The Nature and Classification of Unlabelled Neurons in the Use of Kohonen’s Self-Organizing Map for Supervised Classification

Zhe Li*
Graduate School of Geography
Clark University

J Ronald Eastman
Graduate School of Geography
Clark University

Abstract
Kohonen’s Self-Organizing Map is a neural network procedure in which a layer of neurons is initialized with random weights, and subsequently organized by inspection of the data to be analyzed. The organization procedure uses progressive adjustment of weights based on data characteristics and lateral interaction such that neurons with similar weights will tend to spatially cluster in the neuron layer. When the SOM is associated with a supervised classification, a majority voting technique is usually used to associate these neurons with training data classes. This technique, however, cannot guarantee that every neuron in the output layer will be labelled, and thus causes unclassified pixels in the final map. This problem is similar to but fundamentally different from the problem of dead units that arises in unsupervised SOM classification (neurons which are never organized by the input data). In this paper we specifically address the problem and nature of unlabelled neurons in the use of SOM for supervised classification. Through a case study it is shown that unlabelled neurons are associated with unknown image classes and, most particularly, mixed pixels. It is also shown that an auxiliary algorithm proposed here for assigning classes to unlabelled neurons performs with the same success as that experienced with Maximum Likelihood.

Address for correspondence: Zhe Li, Graduate School of Geography, Clark University, 950 Main Street, Worcester, MA 01610, USA. E-mail: zheli@clarku.edu

*Zhe Li won the TGIS Prize for the best student paper presented at the 2005 University Consortium for Geographic Information Science Summer Assembly in Jackson Hole, Wyoming
1 Introduction

In recent years there has been considerable interest in the use of neural networks for the classification of remotely sensed imagery because of their freedom from assumptions about the form and distribution of input data, the ability to generate non-linear decision boundaries and the ability to generalize inputs as well as to learn complex patterns (Foody 1997, Ji 2000, Tso and Mather 2001). As a neural approach, Kohonen’s (1989, 1990) Self-Organizing Map (SOM) has not been explored as thoroughly as the Multi-Layer Perceptron approach for the classification of remotely sensed imagery (Tso and Mather 2001). However, it is of particular interest in that it is capable of being used for both unsupervised and supervised classification (Ji 2000; Villmann et al. 2003; Mannan and Ray 2003; Waldemark 1997; Ito and Omatu 1997, 1998). In this paper we specifically address its use in supervised classification and the problem of unlabelled neurons – neurons that are activated and organized by existing input vector patterns, but which are not associated with training site vectors. These are somewhat analogous to, but fundamentally different from the problem of dead units that arises in unsupervised SOM classification.

The original concept of SOM, as developed by Kohonen (1989, 1990, 2001), is based on competitive learning in which lateral interaction (Bian and Zhang 2000) in the output layer self-adaptively leads to regional organizations of neurons (a topology) that become special detectors for different signals (e.g. land cover classes). Neurons in the competitive layer are associated with input vectors by weights that are randomly assigned at first, but which are iteratively modified by examination of input data. Because adjustment of weights is spread spatially to neighboring neurons using a distance decay function, the process is essentially one of an adaptive $k$-means clustering where neurons become organized into clusters of association with input vectors.

In the more typical context of unsupervised classification, competitive layer neurons are subsequently clustered into information classes that accommodate intra-class variability. In such instances it is common to encounter dead units – neurons assigned random initial weights which subsequently fail to be associated with any input vectors (Kok 2000, Huang et al. 2000, Yang 2002, Sahin 2003). Some solutions were proposed by researchers to avoid dead units such as initializing output weights to samples from the input patterns, updating the weights of the losers, using a conscience mechanism, using a neighborhood function in the input space, updating the weights of all users with a much smaller rate, and so forth. Pal et al (2005) deal with dead units by rejecting them. Ideally, each neuron is expected to be able to win, or be in the neighborhood of a winner, for at least one input from the training data (Huang et al. 2000). In reality, there can always be neurons that are never activated. Depending upon the output clustering procedure used, these may or may not be associated without output clusters.

In supervised classification, dead units are not a problem. After initial organization of the input data, training site pixels are assigned to their most similar competitive layer neurons. Output classes are then based on the most frequently occurring training site class associated with each neuron. The dead units thus never affect the outputs since they are never associated with any input vector. However, a second form of disconnected unit arises – neurons that are indeed associated with input patterns, but which are not associated with any training site vectors. In their disconnectedness, they are similar to dead units. However, they are also fundamentally different. They are very much alive in that they are associated with actual patterns in the input data (which
includes all data, not just the known training data). The problem is that we do not know what to call them – they do not exist in the training data. They are essentially orphans that remain unlabelled. Clearly they could be unknown land cover classes, but we also reasoned that they might also represent mixed pixels. Regardless, they are a problem because they will be triggered by the input data. In this paper we explore the character of these unlabelled neurons (disconnected units) and an algorithm for their assignment to a class.

