Deep support vector machine for hyperspectral image classification

[1]**Onuwa Okwuashi & [2]Christopher E. Ndehedehe**

[1]Department of Geoinformatics and Surveying, University of Uyo, P.M.B 1017, Uyo, Nigeria

[2]Australian Rivers Institute and Griffith School of Environment & Science, Griffith University, Nathan, Queensland 4111, Australia

Abstract

To improve on the robustness of traditional machine learning approaches, emphasis has recently shifted to the integration of such methods with Deep Learning techniques. However, the classification problems, complexity and inconsistency in several spectral classifiers developed for hyperspectral images are some reasons warranting further research. This study investigates the application of Deep Support Vector Machine (DSVM) for hyperspectral image classification. Two hyperspectral images, Indian Pines and University of Pavia are used as tentative test beds for the experiment. The DSVM is implemented with four kernel functions: Exponential Radial Basis Function (ERBF), Gaussian Radial Basis Function (GRBF), neural and polynomial. Stand-alone SVMs form the interconnecting weights of the entire network. The network is trained with one hundred input datasets, and the interconnecting weights of the network are initialised using the regularisation parameter of the model. Numerical results show that the classification accuracies of the DSVM for Indian Pines and University of Pavia based on each DSVM kernel functions are: ERBF (98.87%, 98.16%), GRBF (98.90%, 98.47%), neural (98.41%, 97.27%), and polynomial (99.24%, 98.79%). By comparing the DSVM algorithm against well-known classifiers, Support Vector Machine (SVM), Deep Neural Network (DNN), Gaussian Mixture Model (GMM), K Nearest Neighbour (KNN), and K Means (KM) classifiers, the mean classification accuracies for Indian Pines and University of Pavia are: DSVM (98.86%, 98.17%), SVM (76.03%, 73.52%), DNN (94.45%, 93.79%), GMM (76.82%, 78.35%), KNN (76.87%, 78.80%), and KM (21.65%, 18.18%). These results indicate that the DSVM outperformed the other classification algorithms. The high accuracy obtained with the DSVM validates its efficacy as state-of-the-art algorithm for hyperspectral image classification.

Keywords: Remote sensing; hyperspectral image; deep support vector machine; image classification

1.0 Introduction

Hyperspectral Image (HSI) has been found invaluable because of its numerous applications and ability to obtain remotely sensed information from the visible through the near infrared wavelength ranges thus providing multi-spectral channels from the same location (e.g. Pan *et al*., 2018; Ghamisi et al., 2007). HSIs are highly innovative remote sensing imageries that consist of hundreds of contiguous narrow spectral bands, which unlike the conventional panchromatic and multispectral imageries enable a better distinct discrimination of object classes (Zhang *et al*., 2018). However, the major challenge for scientists is how to efficiently classify HSIs (see, e.g., Li *et al*., 2018). Some of these challenges as detailed by Ghamisi et al., (2007) include increased presence of redundant spectral information and high dimensionality in observed data, among others. Several conventional unsupervised and supervised machine learning classifiers have been used for classifying HSIs and includes prominent unsupervised conventional classifiers such as Fuzzy C-Means (FCM) and K Means (KM). While notable conventional supervised classifiers (e.g., K Nearest Neighbour (KNN) and Gaussian Mixture Model (GMM)) have been used in the classification of HSIs, the use of contemporary classifiers such as Support Vector Machine (SVM) and Artificial Neural Network (ANN) are gradually emerging (Melgani & Bruzzone, 2004; Ratle *et al*., 2010).

But recently, emphasis has shifted from conventional methods to the integration of Deep Learning (DL) and ANN, which scientists have argued grossly enhanced the robustness of the traditional ANN. For example, Paoletti *et al*. (2018) showed that the use of DL to train the conventional ANN significantly increased the efficiency of the ANN for HSI classification. Haut *et al*. (2018) implemented an integrated Deep Convolutional Neural Network (DCNN) using a new Bayesian approach and found that the hybrid of DL and the traditional ANN classifier enhanced the efficiency of the traditional ANN classifier. Furthermore, a novel guided filter based Deep Recurrent Neural Network (DRNN) for HSI classification has proved to be more efficient than the traditional ANN classifier (Guo *et al*. 2018). Zhao *et al*. (2019) recently proposed an integrated Convolutional Neural Network (CNN) and Gray Level Co-occurrence Matrix (GLCM) textural features for HSI classification using limited training sample. More recently (e.g., Li et al., 2019), other DL ANN algorithms such as the use of Deep Belief

Network have been employed in HSI classification with major highlights on their merits as opposed to conventional methods. However, classification problems associated with small-size training dataset in traditional machine learning techniques have been reported (e.g., Chi et al., 2008).

Recent advances in convolutional neural networks, including activation function, loss function, regularization, optimization and fast computation have been documented. Gu et al., (2018) who provided details on these advances also highlighted the weakness of CNN, indicating computational efficiency and the choice of a suitable hyper-parameters (e.g., learning rate, kernel sizes of convolutional filters) are still challenging issues, especially for large-scale data. The use of a new CNN architecture for the classification of hyperspectral images was therefore predicated on the computational constraints of CNN algorithms to high-dimensional data contained in multidimensional data cubes (Paoletti et al., 2018). Although the application of deep learning techniques, especially CNN in image-based cancer detection, diagnosis and other disciplines have shown significant strength (Hu et al., 2018; Li et al., 2018), improvements are required to handle large-scale multi-resolution data cubes. It is against this background, that assessing the skills of other non-parametric deep learning algorithms such as the deep SVM has become necessary.

In addition to other classification methods such as random forests, neural networks, and logistic regression-based techniques, the SVM (Cortes and Vapnik 1995) is another robust classifier that has been used in hyperspectral data classification (e.g., Bigdeli et al., 2013; Ghamisi et al., 2007). Since the introduction of the SVM it has proven to be very efficient in remote sensing (RS) image classification, tide analysis, and prediction of urban land use change (e.g., Okwuashi & Ndehedehe, 2017). Due to SVM's ability to model complex real-world data, they were found to be relatively better predictive models as opposed to agent-based models whose inability to evaluate model constraints and results have been highlighted in previous studies (e.g., Poursaee, 2018; O'Sullivan, 2004; Zhao & Peng, 2012; Okwuashi, 2011). Even though new algorithms such as the Supervised Fuzzy Partitioning are now competitive with the SVM, the latter is still largely characterized as a state-of-the-art algorithm (Ashtari et al., 2020). SVM is intrinsically a binary classifier, it can be modified however, for multi-class problems by using mainly the One Against All (OAA) or One Against One (OAO) technique

(Kang *et al*., 2015). The OAA and OAO techniques have proven to be considerably effective in the classification of remote sensing images (Pal & Mather, 2005).

While classification problems still exist with traditional machine learning techniques, the complexity (e.g., availability of training samples) surrounding the implementation of different classification algorithms are some reasons warranting further research on ideal techniques for HSI. This was echoed in Ghamisi et al., (2007) who noted the inconsistency in several spectral classifiers developed for HSI based on selected metrics. Furthermore, the use of deep belief network in improving classification outputs of hyperspectral images and multi-temporal images has recently been demonstrated (Li et al., 2019; Kussul et al., 2017). Whereas several parametric and prominent non-parametric algorithms have been widely used in image classification (see, e.g., Melgani & Bruzzone, 2004; Ratle *et al*., 2010; Ghamisi et al., 2007), the assessment and accuracy of HSI classification based on Deep Support Vector Machine (DSVM) however, is largely undocumented. One of the key challenges with HSI classification is limited training samples. It is for this reason that the development of new optimization algorithms such as deep learning is increasingly becoming popular and effective in the fields of image recognition and classification, especially for the classification of large multi-spectral and hyperspectral datasets (Paoletti et al 2018). Moreover, the success of these new algorithms in automatic feature extraction, computer vision, language processing and speech recognition have recently been re-echoed (Li et al 2018). There are still some constraints nonetheless, on the application of these methods to multispectral and hyperspectral images. A regularized ensemble framework of deep learning that incorporates SVM is therefore crucial to further explore these challenges.

