classification accuracy for each class, showing that the proposed method significantly outperformed KNN in 5 out of 9 classes, achieving comparable results in 2 classes, while no notable low accuracy was observed in the remaining classes.

Our key findings indicate that the proposed CNN method consistently yields higher classification accuracy for vegetation classes, such as wheat and corn, compared to KNN. This higher accuracy correlates with the ability of CNNs to extract complex spectral features, which is critical for distinguishing subtle variations in crop types. The classification maps (Figures 5 and 6) demonstrate a clear advantage of the proposed method, particularly in identifying and classifying vegetation areas with high precision, as confirmed by the ground truth data.

Table 4. Accuracy for proposed method on Pavia data sets

Method	Accuracy
Our proposed method CNN	0.95
KNN	0.87

Table 5. Accuracy per class for the data of the University of Pavia in comparison of our method with the KNN method

Label	Class	KNN	Porposed method
1	Asphalt	0.92	0.96
2	Meadows	0.94	0.97
3	Gravel	0.75	0.83
4	Trees	0.91	0.96
5	Painted metal sheets	0.95	1.0
6	Bare soil	0.74	0.93
7	Bitumen	0.83	0.91
8	Self-blocking bricks	0.81	0.88
9	Shadows	0.98	1.0

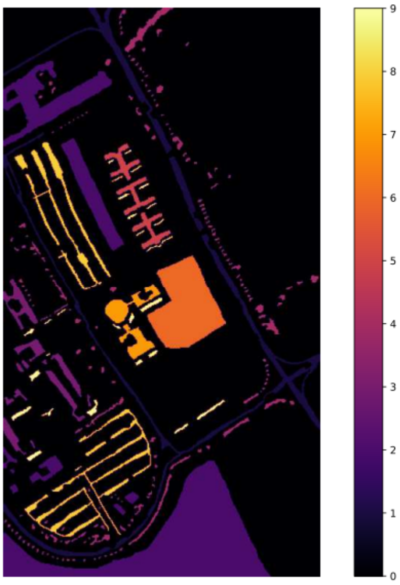


Figure 5. Pavia University classification map

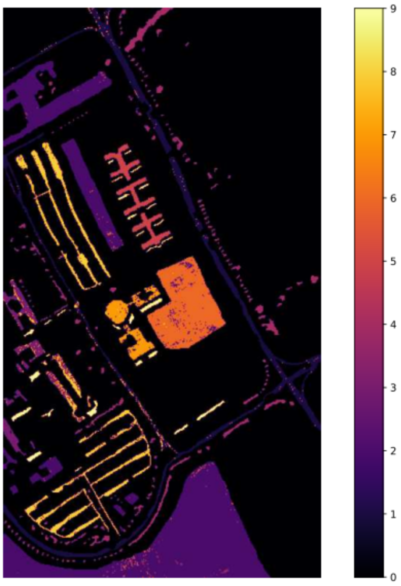


Figure 6. Clasification for our proposed method map

When comparing these results to other studies, our method shows that improved spectral feature extraction does not negatively impact computational efficiency, a key limitation in previous works. For example, unlike studies focusing on spatial features alone, our method exploits spectral domain data, enabling superior performance in classification without the heavy computational burden typically associated with CNNs.

However, this study was limited to the Pavia University dataset, which primarily contains vegetation data. The potential impact of this limitation is that our findings may not generalize to datasets containing more diverse or urban environments. Further research is needed to validate the method's robustness across different types of hyperspectral data and environments. Our results suggest that the proposed CNN method is more resilient to spectral noise and outperforms traditional classifiers in vegetation-related classification tasks. Future studies could explore the application of this architecture to other hyperspectral datasets, focusing on optimizing CNN performance under limited computational resources while maintaining high accuracy.

In conclusion, the findings from this study provide strong evidence that the proposed CNN architecture for HSI classification offers a balanced approach, enhancing classification accuracy in complex spectral datasets without increasing computational costs. These improvements could significantly benefit remote sensing applications, particularly in agriculture and environmental monitoring, where efficient and accurate classification is crucial.

4. CONCLUSION

In this study, a new technique using CNNs for classifying HSIs is presented. First, the data is normalized to retrieve both spatial and spectral features. A resulting HSI image is then combined with the original input HSI and fed into a proposed CNN, which comprises three sets of pooling and convolution layers. To improve the method's accuracy, we have incorporated BN and dropout mechanisms. The classification approach is evaluated on three standard datasets and has been shown to outperform existing state-of-the-art approaches. Future work will concentrate on reducing an algorithm's running time and applying a proposed method to a broader range of HSI datasets using 2D and 3D CNNs.

FUNDING INFORMATION

Authors state no funding involved.

