EARTH OBSERVATION AND GEOMATICS ENGINEERING
website: https://eoge.ut.ac.ir

A Hyperspectral Classification Framework Based on Segmentation and Hybrid Convolutional Neural Network Algorithm

Reza Akbari Dotappeh Sofla[1], Reza Shah-Hosseini [2 *]

[1,2] School of Surveying and Geospatial Engineering, College of Engineering, University of Tehran, Tehran, Iran

Article history:
Received: 2022-04-15, Received in revised form: 2022-09-09, Accepted: 2022-10-07

ABSTRACT

In recent years, deep learning methods based on the convolutional neural network (CNNs) have demonstrated good performance for hyperspectral image classification (HSI). Although, in order to obtain good results, we need a large number of training data in the CNNs to avoid the overfitting problem. This paper aims to establish a segmentation-based method to extend the training data for deep learning-based hyperspectral image classification. First, two unsupervised segmentation methods (K-Means and Multi-resolution) are used for the segmentation of the hyperspectral images. Second, we obtained pseudo-training data which depends on the overlay between segmented hyperspectral images and original training data sets. So, we extend the number of training samples for CNN to avoid the overfitting problem and achieve good results. Finally, a Hybrid-CNN model that is a combination of 2D-Convolution and 3D-Convolution is applied to classify hyperspectral datasets with the training samples consisting of the original and pseudo training sets. The proposed method was tested on two Kennedy Space Center (KSC) and Botswana hyperspectral images and the results are compared with the two methods. The overall accuracy with the proposed method retrieves 100% and 96.11% for KSC and Botswana datasets, respectively. Also, we tested the proposed Hybrid-CNN network with Pavia University data, and the classification results show that the proposed Hybrid-CNN has good performance in the face of complex data. The overall accuracy retrieves 99.66% for the Pavia dataset. Keywords: Hyperspectral Image Classification (HSI), Convolutional Neural Network (CNN), Multiresolution segmentation, K-Means Clustering.

KEYWORDS

Hyperspectral,
Classification,
Convolutional Neural Network

1. Introduction

Hyperspectral images can produce many bands for each image pixel. With the rich information captured in hyperspectral images, this image has been widely used in various fields, such as agriculture, monitoring, and industrial inspection. Hyperspectral image classification (HSI) is one of the most promising techniques for understanding remote-sensing images (Liang et al., 2015).

Over the past several years, many methods have been proposed to classify HSIs. Most methods have focused on exploring the role of the spectral signatures of HSIs for classification. Thus, numerous pixel-wise classification methods, such as neural networks (Zhong & Zhang, 2012), support vector machines (SVM) (Melgani & Bruzzone,

* Corresponding Author:
E-mail addresses: Rshahosseini@ut.ac.ir (R.Shahhosseini)

2004), and multinomial logistic regression (Li et al., 2010) have been proposed. However, there exist some challenges. For example, the large spatial variability of spectral signatures and the limited available training samples versus the high dimensionality of hyperspectral data are the main problems in the HSI classification (Ghamisi et al., 2017).

The classification results obtained by these pixel-wise classifiers are unsatisfactory since the spatial features are not considered. To improve the performance of classification, many researchers have focused on spectral-spatial classification which can incorporate spatial features into pixel-wise classifiers. For example, in (Benediktsson et al., 2005), extended morphological profiles (EMPs) were used to exploit spatial information via multiple morphological operations. In (Kang et al., 2017), principal component analysis (PCA) based edge-preserving features (PCA-EPFs) is proposed to capture the multi-scale structural information of hyperspectral images. In (Kang et al., 2015), extended random walkers serving as a powerful optimization tool are used to refine the pixel-wise probability maps obtained by the SVM.

In this research, we attempted to mitigate the problems associated with the classification of hyperspectral images including the lack of training data and extracting spectral and spatial information. To solve the problem of the lack of training data, we took advantage of K-Means and Multi-resolution methods for segmentation to increase training data. Also, we introduced a hybrid-CNN network to classify hyperspectral images by extracting both spectral and spatial data. This proposed network was a combination of 3D and 2D Convolution.

2. The Related Work

Deep learning methods which exploit the non-linear transformation of data via several layers have attracted a lot of attention in remote sensing areas in recent years. In the context of feature extraction, deep learning automatically extracts significant and discriminative features from a hierarchy of hidden layers. In the case of HSI, deep learning-based methods, e.g. autoencoders (AEs) (Feng et al., 2018), stacked autoencoders (SAEs) (Zabalza et al., 2016), deep belief networks (DBNs) (Chen et al., 2015), recurrent neural networks (RNNs) (Mou et al., 2017), and convolutional neural networks (CNNs) (Hu et al., 2015; Yu et al., 2017), have been demonstrated to be very efficient in extracting robust and invariant features. Notably, the classification accuracy of deeper networks tends to reduce with the limited training samples available for HSIs. This problem is more serious when fully connected models such as AEs and DBNs (Ghamisi et al., 2018) are used. In CNNs, the number of parameters is reduced by the properties of shared weights and local connections, which makes it feasible to obtain high classification accuracy for hyperspectral data even when limited training samples are available. CNNs usually require large amounts of training samples in order to avoid overfitting. Data augmentation is a technique that synthetically generates new samples by applying a set of domain-specific transformations over the original input dataset to improve the generalization capabilities of a classification model. Several data augmentation techniques applicable to HSIs have been proposed. Recently, (Zhang et al., 2018) described hyperspectral data augmentation techniques where pixels are grouped in blocks and different block pairs are used as the input to a CNN. In (Nalepa et al., 2012), samples in the original dataset were shifted along with its first principal component or based on the average value in each band. Augmentation based on randomly erasing parts of the input patches has also been proven effective for HSI classification (Haut et al 2019). Finally, generative adversarial networks have been proposed recently as a data augmentation technique in order to generate new samples mimicking the distribution of the original data (Arefi et al., 2019). In (Kang et al., 2019), a novel semi-supervised deep learning method is proposed for hyperspectral image classification. They extend the original training set with the principal component analysis-based edge-preserving features (PCA-EPFs) and extended morphological attribute profiles (EMAPs) methods and with the proposed decision fusion strategy, the accuracy of the existing deep learning-based hyperspectral image classification method can be improved dramatically. (Wu and Prasad, 2018) proposed a semi-supervised method for hyperspectral image classification. First, they obtained pseudo-training samples by clustering methods and then train the network with pseudo and original training data. Given this fact, we emphasize that CNNs are a promising and powerful method for the classification of HSIs. However, these methods often cause an over-fitting phenomenon when only a few numbers of training samples are available. Therefore, the lack of a large number of training samples is the main challenge of deep learning-based classification methods.

This paper is organized as follows. Part 2, describes the unsupervised segmentation methods and the Hybrid-CNN model and then the details of the proposed algorithm for HSI are presented. Part 3, reports the experimental results of the proposed method and in part 4, the conclusions are presented.

3. The Proposed Method

3.1. Segmentation Methods

3.1.1 Multiresolution Segmentation

The multiresolution segmentation approach was used to partition the image layer into homogeneous objects in this paper (Baatz & Schäpe, 2000). This approach is one of the most widely used image segmentation methods in the remote sensing community (G. Mallinis et al., 2008). Multiresolution segmentation is a bottom-up region-growing algorithm, which starts by considering each pixel as a separate object and subsequently. In this method, pixels or existing objects are merged into bigger ones based on three parameters: scale, color (spectral properties), and shape (smoothness and compactness). The scale is a crucial parameter that determines when the optimization process stops. The higher scale parameter leads to larger homogeneous objects. The segmentation process stops when the smallest growth exceeds a user-defined threshold (e.g., scale parameter), which determines the maximum increase of heterogeneity when objects are merged. The heterogeneity criterion f is a combination of the spectral heterogeneity (Δh_{color}) and the shape heterogeneity (Δh_{shape}), which can be formulated as follows:

$$f = (w_{color} \times \Delta h_{color}) + (w_{shape} \times \Delta h_{shape}) \quad (1)$$

$$w_{color} + w_{shape} = 1 ;\; w_{color}, w_{shape} \in [0,1] \quad (2)$$

Where w_{color} and w_{shape} are the weights of spectral heterogeneity and shape heterogeneity, which allow us to adapt the definition of heterogeneity for a given application.

3.1.1. K-Means Clustering

Clustering is a method to divide a set of data into a specific number of groups. One of the popular methods is K-Means clustering. In K-Means clustering, it partitions a collection of data into a K number group of data (Shehroz et al., 2004). K-Means clustering is a type of unsupervised method, which is used when you have unlabeled data (i.e., data without defined categories or groups). The goal of this algorithm is to find similar groups in the data, with the number of groups represented by the variable K. The algorithm works iteratively to assign each data to at least one of the K groups supported by the features that are provided. Data points are clustered and supported by feature similarity. The results of the K-Means clustering algorithm are:

(1) The centroids of the K clusters, which may be accustomed label new data
(2) Labels for the training data (each data is assigned to one cluster)

The K-Means clustering algorithm uses iterative refinement to produce a segmentation map. The algorithm inputs are the number of clusters K and the data set. The data set may be a collection of features for every data point. The algorithms start with initial estimates for the K centroids, which can either be randomly generated or randomly selected from the data set.

Although K-Means has the good advantage of being easy to implement, it has some drawbacks. The quality of the final clustering results depends on the arbitrary selection of the initial centroid. The initial centroid is randomly chosen; it will get different results for different initial centers. And also computational complexity is another term that we'd like to think about while designing the K-Means clustering. It relies on the number of data elements, the number of clusters, and the number of iterations (Dhanachandra et al., 2015).

3.2. Convolutional Neural Network

Convolutional Neural Network (CNN) is a part of deep neural networks, which can be used in conjunction with a deep learning platform. A CNN is a network of processing layers accustomed reduce an image to its key features in order that it is often more easily classified. The advantage of CNNs over other uses of classification algorithms is the ability to learn key characteristics on their own, reducing the need for hyperparameters, and hand-engineered filters. These algorithms are increasingly getting used for tasks like face recognition, image classification, video analysis, and automatic caption generation. In this paper, we address the hyperspectral image classification problem using a CNN model. A CNN operates in three layers:

(1) *Convolution Layer*: This layer is where images are transformed into processable data by kernels, a filter layer consisting of knowledgeable parameters. Each kernel filters for a special feature and multiple kernels are utilized in each analysis. In a convolution, small areas of an image are scanned and the probability that they belong to a filter class is assigned and translated to an activation map, a representation of the image layers. In a 3D CNN, the kernels move through three dimensions of knowledge (height, length, and depth) and produce 3D activation maps.
(2) *Pooling Layer:* Pooling, or downsampling, is done on the activation maps created during convolution. During pooling, a filter moves across an activation map evaluating a small section at a time, similar to the convolution process. This filter takes either the type of the scanned area, a weighted average supported by the central pixel, or the max value and abstracts that value to a replacement map. The max-pooling method, where the highest value from the scanned area is taken, is the most commonly used. This abstraction is done to decrease the processing time and it evaluates each map by eliminating unimportant features and allows for spatial variance, the ability to detect features regardless of rotation or tilting.
(3) *Fully Connected (FC) Layer*: After multiple iterations, sometimes thousands of convolutions and pooling of the output layers are flattened, the probabilities identified are analyzed, and the output is assigned a value. This analysis is completed by

the Fully Connected layer, during which each flattened output layer is processed by interconnected nodes, almost like a totally connected neural network (FCNN). The difference is that in a CNN the convolutional and pooling layers are independent of the FC layer. By isolating features of an image before feeding the output to the FC layer, CNN is in a position to limit the necessity for higher processing power to the final steps.

