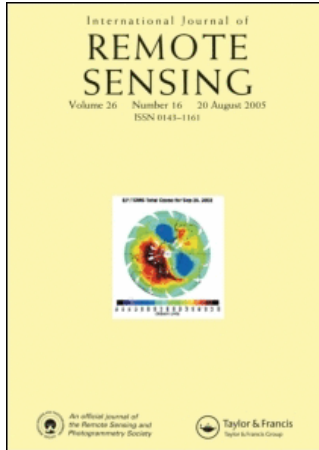


This article was downloaded by:[University of South Florida]
On: 21 February 2008
Access Details: [subscription number 768615674]
Publisher: Taylor & Francis
Informa Ltd Registered in England and Wales Registered Number: 1072954
Registered office: Mortimer House, 37-41 Mortimer Street, London W1T 3JH, UK



International Journal of Remote Sensing

Publication details, including instructions for authors and subscription information:
<http://www.informaworld.com/smpp/title~content=t713722504>

Multispectral landuse classification using neural networks and support vector machines: one or the other, or both?

B. Dixon^a; N. Candade^a

^a University of South Florida, FL 33701

Online Publication Date: 01 February 2008

To cite this Article: Dixon, B. and Candade, N. (2008) 'Multispectral landuse classification using neural networks and support vector machines: one or the other, or both?', International Journal of Remote Sensing, 29:4, 1185 - 1206

To link to this article: DOI: 10.1080/01431160701294661

URL: <http://dx.doi.org/10.1080/01431160701294661>

PLEASE SCROLL DOWN FOR ARTICLE

Full terms and conditions of use: <http://www.informaworld.com/terms-and-conditions-of-access.pdf>

This article maybe used for research, teaching and private study purposes. Any substantial or systematic reproduction, re-distribution, re-selling, loan or sub-licensing, systematic supply or distribution in any form to anyone is expressly forbidden.

The publisher does not give any warranty express or implied or make any representation that the contents will be complete or accurate or up to date. The accuracy of any instructions, formulae and drug doses should be independently verified with primary sources. The publisher shall not be liable for any loss, actions, claims, proceedings, demand or costs or damages whatsoever or howsoever caused arising directly or indirectly in connection with or arising out of the use of this material.

Multispectral landuse classification using neural networks and support vector machines: one or the other, or both?

B. DIXON* and N. CANDADE

University of South Florida, St. Petersburg, 140 Seventh Ave South, St. Petersburg, FL 33701, USA

(Received 10 May 2005; in final form 10 February 2007)

Land use classification is an important part of many remote sensing applications. A lot of research has gone into the application of statistical and neural network classifiers to remote-sensing images. This research involves the study and implementation of a new pattern recognition technique introduced within the framework of statistical learning theory called Support Vector Machines (SVMs), and its application to remote-sensing image classification. Standard classifiers such as Artificial Neural Network (ANN) need a number of training samples that exponentially increase with the dimension of the input feature space. With a limited number of training samples, the classification rate thus decreases as the dimensionality increases. SVMs are independent of the dimensionality of feature space as the main idea behind this classification technique is to separate the classes with a surface that maximizes the margin between them, using boundary pixels to create the decision surface. Results from SVMs are compared with traditional Maximum Likelihood Classification (MLC) and an ANN classifier. The findings suggest that the ANN and SVM classifiers perform better than the traditional MLC. The SVM and the ANN show comparable results. However, accuracy is dependent on factors such as the number of hidden nodes (in the case of ANN) and kernel parameters (in the case of SVM). The training time taken by the SVM is several magnitudes less.

1. Introduction

Mapping land cover using remotely sensed images most commonly involves use of the reflectance and radiances of each pixel to assign it to a number of land cover classes (Huang *et al.* 2001). The assumption is made that the information content of a pixel originates solely from within its footprint. Spectral reflectance characteristics of various land covers shows that cover type identification should be possible if the sensor gathers data at several wavelengths (Richards 1993). For each pixel the set of samples are analysed to provide a label that associates the pixel with a particular land cover.

Numerous classification methodologies have been applied to digital image processing and various degrees of success are noted. The majority of remote sensing approaches are based on classical pattern recognition techniques, mostly Maximum Likelihood Classifiers (MLCs), k-nearest neighbour or a combination of maximum likelihood and clustering (Jensen 2000). The MLC requires representative pixel

*Corresponding author. Email: bdixon@stpt.usf.edu

samples to have normal distribution (Mather 2001). This is clearly not true for remotely sensed images.

One of the favoured alternatives to the statistical classifiers is the Artificial Neural Network (ANN) (Benediktsson and Sveinsson 1997, Kanellopoulos *et al.* 1992, Baraldi and Parmiggiani 1995, Hara *et al.* 1995, Paola and Schowengerdt 1995, Austin 1997, Clastres *et al.* 1997, Carpenter *et al.* 1999a., 1999b). A lot of research has been carried out on the use of back propagation ANN and a number of guidelines have been presented by research regarding fixing ANN parameters for land use classification (Riedmiller and Braun 1993, Kanellopoulos and Wilkinson 1997, Paola and Schowengerdt 1995).

ANN, with its non-parametric approach, helps avoid some of the problems of MLC (Huang *et al.* 2002). This classification technique is capable of handling multitudes of data and is free of distributional assumptions common with statistical analysis. It is however noted that improvement of accuracy by using ANN is generally marginal—accuracy rarely increases beyond 80% (Atkinson and Tate 2000). Although ANN is computationally efficient, training is time consuming. The amount of training data required for successful classification increases exponentially with increased dimensionality of the input data.

Recent application of Support Vector Machines (SVM) shows that they can be successfully applied to the problems of image classification with large input dimensionality (Roli and Fumera 2000). SVMs are also known to generalize better. In comparison to ANNs, SVMs offer a solid mathematical foundation that provides a probabilistic guarantee on how well the classifier will generalize on unseen data (Perkins *et al.* 2001). While ANNs are based on the idea of minimizing the error on training data (empirical risk), SVMs operate on another induction principle, called Structural Risk Minimization (SRM), which minimizes an upper bound on the generalization error or the error on unseen data (Shankar and Deshwal 2002). The complexity of the resulting classifier is characterized by the number of support vectors rather than the dimensionality of the transformed space. As a result, SVMs tend to be less prone to problems of over fitting than other methods (Duda *et al.* 2002).

SVMs have been successfully used in a number of applications ranging from text characterization, face identification to remote sensing image classification. Xu *et al.* (2003) used SVM for the segmentation of aerial grey images in combination with pyramid image and decision tree. Water boundaries were modified using the snake method (Xu *et al.* 2003). Inglada and Giros (2004) present an image processing chain for the detection of man-made objects in high-resolution remote sensing images. The paper discusses SVM based classification techniques using image geometry rather than spectral signatures. Huang *et al.* (2003) have used remote sensing techniques to investigate soil erosion based on expert knowledge and SVM classification. Melgani and Bruzzone (2004) have successfully applied SVMs for the classification of hyperspectral imagery. This paper shows that SVMs are a valid alternative to conventional pattern recognition approaches (feature-reduction procedures combined with classification methods) for the classification of hyperspectral remote sensing data.

The objective of this research was to compare two classification algorithms (*viz.* ANN and SVM) against traditional statistical classifiers like the MLC and assess their accuracy in remote sensing image classification. There is no shortage of availability of data from remotely sensed sources; however, extracting meaningful

information from these data in an efficient way (cost effective and timely fashion) with reasonable accuracy remains a challenge. Therefore, there is a need to explore the suitability and sensitivity of these classifiers (ANN, SVM and MLC) in extracting information from the remotely sensed data. It is hoped that this research will shed light on the suitability of the algorithm(s) (classifier/model) selection for a given application. This type of research and resultant information is critical for utilization of remotely sensed data to the fullest extent. Please refer to Appendix A for a description and comparison of these classifiers as well as general background of algorithms.