Arguably, deep learning algorithms have attracted the attention of the remote sensing community and several other experts in the fields of speech recognition, computer vision, and natural language processing among others. To explore the application of deep learning methods in hyperspectral image classification, a multi-grained network that appears to be an ensemble deep learning method has been proposed (Pan et al., 2018). Another hybrid model that integrates unsupervised deep belief network with a one-class SVM were found to be scalable and computationally efficient in an earlier study (Erfani et al., 2016). While theoretical foundations and optimization techniques for learning deep CNN

architectures are required (Gu et al., 2018), ensemble deep learning methods have shown considerable potentials in the classification of HSI classification. For example, the coupling of deep belief network with a SVM algorithm addressed complexity and scalability issues of SVM in large datasets (Erfani et al., 2016). The use of hybrid models is therefore emerging as efficient, accurate and scalable techniques that can improve the classification accuracy of large-scale and high-dimensional data.

The aim of this research therefore is to integrate DL and SVM to formulate a hybrid DSVM. The architecture of the DSVM proposed for this experiment imitates the Deep Neural Network (DNN) that consists of multiple hidden layers. Normally the hidden layer neurons are connected by series of weights; but instead the weights are initialised by several SVM functions modified with the SVM regularisation parameter. The optimal DSVM output for each input is found by updating all the connecting SVM functions in the hidden layer. Two HSIs, Indian Pines and University of Pavia are used as experimental and tentative test beds for the study. The OAA multi-class technique are used to modify the SVM for multi-class separation. To assess the robustness of our hybrid DSVM model, the results of the DSVM are compared to those of the SVM, DNN, GMM, KNN, and KM.

2.0 Materials and method

2.1.1 Deep support vector machine framework

SVM classifies a binary problem using a linear hyperplane by assuming that the training set has *n*-training samples, that is, $(x_1, y_1),(x_2, y_2),...,(x_n, y_n)$, where $x_i \in \Re^N$ is an N dimensional vector that belongs to one of classes $y_i \in \{-1,+1\}$ (Guyon *et al*., 2002). The stated binary classification problem can be separated using a linear decision function,

$$f(x) = w \cdot x + b \quad (1)$$

where $w \in \Re^N$ is a vector that determines the orientation of the desired hyperplane required for the separation, and $b \in \Re$ is called the "bias." The optimal hyperplane needed to separate the two objects is,

$$y_i(w \cdot x + b) \geq 1 \quad (2)$$

The solution to this problem can be found by solving the following constrained optimization problem (or primal problem),

$$\text{minimise}\ \frac{1}{2} w \cdot w + C\sum_{i=1}^{n}\xi_i \tag{3}$$

subject to: $y_i(w \cdot x + b) \geq 1 - \xi_i$, $\xi_i > 0$, and for $\forall i = 1,...,n$; where C, $0 < C < \infty$, is called the penalty value or regularisation parameter; while ξ_i are the slack variables (Chen *et al*., 2018). For a nonlinear case the optimisation problem can be written as,

$$\text{maximise}\ \sum_{i=1}^{n}\alpha_i - \frac{1}{2}\sum_{i=1}^{n}\sum_{j=1}^{n}\alpha_i\alpha_j y_i y_j K(x_i, x_j) \tag{4}$$

subject to: $\sum_{i=1}^{n}\alpha_i y_i = 0$, and, $0 \leq \alpha_i \leq C$, for $i = 1,...,n$. While the resulting decision function is,

$$f(x) = sign\left[\sum_{i=1}^{n} y_i \alpha_i^0 K(x_i, x) + b^0\right] \text{ (Pirra \& Diana, 2019).} \tag{5}$$

α_i^0 is the support vector while $K(x_i, x)$ is the kernel function or kernel trick. DSVM can be formulated by using a multi-layer architecture that contains multiple hidden layers (Figure 1).

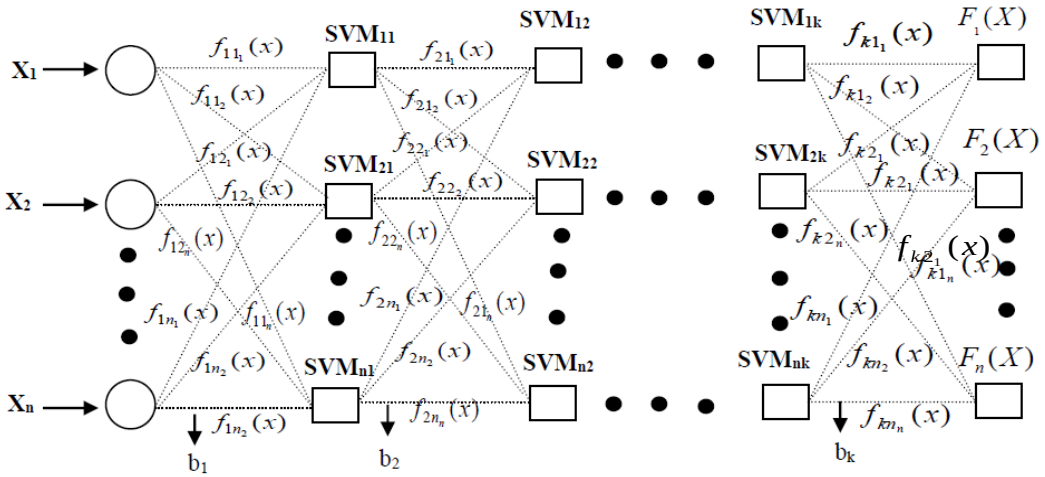


Figure 1: Architecture of the DSVM implemented in this study.

X_1, X_2, …, X_n represent the input layer data points. The multiple hidden layers consist SVM_{11}, SVM_{12}, …, SVM_{1k}, SVM_{21}, SVM_{22}, …, SVM_{2k}, and SVM_{n1}, SVM_{n2}, …, SVM_{nk}; while $F_1(X)$, $F_2(X)$,…, $F_n(X)$ represent the output layer points. For X_1, the output for training SVM_{11}, SVM_{12}, …, SVM_{1k} is $F_1(X)$. For X_2, the output for training SVM_{21}, SVM_{22}, …, SVM_{2k} is $F_2(X)$. For X_n, the output for training SVM_{n1}, SVM_{n2}, …, SVM_{nk} is $F_n(X)$. The network weights are represented as $f(x)$. All the various $f(x)$ are evaluated in the hidden layers, which are multiple layers that connect all the input

neurons with the output neurons (Fig. 1). The total net input to each hidden layer neuron can be expressed as,

$$\begin{aligned} net_{h1} &= f_{11_1}(x)\cdot X_1 + f_{11_2}(x)\cdot X_2 + \cdots + f_{11_n}(x)\cdot X_n + b1 \\ net_{h2} &= f_{12_1}(x)\cdot X_1 + f_{12_2}(x)\cdot X_2 + \cdots + f_{11_n}(x)\cdot X_n + b1 \\ &\bullet \\ &\bullet \\ &\bullet \\ net_{hn} &= f_{1n_1}(x)\cdot X_1 + f_{1n_2}\cdot X_2 + \cdots + f_{11_n}(x)\cdot X_n + b1 \end{aligned} \quad (6)$$

The logistic activation function is used to compute the output for each input neuron as,

$$\begin{aligned} out_{h1} &= \frac{1}{1+e^{-net_{h1}}} \\ out_{h2} &= \frac{1}{1+e^{-net_{h2}}} \\ &\bullet \\ &\bullet \\ &\bullet \\ out_{hn} &= \frac{1}{1+e^{-net_{hn}}} \end{aligned} \quad (7)$$