In this paper, we address the hyperspectral image classification problem using a CNN model. As you know there is three-way to HSI by CNNs:

(1) *1-D CNN*: this model Extract only spectral information and neglect the spatial components. 1-D CNN architecture receives $M \times 1$ input vectors, where *M* is the number of spectral bands. (Ghamisi et al. 2017)

(2) *2-D CNN*: these models consider the neighboring pixels of a certain pixel in the original remote-sensing image in order to extract only the spatial information. The input data of 2-D CNN architectures is a patch of $P \times P$ neighboring and it cannot extract good discriminating features from spectral dimension.

(3) *3-D CNN:* this model extracts spatial-spectral information from each patch and improves the classification accuracy. But 3-D CNN has a few problems such as computationally complex.

As you see, all CNN models have a few problems. 1-D CNN neglects the spatial information, 2-D CNN has a problem at extracting spectral information, and 3-D CNN has computationally complex and may cause to misclassify the pixels having similar textures over many spectral bands. This is the cause to propose a CNN model that hasn't the above shortcomings. This model uses 2-D CNN and 3-D CNN to utilize both the spectral as well as spatial features to achieve high classification accuracy. The details of Hybrid-CNN are presented in the proposed method section.

3.3. Proposed Method

As mentioned in previous sections, there are some limitations to classifying hyperspectral images using CNNs, including the lack of training data and selecting an appropriate CNN model for extracting spectral and spatial information. In this section, we introduce a method that solves both the mentioned limitations and we can classify hyperspectral images with high accuracy. In the proposed method, shown in Figure 1, we first use Multi-resolution and K-Means Clustering algorithms to extend training data. In fact, these two algorithms are used for input image segmentation purposes. Then, the segmentation image is placed on the original ground truth image of the same data and it is observed that the original labeled samples fall into the segments. Based on this observation, we consider the whole segment as corresponding labeled samples. For example, if there are grass samples in a segment, the whole segment is considered as the labeled samples of grass and extended labeled data. It should be noted that the whole labeled samples should be fell into a single segment to select that segment as labeled data. In this research, the parameter number of clusters in the K-Means algorithm and the parameters scale, shape, and compactness are very effective in the accuracy of the classification results.

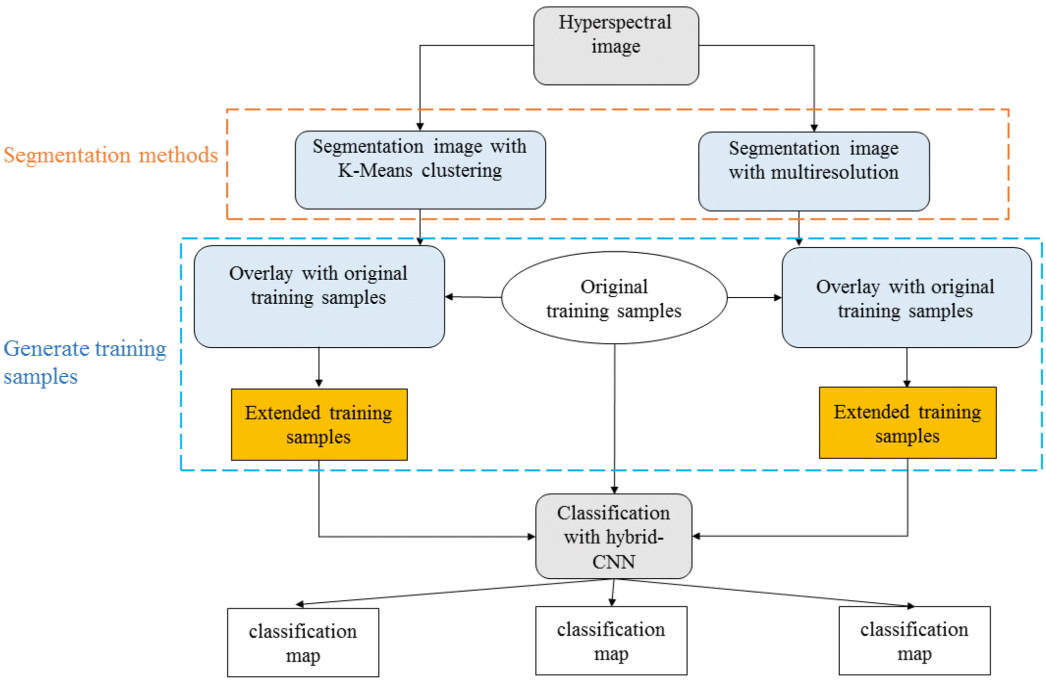


Figure 1. Schematic of the proposed method for HIC with extended labeled samples

We have three kinds of labeled data in this study, the extended labeled data obtained by the Multi-resolution algorithm (T_M), the extended labeled data obtained by the K-Means algorithm (T_{KM}), and the original labeled data (T_0). In fact, we employ all of the three kinds of labeled data to classify hyperspectral images using the proposed CNN model and compare their results.

To classify hyperspectral images using CNN, it is required to present a model without the limitations mentioned in the previous sections. As shown in Figure 2, the proposed model is a combination of the 2-D CNN and 3-D CNN models used for classification. The 2-D CNN models are not able to extract spectral information and only use spatial information. On the other hand, although 3-D CNN models can extract both spectral and spatial information, they have two defects; first, those models are complicated with various parameters; second, it is possible that they fail to distinguish the pixels with similar textural features in different bands. Therefore, our proposed method is a combination of both models. In the proposed method, first, three 3-D Convolutions are applied to input data, the lengths and heights of the three kernels of Convolution3D are the same as to classify hyperspectral mages using CNN, it is required to present a model without the limitations mentioned in the previous sections. As shown in Figure 2, the proposed model is a combination of the 2-D CNN and 3-D CNN models used for classification. The 2-D CNN models are not able to extract spectral information and only use spatial information. On the other hand, although 3-D CNN models can extract both spectral and spatial information, they have two defects; first, those models are complicated with various parameters; second, it is possible that they fail to distinguish the pixels with similar textural features in different bands. Therefore, our proposed method is a combination of both models.

In the proposed method, first, three 3-D Convolutions are applied to input data, the lengths and heights of the three kernels of Convolution3D are the same as 3×3 but their depths are 3, 5, and 7 respectively. In Convolution2D, different spatial data can be extracted by changing the size of the kernel. Now, we attempt to change the depths of the kernels instead of changing their lengths and heights to extract different spectral and spatial features from the input patch of the network and improve classification accuracy by concatenating these features.

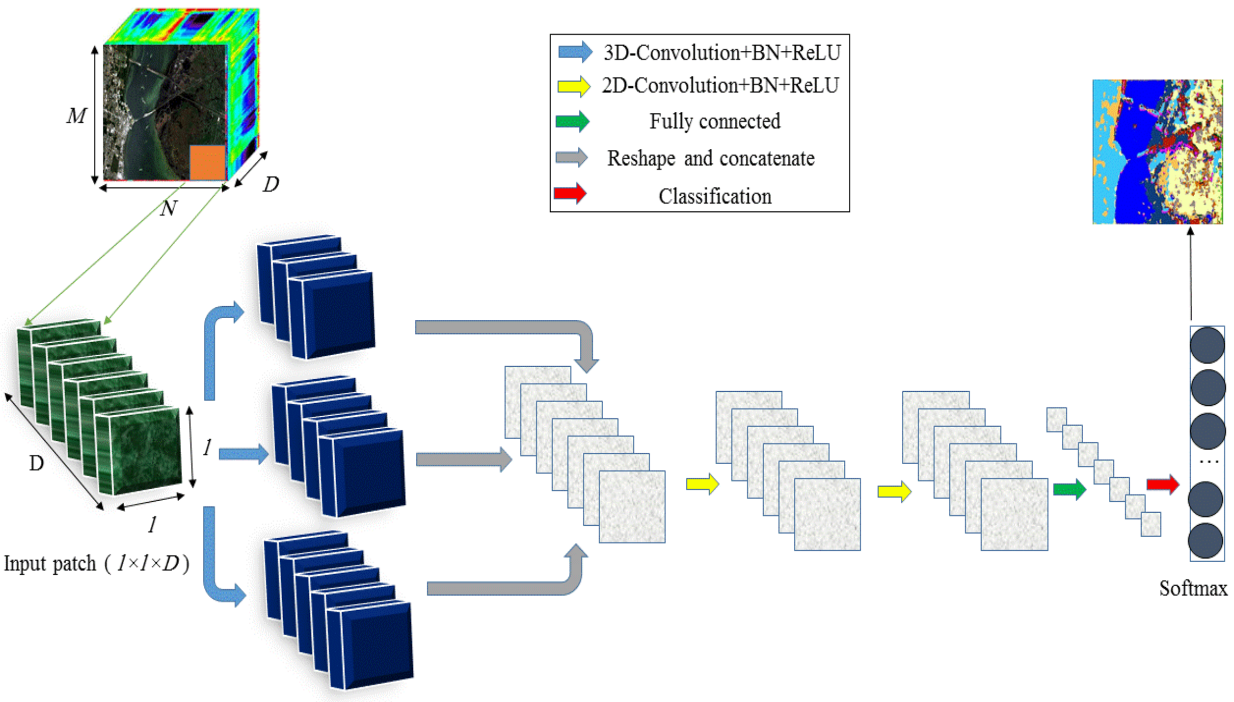


Figure 2. Proposed hybrid-CNN architecture

The size of a hyperspectral image is shown by $M \times N \times D$ where M and N are the length and height of the image respectively and D is the number of the spectral bands of the image. For classification, the hyperspectral image is first converted into patches with $l \times l \times D$ sizes to enter the proposed hybrid-CNN model. The proposed model has three Convolution3D layers, two Convolution2D layers, one Fully-Connected layer, and one Softmax layer at the end of the model. The three Convolution3D layers are used to maintain spectral information and the two Convolution2D layers are then employed to distinguish spatial information in different bands without losing spectral information. The input image enters the three Convolution3D layers simultaneously and the features obtained from these three layers are concatenated with each other to be used as the input of the Convolution2D layer. There is a batch normalization and activation function ReLU after each convolution layer. After the two Convolution2D layers, there are a flattened layer, a fully-connected layer, and a softmax layer to classify pixels. The details of the network are summarized in Table 1. In this network, weights are trained by a back-propagation algorithm using Adam optimizer and categorical-cross-entropy loss function. Also, a dropout = 0.5 layer is placed after the fully-connected layer to prevent the overfitting problem.

Table 1. Configuration of the CNN architecture for the KSC and Botswana datasets. (*l* = *s* and *D* is the number of channels)

	Layer	Filter Size	Output Size	Padding	Stride
(1)	Input	-	$l \times l \times D \times 1$	-	-
(2)	Conv3D+BN+Relu	$(3\times3\times3, 8)$	$l \times l \times D \times 8$	Same	1
	Conv3D+BN+Relu	$(3\times3\times3, 16)$	$l \times l \times D \times 16$	Same	1
	Conv3D+BN+Relu	$(3\times3\times7, 32)$	$l \times l \times D \times 32$	Same	1
(3)	Conv2D+BN+Relu	$(3\times3, 32)$	$l \times l \times 32$	Same	1
	Conv2D+BN+Relu	$(3\times3, 16)$	$l \times l \times 16$	Same	1
(4)	Flatten	-	-	-	-
	Dense 1 (128, Dropout=0.5)	-	-	-	-
	Dense 2 (number of classes, softmax)	-	-	-	-

4. Experiments

4.1. Datasets

In this study, we used the Botswana and Kennedy Space Center (KSC) datasets[2]. Since both datasets have little training data, they can challenge our research.

