A SUPERVISED SPECTRAL SUBSTRATUM CLASSIFIER TO CLASSIFY IMAGES WITH FUZZY MEMBERSHIPS

Z. Sha[a], Y. Xie[b,] *

[a] Wuhan University, International School of Software, Wuhan, Hubei Province, China - zongyaosha@yahoo.com.cn
[b] Eastern Michigan University, Institute for Geospatial Research & Education, 125 King Hall, Ypsilanti, MI 48197, United States - yxie@emich.edu

KEY WORDS: Classification, Land Cover, Fuzzy Logic, Multispectral, Vegetation, Landsat

ABSTRACT:

Remotely sensed images often display spectral variations over heterogeneous regions in the context of land cover classes (LCCs), which imposes challenges to information extraction from the images. In this paper, an easy-to-apply image classification model, supervised spectral substratum classifier, is proposed. The classifier first builds spectral LCCs (SLCCs) from a training dataset (*TD*). A SLCC comprises the spectral signals of a labeled LCC in *TD* based on the ground truth. This SLCC is further marked as homogeneous or heterogeneous according to the statistical properties of the mean value and the standard deviation of all spectral cases in this SLCC. When this SLCC is marked as heterogeneous, the spectral space of the SLCC will be disaggregated (or clustered) into substrata by applying statistical cluster analysis. A membership function is then defined for each substratum. To classify images, fuzzy membership functions are applied to measure similarities between corresponding spectral substrata and any new to-be-classified cases (pixels). The new cases are classified to the most comparable substrata as determined by the membership functions. As a case study, a vegetation cover classification over a typical grassland in Inner Mongolia from Landsat ETM+ is conducted. The result shows that the proposed classification model obtains an overall accuracy of 79.3% and kappa of 0.76. As comparison, a hybrid fuzzy classifier and a conventional and hard classification of maximum likelihood were applied as references.

1. INTRODUCTION

Remote sensing technology has been proved to be practical and economical means to study land cover changes and to assess natural resources, especially over large areas (Langley et al., 2001; Nordberg and Evertson, 2003). Image classification is widely used to derive useful information from remotely sensed datasets. Various models, or image classifiers, have been developed to extract land cover information from remote sensed images. Image classifiers can be broadly divided into unsupervised ones and supervised ones. Unsupervised approaches are often used in thematic mapping from imagery, and available in most of the image processing and statistical software packages (Langley et al., 2001). For supervised classification, a maximum likelihood (ML) classifier is usually viewed as a classic and most widely used method (Sohn and Rebello, 2002; Xu et al., 2005). More advanced classification models, such as artificial neural network (ANN) and support vector machine (SVM), have been attempted in recent years (Černá and Chytrý, 2005; Cristianini and Shawe-Taylor, 2000; Du and Sun, 2008; Gustavo and Lorenzo, 2009). Fuzzy logic classification, a kind of probability-based classification, also gets good attentions in recent years (Triepke et al., 2008).

To get a better classification result, there have been a few attempts to incorporate different image classification methods. Lo and Choi (2004) developed a hybrid classification method that incorporated the advantages of supervised and unsupervised approaches as well as hard and soft classifications for mapping the land use/cover of the Atlanta metropolitan area using Landsat 7 Enhanced Thematic Mapper Plus (ETM +) data. They applied a supervised fuzzy classification to the mixed pixels, and got a slightly better result than other methods (unsupervised ISODATA, supervised fuzzy, and supervised maximum likelihood classification methods) in terms of land use/cover classification accuracy. Laba et al. (2002) compared the accuracy of a regional-scale thematic map of land cover at taxonomic resolutions (i.e., different classification levels). The study showed that the map produced by the fuzzy-method had an obvious improvement in accuracy at both low and high taxonomic resolutions. In general, fuzzy image classifiers are more suitable for heterogeneous areas, while hard classifications are widely applied in homogeneous areas (Sha et al., 2008).

We propose in this paper an easy-to-apply fuzzy classification model (classifier) to extract land cover classes (LCCs) from remotely sensed images. The classifier first builds spectral LCCs (SLCCs) from a training dataset. A SLCC will be marked as heterogeneous if the statistical properties (mean value and standard deviation) of the cases labeled with this SLCC meet certain criteria. The spectral space of this SLCC will be disaggregated (or clustered) into substrata by applying statistical cluster analysis. Fuzzy membership functions are defined for the substrata based on the training dataset and then applied to measure similarities between the new cases and these spectral substrata and to determine their classifications.