2. Methods

2.1 Study area and source data

This work focused on the land use classification of a subset image of South West Florida, from Landsat TM 5 data. The study area is shown in figure 1 and the corresponding satellite image in figure 2. The source image was in UTM Zone 17N projection, Spheroid WGS 84, Datum WGS 84, 30 m resolution and in Geo TIFF format. However, it had to be rectified in order to assign the right map coordinates. Rectification was performed using a high resolution Digital Ortho Quarter Quad (DOQQ). Subsetting was performed post-rectification. Figure 3 shows the subset region of Joshua Creek Watershed, Desoto County, Florida. The Landsat TM image consisted of seven bands, out of which bands 1–5 and band 7 were used for analysis. The thermal IR band on TM (band 6) is designed to assist in thermal mapping and is used generally for soil moisture and vegetation studies. Therefore, this band was not included in this study.



Figure 1. Study area—South West Florida.



Figure 2. Landsat TM image of study area.

2.2 Signature evaluation and training data creation

This study attempted to classify seven broad categories of land use/land cover (LULC) according to the Florida Land Use and Cover Classification System (FLUCCS) (Florida Topographic Bureau, 1985). These categories were: citrus (C), pasture (P), sod (S), timber (T), urban (U), water (WR) and wetland (WT).

Once training sites were identified, these pixels were converted to ASCII on a class-by-class basis (to maintain class identity) to bring it into the data format of SVM and ANN algorithms. The ASCII output file consisted of the Digital Numbers (DN) in the six bands along with the UTM *X* and *Y* coordinates. Only the DNs were extracted to create the training dataset, since the training process is based on merely the spectral signatures and independent of training site location. Thus, the input feature space consisted of six-dimensional vectors, where each vector represented the DNs of the pixels in the six bands. Initial data statistics and analysis was necessary to ensure class separability and validity of training sites for classification purposes.

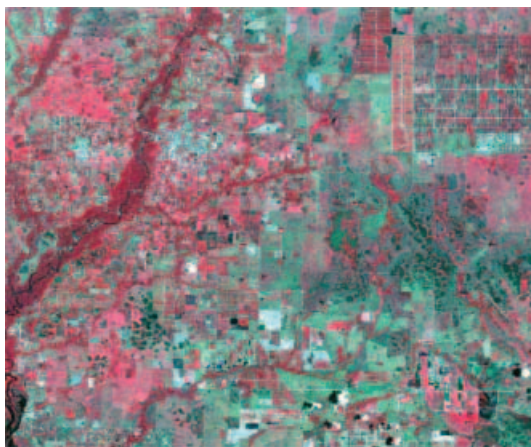


Figure 3. Subset image—Joshua Creek Watershed, Desoto County, Florida.

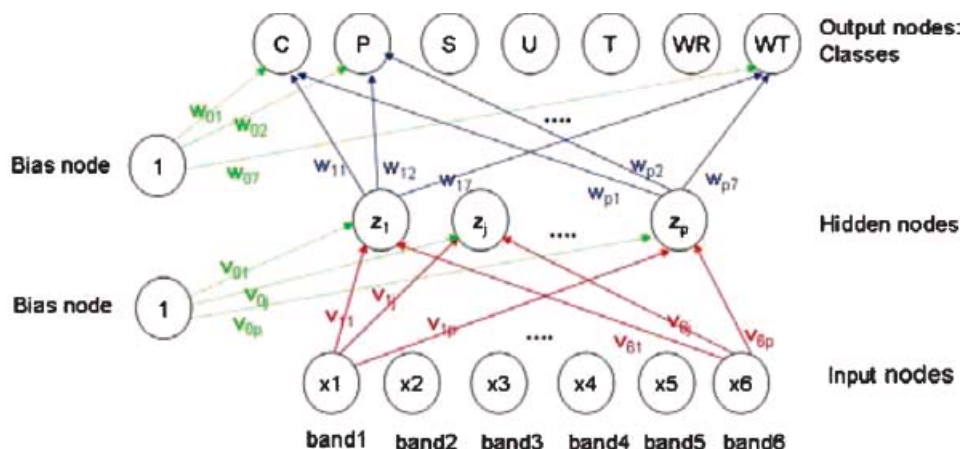


Figure 4. Neural network architecture used in this study.

Once the training data was created and evaluated, it was given as input to the MLC, ANN and SVM. The classes were assigned equal prior probabilities.

2.3 Artificial neural network (ANN)

The schematic of the ANN used is shown in figure 4. The SBP architecture provided by PREDICT software was used to perform classification. A summary of the ANN architectures constructed is given in table 1. All networks consisted of a single layer of 8, 15, 30, 40 and 50 hidden nodes respectively. Other parameters of the ANNs were kept the same for all the networks. A summary of these parameters is given in table 2. The ANN weights were initialized using a uniform distribution. Learning rate was set to 100 for the hidden layer and 0.01 for the output layer. Multiple regression was used as the variable selection method. The hidden nodes were trained using the tan hyperbolic (tanh) transfer function while softmax was used for the

Table 1. Architecture of ANNs used in this study.

Architecture			
No. of hidden units	Max. layer size	Min. increment	Max. increment
8	20	1	1
15	30	5	5
30	50	15	15
40	100	30	45
50	200	50	50

Table 2. Parameters of the ANNs.

Learning Rate	Hidden layer=100 Output layer=0.01
Variable selection model	Multiple regression
Transfer functions	Hidden=tanh Output=Softmax

output nodes. Stopping criteria was set to 0.001. This is the value on the objective function which must be exceeded for an improvement to be counted.

2.4 Support vector machine (SVM)

The SVM was implemented using LIBSVM Version 2.6. Three SVM kernels were used—the Polynomial, radial basis function (RBF) and linear kernels. Kernel parameters were varied to observe their effect on accuracy. The parameters that were varied were the error penalty (C), RBF kernel radius and degree of polynomial kernel. C is the constant of constraint violation which observes the occurrence of a data sample on the wrong side of the decision boundary. RBF kernel radius denotes the width of the RBF function.

2.5 Accuracy assessment

Classification was performed using MLC, ANN (using 8, 15, 30, 40 and 50 hidden nodes in a single layer) and SVM (using linear, polynomial and RBF kernels). Once classification was complete, accuracy assessment of all the output maps was performed with respect to ground data points. Stratified random sampling was used to generate these ground truth points which were assigned reference values identified from a high resolution Digital Ortho Quarter Quad (DOQQ) in conjunction with the South West Florida Water Management District (SWFWMD) land use map. Finally, the classifiers that showed highest accuracies in each category (ANN and SVM) were used for comparison purposes.

Contingency matrices help evaluate the classifiers based on their performance on a class-by-class basis. For example, they show the number of citrus pixels misclassified as pasture, sod etc., and also give the count of other classes misclassified as citrus. This leads us to find the user's and producer's accuracy. User's accuracy calculates correctly classed from the trace variable over the row total and provides an indication of errors of case omission. Producer's accuracy is the calculation of correctly classed from the trace value over the column total (Congalton 1991). Producer's accuracy gives an indication of the accuracy of what the model was able to itself predict, whereas user's accuracy relates to how well the training data was discerned.

3. Results and discussion

3.1 Training data analysis

The training dataset used in this study consisted of 4965 samples (pixels). This is about 1% of the pixels in the entire sub-scene. Tables 3–5 give the statistics of

Table 3. Distribution of classes in training dataset.

Class (abbreviation)	# of cases
1. citrus (C)	1284 cases
2. pasture (P)	176 cases
3. sod (S)	118 cases
4. timber (T)	1016 cases
5. urban (U)	65 cases
6. water (WR)	942 cases
7. wetland (WT)	1364 cases

Table 4. Training data characteristics.

Variable	Mean	SD	Minimum	Maximum
band1 (B1)	82.6	9	68	166
band2 (B2)	32	6.9	20	85
band3 (B3)	32.3	11.5	16	128
band4 (B4)	72.4	32.7	7	136
band5 (B5)	75.9	40.3	3	249
band7 (B7)	27.6	19.1	1	159

Table 5. Testing data characteristics.