4.1.1. Botswana Dataset

The first dataset was collected by a Hyperion sensor on EO-1 over the Okavango Delta, Botswana in 2001. The acquired data originally consisted of 242 bands covering the 400-2500 nm portion of the spectrum in 10 nm windows with 30 m pixel resolution. Only 145 bands were used after uncalibrated and noisy bands that cover water absorption were removed. The data used in this paper consist of pixels with observations from 14 identified classified classes representing the land cover types (Table 2). These labeled samples were shown in Table 5. The RGB images and ground truths (GT) of the HS data are shown in Figure 3.

4.1.2. KSC Dataset

It is a NASA AVIRIS data that has been acquired over the Kennedy Space Center, Florida on March 23, 1996. It has 224 bands of 10 nm width in visible and near-infrared spectrum (400-2500 nm) with a spatial resolution of 18 m. for our experiment, low signal-to-noise and water absorption bands are removed. The data used this paper consist of pixels with 13 classes that representing the land cover types (Table 2). The RGB images and ground truths (GT) of the HS data are shown in Figure 4.

[2] http://www.ehu.eus/ccwintco/index.php?title=Hyperspectral_Remote_Sensing_Scenes

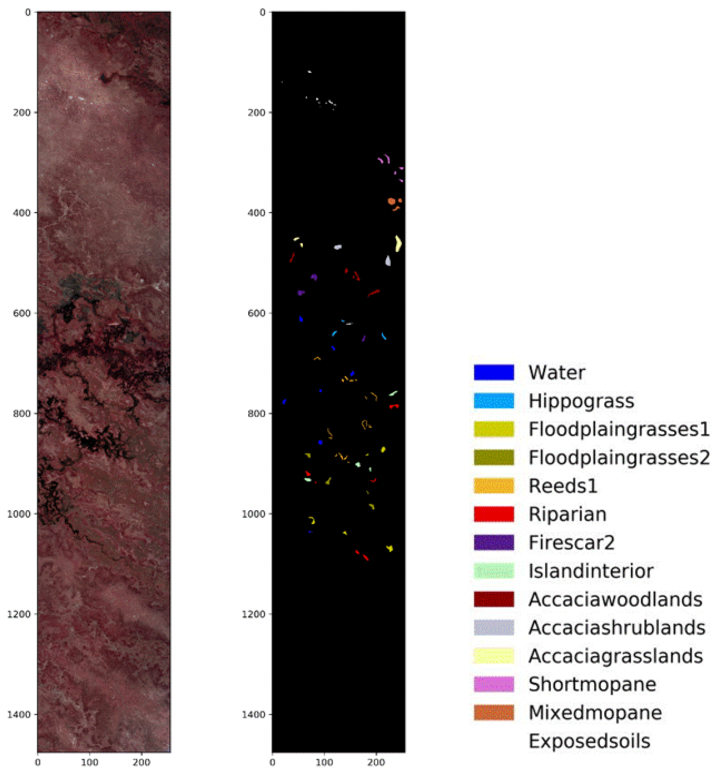


Figure 3. Original RGB and ground truth image of the Botswana scene and name of samples from left to right, respectively

Table 2. Original labeled samples (T_0) for KSC and Botswana datasets

Class	**KSC**		**Botswana**	
	Class Name	**Samples**	**Class Name**	**Samples**
1	Scrub	761	Water	270
2	Willow swamp	243	Hippo grass	101
3	Cabbage palm hammock	256	Floodplain grasses 1	251
4	Cabbage palm/oak hammock	252	Floodplain grasses 2	215
5	Slash pine	161	Reeds	269
6	Oak/broadleaf hammock	229	Riparian	269
7	Hardwood swamp	105	Fires car	259
8	Graminoid marsh	431	Island interior	203
9	Spartina marsh	520	Acacia woodlands	314
10	Cattail marsh	404	Acacia shrub lands	248
11	Salt marsh	419	Acacia grasslands	305
12	Mudflats	503	Short mopane	181
13	Water	927	Mixed mopane	268
14			Exposes soils	95

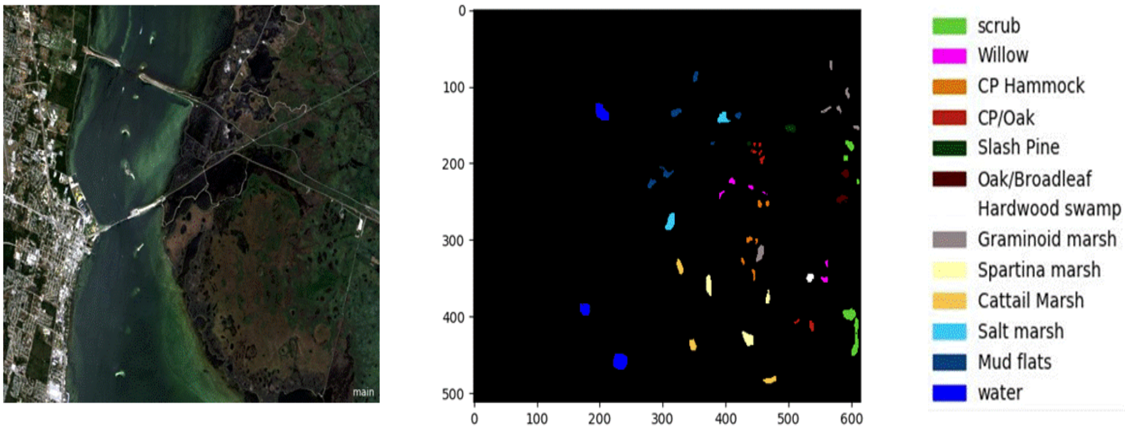


Figure 4. Original RGB and ground truth image of the KSC scene and name of samples from left to right, respectively

4.2. Experimental Setup

In this research, the two methods of 2D-CNN (Makantasis et al., 2015) and 3D-CNN (Hamida et al., 2018) were used to evaluate the proposed method and compare our method with them. Also, a python programming language with Keras library was used to evaluate and train the proposed method and the related codes were run on the Google Colab environment, a public and free service provided by Google. The number of epochs for Botswana and KSC datasets was set to 100 and 200 respectively. In the end, the proposed method was also compared with two other methods in terms of classification accuracy and computational time.

4.3. Experimental Results and Discussion

In this research, we increased the number of labeled samples using K-Means and Multi-resolution methods (Figure 5). For this purpose, we should find the appropriate parameters of the two mentioned algorithms to extract labeled samples with proper accuracy and very low error. The K-means method has the parameter of the number of clusters which was fixed to 20 for both datasets after testing with a different number of clusters. If the number of clusters are too large, the segmentation map will have very small pieces, and a barricade to increase the number of training data. On the other hand, if the number of clusters is considered too small, the pieces of the segmentation map will be very large so that it is possible a single piece includes two classes and erroneous extended labeled data will be obtained.

In the Multi-resolution method, three parameters of scale, shape, and compactness should be defined with proper accuracy to take advantage of that to create new labeled data. Different values of these three parameters were tested for Botswana and KSC datasets to obtain the best-segmented image. For KSC image segmentation, the parameters were set as Scale = 30, Shape = 0.3, and Compactness = 0.5 and for Botswana image segmentation, they were set to 50, 0.2, and 0.5 respectively.

In this way, we increased our labeled data using these two segmentation methods. The number of extended labeled data using K-Means and Multi-resolution methods is shown in Tables 3 and 4 respectively

Images are $(512 + s - 1, 614 + s - 1, n)$ and $(1467 + s - 1, 256 + s - 1, n)$ respectively.

To implement the proposed method, we first considered 60% of labeled samples as training data and 40% of them as test data (Table 5). In this research, the parameter s was tested with four values of 5, 7, 11, and 13 for the Botswana dataset. It was also tested with four values of 5, 7, 9, 11 for the KSC dataset. By choosing a large s, the network can take advantage of more spatial features but it may reduce the effect of the central pixel and makes the final result erroneous. Also, if the value of the parameter s is too small, the possibility of using spatial features is minimized. To test the different values of s, 60% of labeled samples are considered as training data and 40% of them are considered as test data.

.

Table 3. Extended labeled samples with Multi-resolution segmentation (T_M)

Class	KSC		Botswana	
	Class Name	**Samples**	**Class Name**	**Sample**
1	Scrub	1079	Water	404
2	Willow swamp	388	Hippo grass	169
3	Cabbage palm hammock	390	Floodplain grasses 1	421
4	Cabbage palm/oak hammock	350	Floodplain grasses 2	390
5	Slash pine	288	Reeds	401
6	Oak/broadleaf hammock	321	Riparian	388
7	Hardwood swamp	192	Fires car	401
8	Graminoid marsh	724	Island interior	340
9	Spartina marsh	795	Acacia woodlands	520
10	Cattail marsh	684	Acacia shrub lands	396
11	Salt marsh	673	Acacia grasslands	475
12	Mud flats	820	Short mopane	301
13	Water	1564	Mixed mopane	383
14			Exposes soils	134

Table 4. Extended labeled samples with K-Means segmentation (T_{KM})

Class	KSC		Botswana	
	Class Name	**Samples**	**Class Name**	**Samples**
1	Scrub	1033	Water	420
2	Willow swamp	398	Hippo grass	192
3	Cabbage palm hammock	401	Floodplain grasses 1	456
4	Cabbage palm/oak hammock	376	Floodplain grasses 2	395
5	Slash pine	295	Reeds	416
6	Oak/broadleaf hammock	310	Riparian	402
7	Hardwood swamp	195	Fires car	397
8	Graminoid marsh	750	Island interior	349
9	Spartina marsh	801	Acacia woodlands	569
10	Cattail marsh	702	Acacia shrub lands	411
11	Salt marsh	710	Acacia grasslands	502
12	Mud flats	846	Short mopane	310
13	Water	1456	Mixed mopane	414
14			Exposes soils	142

Table 5. Train and test samples at original labeled samples (T_0)

Class	KSC			Botswana		
	Class Name	**Train**	**Test**	**Class Name**	**Train**	**Test**
1	Scrub	457	304	Water	162	108
2	Willow swamp	146	97	Hippo grass	61	40
3	Cabbage palm hammock	154	102	Floodplain grasses 1	151	100
4	Cabbage palm/oak hammock	151	102	Floodplain grasses 2	129	86
5	Slash pine	97	64	Reeds	161	108
6	Oak/broadleaf hammock	137	92	Riparian	161	108
7	Hardwood swamp	63	42	Fires car	155	104
8	Graminoid marsh	259	172	Island interior	122	81
9	Spartina marsh	312	208	Acacia woodlands	188	126
10	Cattail marsh	242	162	Acacia shrub lands	149	99
11	Salt marsh	251	168	Acacia grasslands	183	122
12	Mud flats	302	201	Short mopane	109	72
13	Water	556	371	Mixed mopane	161	107
14				Exposes soils	57	38