The output of the hidden layer neurons are used as input to compute the output layer neurons $net_{o1_1}...,net_{o1_n}$, $net_{o2_1}...,net_{o2_n}$, and $net_{on_1}...,net_{on_n}$ as,

$$\begin{aligned} net_{o1_1} &= f_{21_1}(x)\cdot out_{h1} + f_{21_2}(x)\cdot out_{h2} + \cdots + f_{21_n}(x)\cdot out_{hn} + b_2 \\ &\bullet \\ &\bullet \\ &\bullet \\ net_{o1_n} &= f_{k1_1}(x)\cdot out_{h1} + f_{k1_2}(x)\cdot out_{h2} + \cdots + f_{k1_n}(x)\cdot out_{hn} + b_2 \end{aligned} \quad , \quad (8)$$

$$\begin{aligned} net_{o2_1} &= f_{22_1}(x)\cdot out_{h1} + f_{22_2}(x)\cdot out_{h2} + \cdots + f_{22_n}(x)\cdot out_{hn} + b_2 \\ &\bullet \\ &\bullet \\ &\bullet \\ net_{o2_n} &= f_{k2_1}(x)\cdot out_{h1} + f_{k2_2}(x)\cdot out_{h2} + \cdots + f_{k2_n}(x)\cdot out_{hn} + b_2 \end{aligned} \quad , \text{ and} \quad (9)$$

$$\begin{aligned} net_{on_1} &= f_{2n_1}(x)\cdot out_{h1} + f_{2n_2}(x)\cdot out_{h2} + \cdots + f_{2n_n}(x)\cdot out_{hn} + b_k \\ &\bullet \\ &\bullet \\ &\bullet \\ net_{on_n} &= f_{kn_1}(x)\cdot out_{h1} + f_{kn_2}(x)\cdot out_{h2} + \cdots + f_{kn_n}(x)\cdot out_{hn} + b_k \end{aligned} \quad . \quad (10)$$

For simplicity let us consider only the case of $net_{o1_1}...,net_{o1_n}$. Its output can be computed with the logistic activation function as,

$$out_{o1_1} = \frac{1}{1+e^{-net_{o1_1}}}$$
$$\bullet$$
$$\bullet \qquad . \qquad (11)$$
$$\bullet$$
$$out_{o1_n} = \frac{1}{1+e^{-net_{o1_n}}}$$

The error for computing the output $output_{o1}$ for only X_1 , can be calculated by subtracting the computed output $output_{o1}$ from the known value of $F_1(X)$ as,

$$E_{o1} = \sum_{i=1}^{n} \tfrac{1}{2}\left(F_1(X) - output_{o1_i}\right) \qquad (12)$$

In like manner the total error can be computed by summing all the computed errors E_{o1}, E_{o2},…, E_{on} as,

$$E_{total} = E_{o1} + E_{o2} + \cdots + E_{on} \qquad (13)$$

By applying the method of backpropagation we can update each $f(x)$ in the network so that they will ensure that the actual output becomes closer to the target output $F(X)$, thereby minimising the error for each of the output neurons as well as the entire network. For example, $f_{11_1}(x)$ can be computed as the gradient of ∂E_{total} as,

$$\frac{\partial E_{total}}{f_{11_1}(x)} = \frac{\partial net_{o1_1}}{\partial f_{11_1}(x)} * \frac{\partial out_{o1_1}}{\partial net_{o1_1}} * \frac{\partial E_{total}}{\partial out_{o1_1}} \qquad (14)$$

The updated function $f_{11}^{(new)}(x)$ can be computed as,

$$f_{11_1}^{(new)}(x) = f_{11_1}(x) - \lambda * \frac{\partial E_{total}}{\partial f_{11_1}(x)} \qquad (15)$$

Where λ denotes the learning rate for adjusting the weights of the network. In like manner all the weights in the network that is $f(x)$ will be updated, and the process will be repeated iteratively from equation 6 until E_{total} becomes zero or infinitesimal.

2.1.2 Data and implementation

The first HSI dataset is the Indian Pines region in Northwest Indiana, USA acquired by the Airborne Visible/Infrared Imaging Spectrometer (AVIRIS) that covers the agricultural fields with regular geometry. It has a 145 X 145 pixels scene with 20m spatial resolution and 220 spectral bands in the 0.4-2.45 μm region (Fig. 2). The image contains 16 ground-truth classes. The second HSI dataset is the University of Pavia, Italy acquired by Reflective Optics Spectrographic Image System (ROSIS-03) sensor over the urban area of the University of Pavia, Italy. It generates 115 spectral bands. It has a spatial resolution of 1.3 m and a scene that contains 610 × 340 pixels. The image contains 9 ground-truth classes (Fig. 2). For Indian Pines 10,114 pixels were extracted which consisted of 1,022 training datasets ($X_1, X_2, \ldots, X_{100}$) and 9,092 test datasets (Table 1). For University of Pavia a total of 42,779 pixels were extracted that consisted of 441 training datasets ($X_1, X_2, \ldots, X_{100}$) and 42,338 testing datasets (Table 2). For Indian Pines, the training and test samples were extracted from each of the sixteen classes, while for University of Pavia the training and test samples were extracted from each of the nine classes. The Exponential Radial Basis Function (ERBF) kernel $K(x_i, x_j) = \exp\left(-\frac{\|x-y\|}{2\gamma^2}\right)$, Gaussian Radial Basis Function (GRBF) kernel $K(x_i, x_j) = \exp\left(-\frac{\|x-y\|^2}{2\gamma^2}\right)$, neural kernel $K(x_i, x_j) = \tanh(ax_i \cdot x_j + b)$, and polynomial kernel $K(x_i, x_j) = (x_i \cdot x_j + c)^d$ were implemented for the DSVM. The designated output labels were -1 and +1. The optimal model parameters for ERBF, GRBF, neural and polynomial kernels were obtained using a cross-validation procedure (He & Fan, 2019). For $f_{11_1}(x)$, for ERBF and GRBF, the optimal Gamma value was determined as γ = 0.6, for the neural kernel a = 0.4 and b = 0, for polynomial kernel c = 0, and d = 2 (Fig. 3).