2. METHODOLOGY

Many heterogeneous regions show obvious spectral variations over LCCs in remotely sensed images. Specifically, the cases labeled as a single LCC may demonstrate distince spectral deviations. Under such a condition, the cases labeled as the

* Corresponding author

LCC can be clustered into several sub-groups (substrata) with smaller within-group spectral deviations based on the spectral properties of the LCC labeled cases. This classification process is called a spectral substratum classifier.

Five steps are involved in implementing the proposed classification model. Step 1 creates a classification information system from the training dataset. Step 2 builds spectral substrata (SS) space for each SLCC from the training dataset. Step 3 defines membership functions for assigning new cases. Step 4 evaluates the classifier's performance through a testing dataset. Step 5 classifies new cases by applying the derived classifier.

Step 1: Creating a classification information system from the training dataset

For a given nonempty finite set of cases U={x_t} (t=1,2,……,n) where x_t indicates case t. Each case x_t in U is depicted by a set of attribute variables B_i={$b_{\cdot i}$} (i=1,2,…,m) and labeled by a class $C_{j.}$ (j=1, 2, …, n), $C_{j.} \in$ C={C_1, C_2, …, C_k}, where $b_{\cdot i}$ has a continuous value domain, C is a priori class label set, and the symbol "." in $C_{j.}$ indicates one of the candidate classes from C. That is to say, x_t = {b_{t1} , b_{t2}, …, b_{tm}, $C_{j.}$} Therefore, U can also be viewed as a matrix M with n rows and m+1 columns,

$$M_{n,m+1}(B,C)=\begin{pmatrix} b_{11} & b_{12} & \cdots & b_{1m} & C_{1\cdot} \\ b_{21} & b_{22} & \cdots & b_{2m} & C_{2\cdot} \\ b_{31} & b_{32} & \cdots & b_{3m} & C_{3\cdot} \\ \cdots & \cdots & \cdots & \cdots & \cdots \\ b_{n1} & b_{n2} & \cdots & b_{nm} & C_{n\cdot} \end{pmatrix} \qquad (1)$$

whereas the first m columns are called condition variables and the last column is a decision result. $C_{j.}$ denotes different elements from the candidate class set C. Note that different B_i may have the same class label.

Any object in U is uniquely determined by the values of its attributes. In other words, for any object $x_t \in$ U with an attribute set b_i describing x_t, the object can be uniquely classified (labeled) as C_j. This form of U (or M) is usually referred to as a classification information system (CIS). The notion of classification information systems (sometimes called data tables, information tables, attribute-value systems, knowledge representation systems, etc.) provides a convenient tool for the representation of objects in terms of their attribute values. The training dataset is taken to build a classifier and the testing dataset used to test the accuracy of a derived classifier that has the form of CIS.

Step 2: Building SS-Space for each SLCC from the training dataset

Let x_{i,j_z} denotes the j_z^{th} observation within class C_j for variable b_i, with $1 \le j_z \le$ n_j, 1 ≤ n_j ≤ n, n_j being the number of observations in class C_j, and $\sum_{j=1}^{k} n\_j$ =n. For all cases with such a unique label as C_j in U, calculate the mean values ($\overline{x_{i,j}}$) and the standard deviations (denoted as $\sigma_i(C_j)$) of the observations labeled as C_j for each variable b_i (i=1,2,…, m),

$$\overline{x_{i,j}} = \frac{1}{n\_j}\sum_{j_z=1}^{n\_j} x_{i,j_z} \qquad (2)$$

$$\sigma_i(C_j) = \sqrt{\frac{(x_{i,j} - \overline{x_{i,j}})^2}{n\_j}} \qquad (3)$$

Let $\overline{\sigma_i}$ be the mean value of $\sigma_i(C_j)$ of all C_j (j=1, 2, … k) for variable b_i (i=1, 2, …, m),

$$\overline{\sigma}_i = \frac{1}{k}\sum_{i=1}^{m}\sum_{j=1}^{k}\sigma_i(C_j) \qquad (4)$$