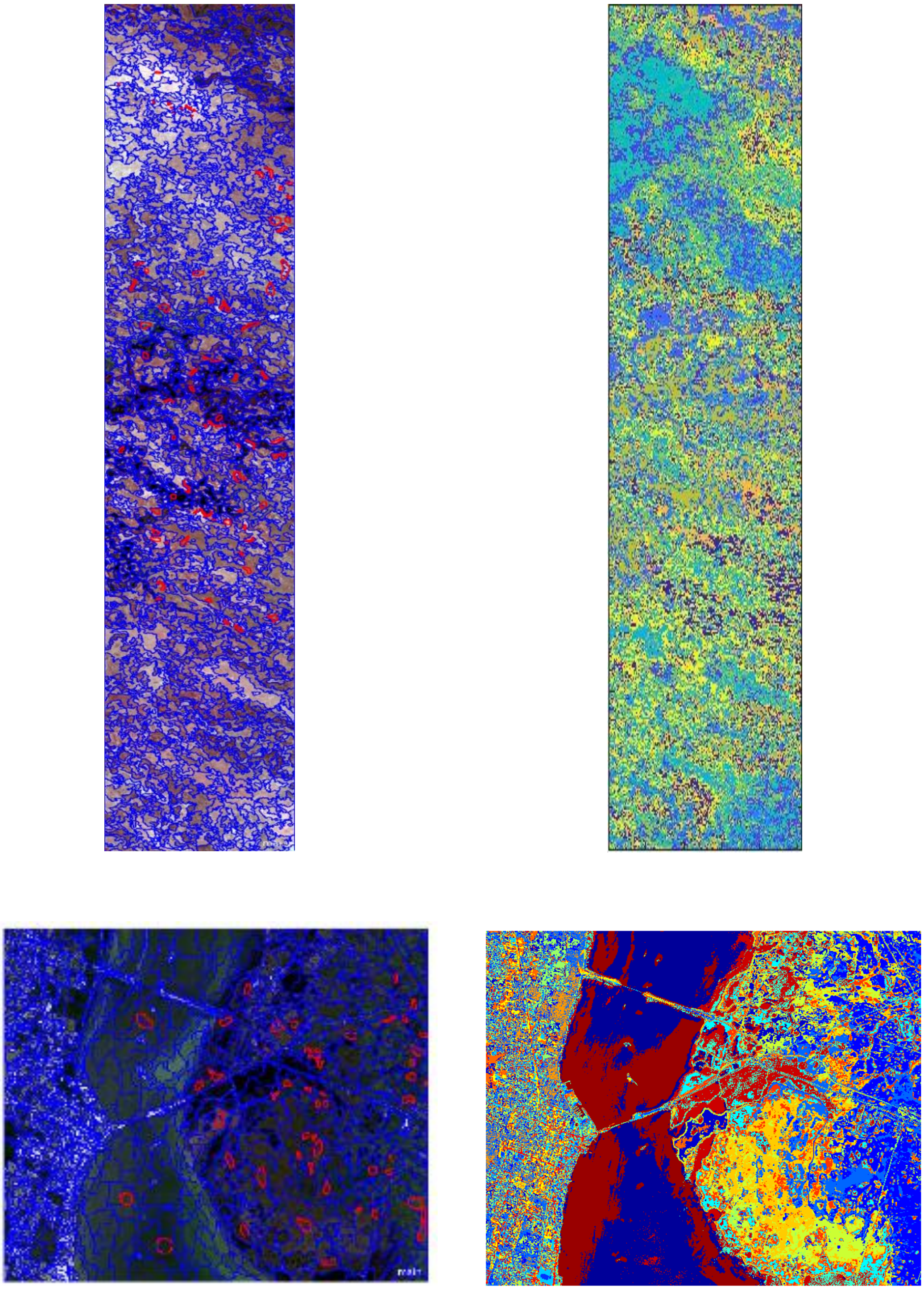
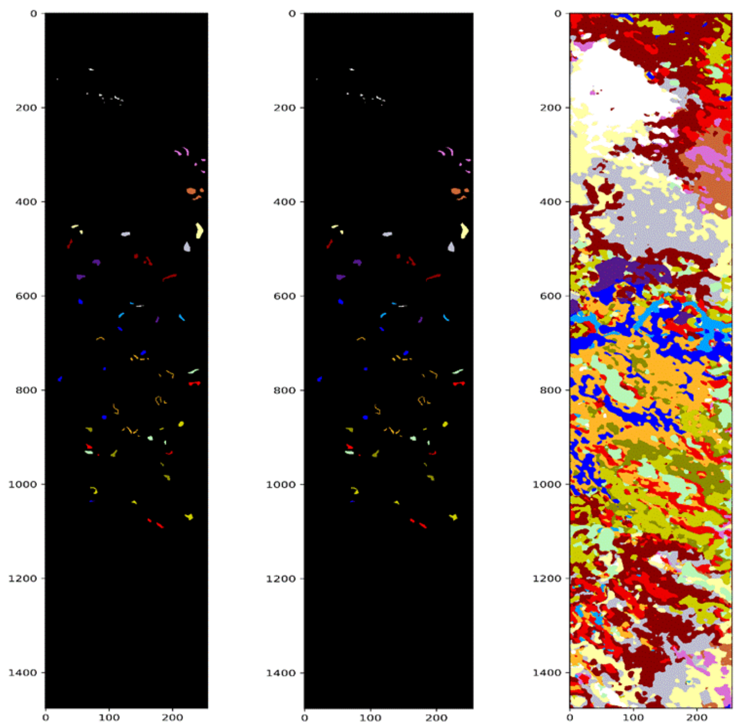


Figure 5. Segmentation results for KSC and Botswana image. The upper row displays an overlay between the original ground truth (red shapes) and Multi-resolution segmentation result of the Botswana image and the K-Means segmentation result from left to right, respectively, and the lower row displays the same results for the KSC image

The proposed hybrid-CNN was first tested with different values of s on the Botswana dataset (three times for each s). As shown in Table 6, it is obvious that the proposed method is appropriate to classify this dataset. For s = 13,11 and 7, it is observed that the proposed method has been able to classify the Botswana dataset with maximum accuracy so that the overall accuracy is 100% for $s = 7, 11$. However, the overall accuracy slightly decreased for s = 5. As shown in Figure 6 (for s = 13, 11) and Figure 7 (for s = 7, 5), we have classified the Botswana image with background and without background for different values of s. A zoomed area of classification images with different s is shown in Figure 8.

Table 6. Classification accuracies obtained by our hybrid-CNN (with patch sizes of s = 5, s = 7, s = 11, and s = 13) for the Botswana hyperspectral dataset

Patch Size	***s* = 5**	***s* = 7**	***s* = 11**	***s* = 13**
Training Data	**60% Samples**	**60% Samples**	**60% Samples**	**60% Samples**
Water	100.00 (0.00)	100.00 (0.00)	100.00 (0.00)	100.00 (0.00)
Hippo grass	100.00 (0.00)	100.00 (0.00)	100.00 (0.00)	100.00 (0.00)
Floodplain grasses 1	99.90 (0.24)	100.00 (0.00)	100.00 (0.00)	100.00 (0.00)
Floodplain grasses 2	100.00 (0.00)	100.00 (0.00)	100.00 (0.00)	100.00 (0.00)
Reeds	99.00 (0.24)	100.00 (0.00)	100.00 (0.00)	100.00 (0.00)
Riparian	96.12 (1.35)	100.00 (0.00)	100.00 (0.00)	99.80 (0.24)
Fires car	100.00 (0.00)	100.00 (0.00)	100.00 (0.00)	100.00 (0.00)
Island interior	100.00 (0.00)	100.00 (0.00)	100.00 (0.00)	100.00 (0.00)
Acacia woodlands	100.00 (0.00)	100.00 (0.00)	100.00 (0.00)	100.00 (0.00)
Acacia shrub lands	100.00 (0.00)	100.00 (0.00)	100.00 (0.00)	100.00 (0.00)
Acacia grasslands	100.00 (0.00)	100.00 (0.00)	100.00 (0.00)	100.00 (0.00)
Short mopane	100.00 (0.00)	100.00 (0.00)	100.00 (0.00)	100.00 (0.00)
Mixed mopane	100.00 (0.00)	100.00 (0.00)	100.00 (0.00)	100.00 (0.00)
Exposes soils	99.27 (0.16)	100.00 (0.00)	100.00 (0.00)	100.00 (0.00)
Overall accuracy	99.14 (0.17)	100.00 (0.00)	100.00 (0.00)	99.99 (0.01)
Average accuracy	99.59 (0.11)	100.00 (0.00)	100.00 (0.00)	99.98 (0.01)



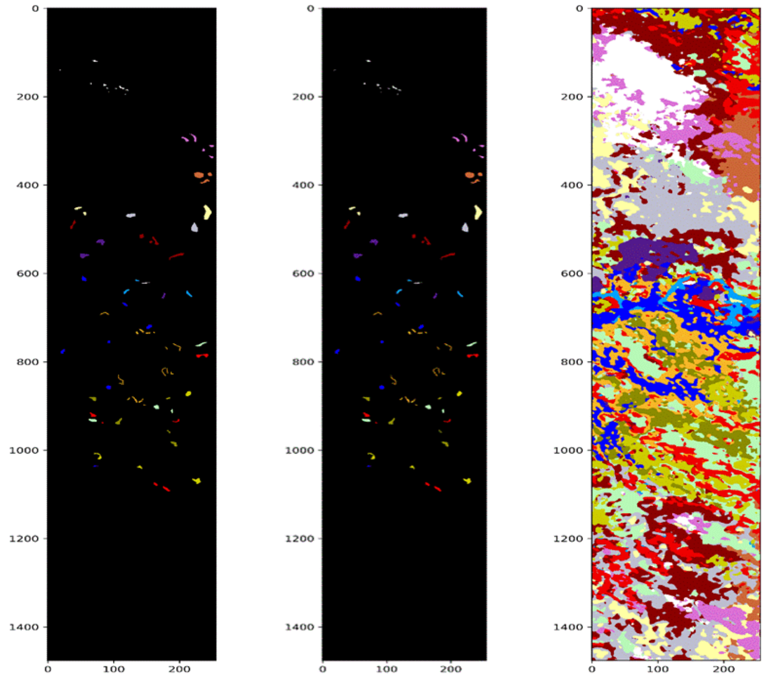
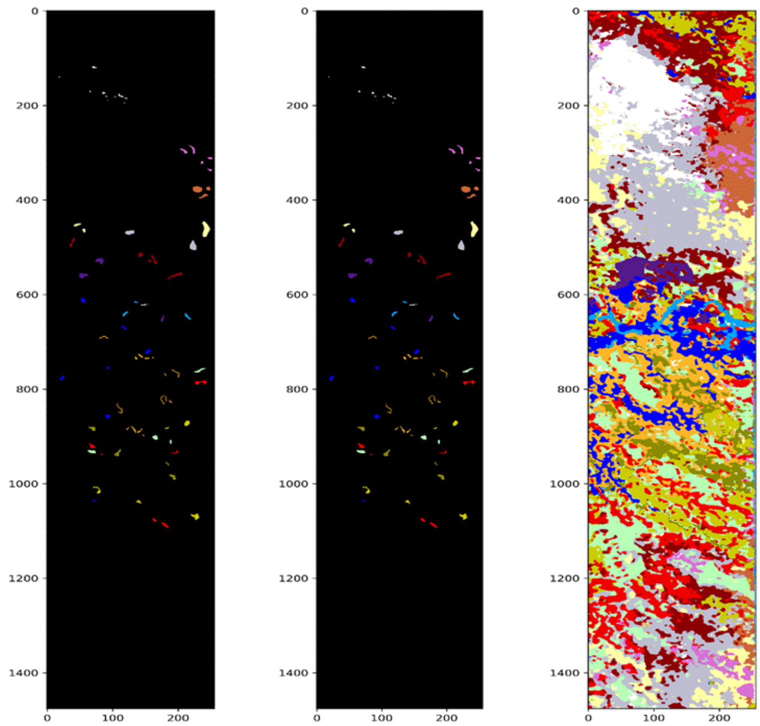


Figure 6. Classification results for the Botswana data set with *s* = 13 (upper row) and *s* = 11 (lower row), that each row displays original ground truth, classification result without background, and classification result with background from left to right, respectively