Variable	Mean	SD	Minimum	Maximum
band1	89.7	10.1	65.0	224.8
band2	37.5	6.1	20.0	143.4
band3	40.6	10.5	12.0	174.9
band4	86.5	13.6	6.0	184.2
band5	112.0	28.2	2.0	249.1
band6	44.6	16.7	0.0	190.8

training and testing data. Table 3 shows the distribution of classes in the training dataset. The number of training samples were in proportion to the abundance of the classes in the study area. The Joshua Creek basin being a predominantly agricultural landscape had a very small percentage of urban land use. Therefore, relatively fewer training pixels were selected for urban land cover.

Table 4 shows the overall training data characteristics for all classes. These values give an indication of the range of DNs in each band. Band 5 shows maximum standard deviation while band 2 shows the least. The mean values of band 2 and band 3, band 4 and band 5 are very close. However, the signature mean plot (figure 5) would give a better picture of class separability based on mean values. It is seen that the mean values for the seven classes are distinctly different in band 5. Table 5 shows the testing data characteristics.

The most effective means by which multispectral training data can be visualized is to plot them in pattern space, or multispectral vector space, with as many dimensions as there are spectral components. However, a two-dimensional representation is the easiest to achieve. Figure 6 shows a depiction of the training classes (also known as information classes) in two-dimensional space. The training classes appear distinct and suggest that they would be suitable for classification. This being a two-dimensional plot is only a visual cue and not a complete representation of class separability.

3.2 Classification and accuracy assessment

Once trained, the MLC, ANN and SVM were used to classify the entire subset image of the study area. They were then converted to pixels in ERDAS. Accuracy assessment with known points was performed in order to evaluate these maps. 245 ground truth points identified from a DOQQ were used as reference. The contingency matrix and kappa statistic were used as evaluation measures (figures 7a and b).

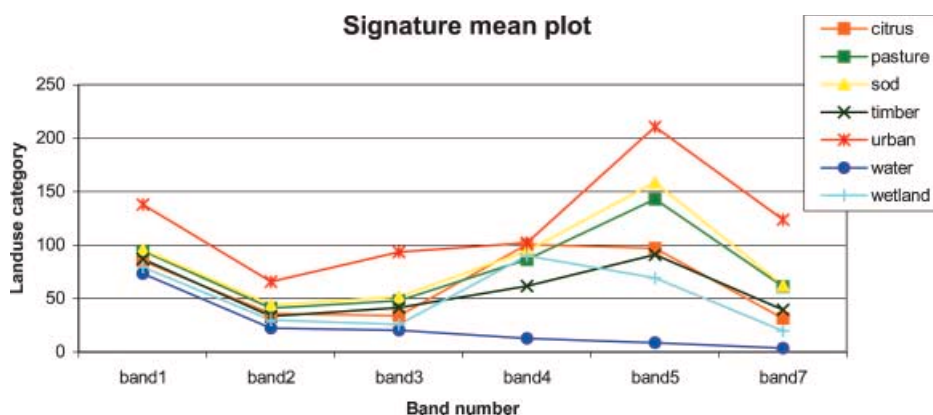


Figure 5. Signature mean plot.

The MLC was performed in ERDAS Imagine. The output map is shown in figure 8. The resulting accuracy was 50.6% with a kappa of 0.4294. This map was used as a reference map for comparison with ANN and SVM classifiers. For the MLC, parameter optimization is not a crucial issue. Thus only training and testing data is necessary. However, for the ANN and SVM, parameters need to be optimized and require preliminary analysis using validation data. The objective of preliminary evaluation was to optimize these parameters.

3.3 ANN classification

For a preliminary evaluation of ANN architecture, the given training data that consisted of 4965 samples was divided into training and testing datasets. The training dataset now consisted of 350 samples, 50 pixels/class. The remaining 4615 pixels were used for validation. Five ANNs were constructed with architectures

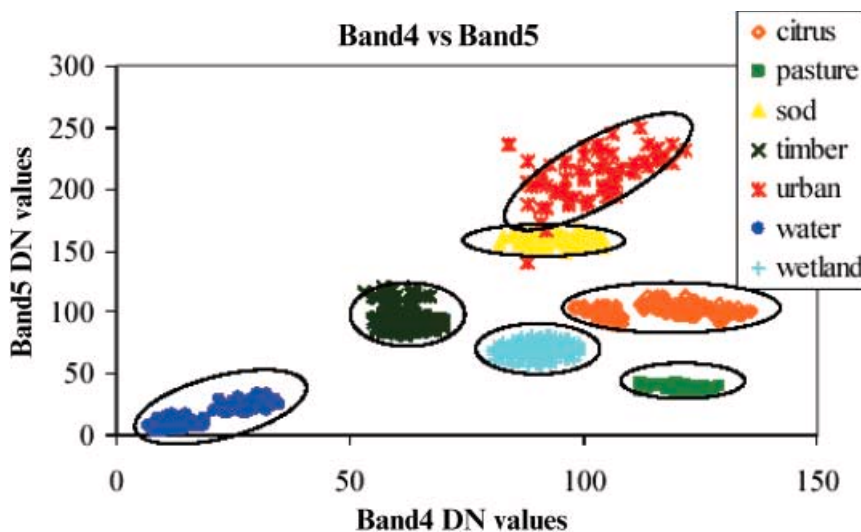


Figure 6. Depiction of training classes in two-dimensional space (band 4 versus band 5).

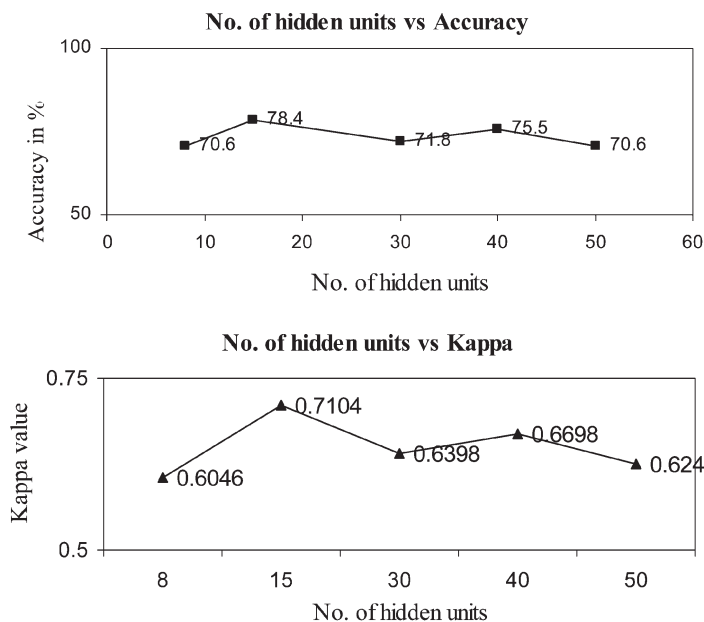


Figure 7a. Change in accuracy and kappa with ANN architecture (hidden units).

and parameters given in tables 1 and 2 respectively. Preliminary results using 350 training and 4615 testing samples are shown in table 6. The accuracy and relative entropy for the training and testing dataset are presented. The relative entropy measure ensures that the outputs of the ANN enforce the mutual dependence of the outputs. It maximizes the probability of successful classification. Ideally a very low value of relative entropy indicates a good fit of the model to the data. The accuracy gives the fraction of records whose prediction is within a specified tolerance of the desired output. By default, the accuracy tolerance is set to 20% of the range of the output.

It is seen that the training accuracy of all the classifiers is 100%. However, the accuracy on the validation data changes with change in ANN architecture. The highest accuracy is about 99% for an ANN with 15 nodes in a single layer. The network with the maximum number of hidden nodes (50) shows the least test accuracy of 91.8%, indicating that too many hidden nodes lead to a decrease in ANN generalization performance (table 6).