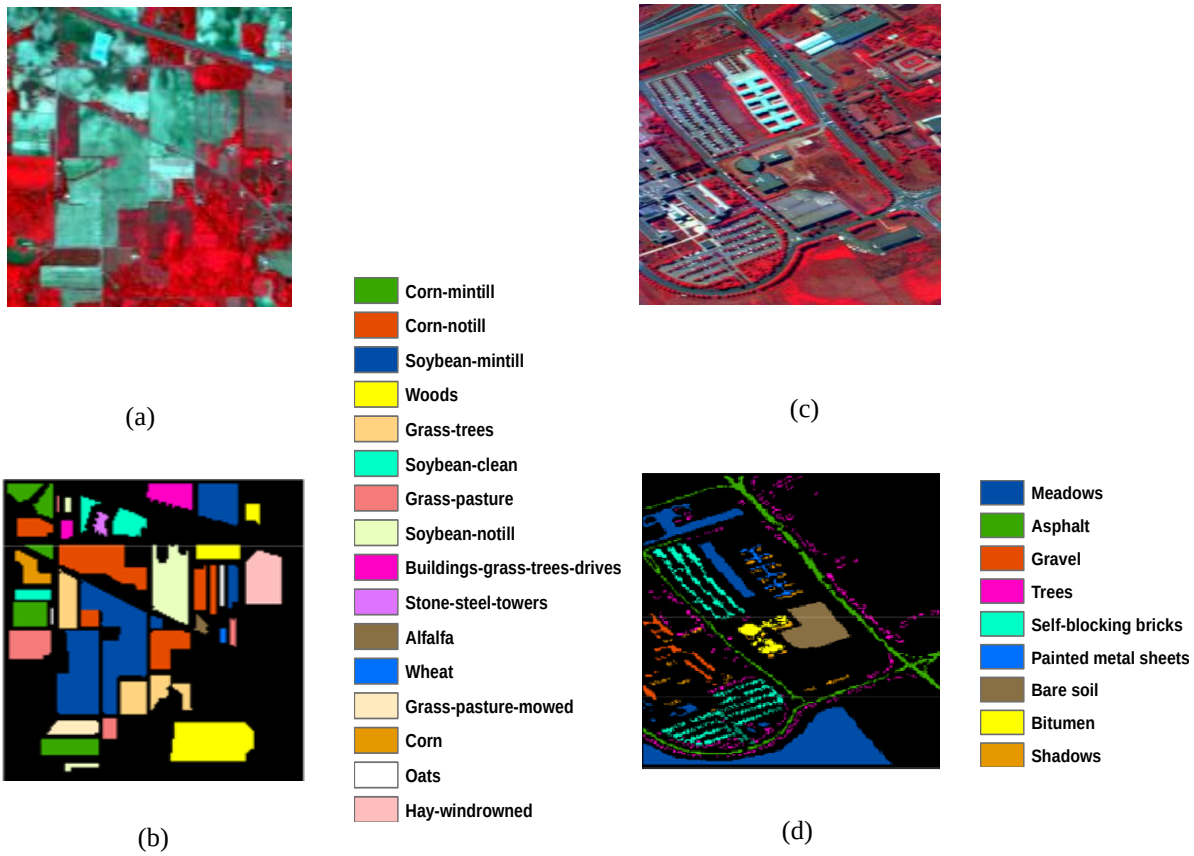


Figure 2: (a) False colour image for Indian Pines (b) Ground truth labels for Indian Pines (c) False colour for University of Pavia (d) Ground truth labels for University of Pavia.

Table 1: Training and testing sets for the Indian Pines.

Class	Name	Training set			Testing set
		X_1	X_2	$\cdots$ X_{100}	
1	Corn-mintill	86	82	84	755
2	Corn-notill	142	140	139	1283
3	Soybean-mintill	247	248	247	2210
4	Woods	130	129	130	1143
5	Grass-trees	75	76	74	658
6	Soybean-clean	55	56	57	530
7	Grass-pasture	49	45	48	435
8	Soybean-notill	96	93	95	875
9	Buiding-grass-trees-drives	39	39	42	349
10	Stone-steel-towers	10	14	10	86
11	Alfalfa	5	7	6	41
12	Wheat	6	9	8	43
13	Grass-pasture-mowed	7	7	8	25
14	Corn	23	25	25	212
15	Oats	5	6	7	16
16	Hay-windrowned	47	46	42	431
Total		1,022	1,022	1,022	9,092

Table 2: Training and testing sets for the University of Pavia.

Class	Name	Training set				Testing set
		X_1	X_2	. . .	X_{100}	
1	Meadows	191	189		190	18,450
2	Asphalt	67	70		66	6,564
3	Gravel	22	20		21	2,081
4	Trees	33	33		32	3,030
5	Self-blocking bricks	39	40		42	3,644
6	Painted metal sheets	15	16		15	1,333
7	Bare soil	50	48		49	4,980
8	Bitumen	14	14		15	1,319
9	Shadows	10	11		11	937
Total		441	441	. . .	441	42,338

The experiment was based on a multi-class OAA approach. The idea of the OAA approach is to classify a multi-class data by classifying each of the classes of interest against the remaining classes. The OAA technique is a traditional and the most widely used approach to extend SVM binary classification to a multi-class scenario. Different regularisation values $0 \le C \le \infty$ were used to set the initial weights of the network for $f_{11_1}(x)$, $f_{11_2}(x)$, $f_{12_1}(x)$, $f_{12_2}(x)$, …, $f_{kn_n}(x)$. The next procedure was the training of the network by backpropagation which simply updates the network weights $f(x)$. The error function or loss function $E = \sum \frac{1}{2}\left(F(X) - output\right)$ was used to determine when to terminate the iteration. The learning of the network terminates when the error function is zero or infinitesimal. In terms of computational cost of any algorithm that solves the SVM problem for arbitrary kernel matrices, there are two complexities involved: training and testing time. Generally, experimental results could indicate that the running-time associated with testing is smaller than training. Although characterizing the time complexity of DSVM is somewhat complex, in running a traditional SVM, space and time complexity are linear with respect to the number of support vectors. Given that asymptotical number of support vectors grows linearly with the number of examples, the computational cost of solving the SVM problem has both a quadratic and a cubic component.

3.0 Results

The DSVM presented in this study was designed to mimic the operation of the DNN. The individual SVMs that is, $f(x)$ were made to function as interconnecting weights of the network. The resulting outputs were compared against the target output $F(X)$ based on the backpropagation technique. One

hundred distinct inputs $X_1, X_2, \ldots, X_{100}$ were used to obtain one hundred distinct outputs $F_1(X)$, $F_2(X), \ldots, F_{100}(X)$. The regularisation parameter was used to initialise the network in order to ensure that the network was initialised with different weights. Starting with an initial weight from +1 to -1 for $f_{11_1}(x)$, Table 3 shows some of the results of the updated weights of $f_{11_1}(x)$ and some of the predicted results of $F_1(X)$. The weights of $f_{11_1}(x)$ yielded new results in each iteration $f_{11_1}(x)(1)$, $f_{11_1}(x)(2)$, $f_{11_1}(x)(3)$, $f_{11_1}(x)(4)$. The support vectors are the non-zero values 3.1958 and 5.1466, hence the predicted $f_{11_1}(x)(1)$ that corresponds to each support vector must either be -1.0000 or +1.0000. The results of $F_1(X)$, $F_2(X), \ldots, F_{100}(X)$ is depicted in Fig. 4. It can be observed that the accuracies of $F_1(X)$, $F_2(X), \ldots, F_{100}(X)$ vary relatively significantly (Fig 4). Based on all the DSVM kernels, classification accuracies for Indian Pines are generally higher than those of University of Pavia (Fig 4).

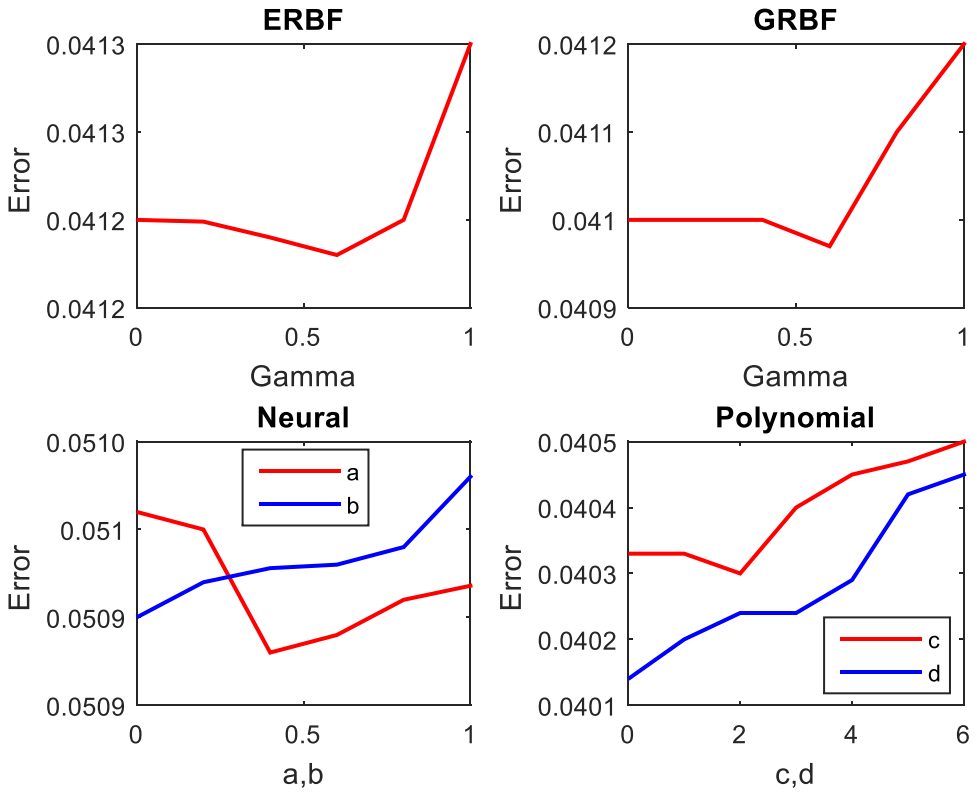


Figure 3: Selection of optimal model parameters for ERBF, GRBF, neural, and polynomial kernels.