AUTHOR CONTRIBUTIONS STATEMENT

Name of Author	C	M	So	Va	Fo	I	R	D	O	E	Vi	Su	P	Fu
Assia Nouna	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓		
Soumaya Nouna	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓		
Mohamed Mansouri				✓	✓	✓	✓	✓		✓	✓	✓		
Achchab Boujamaa				✓	✓	✓	✓	✓		✓	✓	✓		

C	: **C**onceptualization	I	: **I**nvestigation	Vi	: **Vi**sualization
M	: **M**ethodology	R	: **R**esources	Su	: **Su**pervision
So	: **So**ftware	D	: **D**ata Curation	P	: **P**roject Administration
Va	: **Va**lidation	O	: Writing - **O**riginal Draft	Fu	: **Fu**nding Acquisition
Fo	: **Fo**rmal Analysis	E	: Writing - Review & **E**diting		

CONFLICT OF INTEREST STATEMENT

Authors state no conflict of interest.

DATA AVAILABILITY

he data that support the findings of this study are openly available in the Pavia University repository at [https://www.ehu.eus/ccwintco/index.php/Hyperspectral Remote Sensing Scenes].

REFERENCES

[1] D. Landgrebe, "Hyperspectral image data analysis," *IEEE Signal Processing Magazine*, vol. 19, no. 1, pp. 17–28, 2002, doi: 10.1109/79.974718.

[2] G. M. Foody and A. Mathur, "A relative evaluation of multiclass image classification by support vector machines," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 42, no. 6, pp. 1335–1343, Jun. 2004, doi: 10.1109/TGRS.2004.827257.

[3] Y. Tarabalka, J. A. Benediktsson, and J. Chanussot, "Spectral-spatial classification of hyperspectral imagery based on partitional clustering techniques," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 47, no. 8, pp. 2973–2987, Aug. 2009, doi: 10.1109/TGRS.2009.2016214.

[4] J. A. Gualtieri and S. Chettri, "Support vector machines for classification of hyperspectral data," in *International Geoscience and Remote Sensing Symposium (IGARSS)*, 2000, vol. 2, pp. 813–815, doi: 10.1109/igarss.2000.861712.

[5] G. Mountrakis, J. Im, and C. Ogole, "Support vector machines in remote sensing: A review," *ISPRS Journal of Photogrammetry and Remote Sensing*, vol. 66, no. 3, pp. 247–259, May 2011, doi: 10.1016/j.isprsjprs.2010.11.001.

[6] J. Li, J. M. Bioucas-Dias, and A. Plaza, "Spectral–spatial classification of hyperspectral data using loopy belief propagation and active learning," *IEEE Transactions on Geoscience and remote sensing*, vol. 51, no. 2, pp. 844–856, 2012.

[7] P. M. Atkinson and A. R. L. Tatnall, "Introduction neural networks in remote sensing," *International Journal of Remote Sensing*, vol. 18, no. 4, pp. 699–709, Mar. 1997, doi: 10.1080/014311697218700.

[8] F. Ratle, G. Camps-Valls, and J. Weston, "Semisupervised neural networks for efficient hyperspectral image classification," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 48, no. 5, pp. 2271–2282, May 2010, doi: 10.1109/TGRS.2009.2037898.

[9] G. E. Hinton and R. R. Salakhutdinov, "Reducing the dimensionality of data with neural networks," *Science*, vol. 313, no. 5786, pp. 504–507, Jul. 2006, doi: 10.1126/science.1127647.

[10] K. Fukushima, "Neocognitron: A hierarchical neural network capable of visual pattern recognition," *Neural Networks*, vol. 1, no. 2, pp. 119–130, Jan. 1988, doi: 10.1016/0893-6080(88)90014-7.

[11] Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner, "Gradient-based learning applied to document recognition," *Proceedings of the IEEE*, vol. 86, no. 11, pp. 2278–2323, 1998, doi: 10.1109/5.726791.

[12] P. Y. Simard, D. Steinkraus, and J. C. Platt, "Best practices for convolutional neural networks applied to visual document analysis," in *Proceedings of the International Conference on Document Analysis and Recognition, ICDAR*, 2003, vol. 2003-January, pp. 958–963, doi: 10.1109/ICDAR.2003.1227801.

[13] P. Sermanet and Y. Lecun, "Traffic sign recognition with multi-scale convolutional networks," in *Proceedings of the International Joint Conference on Neural Networks*, Jul. 2011, pp. 2809–2813, doi: 10.1109/IJCNN.2011.6033589.

[14] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "ImageNet classification with deep convolutional neural networks," *Advances in Neural Information Processing Systems*, vol. 2, pp. 1097–1105, 2012.

[15] D. Ciregan, U. Meier, and J. Schmidhuber, "Multi-column deep neural networks for image classification," in *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, Jun. 2012, pp. 3642–3649, doi: 10.1109/CVPR.2012.6248110.

[16] O. Abdel-Hamid, A. R. Mohamed, H. Jiang, and G. Penn, "Applying convolutional neural networks concepts to hybrid NN-HMM model for speech recognition," in *ICASSP, IEEE International Conference on Acoustics, Speech and Signal Processing - Proceedings*, Mar. 2012, pp. 4277–4280, doi: 10.1109/ICASSP.2012.6288864.