Once preliminary evaluation was complete, the networks were trained with all the 4965 samples and now tested on the entire image of the study area (test image). The results on test data using these five architectures are shown in table 7. These observations are depicted graphically in figure 9a. It is observed that final accuracy is sensitive to the number of hidden nodes. The network with an optimum number of 15 hidden nodes showed the highest accuracy of 78.4% and kappa of 0.7104. This network also outperformed other architectures in our preliminary analysis. As stated earlier, too few hidden nodes lead to underfitting and too many lead to overfitting and hence poor generalization. The ANN with 15 hidden nodes showed the highest accuracy of 78.4% and was used for comparison with ML and SVM classifiers. The output map for this classifier is shown in figure 8b.

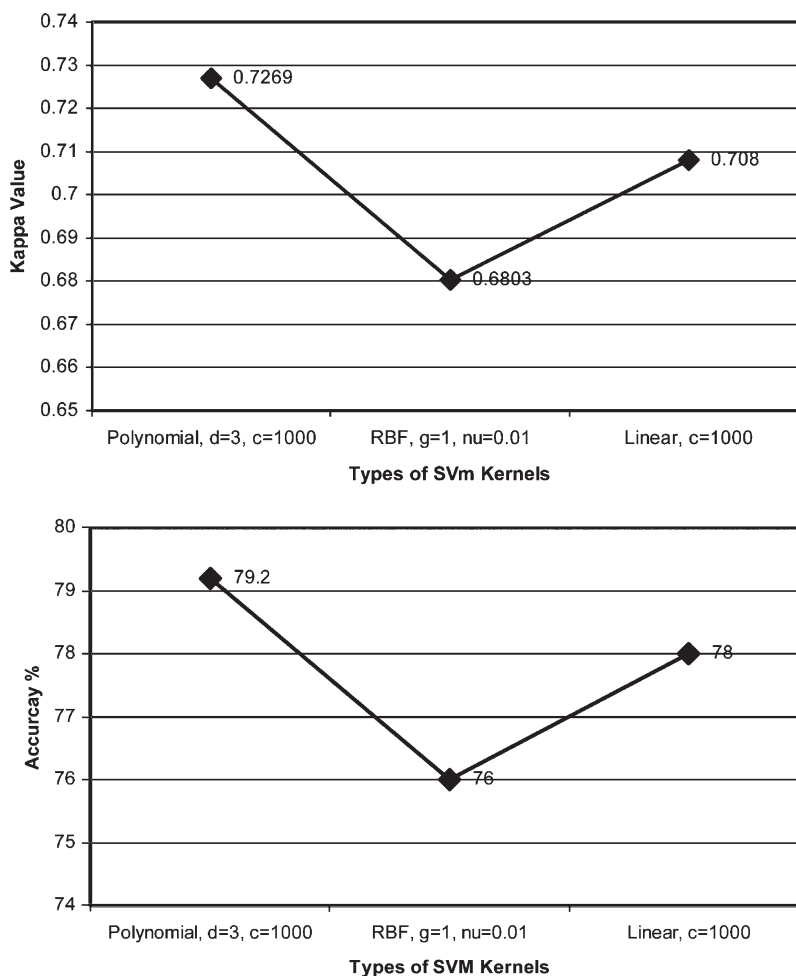


Figure 7b. Change in accuracy and kappa with SVM models (Kernels).

3.4 SVM classification

A similar preliminary evaluation of SVM kernel performance was performed by dividing the given dataset that consisted of 4,965 samples into training and testing datasets. The training dataset consisted of 350 samples, 50 pixels/class. The remaining 4,615 pixels were used for testing.

Three kernels of the SVM were used and the results are as summarized in table 8. The Polynomial kernel of degree 3 ($d=3$) and cost ($C=1000$) showed the highest accuracy on test data. C refers to the cost or error penalty. A high value of error penalty will force the SVM training to avoid classification errors. A large value of C will result in a larger search space for the QP optimizer. However, some experiments fail to converge for $C > 1000$. In the kernels under study, a value of $C=1000$ was optimum. For the RBF kernel, gamma (radius) was set to 1. This gives the area of influence the particular support vector has over the data space. The RBF kernel was experimented with different values of nu. nu-SVC is the same as C-SVC except that the range of nu is always between $[0, 1]$ while C is from zero to infinity. nu is related to the ratio of support vectors and the ratio of the training error.

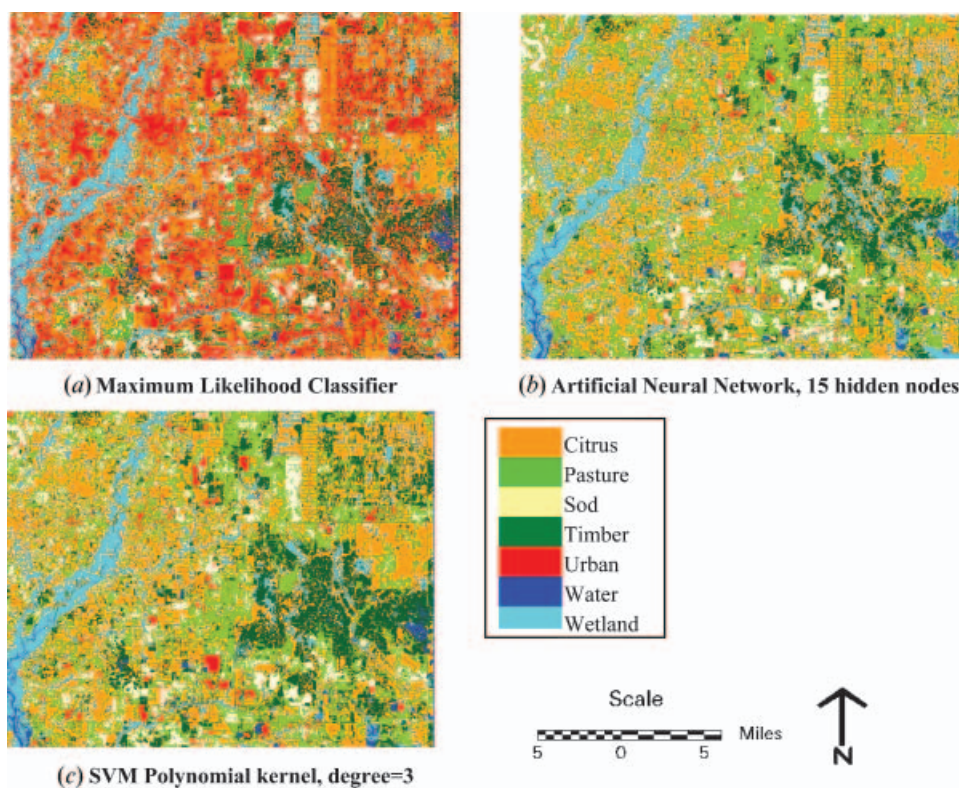


Figure 8. Classified maps.

Table 6. Effect of ANN architecture on accuracy: preliminary analysis.

ANN architecture	Training data (350 cases)		Testing data (4615 cases)	
	Accuracy (%)	Rel. Entropy	Accuracy (%)	Rel. Entropy
No. of hidden nodes				
8	100	0.0001	97.9	0.0103
15	100	0.0002	99.1	0.0127
30	100	0.0001	96.7	0.0195
40	100	0.0021	97.7	0.0195
50	100	0.0019	91.8	0.0234

Table 7. Result of ANN on test data.

no. of hidden units	8	15	30	40	50
Accuracy	70.6	78.4	71.8	75.5	70.6
kappa	0.6046	0.7104	0.6398	0.6698	0.624

Table 8 shows that the polynomial kernel of degree 3 had the highest accuracy of 98.48% while RBF ($\nu=0.01$) and linear kernels had an accuracy of 95.77% and 95.79%, respectively. These models that showed high accuracies were used for further study. The cost for the polynomial and linear kernels was set to 1000 (Figure 7b).