Table 3: Some of the SVM-based training and testing results.

$f_{11_1}(x)$ (initial)	Support vectors	$f_{11_1}(x)$ (1)	$f_{11_1}(x)$ (2)	$f_{11_1}(x)$ (3)	$f_{11_1}(x)$ (4)	. . .	$F_1(X)$ (initial)	$F_1(X)$ (predicted)
+1	0	+8.6473	+8.6291	+8.6307	+8.6602	. . .	+1	+8.6595
+1	3.1958	+1.0000	+1.0436	+1.0373	+1.0268	. . .	+1	+1.0181
+1	0	+5.1327	+5.1554	+5.1641	+5.1600		+1	+5.1462
.	.	.	.	.	.	.	.	.
.	.	.	.	.	.	.	.	.
.	.	.	.	.	.	.	.	.
-1	0	-9.2946	-9.2808	-9.2509	-9.2773	. . .	-1	-9.2840
-1	0	-6.3453	-6.3597	-6.3810	-6.3702	. . .	-1	-6.3678
-1	5.1466	-1.0000	-1.0088	-1.0212	-1.0197		-1	-1.0141
.	.	.	.	.	.	.	.	.
.	.	.	.	.	.	.	.	.
.	.	.	.	.	.	.	.	.

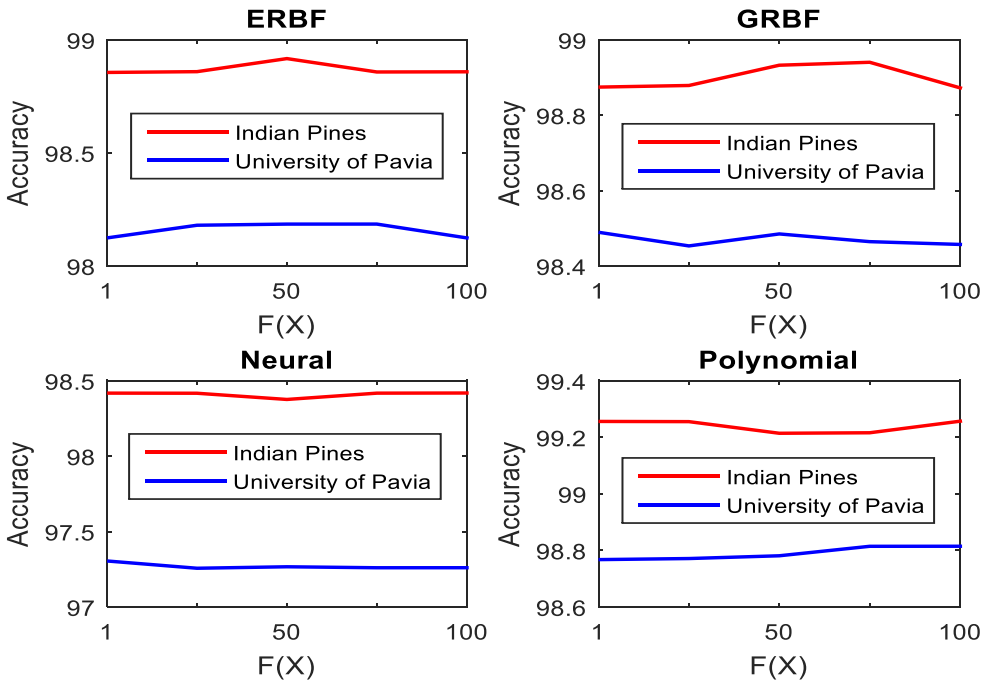
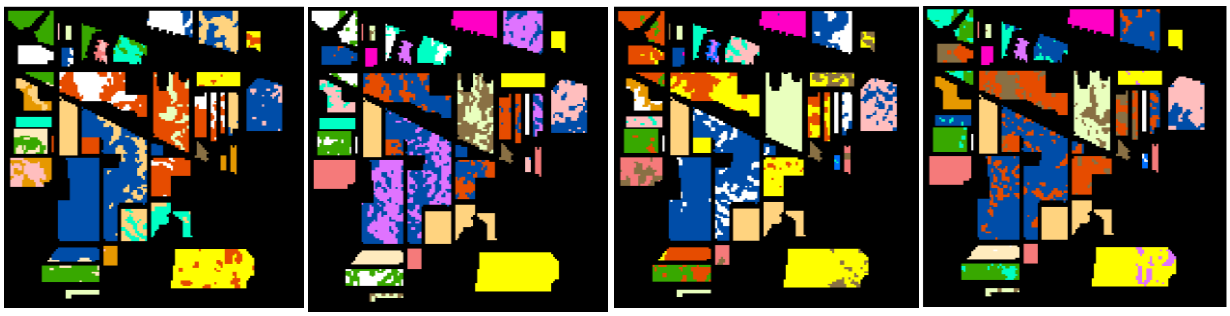


Figure 4: Mean classification accuracy for Indian Pines and University of Pavia for DSVM.

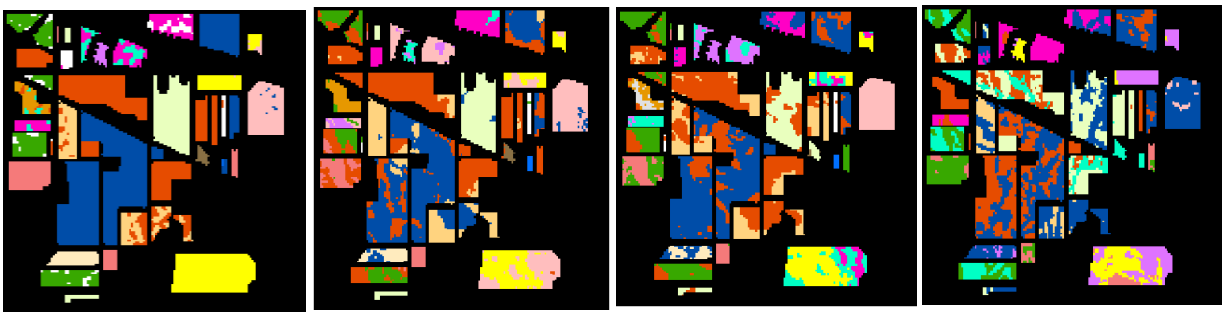
The DSVM classification results for Indian Pines and University of Pavia for the four kernel functions are indicated in Figs. 5a-l and 6a-l. The results of a second set of experiment implemented with SVM, DNN, GMM, KNN, and KM for Indian Pines and University of Pavia are given in Figs. 5 and 6. Generally, the DSVM technique showed little misclassification of land cover states in Indian Pines and University of Pavia as opposed to other methods where misclassifications are obvious and somewhat widespread. The classification accuracy results for Indian Pines are summarised in Table 4. It shows

that GRBF, neural, and polynomial kernels achieved 100% classification accuracy for Corn-mintill; while only ERBF and polynomial kernels achieved 100% classification accuracy for Corn-notill. The results further showed that all the four kernels achieved 100% classification accuracy for Soybean-mintill (Table 4). Notably, GRBF and polynomial kernels achieved 100% classification accuracy for Woods. But only GRBF kernel achieved 100% classification accuracy for Grass-trees; while only ERBF kernel achieved 100% classification accuracy for Soybean-clean. The polynomial kernel achieved 100% classification accuracy for Grass-pasture while only ERBF and GRBF kernels yielded 100% classification accuracy for Soybean-notill.