[17] Y. Chen, Z. Lin, X. Zhao, G. Wang, and Y. Gu, "Deep learning-based classification of hyperspectral data," *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, vol. 7, no. 6, pp. 2094–2107, Jun. 2014, doi: 10.1109/JSTARS.2014.2329330.

[18] L. He, J. Li, C. Liu, and S. Li, "Recent advances on spectral–spatial hyperspectral image classification: An overview and new guidelines," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 56, no. 3, pp. 1579–1597, 2017.

[19] D. Chutia, D. K. Bhattacharyya, K. K. Sarma, R. Kalita, and S. Sudhakar, "Hyperspectral remote sensing classifications: a perspective survey," *Transactions in GIS*, vol. 20, no. 4, pp. 463–490, 2016, doi: 10.1111/tgis.12164.

[20] S. Li, W. Song, L. Fang, Y. Chen, P. Ghamisi, and J. A. Benediktsson, "Deep learning for hyperspectral image classification: an overview," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 57, no. 9, pp. 6690–6709, 2019.

[21] A. Signoroni, M. Savardi, A. Baronio, and S. Benini, "Deep learning meets hyperspectral image analysis: a multidisciplinary review," *Journal of Imaging*, vol. 5, no. 5, p. 52, 2019, doi: 10.3390/jimaging5050052.

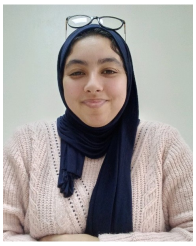
[22] D. Datta, P. K. Mallick, A. K. Bhoi, M. F. Ijaz, J. Shafi, and J. Choi, "Hyperspectral image classification: Potentials, challenges, and future directions," *Computational intelligence and neuroscience*, vol. 2022, no. 1, p. 3854635, 2022.

[23] M. Zhang, W. Li, and Q. Du, "Diverse region-based CNN for hyperspectral image classification," *IEEE Transactions on Image Processing*, vol. 27, no. 6, pp. 2623–2634, 2018, doi: 10.1109/TIP.2018.2809606.

[24] M. E. Paoletti, J. M. Haut, J. Plaza, and A. Plaza, "A new deep convolutional neural network for fast hyperspectral image classification," *ISPRS Journal of Photogrammetry and Remote Sensing*, vol. 145, pp. 120–147, 2018, doi: 10.1016/j.isprsjprs.2017.11.021.

[25] B. Liu, X. Yu, P. Zhang, X. Tan, A. Yu, and Z. Xue, "A semi-supervised convolutional neural network for hyperspectral image classification," *Remote Sensing Letters*, vol. 8, no. 9, pp. 839–848, 2017.

BIOGRAPHIES OF AUTHORS



Assia Nouna is a researcher at the Systems Analysis and Modelling and Decision Support Research Laboratory at Hassan First University in Settat. A doctoral researcher in mathematics and computer science. She is currently working on deep learning and satellite imagery for agricultural applications. Her research aims to enhance agricultural practices through precise soil analysis, improving crop management and yield predictions. Additionally, she has contributed to various projects and publications in the field, demonstrating her expertise in applying advanced computational techniques to solve real-world problems. She can be contacted at email: a.nouna@uhp.ac.ma.

Soumaya Nouna is a researcher at the Systems Analysis and Modelling and Decision Support Research Laboratory at Hassan First University in Settat. She is an expert in mathematics, machine learning and deep learning. A doctoral researcher in mathematics and computer science, she brings a wealth of experience to her field. Her skills include the analysis of differential equations, and machine learning algorithms. S. Nouna is also the author of numerous research articles and is constantly seeking to advance in her areas of expertise. She can be contacted at email: s.nouna@uhp.ac.ma.

Mohamed Mansouri received the Ph.D. degree in Mechanical Engineering and Engineering Sciences from the faculty of science and technology, Hassan First University, Settat, Morocco, and from L'INSA, Rouen, France, in 2013. He is currently a Professor and researcher at the National School of Applied Sciences in Berrechid, Department of Electrical Engineering and Renewable Energies. His research interests include mechano-reliability study, industrial engineering, optimization of shape and reliability optimization of coupled fluid-structure systems, and energy storage systems. He can be contacted at email: m.mansouri@uhp.ac.ma.

Achchab Boujamaa is a professor and director at ENSA Berrechid, Hassan 1st University, specializing in applied mathematics and computer science. He completed his Ph.D. at Université Claude Bernard Lyon 1 in 1995. His research focuses on numerical analysis, mathematical modeling, and computational finance. Notable works include simulations of the Black-Scholes equation and studies on stochastic processes. Achchab is proficient in various mathematical and simulation software, with strong analytical skills and experience in collaborative research projects. He can be contacted at email: achchab@yahoo.fr.