2 An Overview of the SOM Procedure

The SOM procedure we adopted was closely modeled on that discussed by Kohonen (1990) and Ji (2000). Figure 1 illustrates the basic architecture of a SOM. The input layer represents the input feature vector and thus has neurons for each measurement dimension. In the case of remotely sensed data, this would imply a separate neuron for each reflectance band. The output layer of a SOM is typically organized as a two-dimensional (typically square) array of neurons, although one-dimensional structures are also common. Each output layer neuron is connected to all neurons in the input layer by synaptic weights. In its use for supervised classification the procedure begins with a coarse tuning phase that is effectively a form of unsupervised classification. A

Figure 1  Example of the architecture of a SOM with an input layer (made up of three neurons) and an output layer (made up with $5 \times 5$ neurons equally spaced)
later fine tuning stage refines intra-class decision boundaries using a Learning Vector Quantization (LVQ) procedure. Another critical phase in between these two is the assignment of neurons (sub-clusters) to training classes – a process known as labelling.

### 2.1 Coarse Tuning

Coarse tuning is an unsupervised classification stage in which competitive learning and lateral interaction lead to a fundamental regional organization (a topology) of neuron weights that represent the underlying clusters and sub-clusters in the input data.

Specifically, let \( \mathbf{x} = \{x_1, x_2, \ldots, x_n\} \) be an \( n \)-dimensional feature vector input to the SOM (i.e. a vector of reflectances for a single pixel). The Euclidean distances between a reference vector (weight vector) and an input feature vector can then be calculated, and the neuron in the output layer with the minimum distance to the input feature vector (known as the winner) is then determined as:

\[
\text{Winner} = \arg \min_j \left( \sum_{i=1}^{n} (x_i^t - w_{ji}^t)^2 \right)
\]

where \( x_i^t \) is the input to neuron \( i \) at iteration \( t \), and \( w_{ji}^t \) is the synaptic weight from input neuron \( i \) to output neuron \( j \) at iteration \( t \). The weights of the winner and its neighbors within a radius \( \gamma \) are then altered (while those outside are left unaltered) according to a learning rate \( \alpha^t \) as follows:

\[
\begin{align*}
    w_{ji}^{t+1} &= w_{ji}^t + \alpha^t \cdot (x_i^t - w_{ji}^t) \quad \forall d_{\text{winner},j} \in \gamma^t \\
    w_{ji}^{t+1} &= w_{ji}^t \quad \forall d_{\text{winner},j} \notin \gamma^t
\end{align*}
\]

where \( \alpha^t \) is the learning rate at iteration \( t \) and \( d_{\text{winner},j} \) is the distance between the winner and other neurons in the output layer. The learning rate is a value between 0 and 1 that declines with time between its maximum and minimum values. It can be defined by Equation (4), with constraints \( 0 \leq \alpha, \alpha_{\text{min}}, \alpha_{\text{max}} \leq 1 \).

\[
\alpha^t = \alpha_{\text{max}} \left( \frac{\alpha_{\text{min}}}{\alpha_{\text{max}}} \right)^{\gamma_{\text{max}}^{\gamma^t}}
\]

During the tuning phase, the radius \( \gamma \) decreases from an initial size that can encompass all of the neurons in a single neighborhood to an ending one which includes only the winner. This adjustment of weights thus proceeds from global order to local adjustments (Poth et al. 2001, Tso and Mather 2001).

During the tuning phase, the radius \( \gamma \) decreases from an initial size that can encompass all of the neurons in a single neighborhood to an ending one which includes only the winner. This adjustment of weights thus proceeds from global order to local adjustments (see Poth et al. 2001, Tso and Mather 2001 for additional details).

### 2.2 Labelling

Before a SOM can perform a classification, the output layer neurons (also known as code books) must be labelled. This is to determine which class each output neuron
belongs to. At this stage the basic structure of the input data has been topologically organized within the SOM output layer. The result, however, is similar to a cluster analysis in that the identity of regional groupings of neurons is unknown. The labelling stage is intended to establish the identities of these regional associations by comparison with training data. To do this, training data are fed to the coarse tuned network. The training site class that is assigned most frequently to a neuron then becomes its label. This procedure is known as majority voting (Tso and Mather 2001).

2.3 Fine Tuning

In some cases, one expects to improve accuracy by using a supervised training if there are training data available. The goal of fine tuning is to refine the decision boundaries between classes based on the training site data. In the coarse tuning stage, raw input data are fed into the training process. The topological organization of the feature map is thus based on global image characteristics and it is normal to find that a group of neurons is labelled with a single information class. That is, a group of neurons will commonly cover the range of variability in reflectance associated with the information class. In the fine tuning stage, the specific boundaries between neurons associated with specific information classes are refined using training site data.