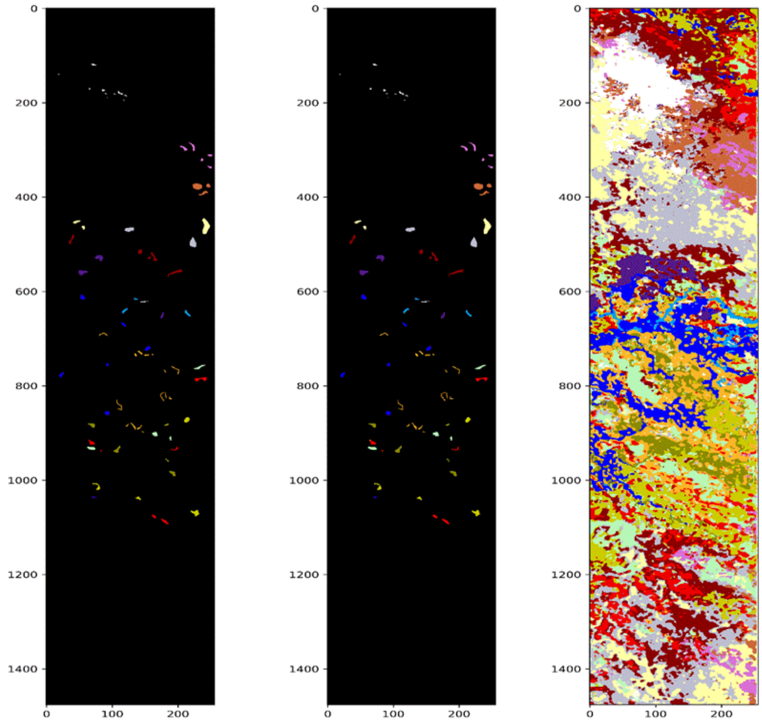


Figure 7. Classification results for the Botswana data set with s = 7 (upper row) and s = 5 (lower row), that each row displays original ground truth, classification result without background, and classification result with background from left to right, respectively

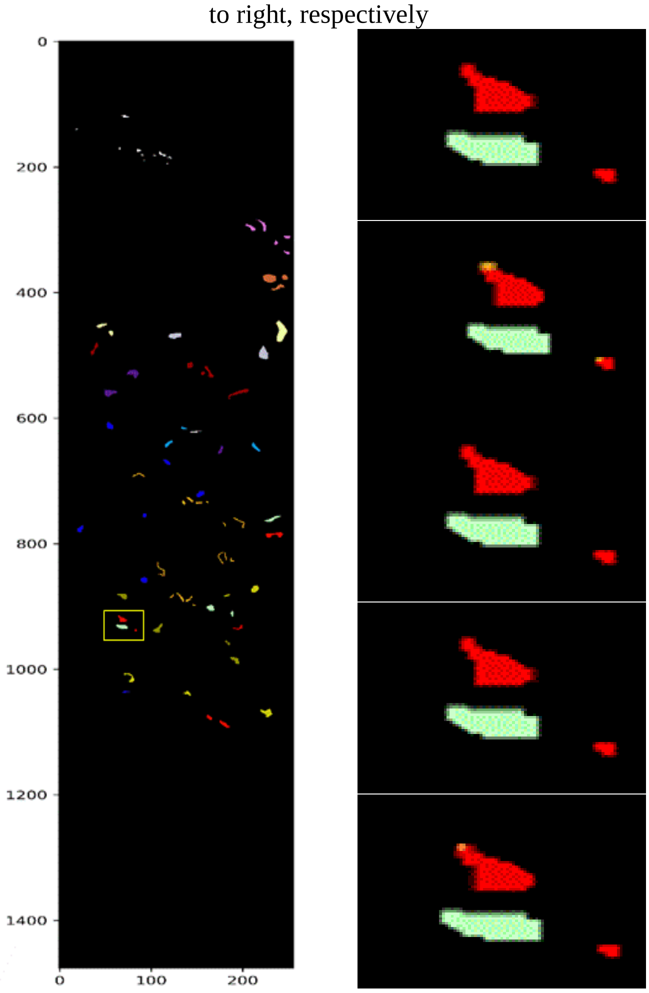


Figure 8. ground truth image of Botswana data set with small yellow square (left side) and zoomed-in yellow square for original ground truth, Classification results with s = 5, s = 7, s = 11 and s = 13 from upper row to lower row, respectively in the right side

The proposed method was also tested on the KSC dataset. Compared to the classification results of the Botswana dataset, the results of the KSC dataset were worse. In this case, the best result (overall accuracy = 96.11) was for s = 7. As shown in Table 7, the result for s = 5 was so bad that it highlights the importance of spatial features in classification. According to the results, the overall accuracy for the water class for all values of *s* is 100%; the reason could be the existence of more training data compared to other classes. In this KSC also, the image was classified with and without a background for all values of s (Figure 9).

.

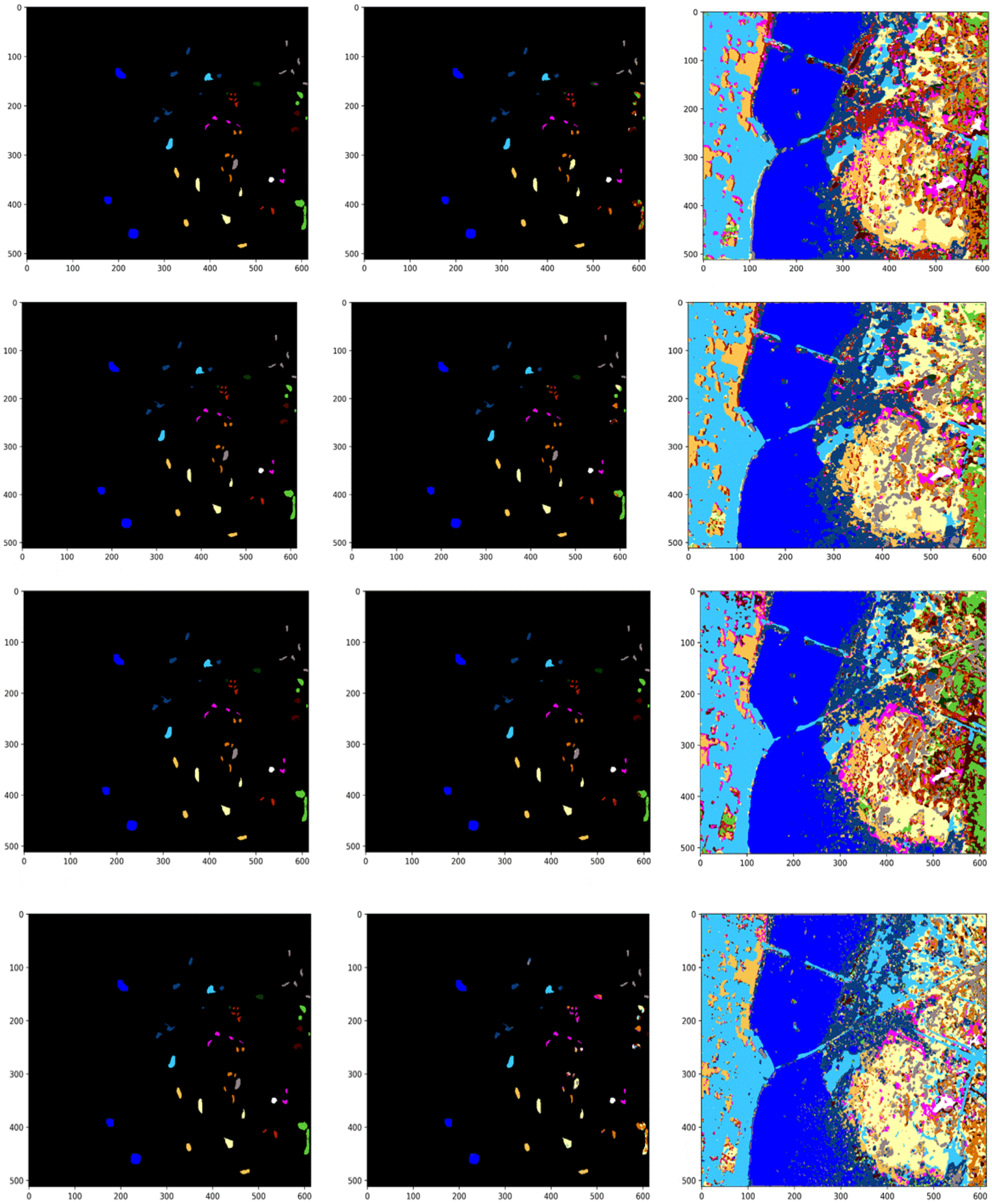


Figure 9. Classification results for the KSC data set, that each row displays original ground truth (left), classification result without background (center), and classification result with background (right) with *s* = 11, *s* = 9, *s* = 7, and *s* = 5 from upper row to lower row, respectively

Table 7. Classification accuracies obtained by our hybrid-CNN (with patch sizes of *s* = 5, *s* = 7, *s* = 9, and *s* = 11) for the KSC hyperspectral dataset

Patch Size	***s* = 5**	***s* = 7**	***s* = 9**	***s* = 11**
Training Data	**60% Samples**	**60% Samples**	**60% Samples**	**60% Samples**
Scrub	07.34 (4.12)	94.56 (1.32)	88.43 (1.05)	76.14 (1.24)
Willow swamp	100.00 (0.00)	97.21 (0.76)	83.38 (3.43)	100.00 (0.00)
Cabbage palm hammock	69.12 (4.76)	95.75 (1.01)	96.00 (1.20)	97.22 (0.80)
Cabbage palm/oak hammock	19.08 (5.67)	83.34 (3.98)	96.15 (1.09)	82.00 (2.10)
Slash pine	12.43 (3.21)	90.95 (1.75)	100.00 (0.00)	96.68 (0.46)
Oak/broadleaf hammock	22.83 (5.13)	99.44 (0.47)	74.98 (4.24)	91.16 (0.22)
Hardwood swamp	88.21 (1.54)	100.00 (0.00)	100.00 (0.00)	100.00 (0.00)
Graminoid marsh	98.74 (0.59)	94.67 (2.06)	100.00 (0.00)	58.47 (8.60)
Spartina marsh	91.48 (1.05)	100.00 (0.00)	98.00 (0.81)	100.00 (0.00)
Cattail marsh	99.19 (0.23)	97.02 (2.14)	100.00 (0.00)	100.00 (0.00)
Salt marsh	100.00 (0.00)	100.00 (0.00)	99.73 (0.29)	98.87 (0.18)
Mudflats	92.12 (2.34)	99.00 (0.94)	100.00 (0.00)	99.60 (0.08)
Water	100.00 (0.00)	100.00 (0.00)	100.00 (0.00)	100.00 (0.00)
Overall accuracy	61.45 (2.13)	96.11 (0.12)	94.73 (0.33)	87.84 (1.14)
Average accuracy	69.27 (4.76)	96.45 (0.42)	95.13 (0.65)	91.93 (0.93)

4.3.1. Comparison Three Labeled Samples (T_O, T_{KM}, T_M)

Now, we used the labeled samples obtained by the two segmentation methods to classify KSC and Botswana images to show the effect of training data increase on classification accuracy. Here, there are three types of labeled samples for both the KSC and Botswana datasets to classify images: T_O (original labeled samples), T_{KM} (originally labeled samples + K-Means labeled samples), and T_M (original labeled samples + Multi-resolution labeled samples). As mentioned earlier, 60% of labeled samples were used as training data, and since the number of labeled samples in T_{KM} and T_M is more than that in T_O, the number of training data is also more in T_{KM} and T_M. For the Botswana image, the size of the input patch for image classification with these three kinds of labeled samples is 5. The classification results are shown in Table 8. The overall accuracy of image classification with T_O, T_M, and T_{KM} data is 99.14, 99.45, and 99.97 respectively.

So, the overall accuracy increases slightly by an increased number of training data. As shown in Figure 10, we have classified the Botswana image with the background using these three kinds of labeled data.