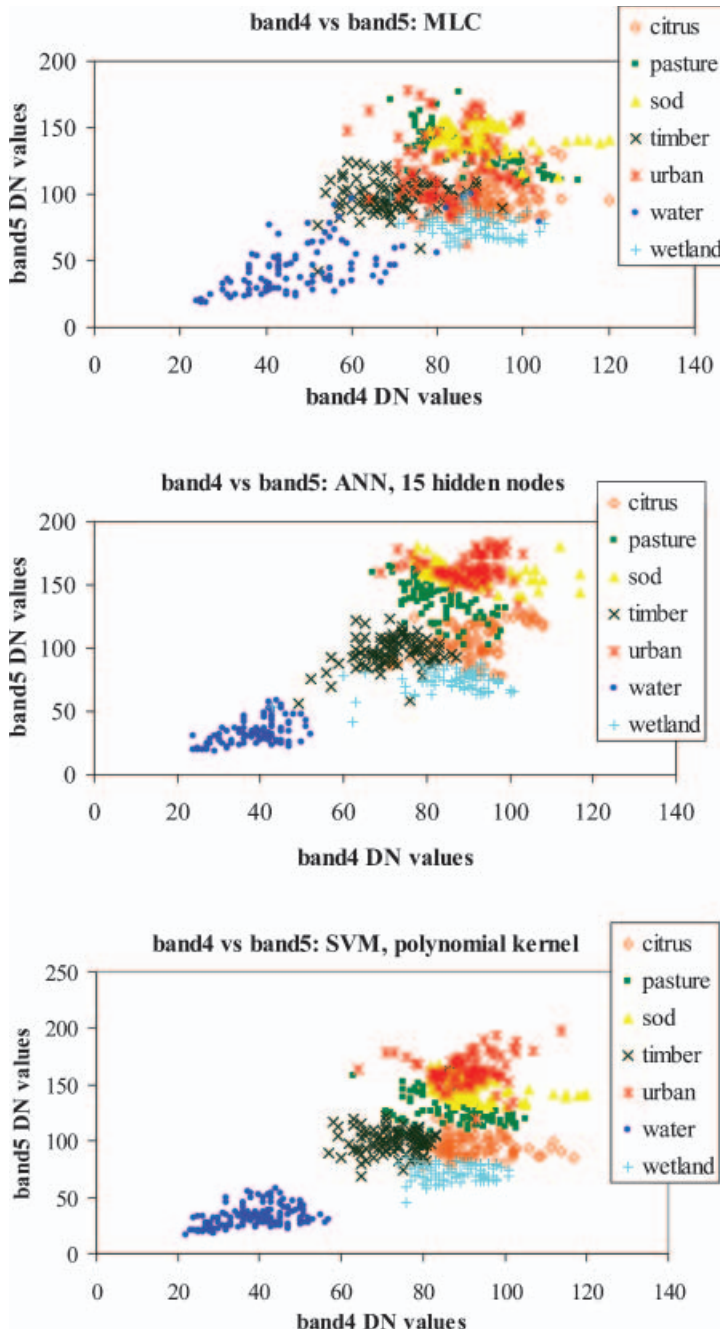


Figure 9. Scatter plots of Band 4 versus Band 5 DN values of predicted (classifier) output.

Once preliminary evaluation was complete, the SVMs were trained with all the 4,965 samples and now tested on the entire image of the study area (test image). Results on test data using these three kernels are shown in table 9. The polynomial kernel of degree 3 showed the highest accuracy of 79.2% and Kappa 0.7269. The RBF and linear kernels showed accuracies of 76% and 78% respectively. It is thus

Table 8. Choice of SVM kernel: preliminary analysis.

Kernel	Train accuracy (%):		Test accuracy (%):	
	350 cases		4615 cases	
Polynomial, degree=1				
$c=10$	99.7		93.5	
$c=1000$	100		95.8	
Polynomial, degree=3				
$c=10$	100		97.5	
$c=1000$	100		98.4	
RBF, gamma=1				
$\nu=0.01$	100		95.7	
$\nu=0.1$	100		92.4	
$\nu=0.5$	99.4		91.9	
Linear				
$c=10$	99.7		93.5	
$C=1000$	100		95.7	

Table 9. Performance of SVM on test data.

Kernel	accuracy	kappa
Polynomial, $d=3, c=1000$	79	0.7269
RBF, $g=1, \nu=0.01$	76	0.6803
Linear, $c=1000$	78	0.7080

observed that selection of the SVM kernel is a crucial step in image classification. The degree 3 polynomial kernel SVM classifier was used for comparison with the MLC and ANN. The output map of this classifier is shown in figure 8c.

3.5 Comparison of classifiers

The following classifiers were used for final comparison: the MLC, the ANN with 15 hidden nodes and the SVM using polynomial kernel as the mapping function. The output maps for these classifiers are shown in figures 8(a-c). The area coverage for these maps is given in table 10. The following section describes the comparison of the above classifiers in terms of percentage area cover, spatial distribution and class separability.

The spatial distribution of urban landuse as classified by the MLC is shown in figure 8a. The MLC classified about 36.2% of the area as ‘urban’. The ANN and SVM classified a significantly lesser area as urban, with area coverage of about 1.7% and 2.7% respectively in similar spatial locations. The maps obtained from the ANN and SVM classifiers look very similar in terms of area coverage and location for

Table 10. Area coverage in % cover for all models.

	Unclassified	Citrus (C)	Pasture (P)	Sod (S)	Timber (T)	Urban (U)	Water (WR)	Wetland (WT)
ML	0.2	23.1	10.5	8.2	7.8	36.2	1	13.1
ANN	0	32.6	34.5	5.9	10.1	1.7	1.2	14
SVM	0	30.2	30.7	8.6	14.9	2.7	1.3	11.5

most of the landuse categories except 'timber'. Timber comprises about 10% of the land cover according to the ANN classification, whereas it covers about 15% with the SVM. The SVM showed timber in the south-east region of the study area, whereas ANN classified only a part of this area as timber and the rest as wetland. The spatial distribution and location of most other classes are very similar (figures 8*b* and 8*c*), though marginal differences in percentage area coverage can be seen (table 10).

The accuracy of these maps was evaluated using 245 ground truth points identified from a high resolution DOQQ. The resultant contingency matrices for each of the classifiers are as shown in tables 11–13. The overall accuracy and kappa statistics for the classifiers is shown in table 14. The MLC shows about 50.6% accuracy versus 78.4% for ANN with 15 hidden nodes and 79.2% for SVM using the polynomial kernel. The ANN and SVM show significantly higher accuracy than the conventional MLC.

As with training data, scatter plots help us visualize how well each classification algorithm has separated the classes. However, these plots are only used as examples and are not to be interpreted as a complete evaluation of accuracy. About 100 pixels from each class were plotted, with their corresponding DN values. For example, figure 9 shows band 4 on the *x*-axis versus band 5 on the *y*-axis. Similar graphs were

Table 11. Contingency Matrix ML Classifier.

Count	C	P	S	T	U	WR	WT	Total	User's accuracy (%)
C	47	0	0	2	25	0	5	79	59.5
P	3	19	7	1	56	0	1	87	21.8
S	0	1	12	0	8	0	0	21	57.1
T	1	0	0	16	7	0	1	25	64.0
U	0	0	0	0	8	0	0	8	100.0
WR	0	0	0	0	1	5	0	6	83.3
WT	1	0	0	1	0	0	17	19	89.5
Total	52	20	19	20	105	5	24	245	
Producer's accuracy (%)	90.4	95.0	63.2	80.0	7.6	100.0	70.8		

Overall classification accuracy=50.6%; overall kappa=0.4294.

Table 12. Contingency matrix ANN classifier, 15 hidden nodes.