(a) DSVM (ERBF) (b) DSVM (GRBF) (c) DSVM (Neural) (d) DSVM (Polynomial)



(e) SVM (ERBF) (f) SVM (GRBF) (g) SVM (Neural) (h) SVM (Polynomial)

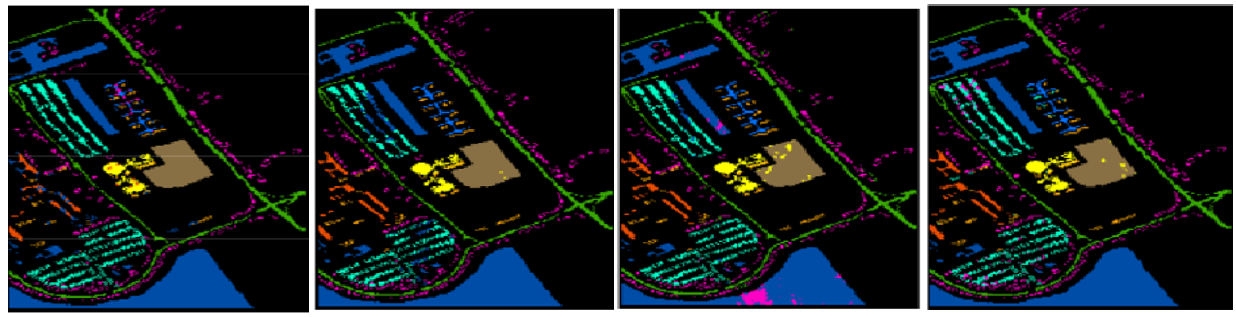


(i) DNN (j) GMM (k) KNN (l) KM

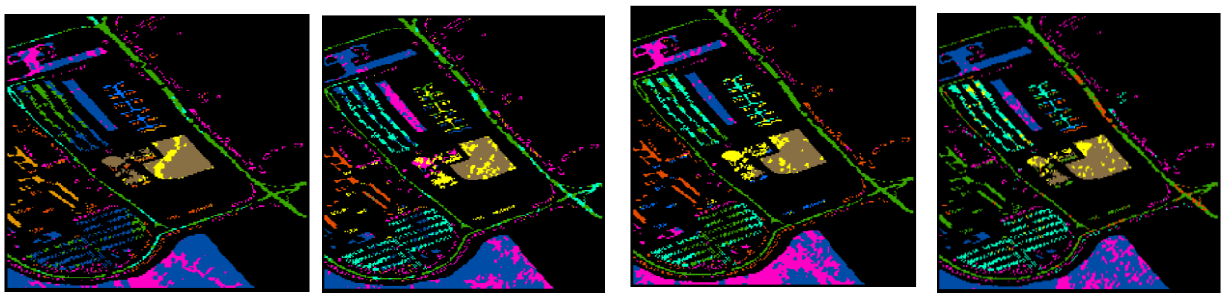
Figure 5: Classified images for Indian Pines. The legend is the same as indicated in Fig. 2.

Moreover, all the four kernels achieved 100% classification accuracy for Building-grass-trees-drives while the neural kernel yielded the highest classification accuracy (98.11%) for Stone-steel-towers. Only the ERBF and polynomial kernels yielded 100% classification accuracy for Alfalfa; while ERBF, GRBF and neural kernels were consistent, and all yielded a 100% classification accuracy for Wheat. ERBF, GRBF and polynomial kernels yielded 100% classification accuracy for Grass-pasture-mowed while only ERBF, neural and polynomial kernels yielded 100% classification accuracy for Corn. ERBF kernel yielded the highest classification accuracy for Oats (99.14%) while GRBF and polynomial kernels performed relatively better as the yielded 100% classification accuracy for Hay-windrowned unlike ERBF and neural kernels (Table 4 and Fig 5).

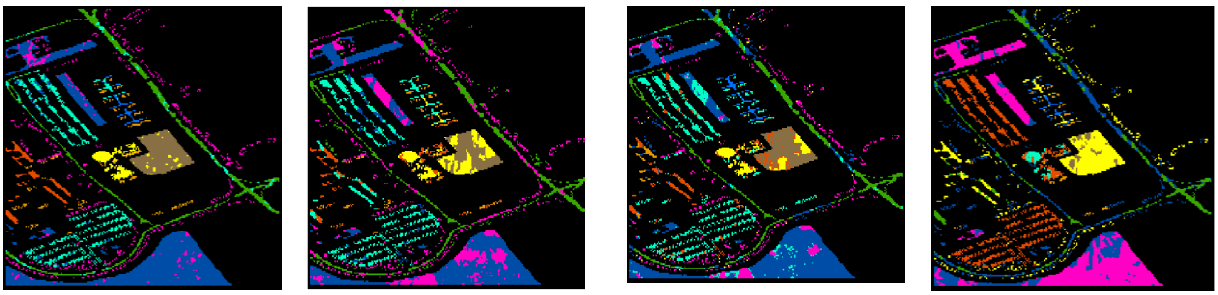
In Table 5 classification accuracy results have been summarised for University of Pavia. Apparently, polynomial kernel achieved the highest classification accuracy for Meadows (99.53%) while GRBF kernel achieved the highest classification accuracy for Asphalt (99.41%). Moreover, the polynomial kernel showed the highest classification accuracy for Gravel, Trees and Self-blocking bricks (Table 5 and Figs. 6a-l). GRBF and polynomial kernels achieved the highest classification accuracy for Painted metal sheets (100%). While ERBF kernel showed a 100% classification accuracy for Bare soil, the polynomial kernel achieved the highest accuracy for Bitumen and Shadows compared to other three kernels (Table 5). The comparison between DSVM and SVM accuracy in Table 6 shows that polynomial kernel yielded the highest mean classification accuracy for both Indian Pines and University of Pavia, followed by GRBF kernel while the neural kernel yielded the lowest mean classification accuracy.



(a) DSVM (ERBF) (b) DSVM (GRBF) (c) DSVM (Neural) (d) DSVM (Polynomial)



(e) SVM (ERBF) (f) SVM (GRBF) (g) SVM (Neural) (h) SVM (Polynomial)



(i) DNN (j) GMM (k) KNN (l) KM

Figure 6: Classified images for University of Pavia.

Table 4: Classification accuracy for Indian Pines for DSVM.

Class	Name	ERBF	GRBF	Neural	Polynomial
1	Corn-mintill	99.34	100	100	100
2	Corn-notill	100	99.21	95.10	100
3	Soybean-mintill	100	100	100	100
4	Woods	98.69	100	98.74	100
5	Grass-trees	98.73	100	98.95	99.43
6	Soybean-clean	100	99.35	99.02	98.89
7	Grass-pasture	95.42	99.26	97.93	100
8	Soybean-notill	100	100	99.22	98.72
9	Buiding-grass-trees-drives	100	100	100	100
10	Stone-steel-towers	95.01	95.26	98.11	95.75
11	Alfalfa	100	95.13	95.47	100
12	Wheat	100	100	100	99.63
13	Grass-pasture-mowed	100	100	97.91	100
14	Corn	100	95.35	100	100
15	Oats	99.14	98.78	98.77	95.45
16	Hay-windrowned	95.55	100	95.36	100

Table 5: Classification accuracy for University of Pavia for DSVM.

Class	Name	ERBF	GRBF	Neural	Polynomial
1	Meadows	99.43	99.53	95.04	99.55
2	Asphalt	99.24	99.41	99.28	99.36
3	Gravel	95.11	99.10	98.49	99.43
4	Trees	98.42	99.05	97.72	99.45
5	Self-blocking bricks	95.28	95.04	95.09	95.64
6	Painted metal sheets	98.89	100	98.58	100
7	Bare soil	100	97.04	95.98	97.66
8	Bitumen	98.03	98.00	97.22	98.92
9	Shadows	99.06	99.07	98.01	99.11

Table 6: Comparison of the classification accuracy between DSVM and SVM.

Type	ERBF	GRBF	Neural	Polynomial
Indian Pines	98.87 (74.87)	98.90 (78.90)	98.41 (70.10)	99.24 (80.24)
University of Pavia	98.16 (73.32)	98.47 (74.00)	97.27 (70.00)	98.79 (76.77)_

Table 7: Comparison of the classification accuracies of DSVM, SVM, DNN, GMM, KNN, and KM.