Table 8. Classification accuracies obtained by our hybrid-CNN (with T_O, T_{KM} and T_M training samples) for the Botswana hyperspectral dataset

Patch size	***s* = 5**	***s* = 5**	***s* = 5**
Training Data	**60%** T_O	**60%** T_{KM}	**60%** T_M
Water	100.00 (0.00)	100.00 (0.00)	100.00 (0.00)
Hippo grass	100.00 (0.00)	100.00 (0.00)	100.00 (0.00)
Floodplain grasses 1	99.90 (0.24)	100.00 (0.00)	100.00 (0.00)
Floodplain grasses 2	100.00 (0.00)	100.00 (0.00)	100.00 (0.00)
Reeds	99.00 (0.24)	99.41 (0.14)	100.00 (0.00)
Riparian	96.12 (1.35)	97.02 (0.55)	99.86 (0.13)
Fires car	100.00 (0.00)	100.00 (0.00)	100.00 (0.00)
Island interior	100.00 (0.00)	100.00 (0.00)	100.00 (0.00)
Acacia woodlands	100.00 (0.00)	100.00 (0.00)	100.00 (0.00)
Acacia shrub lands	100.00 (0.00)	100.00 (0.00)	100.00 (0.00)
Acacia grasslands	100.00 (0.00)	100.00 (0.00)	100.00 (0.00)
Short mopane	100.00 (0.00)	100.00 (0.00)	100.00 (0.00)
Mixed mopane	100.00 (0.00)	100.00 (0.00)	100.00 (0.00)
Exposes soils	99.27 (0.16)	99.62 (0.05)	100.00 (0.00)
Overall accuracy	99.14 (0.17)	99.45 (0.03)	99.97(0.07)
Average accuracy	99.59 (0.11)	99.71 (0.06)	99.99 (0.02)

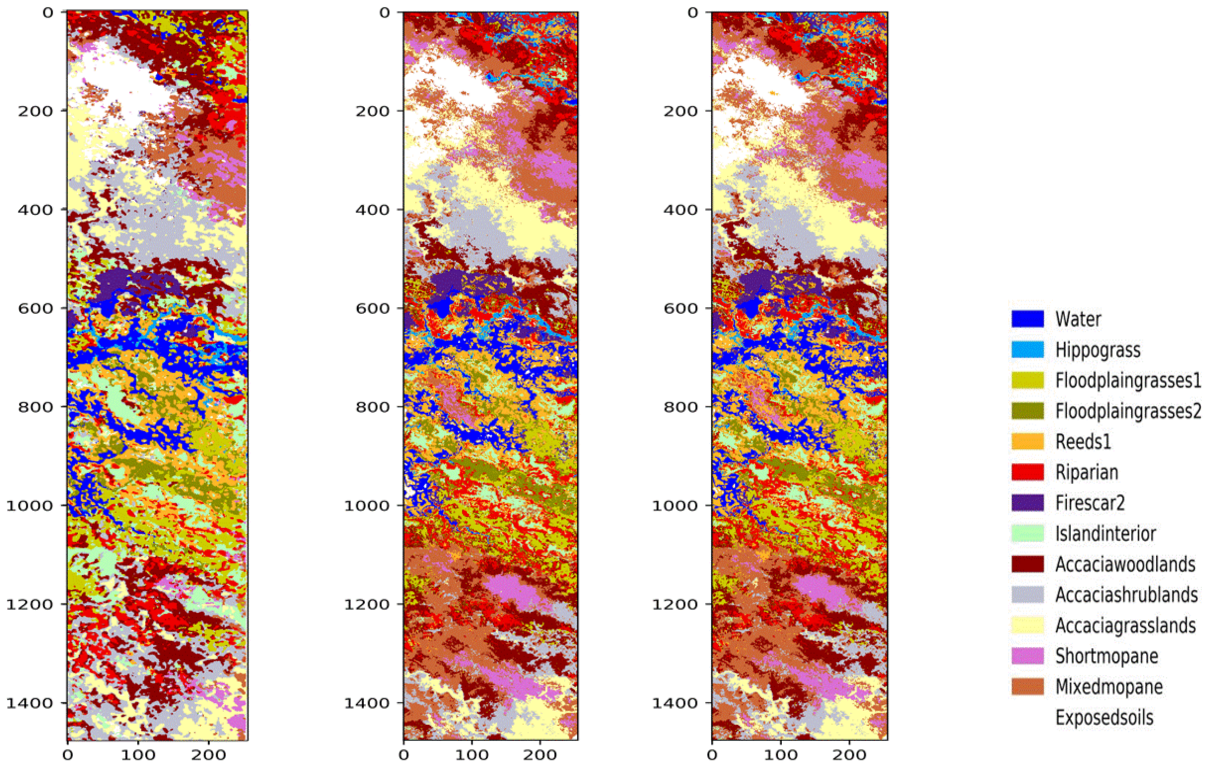


Figure 10. Classification results with background for Botswana data set using original training samples (left side of the upper row), extended training samples with K-Means segmentation (right side of the upper row), and extended training samples with Multi-resolution segmentation (left side of lower row)

The KSC dataset was also classified with these three kinds of labeled data and the classification results for each kind of labeled sample are shown in Table 9. In this case, the size of the input patch is 9. As shown in the table, the overall accuracy for classification with T_o, T_M, and T_{KM} is 94.73, 96.02, and 97.25 respectively. Compared to Botswana dataset, the effect of increased training data is more tangible in the KSC dataset. Figure 11 shows the classification of the KSC image with background for three kinds of data. By visual comparison of raw KSC images with classified ones, it can be found that the maximum classification accuracy is obtained with T_M training data. Totally, it is observed that in the classifications of Botswana and KSC images with the three kinds of labeled data (T_O, T_{KM}, T_M), the classification accuracy with T_M is always better than that with T_{KM}. This demonstrates that the segmentation accuracy of the Multi-resolution method is higher than the K-Means method.

Table 9. Classification accuracies obtained by our hybrid-CNN (with T_O, T_{KM} and T_M training samples) for the KSC hyperspectral dataset

Patch Size **Training Data**	***s* = 9** **60%** T_O	***s* = 9** **60%** T_{KM}	***s* = 9** **60%** T_M
Scrub	88.43 (1.05)	89.24 (0.91)	91.45 (0.34)
Willow swamp	83.38 (3.43)	85.83 (2.11)	88.02 (1.73)
Cabbage palm hammock	96.00 (1.20)	97.23 (1.03)	97.82 (1.11)
Cabbage palm/oak hammock	96.15 (1.09)	96.93 (0.82)	98.02 (0.29)
Slash pine	100.00 (0.00)	100.00 (0.00)	100.00 (0.00)
Oak/broadleaf hammock	74.98 (4.24)	83.19 (1.45)	87.63 (0.81)
Hardwood swamp	100.00 (0.00)	100.00 (0.00)	100.00 (0.00)
Graminoid marsh	100.00 (0.00)	100.00 (0.00)	100.00 (0.00)
Spartina marsh	98.00 (0.81)	98.63 (0.31)	99.07 (0.28)
Cattail marsh	100.00 (0.00)	100.00 (0.00)	100.00 (0.00)
Salt marsh	99.73 (0.29)	100.00 (0.00)	100.00 (0.00)
Mud flats	100.00 (0.00)	98.67 (0.19)	100.00 (0.00)
Water	100.00 (0.00)	100.00 (0.00)	100.00 (0.00)
Overall accuracy	94.73 (0.33)	96.02 (0.03)	97.25(0.07)
Average accuracy	95.13 (0.65)	96.13 (0.14)	97.07 (0.19)

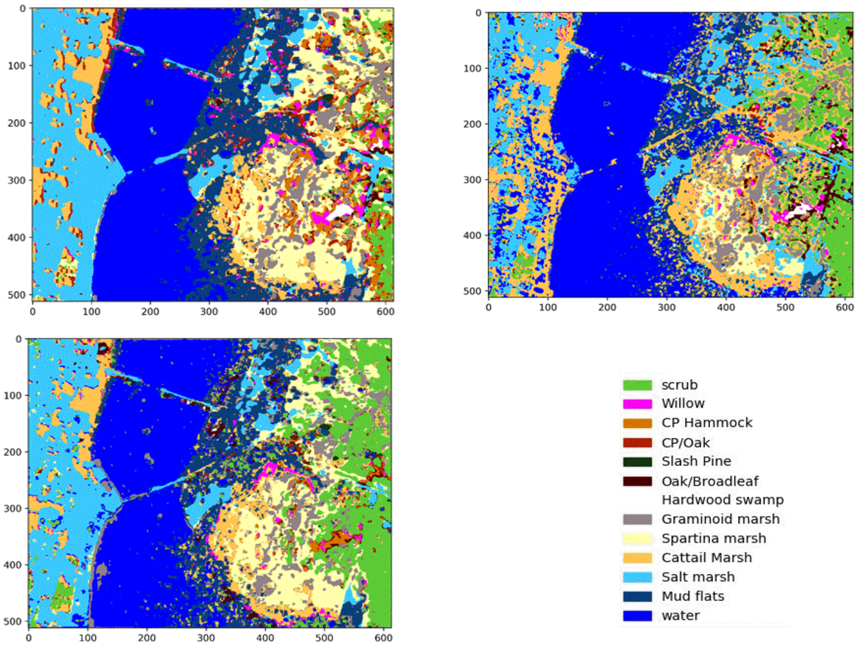


Figure 11. Classification results with background for KSC data set using original training samples (left side of the upper row), extended training samples with K-Means segmentation (right side of the upper row), and extended training samples with Multi-resolution segmentation (left side of lower row)

4.3.2. Comparative Study

In this section, we compared our proposed method with other existing methods. The two methods of 3D-CNN and 2D-CNN were selected for comparison. In all three methods, the size of the input patch for the KSC dataset was set to 7, and 60% of T_O was considered as training data. The results of classification with the three methods can be found in Table 10. As shown in the table, the classification accuracy of the proposed method outperforms that of 3D-CNN slightly and that of 2D-CNN by 5%. However, it should be noted that the the computational time of our proposed method is less than the time needed for the 3D- CNN method.

We also compared our proposed method with the two above-mentioned methods for the Botswana dataset. Here, the size of the input patch was set to 11, and 60% of T_O were used as training data. According to the classification results provided in Table 11, our proposed hybrid-CNN has better accuracy compared to the two other methods. Also, the accuracy of the 3D-CNN method is better than the accuracy of 2D-CNN for both datasets of Botswana and KSC. However, the 3D-CNN method requires more computational time compared to our proposed method and the 2D-CNN method.

Table 10. Classification accuracies were obtained by 2DCNN, 3DCNN and our hybrid-CNN methods (with original training samples) for the KSC hyperspectral dataset.