For m variables (b_i) in CIS, we can get a S.D. vector σ = ($\overline{\sigma_1}$, $\overline{\sigma_2}$, …, $\overline{\sigma_m}$) with dimension m. When $\sigma_i(C_j) \ge \overline{\sigma_i}$ (for variable b_i) happens to a candidate class C_j, there exists a larger deviation of spectral signals among the observations with respect to variable b_i. Therefore, $\sigma_i(C_j) \ge \overline{\sigma_i}$ is used as a judgment to mark "heterogeneous" for these cases. A recursive clustering to these cases labeled as C_j is then performed to determine substrata of the spectral signals until S.D. of each SLCC (subclasses of C_j, denoted as $C_{j\cdot}$ where "·" indicates that the subclasses come from C_j) satisfies $\sigma_i(C_{j.}) < \overline{\sigma_i}$.

A two-step hierarchical clustering analysis is recommended by using the Statistical Package for the Social Sciences (SPSS) (http://www.wright.edu/cats/docs/docroom/spss/), with the original candidate class as priori group and Euclidean distance as the linkage distance measure for variable bi, and the unweighted pair-group centroids as the linkage rule (LR). In addition to the control parameter of $\overline{\sigma_i}$, each subclass has to meet the requirement of a minimum number of cases (*MNC*). When the minimum case requirement is not satisfied, the cases will be merged with its nearest subclass in terms of Euclidean distance.

Accordingly, all the original cases can then be either labeled as C_j if no clustering is needed on the basis of $\overline{\sigma_i}$, or $C_{j\text{-}f}$ (f=1, 2, …, p, where p is the total number of the substrata after the clustered cases are labeled as C_j in the training dataset). In other word, the original k candidate LCC classes can be extended to γ substrata classes where $\gamma = \sum_{j=1}^{k}\sum_{f=1}^{p} 1$. For each $C_{j\cdot}$, the combination of $\overline{x_{i,j\cdot}}$ and $\sigma_i(C_{j.})$, similarly calculated by Function (2) and (3), is referred to as a spectral substratum space (SS space) for variable b_i and class $C_{j\cdot}$. A clustering analysis based on the CIS creates an SS space vector for each b_i (SS_i) with γ elements defined,

$$SS=\{ SS_1, SS_2, \ldots, SS_n\} \quad \text{and } SS_i =\{ss_1, ss_2, \ldots, ss_r\}, \quad \text{where } ss_i= (\bar{x}_{i,j\cdot} , \ \sigma_i(C_{j\cdot})) \tag{5}$$

Step 3: Defining membership functions from the training dataset

For any new object (case) that is attributed by b_i (i=1, 2, …, *m*), the classification task is to find out a substratum ($C_{j\cdot}$) that has the most comparable properties to the object. Thus, for any b_i, the following membership function is defined to calculate the similarity measure (*SM*) between the object and the class $C_{j\cdot}$,

$$SM_j = Max\{SM_{j\cdot}\} \quad \text{and } SM_{j\cdot} = \begin{cases} Max\ (1-|b_i - \bar{x}_{i,j-f}| / \beta \cdot \sigma_i(C_{j-f})), & \text{after clustering} \\ 1-|b_i - \bar{x}_{i,j}| / \beta \cdot \sigma_i(C_j), & \text{if no clustering is made} \end{cases} \tag{6}$$

where j· represents C_j if no clustering analysis is made, or C_{j-f} (f=1, 2, …, p) if C_j is grouped into p substrata after clustering. β is called overlapping coefficient for $\sigma_i(C_{j\cdot})$. For any C_j, $SM_{j\cdot}$ is the largest value of the similarity measure from the subclasses (i.e., substrata from the C_j). It can be seen that $SM_{j\cdot}$ =1 if $b_i = \bar{x}_{i,j\cdot}$, and $SM_{j\cdot}$ =0 if $|b_i - \bar{x}_{i,j\cdot}| \geq \beta \cdot \sigma_i(C_{j\cdot})$. The class label finally assigned to the object will be the label (C_j) that presents the closest similarity through the Max operation defined in Function 6.

For any new case, the defined membership (given by Function 6) compares the similarity between the new case and a number of known class (C_j or C_{j-f}) determined from the training dataset. Instead of directly assigning the new case to a known class, the function selects the most comparable class, to whichthe new case belong based on the calculated similarity measurement ($SM_{j\cdot}$).