Type	DSVM	SVM	DNN	GMM	KNN	KM
Indian Pines	98.86	76.03	94.45	76.82	76.87	21.65
University of Pavia	98.17	73.52	93.79	78.35	79.80	18.18

A breakdown of the accuracies of the four kernels for DSVM and SVM are given in Table 6, while the mean accuracies for DSVM, SVM, DNN, GMM, KNN, and KM are highlighted in Table 7. Table 6 expressed the comparison of the mean classification accuracies between the DSVM and SVM for Indian Pines and University of Pavia. For both DSVM and SVM the polynomial kernel yielded the highest accuracy while the neural kernel yielded the lowest for both the Indian Pines and University of Pavia. From the results of the four kernels, it was obvious that the DSVM outperformed the SVM; this is an indication that the infusion of DL technique significantly enhanced the hybrid DSVM. Table 7 shows the comparison of the classification accuracies of the DSVM, SVM, DNN, GMM, KNN, and KM for Indian Pines and University of Pavia. The DSVM yielded the highest classification accuracy followed by the DNN while KM yielded the least accuracy. We noticed that the KM accuracy is considerably low. Our guess is that this is because the KM is a conventional unsupervised classifier and as with several unsupervised classifiers, they usually perform poorly when applied to hyperspectral data due to the cumbersome nature of the hyperspectral data. The conventional supervised classifiers GMM

and KNN performed slightly better than the SVM, while DSVM performed slightly better than the DNN. Based on the overall classification accuracies of these algorithms (Table 7), it can be discerned that the two DL techniques stood out.

4.0 Discussion and conclusion

Imbalanced, multi-class learning problems have recently been addressed by Yuan, et al., (2018) who used a regularized ensemble framework of DL methods. They showed that DL algorithms are capable of handling multi-class data sets because of the regularization parameter. Although several other sophisticated algorithms have been used to address similar problems, especially in image-based cancer detection and diagnosis (Hu et al., 2018), reduced computational cost and efficiency of ensemble-based approaches have been identified as additional key strength of ensemble DL based methods. Moreover, hybrid-based models have shown increasing potential applications in solving high-dimensional multivariate problems. For example, the coupling of a deep belief network (DBN) with a single class SVM was found useful since it addresses the complexity and scalability issues of the SVM, especially when training with large-scale datasets (Erfani et al., 2016). To leverage upon the unique potential of DL techniques, we integrated DL and SVM to formulate a hybrid DSVM model for the classification of hyperspectral data. Although the architecture of this model imitates the DNN, the optimized DSVM output was found by updating all the connecting SVM functions in the hidden layer. This study shows that the DSVM performed better than the DNN but significantly outperformed the SVM. For Indian Pines and University of Pavia images the DSVM's ERBF, GRBF, neural, and polynomial kernels yielded classification accuracies close to 100%. Insignificant misclassifications were recorded in the classification of Hay-windrowned, Grass-trees, Soybean-mintill, Oats, and Stone-steel-towers. Slight misclassifications were also recorded in the classification of Meadows, Gravel, Self-blocking bricks, and Bare soil for University of Pavia data. However, the misclassification problem was mitigated by the diligent selection of the best spectral bands possible out of the numerous spectral bands. The overall result of this experiment indicated that the DSVM appear to be highly robust and significantly better than the traditional SVM in the classification of hyperspectral remote sensing images.

With the growing applications of remote sensing observations in environmental monitoring, research efforts that focus on improving remote sensing classification algorithms are required to boost feature identification, optimisation, and interpretation of remote sensing images. This has been the prime motivation of the use of advance machine learning (e.g., Deep Learning), multivariate methods, and fourth order cumulant statistics in land cover classification and identification of complex hydrological signals from satellite geodetic systems on a variety of spatial and temporal scales (Ndehedehe & Ferreira, 2020; Kussul et al., 2017). As a proof of concept to the efficacy of such techniques, the recent work by Xu et al (2020) has demonstrated the effectiveness of ensemble learning in the field of object detection and computer vision. Their study noted that ensemble learning has been widely used to improve the performance of single detectors in recent years. Generally, the rapid advancement of these ensemble methods from traditional models such as Neural Networks, Random Forests amongst others to ensemble deep learning are strong indications of their underlying prospects in future applications in improving feature identification in HSI and multi-spectral images. Furthermore, the use of cumulant statistics in the decomposition of time-variable satellite gravity observations has proved to be robust in the identification of complex hydro-geodetic phenomena in regions where interactions between physical processes and climate fluctuations induce considerable large crustal displacements (Ndehedehe & Ferreira, 2020). Improved and advanced algorithms in the analysis of remote sensing observations, be it from optical or geodetic systems are therefore crucial to effective feature identification and interpretation of earth observation.

In the last few years, DL algorithms have emerged as powerful state-of-the-art technique for the analysis and classification of remotely sensed observations at different scales. The recent improved performance of the DBN in the classification of multi-temporal and HSI as opposed to other traditional methods (Li et al., 2019; Kussul et al., 2017) confirm the need to optimize the classification of hyperspectral images using DL techniques. Although Li et al., (2019) argued that the DBN performed better than traditional classification and other DL approaches, computational efficiency and accuracy are some challenges that have been identified with the use of DBN in HSI classification. However, our numerical results indicate that the DSVM outperformed the DNN. The high accuracy obtained with the DSVM validates its efficacy for hyperspectral remote sensing image classification. Notably, the GMM

and KNN, and the KM are two conventional supervised classifiers and unsupervised classifiers respectively that were experimented for the purpose of comparison. The results in this study (Table 7) show that the DSVM substantially outperformed the litany of other classification algorithms assessed in this study. The findings from our analysis also indicate that the GMM and KNN are slightly better than those of the SVM. In terms of robustness of all the algorithms used in the classification of hyperspectral remote sensing imageries, the overall results of this experiment indicate the DSVM is better as opposed to the DNN, SVM, and other renowned conventional classifiers.

The emergence of DL methods has been widely received because of their efficiency and success in several disciplines. For example, past studies (e.g., Langkvist et al., 2014) have argued that they have better representation and classification in several time-series problems as opposed to other methods. Recent review undertaken by Li et al., (2018) highlighted several issues with DL methods, including the tremendous success of DL in visual tracking. Among other observations, they noted that the use of the convolutional neural network model could significantly improve tracking performance.

Given the inefficiency of one-class support vector machines in modelling the variation in large, high-dimensional datasets, a hybrid model that combines DBN with a one-class SVM has significantly reduced training and testing time (Erfani et al., 2016).

In terms of land cover classification based on hyperspectral data, traditional methods appear to be limited. This among other factors have been attributed to the high number of spectral bands. The limited availability of training data or small-size sample training set (Chi et al., 2008) creates the 'curse of dimensionality' problem in hyperspectral observations. So, the current state of HSI classification has been the integration of non-parametric methods to address the complexities associated with hyperspectral data using conventional classification strategies. This has been illustrated, for example, by Moughal (2013) who integrated Minimum Noise Fraction and SVM technique to significantly reduce classification complexity and improved classification accuracy of HSI. In a related study seeking to address the multiclass problem of HSI, a new method for hyperspectral data classification employed a band clustering strategy through a multiple SVM system to improve the classification accuracy of HSI as opposed to the standard SVM (Bigdeli et al., 2013). However, only recently has deep learning based methods emerge as new options in optimising HSI classification. But Pan et al., (2018) argue that

massive parameters and the complex network structure may lead to poor performance when only few training samples are used. To address this, they proposed a small-scale data-based method, multi-grained network (MugNet). The performance of MugNet which was designed to make full use of the spectral and spatial correlations (Pan et al., 2018) has supported the notion that ensemble deep learning based method are excellent techniques in HSI classification. The robustness of our hybrid DSVM model for land cover classification is further demonstrated by analytical comparison with those of the SVM, DNN, GMM, KNN, and KM. As traditional SVMs and other nonparametric algorithms are limited in high-dimensional data, ensemble-based approaches that incorporates DL are novel methods to address the curse of dimensionality problem in HSI classification.