Patch Size	***s* = 7**	***s* = 7**	***s* = 7**
Method	**2DCNN**	**3DCNN**	**Ours**
Scrub	98.05 (0.15)	97.37 (0.41)	94.56 (1.32)
Willow swamp	86.40 (0.33)	98.60 (0.78)	97.21 (0.76)
Cabbage palm hammock	97.44 (0.37)	98.32 (1.11)	95.75 (1.01)
Cabbage palm/oak hammock	77.52 (0.25)	84.51 (0.13)	83.34 (3.98)
Slash pine	83.13 (0.42)	81.42 (0.20)	90.95 (1.75)
Oak/broadleaf hammock	97.61 (1.34)	90.12 (1.18)	99.44 (0.47)
Hardwood swamp	100.00 (0.00)	100.00 (0.00)	100.00 (0.00)
Graminoid marsh	92.90 (0.87)	98.45 (0.39)	94.67 (2.06)
Spartina marsh	100.00 (0.00)	100.00 (0.00)	100.00 (0.00)
Cattail marsh	69.51 (1.52)	98.19 (0.31)	97.02 (2.14)
Salt marsh	99.12 (0.41)	100.00 (0.00)	100.00 (0.00)
Mudflats	95.90 (0.17)	95.36 (0.17)	99.00 (0.94)
Water	87.66 (0.47)	100.00 (0.00)	100.00 (0.00)
Overall accuracy	91.01 (0.22)	95.92 (0.03)	96.11 (0.12)
Average accuracy	91.17 (0.42)	96.56 (0.14)	96.45 (0.42)

4.3.3. Comparative Study

The computational times needed for the training and test of both Botswana and KSC datasets are presented in Table 12. These results were obtained in the case that the size of the input patch is 7 and the number of epochs in the KSC and Botswana datasets is 200 and 100 respectively. According to the table, it can be said that the 2D-CNN method has the least computational time in training. As mentioned earlier, although the accuracy difference between our proposed method and the 3D-CNN method is slight, our proposed hybrid-CNN outperforms the 3D-CNN method in terms of computational time.

Table 11. Classification accuracies obtained by 2DCNN, 3DCNN and our hybrid-CNN methods (with original training samples) for the Botswana hyperspectral dataset

Patch size	***S*=11**	***S*=11**	***S*=11**
Training data	**2DCNN**	**3DCNN**	**Ours**
Water	91.08 (1.65)	100.00 (0.00)	100.00 (0.00)
Hippo grass	100.00 (0.00)	100.00 (0.00)	100.00 (0.00)
Floodplain grasses 1	94.21 (0.64)	100.00 (0.00)	100.00 (0.00)
Floodplain grasses 2	97.34 (0.94)	97.34 (0.65)	100.00 (0.00)
Reeds	89.43 (1.23)	92.70 (0.18)	100.00 (0.00)
Riparian	97.54 (0.54)	91.96 (0.45)	100.00 (0.00)
Fires car	91.57 (0.73)	96.01 (0.08)	100.00 (0.00)
Island interior	85.82 (1.97)	91.54 (1.01)	100.00 (0.00)
Acacia woodlands	90.67 (0.39)	98.05 (0.32)	100.00 (0.00)
Acacia shrub lands	93.32 (1.04)	98.15 (0.51)	100.00 (0.00)
Acacia grasslands	100.00 (0.00)	100.00 (0.00)	100.00 (0.00)
Short mopane	97.29 (0.48)	96.51 (0.09)	100.00 (0.00)
Mixed mopane	100.00 (0.00)	100.00 (0.00)	100.00 (0.00)
Exposes soils	90.13 (0.42)	99.62 (0.05)	100.00 (0.00)
Overall accuracy	93.26 (0.30)	96.75 (0.13)	100.00 (0.00)
Average accuracy	94.17 (0.15)	97.27 (0.21)	100.00 (0.00)

Table 12. The training time in minutes (m) and test time in seconds (s) for KSC and Botswana datasets using 2DCNN, 3DCNN and hybrid-CNN methods

	2DCNN		**3DCNN**		**Ours**	
Data	**Train (m)**	**Test (s)**	**Train (m)**	**Test (s)**	**Train (m)**	**Test (s)**
KSC	2.9	1.3	18.3	10.4	3.5	3.1
Botswana	1.7	1.1	14.7	8.1	2.6	2.7

4.3.4. Impact of the Training Sample Size

We ran the proposed hybrid-CNN algorithm with the different sizes of training data to test the accuracy of the proposed network in different training sample sizes. The results are presented in Figure 12. After increasing the number of training data, the accuracy of the proposed method does not change significantly for the Botswana dataset but it increases in the case of the KSC dataset.

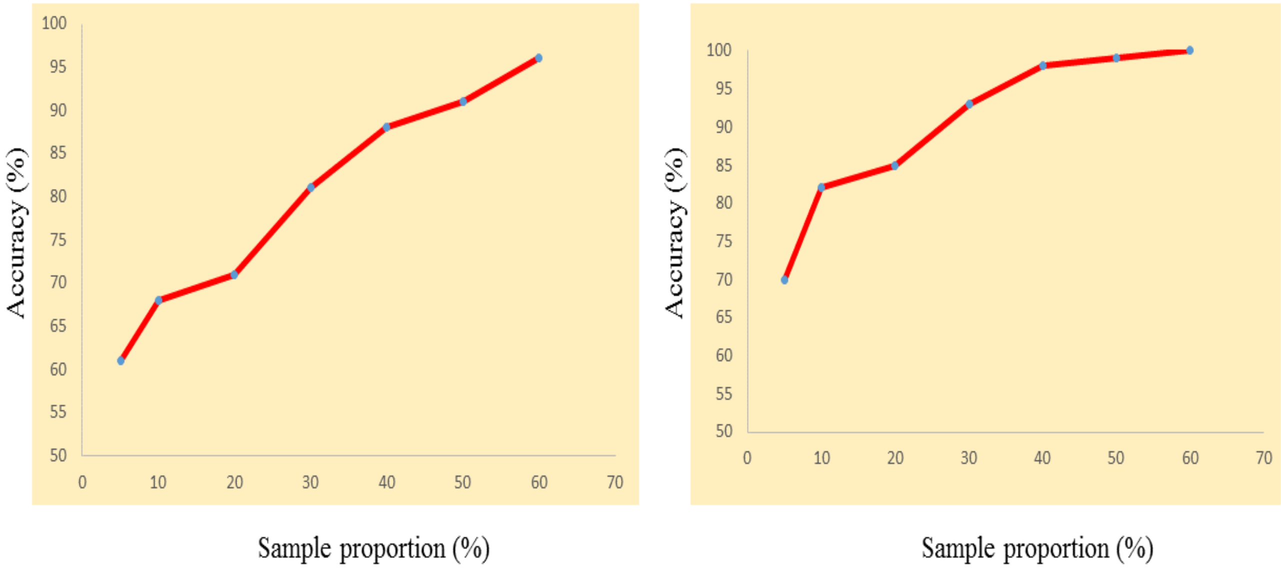


Figure 12. Influence of sample proportion on overall accuracy for KSC (left) and Botswana (right) data set.

4.3.5. Evaluation of the Proposed Hybrid-CNN with Pavia University Dataset

Today, there are many hyperspectral images that allow comparing and evaluating the algorithms. At the end of the research, we want to analyze the proposed hybrid-CNN with complex data. For this reason, we used the Pavia University dataset.

The Pavia University dataset was collected by Rosis sensor. The acquired data originally consisted of 103 bands covering the 430-810 nm portion of the spectrum with 1.3 m pixel resolution. Only 103 bands were used after uncalibrated and noisy bands that cover water absorption were removed. The data used in this paper consists of 610×340 pixels with observations from 9 identified classified classes representing the land cover types. These labeled samples are shown in Table 13. The RGB images and ground truths (GT) of the HS data are shown in Figure 13.

Table 13. Original labeled samples for Pavia University dataset.

Class	Pavia University	
	Class Name	**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

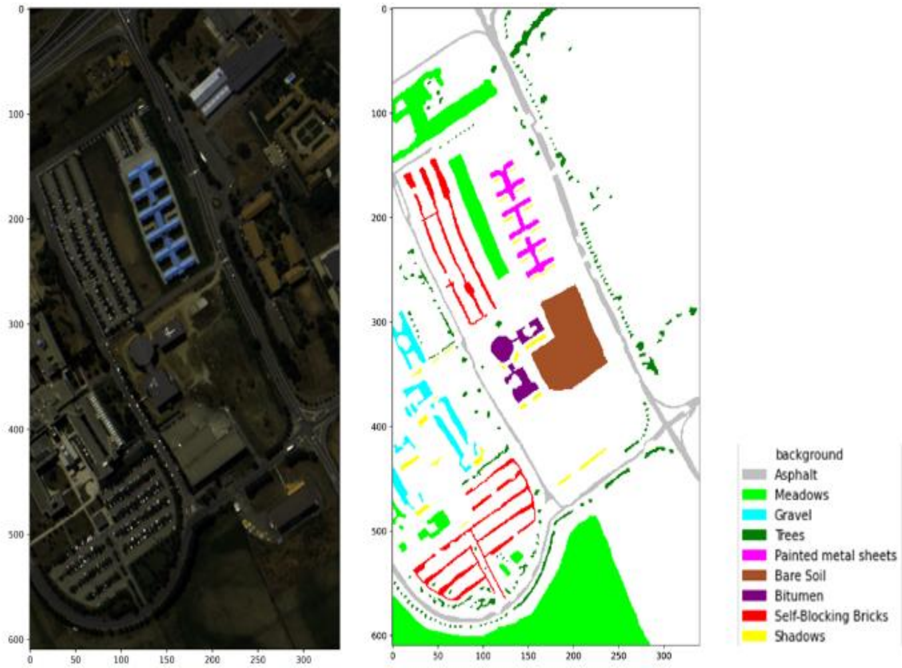


Figure 13. Original RGB and ground truth image of the Pavia University scene and name of samples from left to right, respectively

We also performed classification on the Pavia University dataset with the proposed Hybrid-CNN network. In this classification, we used 10% of the labeled samples as training data, 5% as evaluation data, and 85% as test data. The purpose of selecting little training data is to evaluate the compatibility of the hybrid-CNN network against the lack of training data. The number of epochs in this part was set to 100. And finally, the classification results were obtained in the four values of s (patch size) for the Pavia University data, as shown in Table 14.

In this research, the parameters s were tested with four values of 5, 7, 9, and 11 for the Pavia University dataset. By choosing a large s, the network can take advantage of more spatial features but it may reduce the effect of the central pixel and makes the final result erroneous. Also, if the value of the parameter s is too small, the possibility of using spatial features is minimized.

As the results show in Table 14, the proposed hybrid-CNN network has performed well on Pavia University data. In this dataset, the best result (overall accuracy = 99.66, 99.20) was for s = 11 and s = 9 respectively. As shown in Table 14, the result for s = 5 was so bad that it highlights the importance of spatial features in classification. According to the results, the classification accuracy is high in most classes of this image, and only the Gravel class does not have the proper accuracy, which is also corrected by resizing s. Therefore, the extraction accuracy of this class also depends on the spatial and neighborhood properties of the pixel. The image was classified with and without a background as shown in Figure 14.

Table 14. Classification accuracies were obtained by our hybrid-CNN (with patch sizes of $s = 5$, $s = 7$, $s = 9$, and $s = 11$) for the Pavia hyperspectral dataset.