An Example is provided as an illustration:

Suppose, for variable b_1, we have $\bar{x}_{1,1\cdot}$ =0.50, $\sigma_i(C_{1\cdot})$ =0.05; $\bar{x}_{2,2\cdot}$ =0.55, $\sigma_1(C_{2\cdot})$ =0.02. A given new case has b_1 =0.58 and β takes 3.0.

Scenario 1: According to Function 6, SM_1=0.47 and SM_2=0.50. This is to say, the new case shows more similarity to C_2 than to C_1. Therefore, the new case is more likely to be C_2 if the variable b_1 is taken as the input for consideration.

Scenario 2: When the deviation of sample cases is considered, the cases labeled as C_1 must be subdivided into two substrata through the clustering analysis, C_{1-1} and C_{1-2}, because of the fact that $\sigma_1(C_{1\cdot})$ =0.05 $\geq \bar{\sigma}_1$ =0.035. After the clustering analysis, we get two SS-spaces for C_{1-1} and C_{1-2} respectively, s_{1-1} = ($\bar{x}_{1,1-1}$ = 0.43, $\sigma_1(C_{1-1})$ = 0.02) and s_{1-2} = ($\bar{x}_{1,1-2}$ = 0.57, $\sigma_1(C_{1-2})$ =0.02).

According to Function 6, SM_1=max $\{SM_{1,1-1}, SM_{1,1-2}\}$ = {0, 0.83} = 0.83, and SM_2=0.50. Therefore, the new case is more comparable to C_1 if the variable b_1 is taken as the input for consideration. When there are a set of properties (b_i) to depict an object, a weighted average of SM_j for all considered b_i is used to decide the similarity measure,

$$\overline{SM_j} = \frac{1}{n}\sum_{i=1}^{n}\omega_i \cdot SM_j \tag{7}$$

where $\overline{SM_j}$ is the weighted average SM_j for all considered b_i, and ω_i stands for the weight for variable b_i. We placed equal weights for the 6 image bands in the case study. $\overline{SM_j}$ measures the similarity between any new case and C_j, and is computed to quantify the possibility that the new case belongs to a class C_j. The given object will be assigned to C_j that has the largest $\overline{SM_j}$.

Step 4: Evaluating classifier performance through testing dataset

The classified results derived from remote sensed images should be objectively verified and communicated to users so that they can make informed decisions on whether and how the products can be used. A testing dataset is used to evaluate the degree of 'correctness' of the classified features compared to the actual ones. Though there are a few evaluation methods, we used a confusion (or error) matrix to evaluate the classification result, which describes the goodness of fit between the derived classes and the reference data through using the measures like overall accuracy and kappa coefficient.

The testing dataset has a similar format as that of the training dataset. Any case in the testing dataset has an input vector denoted as $\{b_1, b_2, \ldots, b_n\}$ and an actual output class label C_j. The built fuzzy classifier from Step 3 takes the input vector and outputs (or assigns) a class label. The classifier outcomes are compared with the actual ones to build the error matrix. The overall classification accuracy and Kappa statistic are calculated to quantify the result (de Leeuw et al. 2006).

Step 5: Classifying new cases by applying the derived classifier

For any new case that needs to be classified, the derived classifier, if a reasonable classification accuracy is achieved, is employed to make classification. To classify a remotely sensed image, each pixel is a new case that is taken as input to the derived classifier and an output class label then is determined. Afterwards, a map covering the image area is sually produced to show visualize the classification result.

3. CASE STUDY

The proposed classifier is applied to classify grassland vegetation from Landsat Enhanced Thematic Mapper (ETM+) in Xilin River Basin, Inner Mongolia, China (Figure 1). The study region covers an area of nearly 10, 000 km^2 and strides two Landsat ETM+ scenes. It is one of the most representative steppe zones in China and the world (Xie et al., 2010). It is known from the long-term observations and field samplings that much of the research region is dominated by heterogeneous plant communities. It is confirmed that hard and pixel-based image classifications were not the right ways to map the vegetation cover in this region (Sha et al., 2008).
A vegetation classification system consisting of 11 vegetation communities is determined based on the plant ecological and biological features (Table 1). Two image scenes of Landsat ETM+ (path 124/row 29 and path 124/row 30) on 14 August, 2004, covering the whole region, are obtained and a series of image preprocessing tasks are performed to produce a qualified image for classifying vegetation cover (Sha et al., 2008). Simultaneous 464 ground samples evenly distributed over the study area were collected in the field with a hand-held global positioning system (GPS) with an accuracy of 15m and geo-registered to the image. The ground samples are divided into two groups: the training dataset and the testing dataset.