Count	C	P	S	T	U	WR	WT	Total	User's accuracy (%)
C	66	9	0	2	1	0	1	79	83.5
P	7	67	7	5	1	0	0	87	77.0
S	0	9	9	0	3	0	0	21	42.9
T	2	1	0	20	0	0	2	25	80.0
U	0	2	0	0	6	0	0	8	0.8
WR	0	0	0	0	0	6	0	6	100.0
WT	1	0	0	0	0	0	18	19	94.7
Total	76	88	16	27	11	6	21	245	
Producer's accuracy (%)	86.8	76.1	56.3	74.1	54.5	100.0	85.7		

Overall classification accuracy=78.4%; overall kappa=0.7104.

Table 13. Contingency matrix SVM—polynomial kernel, degree 3.

Count	C	P	S	T	U	WR	WT	Total	User's accuracy (%)
C	65	8	0	5	0	0	1	79	82.3
P	5	64	8	5	4	0	1	87	73.6
S	0	4	12	0	5	0	0	21	57.1
T	1	1	0	23	0	0	0	25	92.0
U	0	1	0	0	7	0	0	8	87.5
WR	0	0	0	0	0	6	0	6	100.0
WT	1	0	0	1	0	0	17	19	89.5
Total	72	78	20	34	16	6	19	245	
Producer's accuracy (%)	90.3	82.1	60.0	67.6	43.8	100.0	89.5		

Overall classification accuracy=79.2%; overall kappa=0.7269.

Table 14. Overall accuracy and kappa for all models.

Algorithm	Accuracy (%)	Kappa
Maximum likelihood	50.6	0.4294
ANN	78.4	0.7104
SVM	79.2	0.7269

created for all bands, however, due to space constraints and observed maximum diversity with band 4 versus 5, only these values have been plotted in figure 9. Also figure 5 shows the uniqueness of band 5. From these graphs, it can be noted that MLC is not the best tool to distinguish between classes. It shows highly scattered points for all the seven classes. For example, urban data points are overlapping with data points of timber, citrus, pasture and wetland. Better class discrimination is observed in the ANN and SVM scatter plots. Between the two, the SVM showed marginally better results. For example, the wetland and timber classes are less scattered in the SVM plot compared with ANN. Additionally, the SVM showed less overlap between pasture and citrus classes.

This study aimed at identifying seven land use categories. Individual class accuracies were evaluated and are presented in figure 10. The ANN and SVM classifiers performed better than the MLC except for classes urban, wetland and sod for which they showed comparable results. Although the MLC showed 100% accuracy for urban (eight out of eight sites were classified correctly), about 36.2% of the region was classified as urban (figure 7a) indicating that the MLC overestimated this category. In figure 10, it is seen that the SVM performed better than the ANN for classes sod, timber and urban while the ANN performed better for citrus, pasture and wetland. From figures 5 and 6 it is evident that spectral proximity observed during the training phase plays a critical role in classification accuracy. For example, classes sod and urban have close spectral proximity. The SVM was able to classify these two classes better than the ANN, indicating good discriminatory power of the SVM in spite of signature proximity. Similarly, water had a very distinct spectral cluster in figure 8. Hence, all the six points belonging to 'water' have been classified correctly by the ANN and SVM.

This research confirms most of the findings published by Huang *et al.* (2002) with some notable differences. The differences are notable because the algorithms were

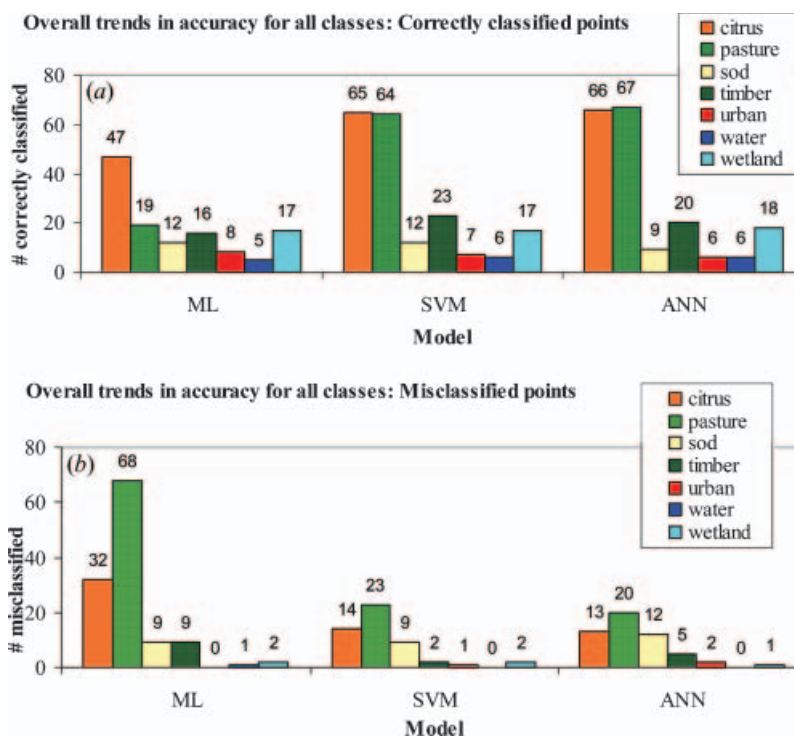


Figure 10. Overall trends in accuracies of individual classes for each model (a) correctly classified and (b) misclassified.

applied to two different satellites data sources. Huang *et al.* (2002) applied MLC, ANN, SVM and Decision Tree to classify TM corresponding MODIS imageries, whereas this study used MLC, ANN and SVM to classify Landsat TM data. Both of these studies show (1) that SVM was more accurate than MLC and (2) the stability and speed of ANN and SVM algorithms were comparable in spite of being applied to two different data sets (MODIS versus TM). It should be noted that Sun Ultra was used for the study conducted by Huang *et al.* (2002); PCs, an inexpensive alternative, were used in our study. Notable findings (and differences) of this research are that (1) SVMs simulations were faster than ANN while using TM data in a PC platform, (2) SVM marginally outperformed ANN while using with TM data, whereas Huang *et al.* (2002) found that SVM didn't provide significantly higher accuracy than ANN.

In summary, the ANN and SVM clearly outperform the traditional MLC especially when the training dataset is small. Almost equal classification accuracy is obtained from the ANN and SVM. However, the performance of these classifiers depends largely on their architectures. ANN's performance depends on the number of hidden units while that of the SVM depends on the kernel function used. In our work the SVM marginally outperformed the ANN. Careful optimization of the ANN, architecture leads to comparable performances by the two classifiers. However, this is a time consuming and cumbersome process. Availability of resources and time may govern whether a project should (can) use SVM, ANN or both. It is well known that the precision of MLC is not excellent, but it is surely the less expensive classification method, and in several cases it still may be suitable since

implementation of ANN and SVM are far more complicated than MLC. When used with TM data, SVM was faster than ANN. However, the difference was marginal to answer to the initial question 'one, the other or both'. It would be recommendable to carry out some other tests in different environmental situations, i.e. different land use, complex morphology etc., to find out which one is the better algorithm for a particular project.

4. Conclusions

The ANN and SVM both being non-parametric classifiers do not assume normality of the data. They are applicable to remote sensing applications with the data used in this study where this assumption may not be true. From our research, both the ANN and SVM significantly outperform the parametric MLC, which fails to show good results for a small training dataset. To answer the question that we started out with: 'Do we use the ANN or SVM, or both?'; our study shows comparable results with the two classifiers. However, the ANN requires extensive tuning with respect to its architecture. This is a relatively cumbersome process. Compared to the ANNs, SVMs are extremely fast and simple in implementation. While the ANN is sensitive to the number of hidden nodes, the SVM is sensitive to the choice of the mapping kernel. By optimizing these factors, it is possible to obtain comparable results with the two classifiers. SVMs have already been used in high dimensional datasets like face recognition. However, its use in remote sensing with fewer bands is new and needs to be researched. Future study would also involve datasets of higher dimensions to explore the effect of dimensionality on the accuracies of ANNs and SVMs. When used with TM data, SVM was faster than ANN. In this study using TM data, SVM was marginally faster than ANN, but not appreciable enough to warrant selecting one over the other without project-specific further investigation. It is recommended that extensive suitability analysis of various algorithms and classification methods must be conducted before undertaking extensive LULC data analysis for environmental modelling and management efforts. This type of research and resultant information is critical for the utilization of remotely sensed data to the fullest extent.