Patch Size	**$s = 5$**	**$s = 7$**	**$s = 9$**	**$s = 11$**
Training Data	**10% Samples**	**10% Samples**	**10% Samples**	**10% Samples**
Asphalt	98.15	100.00	100.00	100.00
Meadows	99.21	100.00	100.00	100.00
Gravel	76.06	94.67	95.34	97.03
Trees	95.46	99.45	99.10	99.34
Painted metal sheets	100	99.87	100.00	100.00
Bare soil	99.00	100.00	100.00	100.00
Bitumen	98.17	100.00	100.00	100.00
Self-Blocking Bricks	98.19	98.65	98.95	99.91
Shadows	99.04	100.00	98.18	99.21
Overall accuracy	97.95	98.28	99.20	99.66
Average accuracy	95.92	99.18	99.06	99.50

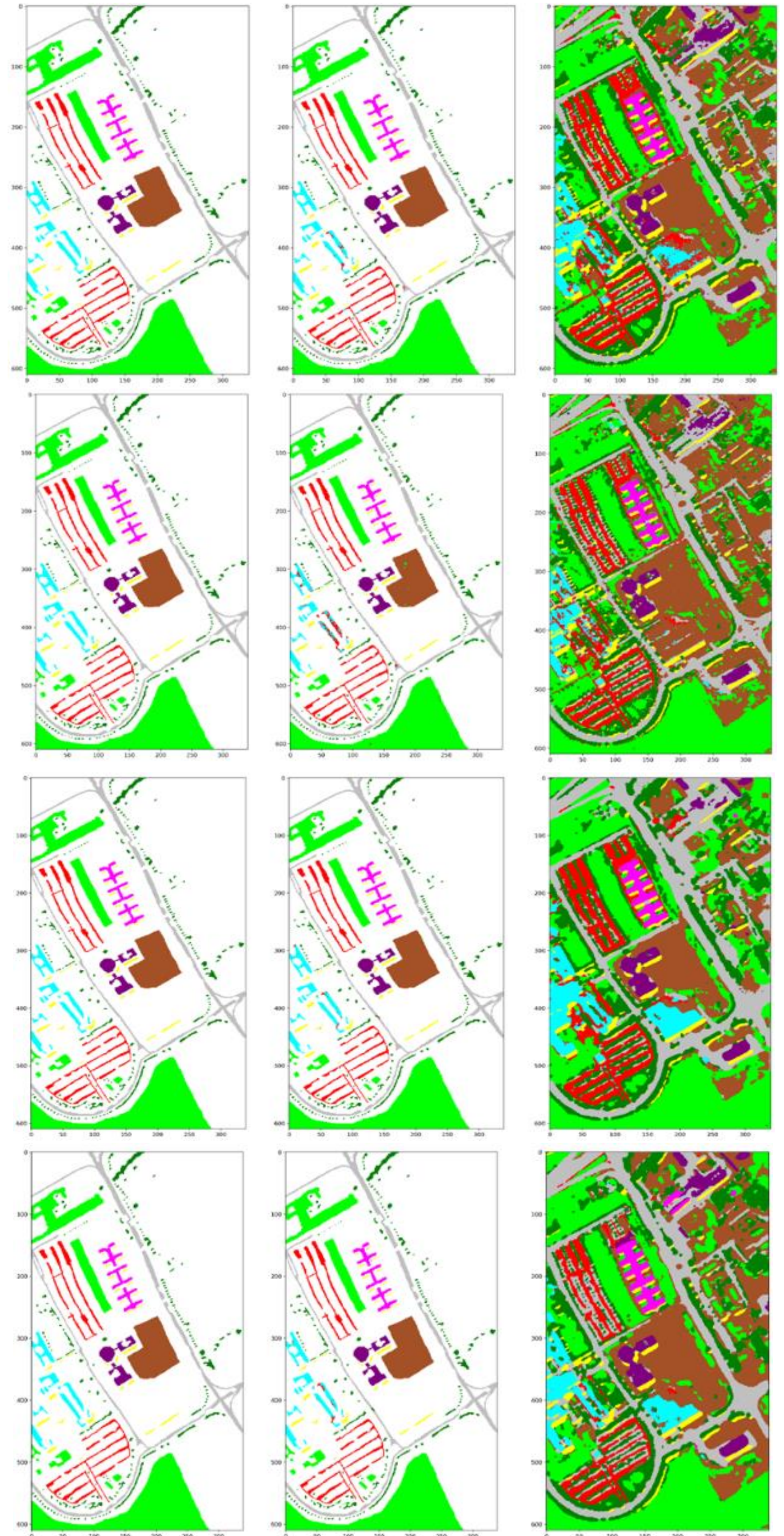


Figure 14. Classification results for Pavia University data set, that each row displays original ground truth (left), classification result without background (center), and classification result with background (right) with s = 5, s =7, s = 9, and s = 11 from upper row to lower row, respectively

5. Conclusion

In this research, we attempted to mitigate the problems associated with the classification of hyperspectral images including the lack of training data and extracting spectral and spatial information. To solve the problem of the lack of training data, we took advantage of K-Means and Multi-resolution methods for segmentation to increase training data. The classification results showed that our results are improved with an increased number of training data. Also, we introduced a hybrid-CNN network to classify hyperspectral images by extracting both spectral and spatial data. This proposed network was a combination of Convolution3D and Convolution2D. Using Convolution3D, the spectral and spatial features were extracted in different depths and passed to the next layers in which Convolution2D was used to prevent computational complexity and to distinguish the pixels with similar textural features in different bands. Also, we compared our proposed method with different methods. The results demonstrated that the hybrid-CNN outperforms other methods in terms of classification accuracy and computational time. Finally, we used the Pavia University dataset to analyze the proposed hybrid-CNN. The results show the proposed hybrid-CNN network has performed well on Pavia University data. In this dataset, overall accuracy was 99.66 for s = 11.

References

Liang L, Di L, Zhang L, Deng M, Qin Z, Zhao S., & Lin L (2015) Estimation of crop LAI using hyperspectral vegetation indices and a hybrid inversion method. Remote Sensing of Environment 165:123–134.

Zhong Y.,& Zhang, L. (2012). An adaptive artificial immune network for supervised classification of multi-hyperspectral remote sensing imagery. IEEE Transactions on Geoscience and Remote Sensing 50 (3): 894–909.

Melgani, F, & Bruzzone L. (2004) Classification of hyperspectral remote sensing imges with support vector machines. IEEE Transactions on Geoscience and Remote Sensing 42 (8): 1778–1790.

Li J, Bioucas-Dias JM, & Plaza A. (2010) Semisupervised hyperspectral image segmentation using multinomial logistic regression with active learning. IEEE Transactions on Geoscience and Remote Sensing 48 (11): 4085–4098.

Ghamisi, P, Yokoya N, Li J, Liao W, Liu S, Plaza J, Rasti B, & Plaza A. (2017) Advances in hyperspectral image and signal processing: A comprehensive overview of the state of the art. IEEE Geoscience and Remote Sensing Magazine 5 (4): 37–78.

Benediktsson JA., Palmason J, & Sveinsson JR. (2005) Classification of hyperspectral data from urban areas based on extended morphological profiles. IEEE Transactions on Geoscience and Remote Sensing 43: 480– 491.

Kang X, Xiang X, Li S, & Benediktsson JA. (2017) PCA-Based Edge-Preserving Features for Hyperspectral Image Classification. IEEE Transactions on Geoscience and Remote Sensing 55 (12):7140–7151.

Kang X, Li S, Fang L, Li M, & Benediktsson JA. (2015) Extended Random Walker-Based Classification of Hyperspectral Images. *IEEE Transactions on Geoscience and Remote Sensing* 53(1): 144–153.

Kang X, Li S, Fang L, Li M, & Benediktsson JA. (2015) Extended Random Walker-Based Classification of Hyperspectral Images. IEEE Transactions on Geoscience and Remote Sensing 53(1): 144–153.

Li, W.; Chen, C.; Zhang, M.; Li, H. & Du, Q. (2018) Data augmentation for hyperspectral image classification with deep CNN. IEEE Geosci. Remote Sens. Lett. 16, 593–597

Nalepa, J.; Myller, M.& Kawulok, M. (2019) Training-and test-time data augmentation for hyperspectral image segmentation. IEEE Geosci. Remote Sens. Lett 17, 292–296.

Haut, J.M.; Paoletti, M.E.; Plaza, J.; Plaza, A .& Li, J. (2019) Hyperspectral image classification using random occlusion data augmentation. IEEE Geosci. Remote Sens. Lett16, 1751–1755.

Alipourfard, T. & Arefi, H. (2019) Virtual Training Sample Generation by Generative Adversarial Networks for Hyperspectral Images Classification. *Int. Arch. Photogramm. Remote Sens. Spat. Inf. Sci*. 42, 63–69.

Feng J, Liu L, Cao X, Jiao L, Sun T, Zhang X (2018) Marginal Stacked Autoencoder with Adaptively-Spatial Regularization for Hyperspectral Image Classification. IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing 11(9): 3297-3311.

Zabalza J, Ren J, Zhao H, Qing C, Yang Z, Du P, Marshall S & Zheng J (2016) Novel segmented stacked autoencoder for effective dimensionality reduction and feature extraction in hyperspectral imaging. Neurocomputing 185: 1-10.

Chen Y, Zhao X & Jia X (2015) Spectral–spatial classification of hyperspectral data based on deep belief network. IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing 8(6):2381-2392.

Mou L, Ghamisi P, & Zhu X (2017) Deep recurrent neural networks for hyperspectral image classification. IEEE Transactions on Geoscience and Remote Sensing 55(7): 3639-3655.

Hu W, Huang Y, Wei L, Zhang F, & Li H (2015) Deep convolutional neural networks for hyperspectral image classification. Journal of Sensors 2015(2): 1-12.

G Yu S, Jia S, & Xu C (2017) Convolutional neural networks for hyperspectral image classification. *Neurocomputing* 219: 88-98.

Ghamisi P, Maggiori E, Li S, Souza R, Tarablaka Y, Moser G, Georgi AD, Fang L, Chen Y, Chi M, Serpico

SB, & Benediktsson JA. (2018) New Frontiers in Spectral-Spatial Hyperspectral Image Classification: The Latest Advances Based on Mathematical Morphology, Markov Random Fields, Segmentation, Sparse Representation, and Deep Learning. IEEE Geoscience and Remote Sensing Magazine 6(3): 10-43.

Kang X, Zhuo B, & Duan P. (2019) Semi-supervised deep learning for hyperspectral image classification. Remote Sensing Letters, vol. 10, no. 4, 353–362.

Wu H, & Prasad S. (2018) Semi-Supervised Deep Learning Using Pseudo Labels for Hyperspectral Image Classification. IEEE Transactions on Image Processing, vol. 27, no. 3, pp 1259 – 1270.

Baatz M, & Schäpe A. (2000) Multiresolution segmentation: An optimization approach for high quality multi-scale image segmentation. In Angewandte Geographische Informations verarbeitung, 12-23. XII. Karlsruhe, Germany: Wichmann Verlag.

Mallinis, G, Koutsias N, Tsakiri-Strati M, & Karteris M. (2008) Object-based classification using Quickbird imagery for delineating forest vegetation polygons in a Mediterranean test site. ISPRS Journal of Photogrammetry and Remote Sensing 63(2): 237–250.

Im J, Jensen J, & Tullis J. (2008) Object-based change detection using correlation image analysis and image segmentation. International Journal of Remote Sensing 29(2): 399–423.

Definiens AG 2004. *eCognition Version 4 User Guide*. München, Germany: Definiens Imaging GmbH.

Khan SS, & Ahmad A. (2004) Cluster Centre Initialization Algorithm for K-means Cluster. Pattern Recognition Letters 25(11): 1293–1302.

Dhanachandra N, Manglem K, & Chanu YJ. (2015) Image Segmentation using K-means Clustering Algorithm and Subtractive Clustering Algorithm. Procedia Computer Science 54: 764 – 771.

Ghamisi P, Plaza J, Chen Y, Li J, & Plaza A. (2017) Advanced supervised spectral classifiers for hyperspectral images: a review. IEEE Geoscience and Remote Sensing Magazine 5(1): 8–32.

Makantasis K, Karantzalos K, Doulamis A, & Doulamis N .(2015) Deep supervised learning for hyperspectral data classification through convolutional neural networks. In IEEE International Geoscience and Remote Sensing Symposium (IGARSS), 4959–4962. IEEE International.

Hamida AB, Benoit A, Lambert P, & Amar CB. (2018) 3-d deep learning approach for remote sensing image classification. IEEE Transactions on geoscience and remote sensing 56(8): 4420–4434.

M, Graña, MA Veganzons, & B Ayerdi, Hyperspectral Remote Sensing Scenes http://www.ehu.eus/ccwintco/index.php?title=Hyperspectral_Remote_Sensing_Scenes