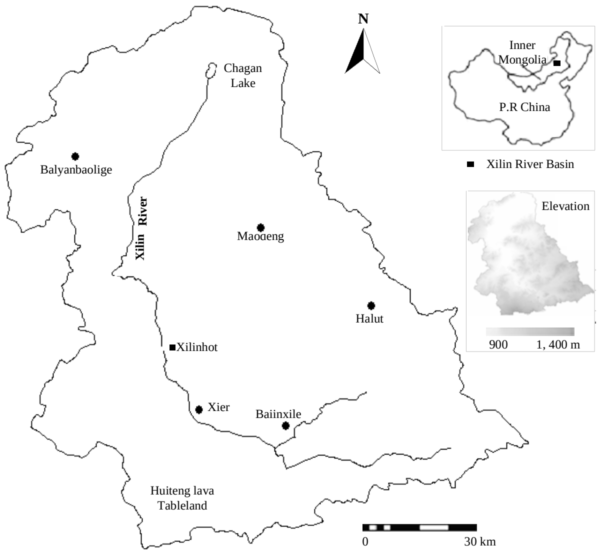

Figure 1. Study area

The training dataset, including 348 samples with six bands (variables) and the class labels (C_1, C_2, …, C_{11}), are taken to build the spectral substratum classifier. The rest 116 samples are used to validate its classification accuracy. All of the samples (training and testing) and the pre-processed image are used to extract spectral data of each ground sample. Spectral data for the samples are analyzed with the six reflective bands of the Landsat ETM+. Bands 1, 2, 3, 4, 5 and 7 of the pre-processed image are analyzed separately to create a SS space vector. The six bands of the pre-processed image are normalized through the following function, respectively,

$$b_i = (b-b_{Min}) / (b_{Max}-b_{Min}) \quad (8)$$

where b_i is the transformed value of any original pixel value (b) for layer i that has maximum and minimum pixel values given by b_{Max} and b_{Min}.

class	Community type (named after dominant species)	Vegetation type
C_1	*Cleistogenes squarrosa*	Typical steppe
C_2	*Stipa grandis*	Typical steppe
C_3	*Achnatherum splendens*	Meadow
C_4	*Stipa krylovii*	Typical steppe
C_5	*Artemisia frigida*	Typical steppe
C_6	*Carex pediformis*	Meadow steppe
C_7	*Carex spp.*	Meadow
C_8	*Caragana microphylla*	Typical steppe
C_9	*Leymus chinensis+Stipa baicalensis*	Meadow steppe
C_{10}	*Leymus chinensis*	Typical steppe
C_{11}	*Salsola collina (Chenopodium glaucum)*	Typical steppe

Table 2. Vegetation classification system

The 'brightness' value of each image band of the samples in the training set was normalized according to Equation 8 to produce a training data matrix of 348 × 7 (six bands + vegetation type). Similarly, all testing samples were processed to form a testing data matrix of 116 × 7. These matrixes are two CISs used to facilitate our analyses. The six variables (bands) along with the class label from the training matrix were analyzed to form a SS space vector {SS_1, SS_2, SS_3, SS_4, SS_5, SS_7}. It was found that the cases labeled with C_1, C_2, C_4, C_5, C_{10}, and C_{11} in the training dataset for most of the bands were marked heterogeneous and thus clustering analysis was applied to form substrata. Take band 1 as an example, a clustering analysis was performed on the cases originally labeled as C_2, C_4, and C_{10} in the training dataset. The results of the clustering analyses were reported Table 3. In other words, the cases in band 1 labeled as C_2, C_4, or C_{10} in the training dataset displayed significant variations, forming seven substrata. Moreover, the spectral variation of the variable within each substratum was decreasing when it was further clustered. As Table 2 revealed, the largest variations occurred with the cases labeled as C_2 since most LCCs in C_2 produced three substrata, $C_{2\text{-}1}$, $C_{2\text{-}2}$, and $C_{2\text{-}3}$ (Table 2).