However, the application of DL methods in several disciplines is not without challenges. An important issue of this DL method in cancer detection was the size variation of target objects within the images. While the need to optimise the performance of deep learning-based cancer detection is an important direction for future research, a proposal to train the same CNN models using a broad range of image data on different scales and fused the outputs of multiple models to gain final result has been advocated in several studies summarised by Hu et al., (2018). Other improvements of DL in time series data analyses focus on improving redundant signals in multivariate input data. To this end, recommendations on developing algorithms for time-series modelling that learn even better features and are easier and faster to train are suggested as future research direction (Langkvist et al., 2014).

553

554

555

556

557

558

559

560

561

562

563

564

565

566

567

568

569

570

571

572

573

574

575

576

577

578

579

580

Acknowledgement

This research did not receive any specific grant from funding agencies in the public, commercial, or not-for-profit sectors.

Conflict of interest

The authors declare that there is no conflict of interest.

References

Ashtari, P., Haredasht, N. F., Beigy, Hamid. (2020). Supervised fuzzy partitioning. *Pattern Recognition*, 97, 1-15.

Bigdeli, B., Samadzadegan, F., & Reinartz, P. (2013). A multiple SVM system for classification of hyperspectral remote sensing data. *Journal of the Indian Society of Remote Sensing*, 41(4):763-776.

Chen, W., Pourghasemi, H. R., & Naghibi, S. A. (2018). A comparative study of landslide susceptibility maps produced using support vector machine with different kernel functions and entropy data mining models in China. *Bulletin of Engineering Geology and the Environment*, *77*(2), 647-664.

Chi, M., Feng, R., & Bruzzone, L (2008). Classification of hyperspectral remote-sensing data with primal SVM for small-sized training dataset problem. Advances in Space Research, 41, 1793-1799.

Cortes, C., & Vapnik, V. (1995). Support vector networks. *Machine Learning*, *20*, 273–297.

Erfani, S. M., Rajasegarar, S., Karunasekera, S., Leckie, C. (2016). High-dimensional and large-scale anomaly detection using a linear one-class SVM with deep learning, *Pattern Recognition*, 58, 121-134.

Gu, J., Wang, Z., Kuen, J., Ma, L., Shahroudy, A., Shuai, B., Liu, T., Wang, X., Wang, G., Cai, J., Chen, T., (2018). Recent advances in convolutional neural networks, *Pattern Recognition*, 77, 354-377.

Guo, Y., Han, S., Cao, H., Zhang, Y., & Wang, Q. (2018). Guided filter based Deep Recurrent Neural Networks for Hyperspectral Image Classification. *Procedia Computer Science*, *129*, 219-223.

Guyon, I., Weston, J., Barnhill, S., & Vapnik, V. (2002). Gene selection for cancer classification using support vector machines. *Machine learning*, *46*(1-3), 389-422.

Ghamisi, P., Plaza, J., Chen, Y., Li, J., and Plaza, A. (2007). Advanced supervised spectral classifiers for hyperspectral images: a review. *Journal of Latex Class Files*, 6(1), 1-23.

Haut, J. M., Paoletti, M. E., Plaza, J., Li, J., & Plaza, A. (2018). Active learning with convolutional neural networks for hyperspectral image classification using a new bayesian approach. *IEEE Transactions on Geoscience and Remote Sensing*, (99), 1-22.

He, J., & Fan, X. (2019). Evaluating the Performance of the K-fold Cross-Validation Approach for Model Selection in Growth Mixture Modeling. *Structural Equation Modelling: A Multidisciplinary Journal*, *26*(1), 66-79.

Hu, Z., Tang, J., Wang, Z., Zhang, K., Zhang, L., Sun, Q. (2018). Deep learning for image-based cancer detection and diagnosis − A survey. *Pattern Recognition*, Volume 83, 134-149.

Kang, S., Cho, S., & Kang, P. (2015). Constructing a multi-class classifier using one-against-one approach with different binary classifiers. *Neurocomputing*, *149*, 677-682.

Kussul, N., Lavreniuk, M., Skakun, S., & Shelestov A. (2017). Deep Learning Classification of Land Cover and Crop Types Using Remote Sensing Data. *IEEE Geoscience and Remote Sensing Letters, 4(5),778-782.*

Längkvist, M., Karlsson, L., Loutfi, A. (2014). A review of unsupervised feature learning and deep learning for time-series modeling, *Pattern Recognition Letters*, 42, 11-24.

Li, P., Wang, D., Wang, L., Lu, H., (2018). Deep visual tracking: Review and experimental comparison, *Pattern Recognition*, 76, 323-338.

Li, L., Wang, C., Li, W., & Chen, J. (2018). Hyperspectral image classification by AdaBoost weighted composite kernel extreme learning machines. *Neurocomputing*, *275*, 1725-1733.

Li, C., Wang, Y., Zhang, X., Gao, H., Ynag, Y., and Wang, J. (2019). Deep Belief Network for spectral-spatial classification of hyperspectral remote sensor data. *Sensors*, 19(204), 2-13.

Melgani, F., & Bruzzone, L. (2004). Classification of hyperspectral remote sensing images with support vector machines. *IEEE Transactions on geoscience and remote sensing*, *42*(8), 1778-1790.

Moughal, T A (2013). Hyperspectral image classification using Support Vector Machine. Journal of Physics: Conference Series, 439, 012042.

Ndehedehe, C E and Ferreira (2020). Assessing land water storage dynamics over South America. *Journal of Hydrology, 580,* 124339.

Okwuashi, O. H. S. (2011). The Application of Geographic Information Systems Cellular Automata Based Models to Land Use Change Modelling of Lagos, Nigeria (Unpublished doctoral dissertation). Victoria University of Wellington, Wellington, New Zealand.

Okwuashi, O. & Ndehedehe, C. (2017). Tide modelling using support vector machine regression. *Journal of Spatial Science, 62*(1), 29–46.

O'Sullivan, D. (2004). Complexity science and human geography. *Transactions of the Institute of British Geographers, 29*(3), 282-295.

Pal, M., & Mather, P. M. (2005). Support vector machines for classification in remote sensing. *International Journal of Remote Sensing*, *26*(5), 1007-1011.

Pan, B., Shi, Z., & Xu, X. (2018). MugNet: deep learning for hyperspectral image classification using limited samples. *ISPRS Journal of Photogrammetry and Remote Sensing*, *145*, 108-119.

Paoletti, M. E., Haut, J. M., Plaza, J., & Plaza, A. (2018). A new deep convolutional neural network for fast hyperspectral image classification. *ISPRS journal of photogrammetry and remote sensing*, *145*, 120-147.

Pirra, M., & Diana, M. (2019). A study of tour-based mode choice based on a Support Vector Machine classifier. *Transportation Planning and Technology*, *42*(1), 23-36.

Poursaee, A. (2018). Application of agent-based paradigm to model corrosion of steel in concrete environment. *Corrosion Engineering, Science and Technology*, *53*(4), 259-264.

Ratle, F., Camps-Valls, G., & Weston, J. (2010). Semisupervised neural networks for efficient hyperspectral image classification. *IEEE Transactions on Geoscience and Remote Sensing*, *48*(5), 2271-2282.

Xu, J., Wang, W., Wang, H., & Guo ,J (2020). Multi-model ensemble with rich spatial information for object detection. *Pattern Recognition, 99,* 107098.

Yuan, X., Xie, L., Abouelenien, M. (2018). A regularized ensemble framework of deep learning for cancer detection from multi-class, imbalanced training data. *Pattern Recognition*, 77, Pages 160-172,

Zhang, M., Li, W., & Du, Q. (2018). Diverse Region-Based CNN for Hyperspectral Image Classification. *IEEE Transactions on Image Processing*, *27*(6), 2623-2634.

Zhao, W., Li, S., Li, A., Zhang, B., & Li, Y. (2019). Hyperspectral images classification with convolutional neural network and textural feature using limited training samples. *Remote Sensing Letters*, *10*(5), 449-458.

Zhao, L., & Peng, Z. (2012). LandSys: an agent-based cellular automata model of land use change developed for transportation analysis. *Journal of Transport Geography*, *25*, 35-49.