Acknowledgements

Funding for this research was provided by FWRRRC. The authors would like to thank YS for the continual support.

References

- AUSTIN, J., 1997, High speed image segmentation using binary neural network. In *Neuro-Computation in Remote Sensing Data Analysis*, I. Kanellopoulos, G.G. Wilkinson, F. Roli and J. Austin (Eds), pp. 202–213 (Berlin: Springer-Verlag, 1997).
- ANAND, 1999, The back propagation algorithm. From <http://www.speech.sri.com/people/anand/771/html/node37.html>.
- ATKINSON, P.M. and TATE, N.J., 2000, *Advances in Remote Sensing Image Analysis* (New York: John Wiley & Sons, Inc.).
- BARALDI, A. and PARMIGGIANI, F., 1995, A neural network for unsupervised classification of multivalued input patterns: an application to satellite image clustering. *IEEE Transactions on Geoscience and Remote Sensing*, **33**, pp. 305–316.
- BAUM, E.B. and HAUSSLER, D., 1989, What size net gives valid generalization? In *Advances in Neural Information Processing Systems I*, D.S. Touretzky (Ed.), pp. 81–90 (San Mateo, CA: Morgan Kaufmann Publishers).

- BENEDIKTSSON, J.A. and SVEINSSON, J.R., 1997, Feature extraction for multisource data classification with artificial neural networks. *International Journal of Remote Sensing*, **18**, pp. 727–740.
- CARPENTER, G.A., GOPAL, S., MACOMBER, S., MARTENS, S., WOODCOCK, C.E. and FRA, J., 1999a, A neural network method for efficient vegetation mapping. *Remote Sensing of Environment*, **70**, pp. 326–338.
- CARPENTER, G.A., GOPAL, S., MACOMBER, S., MARTENS, S. and WOODCOCK, C.E., 1999b, A neural network method for mixture estimation for vegetation mapping. *Remote Sensing of Environment*, **70**, pp. 138–152.
- CLASTRES, X., SAMUELIDES, M. and TARR, G.L., 1997, Dynamic segmentation of satellite images using pulsed coupled neural networks. In *Neuro-computation in Remote Sensing Data Analysis*, I. Kanellopoulos, G.G. Wilkinson, F. Roli and J. Austin (Eds), pp. 160–167 (Berlin: Springer-Verlag).
- CLAUDIO, C., CIOCCA, G. and SCETTINI, R., 2004, Image annotation using SVM. *IS&T and SPIE's Electronic Imaging*, San Jose.
- CONGALTON, 1991, A Review of Assessing the Accuracy of Classifications of Remotely Sensed Data. *Remote Sensing of Environment*, **37**, pp. 35–46.
- CRISTIANINI, N. and SHAW-TAYLOR, J., 2000, *An Introduction to Support Vector Machines and other kernel-based learning methods* (Cambridge: Cambridge University Press).
- DUDA, R. and HART, P., 1973, *Pattern Classification and Scene Analysis* (New York: John Wiley & Sons).
- DUDA, R., HART, P. and STORK, D.G., 2002, *Pattern Classification*, 2nd Ed. (New York: John Wiley & Sons).
- FLORIDA TOPOGRAPHIC BUREAU, T.M.S., 1985, Florida Land Use, Cover and Forms Classification System. 550-010-001-a, Florida Department of Transportation.
- FOODY, G.M. and ARORA, M.K., 1997, An evaluation of some factors affecting the accuracy of classification by an artificial neural network. *International Journal of Remote Sensing*, **18**, pp. 799–810.
- HARA, Y., ATKINS, R.G., SHIN, R.T., KONG, J.A., YEUEH, S.H. and KWOK, R., 1995, Application of neural networks for sea ice classification in polarimetric SAR images. *IEEE Transactions on Geoscience and Remote Sensing*, **33**, pp. 740–748.
- HUANG, C., DAVIS, L.S. and TOWNSHEND, J.R.G., 2002, An assessment of support vector machines for land cover classification. *International Journal of Remote Sensing*, **23**, pp. 725–749.
- HUANG, C., TOWNSHEND, J.R.G., LIANG, S., KALLURI, S.N.V. and DEFRIES, R.S., 2001, Impact of sensor's point spread function on land cover characterization: assessment and deconvolution. *Remote Sensing of Environment*, **80**, pp. 203–212.
- HUANG, Y., WANG, G., SU, L. and LIU, Z., 2003, An automatic recognition system for soil erosion based on knowledge and Support Vector Machine. *Proceedings of the Geoscience and Remote Sensing Symposium*, pp. 3444–3446.
- INGLADA, J. and GIROS, A., 2004, Automatic man-made object recognition in high resolution remote sensing images. *Proceedings of the Geoscience and Remote Sensing Symposium*, pp. 2011–2013.
- JENSEN, J.R., 2000, *Remote Sensing of the Environment, an Earth Resource Perspective* (Upper Saddle River, NJ: Princeton Hall).
- KANELLOPOULOS, I., VARFIS, A., WILKINSON, G.G. and MEGIER, J., 1992, Land-cover discrimination in SPOT HRV imagery using an artificial neural network- a 20-class experiment. *International Journal of Remote Sensing*, **13**, pp. 917–924.
- KANELLOPOULOS, I. and WILKINSON, G.G., 1997, Strategies and best practice for neural network image classification, **18**, pp. 711–725.
- MATHER, P.M., 2001, *Computer processing of Remotely-Sensed images: An Introduction* (New York: John Wiley & Sons).

- MELGANI, F. and BRUZZONE, L., 2004, Classification of hyperspectral remote sensing images with support vector machines. *IEEE Transactions on Geoscience and Remote Sensing*, **42**, pp. 1778–1790.
- PAL, M. and MATHER, P.M., 2003, Support Vector classifiers for Land Cover Classification, URL: <http://www.gisdevelopment.net/technology/rs/pdf/23.pdf>, GIS Development, Haryana, India (last date accessed: 14 December 2004).
- PAOLA, J.D. and SCHOWENGERDT, R.A., 1995, A review and analysis of back propagation neural networks for classification of remotely sensed multi-spectral imagery. *International Journal of Remote Sensing*, **16**, pp. 3033–3058.
- PERKINS, S., HARVEY, N.R., BRUMBY, S.P. and LACKER, K., 2001, *Support Vector Machines for Broad Area Feature Extraction in Remotely Sensed Images*, Proc. SPIE 4381, April.
- RICHARDS, J.A., 1993, *Remote Sensing Digital Image Analysis: an Introduction*, 2nd Ed. (Berlin: Springer-Verlag).
- RIEDMILLER, M. and BRAUN, H., 1993, A direct adaptive method for faster backpropagation learning: The RPROP algorithm. *Proceedings of the IEEE International conference on Neural Networks*, 1993.
- ROLI, F. and FUMERA, G., 2000, *Support Vector Machines for Remote-Sensing Image Classification*, Department of Electrical and Electronic Engineering, University of Cagliari, Cagliari, Italy.
- SHANKAR, A. and DESHWAL, P.S., 2002, *Face Detection in images: Neural networks & Support Vector Machines*, Computer Science and Engineering Department, Indian Institute of Technology, Kanpur, India.
- STATSOFT INC., 1984–2003, Neural Networks. From <http://www.statsoft.com/textbook/multilayerb>.
- VAN KHUU, H., LEE, H.K. and TSAI, J.L., 2003, *Machine Learning with Neural Networks and Support Vector Machines* (Unpublished).
- VAPNIK, V., 1995, *The Nature of Statistical Learning Theory* (New York: Springer Verlag).
- WESTON, J. and WATKINS, C., 1999, Support Vector Machines for Multi-Class Pattern Recognition. *ESANN '1999 Proceedings* (Bruges, Belgium: European Symposium on Artificial Neural Networks) ISBN 2-600049-9-X, pp. 219–224.
- XU, F., LI, X. and YAN, Q., 2003, Aerial Images Segmentation Based on SVM. *Proceedings of the Second International Conference on Machine Learning and Cybernetics*, Xi'an, pp. 2207–2211.