The constructed classifier was then applied to the testing dataset for an accuracy evaluation. The result of the accuracy test showed that most reference classes with large sizes of cases could be well predicted by the classifier. The overall accuracy of the classifier reaches 79.3% with Kappa valued of 0.76 (Table 3). Considering the accuracy obtained from the classifier, the spectral substratum classifier could be applied to classify the whole image according to Equations 6)and 7 to derive the final vegetation cover map over the study region.

SS_{band}	SS_i list	γ
SS_1	C_1(0.52, 0.03) C_{2-1}(0.18, 0.04) C_{2-2}(0.31, 0.04) C_{2-3}(0.63, 0.04) C_3(0.16, 0.02) C_{4-1}(0.12, 0.02) C_{4-2}(0.67, 0.02) C_5(0.49, 0.04) C_6(0.62, 0.02) C_7(0.55, 0.03) C_8(0.96, 0.02) C_9(0.40, 0.02) C_{10-1}(0.20, 0.04) C_{10-2}(0.50, 0.03) C_{11}(0.52, 0.02)	15
SS_2	C_{1-1}(0.38, 0.03) C_{1-2}(0.64, 0.02) C_{2-1}(0.21, 0.02) C_{2-2}(0.31, 0.03) C_{2-3}(0.53, 0.03) C_3(0.67, 0.02) C_{4-1}(0.37, 0.03) C_{4-2}(0.57, 0.04) C_{5-1}(0.35, 0.02) C_{5-2}(0.73, 0.03) C_6(0.36, 0.03) C_7(0.54, 0.02) C_8(0.70, 0.02) C_9(0.27, 0.02) C_{10-1}(0.20, 0.04) C_{10-2}(0.24, 0.04) C_{11}(0.42, 0.02)	17
SS_3	C_1(0.61, 0.03) C_{2-1}(0.30, 0.02) C_{2-2}(0.60, 0.02) C_3(0.37, 0.02) C_{4-1}(0.24, 0.03) C_{4-2}(0.50, 0.03) C_5(0.46, 0.02) C_6(0.28, 0.02) C_7(0.76, 0.04) C_8(0.20, 0.03) C_9(0.53, 0.02) C_{10-1}(0.15, 0.04) C_{10-2}(0.33, 0.02) C_{11-1}(0.29, 0.03) C11-2(0.34, 0.02)	15
SS_4	C_1(0.32, 0.02) C_{2-1}(0.19, 0.03) C_{2-2}(0.50, 0.02) C_{2-3}(0.69, 0.03) C_3(0.44, 0.03) C_{4-1}(0.37, 0.03) C_{4-2}(0.86, 0.02) C_{5-1}(0.27, 0.02) C_{5-2}(0.60, 0.03) C_6(0.65, 0.02) C_7(0.23, 0.02) C_8(0.36, 0.02) C_9(0.30, 0.02) C_{10-1}(0.28, 0.02) C_{10-2}(0.53, 0.04) C_{11}(0.45, 0.03)	15
SS_5	C_{1-1}(0.28, 0.02) C_{1-2}(0.39, 0.04) C_{2-1}(0.25, 0.04) C_{2-2}(0.44, 0.02) C_{2-3}(0.83, 0.03) C_3(0.49, 0.03) C_{4-1}(0.28, 0.03) C_{4-2}(0.77, 0.03) C_5(0.39, 0.04) C_6(0.30, 0.02) C_7(0.55, 0.02) C_8(0.74, 0.03) C_9(0.68, 0.02) C_{10-1}(0.26, 0.03) C_{10-2}(0.72, 0.03) C_{11}(0.31, 0.02)	16
SS_7	C_{1-1}(0.26, 0.02) C_{1-2}(0.59, 0.03) C_{2-1}(0.52, 0.03) C_{2-2}(0.78, 0.02) C_3(0.45, 0.02) C_{4-1}(0.32, 0.04) C_{4-2}(0.54, 0.03) C_{5-1}(0.23, 0.02) C_{5-2}(0.45, 0.03) C_6(0.35, 0.04) C_7(0.23, 0.03) C_8(0.51, 0.04) C_9(0.23, 0.03) C_{10-1}(0.18, 0.02) C_{10-2}(0.31, 0.03) C_{11}(0.45, 0.03)	15

Table 3. SS space vector derived from the training dataset (*MNC* (minimum number of cases) =5)

Map class	Reference class											Total	User's
	C_1	C_2	C_3	C_4	C_5	C_6	C_7	C_8	C_9	C_{10}	C_{11}		
C_1	5	2		1								8	62.5
C_2	1	23	1	1		1			1	1		29	79.3
C_3			6								1	7	85.7
C_4		1		15								16	93.8
C_5	2				4							6	66.7
C_6						2						2	100
C_7					1		2					3	66.7
C_8								6		1		7	85.7
C_9					1				4			5	80
C_{10}		2		2			1			15		20	75
C_{11}				1		1				1	10	13	76.9
Total	8	28	7	20	6	4	3	6	5	18	11	116	
P*	63	82	86	75	67	50	67	100	80	83	91		

Overall accuracy: 79.3%; kappa=0.76
P* standards for Producer's

Table 4. Error matrix for the proposed classifier

The classification performance was further assessed in comparison with other classification models (Table 5). It was found that the result from the proposed substratum classifier produced a comparable accuracy as the hybrid fuzzy classifier (HFC) did in the same study area (Sha, et al., 2008) and had a much better performance than the conventional supervised classifier (CSC) model. In addition, in terms of the procedures involved, the proposed substratum model was relatively easier to carry out.

Classification method	Percentage classified	Kappa
HFC(Sha et al., 2008)	80.2	0.77
CSC*	69.0	0.63
Spectral Substratum Model	79.3	0.76

*CSC: Conventional supervised classification on the basis of maximum likelihood.

Table 5. Result comparison with other models

4. CONCLUSION AND DISCUSSIONS

The proposed spectral substratum classifier essentially adopts a fuzzy or soft classification strategy. Fuzzy classifiers have been studied for years and proved to produce more accurate classifications compared to the hard methods especially over a hetegeneous environment. Under such a condition,, a pixel in an image may not display an overwhelming similarity to a LCC. Instead, it would be better to say that the pixel is more likely belonging to a LCC.

In the applications of environmental mapping from remote sensed images, two considerations are usually taken into account to develop a new classification system, when the sampled cases in the training dataset show distinct spectral variations even if they belong to the same LCC. First, if the spectral variations are within a reasonable limit, all cases can be treated as a class corresponding to a LCC. Second, if the variations within a spectral LCC (SLCC) are too large, these cases can be split into two or more subclasses (substrata). The derived substrata will be used to replace the original SLCC. Compared to SLCC, these substrata show much smaller within-group spectral variations. In our research, a hierarchical clustering analysis is performed with all the variables in the training dataset as an input vector to derive substrata for the LCC cases. Each substratum has a membership function defined by the statistical properties (mean value and standard deviation) of the cases labeled with this substratum. During the clustering process for variable *i* and class C_j, two parameters are examined to control its running:

1) First, when the standard deviation of all the subclasses ($\sigma_i(C_{j\cdot})$) is smaller than a predetermined parameter, i.e., the average standard deviation ($\bar{\sigma}_i$) for all the original classes, the clustering stops. This strategy assumes that only a few classes displaying significant spectral variations among he labeled cases, are to be generated. In other words, $\bar{\sigma}_i$ is the controlling parameter. Though the value of this predefined parameter can be manually set, setting it too

small will lead to too many subclasses (substrata) to be generated, which may make the classifier over-fitting or having too many noises and thus lacking prediction power. In the current case study, we compared a few results by setting different values for the parameter (i.e., $\bar{\sigma}_i$, $1.5\times\bar{\sigma}_i$, and $0.5\times\bar{\sigma}_i$) and found that $1.0\times\bar{\sigma}_i$ performed best in terms of the classification accuracy test. However, the trial-by-error method for setting up a value for $\bar{\sigma}_i$ is neither robotic nor the best for accuracy assessment. Future efforts should be made to explore better strategies for setting up the parameter.

2) Second, when the number of cases in a subclass is smaller than a predefined parameter, *MNC* (minimum number of cases), the analysis also stops. For the similar reason, a substratum containing too few cases in the training dataset will also lead to over-fitting or having too many noises.