Appendix A

General background of algorithms

Maximum likelihood classifier (MLC). The MLC is the most commonly used method of classification in remote sensing. In this method, a pixel with the maximum likelihood is classified into the corresponding class. For a detailed explanation of MLC, please refer to Mather (2001). The MLC is a classical parametric classifier that relies on second order statistics of Gaussian probability density model for each class (Duda and Hart 1973). The equation for this classifier is given below (equation (1)):

$$D = \ln(a_c) - [0.5 \ln(|Cov_c|)] - \left[0.5 (X - \mu_c)^T Cov_c^{-1} (X - \mu_c) \right] \quad (A1)$$

where D = likelihood

c = particular class

X = measurement vector of the candidate pixel

μ_c = mean vector of the sample of class c

- a_c =percent probability that a candidate pixel is a member of class c (or a priori knowledge)
 Cov_c =covariance matrix of class c
 $|Cov_c|$ =determinant of Cov_c
 Cov_c^{-1} =inverse of Cov_c

A pixel is assigned to class c , for which the likelihood is the highest. The MLC has certain disadvantages. A good estimation of the mean vector and covariance of the population is essential, hence sufficient ground truth data should be sampled. MLC being a parametric classifier relies heavily on a normal distribution of the data in each input band. In cases where there is high correlation between bands or ground truth data is homogeneous, the inverse of the variance–covariance matrix becomes unstable. The MLC training is long and time consuming since it involves two matrix multiplications for each pixel and for each class.

Artificial neural network (ANN). The Multi Layer Perceptron (MLP) model using the Standard Back Propagation (SBP) algorithm is one of the well-known ANN classifiers. SBP is a method for assigning responsibility for mismatches to each of the processing elements in the network; this is achieved by propagating the gradient of the objective function back through the network to the hidden units (Anand 1999). Based on the degree of responsibility, the weights of each individual processing element are modified iteratively to improve the objective function. Inputs are supplied to the network and each input is given a weight, W . This weight is combined with other weights at the hidden layer node and a new weight is calculated. Weight modifications are made at all nodes then sent back between the first and second layers, until it reaches the designated output error rate. An error rate is set to help evaluate the actual value against the predicted value. One node is assigned to each input data. Two parameters, momentum and learning rates also affect the network. The SBP is mathematically defined as (equation (2)):

$$\Delta W_{ij} = \eta \delta_j O_i \quad (\text{A2})$$

If unit j is an output unit, then $\delta_j = f'_j(\text{net}_j)(t_j - O_j)$

If unit j is a hidden unit then $\delta_j = f'_j(\text{net}_j) \sum_k \delta_k W_{jk}$

η = learning parameter that specifies the step width of gradient descent

$d_j = t_j - O_j$ = difference between teaching value t_j and an output O_j of an output unit which is propagated back.

$f'_j(\text{net}_j)$ indicates ‘function of’ which in this case would be defined by the delta rule.

This algorithm updates the weights at every training pattern. A least squares objective function is minimized in a feed forward step followed by an error back propagation step during which the output and middle layer weights are adjusted to reduce the size of the error. This process is continued in an iterative fashion for each observation in the dataset until some desired degree of error minimization or convergence is reached (Statsoft Inc., 1984–2003). In this paper, the SBP was used to train the MLP. More details on the SBP can be found in Anand (1999).

Performance of the ANN can be significantly influenced by its architecture (Foody and Arora 1997). The number of hidden units is one of the most important factors that defines the capacity of a network (Baum and Haussler 1989). An optimal number of nodes need to be found and adjusted to create a better trained

network. In general, a small number of hidden nodes cause high training and generalization error due to underfitting. If the number of hidden nodes is too large, overfitting occurs which leads to low training but high generalization error.

The ANN has certain disadvantages. It minimizes only the empirical risk i.e. the network is trained to minimize the error on the training set. The most important manifestation of this problem is overfitting. The goal of machine learning is not only to fit the model to training data but also to minimize generalization error or the error on unseen data (Van Khuu *et al.* 2003). The learning process in an ANN involves iterative training, leading to long training times. The problem of overfitting might then get stuck in local minima.

Support vector machine (SVM). In recent years, the SVM has become an effective tool for pattern recognition, machine learning and data mining. The foundations of SVMs developed by Vapnik (1995) are gaining popularity due to many attractive features such as ability to find global optimum, increased speed of training and promising generalization performance.

One of the main results of Statistical Learning Theory is that the error probability of a classifier is upper bounded by a quantity depending not only on the error rate achieved on the training set, but also on an intrinsic property of the classifier, which is a measure of the “richness” of the set of decision functions it can implement (Roli and Fumera 2000). This property is named “capacity”, or Vapnik–Chervonenkis dimension. The more the set of decision functions is rich, the higher the classifier’s capacity. SVMs, developed by Vapnik (1995) and co-workers, are based on the SRM principle, which aims at reaching the minimum of the upper bound on the error probability of the classifier by achieving a trade-off between the training set and the capacity.

This technique consists of finding the optimal separation surface between classes due to the identification of the most representative training samples called the support vectors. If the training dataset is not linearly separable, a kernel method is used to simulate a non-linear projection of the data in a higher dimensional space, where the classes are linearly separable. A complete mathematical formulation of SVM can be found in Cristianini and Shawe-Taylor (2000). A brief description is given below (Pal and Mather 2003).

In the two-class case, the SVM attempts to locate a hyper plane that maximizes the distance from the members of each class to the optimal hyper plane. Assume that the training data with k number of samples is represented by $\{X_i, Y_i\} i=1, X \in R^n, k$, where X is an n -dimensional vector and $Y \in \{-1, +1\}$ is the class label. These training patterns are said to be linearly separable if a vector w (which determines the orientation of a discriminating plane) and a scalar b (determines offset of the discriminating plane from origin) can be defined so that inequalities (equation (3)) and (Equation (4)) are satisfied.

$$w \cdot x_i + b \geq +1 \quad \forall y = +1 \quad (\text{A3})$$

$$w \cdot x_i + b \leq -1 \quad \forall y = -1 \quad (\text{A4})$$

The training vectors x are solely used in inner products which can be replaced by a kernel function $K(x, y)$ that obeys Mercer’s condition. Mercer’s condition states that any positive semi-definite kernel $K(x_i, x_j)$ can be expressed as a dot product in high-dimensional space. Thus we avoid translating the input data to feature space first

and then finding their inner products. This is equivalent to mapping the feature vectors into a high-dimensional feature space before using a hyper plane classifier there. The use of kernels makes it possible to map the data implicitly into a feature space and to train a linear machine in such a space, potentially side-stepping the computational problems inherent in evaluating the feature map (Cristianini and Shawe-Taylor 2000).

SVMs were originally designed for binary analyses as discussed but generally land-use classification is a multi-class problem. One approach to solving k -class pattern recognition problems is by considering the problem as a collection of binary classification problems. k classifiers can be constructed, one for each class. The k^{th} classifier constructs a hyperplane between class n and the $k-1$ other classes (Weston and Watkins 1999). The one-versus-one approach is implemented in which $k(k-1)/2$ binary classifiers are trained. Each of these is an SVM trained to discriminate between two classes. To classify a case, this method combines the discrimination functions of these $k(k-1)/2$ classifiers using a voting scheme (Claudio *et al.* 2004).