To measure the similarity between any given case and a substratum from LCCs, a set of membership functions are defined based on each SS space of each SLCC. We first define a membership function for each variable for each substratum and then combine the effect of all the variables to make an overall determination function given by band weight vector ω_i. A trial-and-test method is employed to set the suitable weight vector. The membership function defined to measure the similarity of a given object to a substratum for a single variable takes the mean value and the standard deviation of the variable from the substratum labeled cases in the training dataset. The empirical parameter that affects the measured similarity value is the overlapping coefficient (β). As can be seen from Equation 6, the increase of β also increases the similarity measure value. While increasing β may lead to more cases to be classified, misclassification may occur when doing so. On the contrary, decreasing β may lead to some cases that actually belong to a substratum get lower similarity values and thus may be misclassified to other classes. In the case study, we adopt a value of 3 for β . This value considered the statistical properties (mean value and standard deviation) of the proximity between different classes and had the best accuracy when compared to other settings.

Although a moderate classification accuracy is obtained by the spectral substratum classifier in our case study, it is one of the best classification results among the present literature, considering the complicated vegetation cover and strong human influences in this region. In addition, the classifier is also easy to build and could be widely applicable to different environment conditions. Therefore, we suggest that the spectral substratum classifier should be further tested to extract information from remotely sensed images in other heterogeneous regions.

REFERENCES

Černá, L. and Chytrý, M., 2005, Supervised classification of plant communities with artificial neural networks. Journal of Vegetation Science, 16(4), pp. 407-414.

Cristianini, N. and Shawe-Taylor, J., 2000, An introduction to Support Vector Machines and other kernel-based learning methods. Cambridge University Press.

De Leeuw J., Liu, X., Schmidt, K., Skidmore, A.K., Jia, H. And Yang, L., 2006, Comparing accuracy assessments to infer superiority of image classification methods. International Journal of Remote Sensing, 27(1), pp. 223-232.

Du, C.J., Sun, D.W., 2008. Multi-classification of pizza using computer vision and support vector machine. Journal of Food Engineering 86(2), pp. 234-242

Gustavo, C.V., and Lorenzo, B., 2009. Kernel Methods for Remote Sensing Data Analysis. Wiley, Singapore, pp.125-148.

Triepke, F.J., Brewer, C. K., Leavell, D. M., Novak, S. J., 2008. Mapping forest alliances and associations using fuzzy systems and nearest neighbor classifiers. Remote Sensing of Environment, 112(3), pp. 1037-1050.

Laba, M., Ogurcak, D., Hill, E., Fegraus, E., Fiore, J., Degloria, S.D., Gregory, S.K. And Braden, J., 2002, Conventional and fuzzy accuracy assessment of the New York Gap Analysis Project land cover map. Remote Sensing of Environment, 81(2-3), pp. 443-455.

Langley, S.K., Cheshire, H.M., Humes, K.S., 2001. A comparison of single date and multitemporal satellite image classifications in a semi-arid grassland. J Arid Environ., 49(2), pp. 401-11.

Lo, C.P. And Choi, J., 2004, A hybrid approach to urban land use/cover mapping using Landsat 7 Enhanced Thematic Mapper Plus (ETM +) images. International Journal of Remote Sensing, 25(14), pp. 2687-2700.

Nordberg, M.L., Evertson, J., 2003. Vegetation index differencing and linear regression for change detection in a Swedish mountain range using Landsat TM and ETM+ imagery. Land Degradation & Development, 16(2), pp. 139–149.

Sha, Z., Bai Y., Xie, Y., Yu, M., Zhang, L., 2008. Using a hybrid fuzzy classifier (HFC) to map typical grassland vegetation in Xilin River Basin, Inner Mongolia, China. International Journal of Remote Sensing 29(8), pp. 2317-2337.

Sohn, Y., Rebello, N.S., 2002, Supervised and unsupervised spectral angle classifiers. Photogrammetric Engineering and Remote Sensing, 68(12), pp. 1271–1280.

Xie, Y., Sha, Z., and Bai, Y., 2010. Classifying historical remotely sensed imagery using a tempo-spatial feature evolution (T-SFE) model. ISPRS Journal of Photogrammetry and Remote Sensing, 65(2): 182-190.

Xu, M., Watanachaturaporn, P, Varshney P.K., et al., 2005. Decision tree regression for soft classification of remote sensing data. Remote Sens Environ., 97(3), pp. 322–36.