In the procedure outlined by Kohonen (1990), fine tuning is achieved through the use of Learning Vector Quantizations (LVQs) (Kohonen 1989, 1990; also described in Ji 2000 and Tso and Mather 2001). At present we have implemented the LVQ1 and LVQ2 procedures. With LVQ1, known patterns (from training sites) are fed again to the SOM. The neuron with the most similar weights to the input vector of reflectances, also known as the “best matching unit” (BMU) is determined and its weights are adjusted as follows:

\[
\begin{align*}
\text{if } x & \text{ is correctly classified: } \\
\text{if } x & \text{ is incorrectly classified: }
\end{align*}
\]

\[w_{c}^{t+1} = w_{c}^{t} + \delta'(x_{i} - w_{c}^{t}) \quad \text{if } i \neq c\]

where \(w_{c}^{t}\) is the weight vector of the winner, and \(\delta'\) is a gain term that will decrease with time (using the same logic as that which governs the decay of the learning rate and as expressed in Equation 4) and is within the range of (0, 1). In this manner weights move closer to an input feature if they match the label of the corresponding winner (Equation 5), while moving farther from it if it does not (Equation 6).

With LVQ2, weight vectors are updated in the following way:

\[w_{i}^{t+1} = w_{i}^{t} - \delta'(x_{i} - w_{i}^{t}) \quad \text{and} \]

\[w_{i}^{t+1} = w_{i}^{t} + \delta'(x_{i} - w_{i}^{t}) \quad \text{if } C_{i} \text{ is the nearest class, but } x \text{ belongs to } C_{i} \neq C_{i} \]

\[w_{k}^{t+1} = w_{k}^{t} \quad \text{in all other cases}\]

The primary controls for the fine tuning are thus the number of iterations and the gain factor \(\delta'\) setting. See Tso and Mather (2001) and Ji (2000) for useful guidelines.
2.4 The Feature Map

With a converged status, SOM can characterize the distribution of input samples, and thus generate a two-dimensional map from a multi-dimensional feature space (Mather 1999). The feature map of the SOM is analogous to the mapping cortex of the human brain upon which processing of spatial information is based. After either the coarse tuning and labelling or fine tuning stages, the feature map can be viewed. The feature map is color coded by information classes. Thus all neurons associated with a particular class share the same color. As can be seen in Figure 2a, classes with substantial variability (such as the deciduous vegetation class symbolized with a forest green color) will have many neurons associated with it while those with little variability will be associated with only a few neurons. For example, the shallow water class (in cyan) and deep water (dark blue) occupy only one neuron each. Topology preservation is a key property of SOM, neurons that represent classes with similar properties are therefore likely grouped together, such as the deep water and shallow water. The crop (light green) land class and grass class are very similar in terms of reflectance on the three bands, so in the feature map they are close neighbors. It is also quite reasonable to see that high/low density residential, industrial/commercial and transportation are grouped together. Interestingly, classes where spatially separate neurons occur indicate cases of multimodal distributions, e.g. the high density residential (orange) and low density residential (yellow) classes exhibit clear multimodal distributions in the NIR band. Similarly the reed class (light magenta) is adjacent to both deep water and shallow water, which suggests that it may relate to shallow areas where submerged vegetation can be seen. Neurons colored in black in the feature map are unlabelled. We find the feature map to be a useful adjunct to the classification process.

2.5 Classification

The final classification of the input image is normally performed after the fine tuning stage (but can be undertaken after coarse tuning and labelling). Each input pixel is now assigned the class label of the neuron most similar in its weight structure to the pixel vector of reflectances.

2.6 The Problem of Unlabelled Neurons (disconnected units)

As discussed earlier, because the unsupervised stage of SOM uses all image pixels, unknown classes or characteristic mixed pixel assemblages will lead to the presence of neuron clusters that are not represented in the training site data and these clusters thus generate unlabelled neurons in the feature map that are additional to any dead units. During the classification stage, when input pixels to be classified trigger labelled neurons, they are assigned to the class associated with them, while if they trigger unlabelled neurons (disconnected units), they cannot be assigned to any class and thus are left unclassified.

To illustrate the extent of the problem, Figure 2a shows the feature map from the experimental study that will be described below. Unlabelled neurons can be seen as the black cells in the feature map, which account for 42.6% of the total (289) neurons. Since both dead units and the second form of disconnected units may be left unlabelled, it is meaningful to distinguish them and determine their relative proportions. To do this,
Figure 2  (a) A $17 \times 17$ feature map with unlabelled neurons including both dead units and disconnected units; (b) a $17 \times 17$ feature map with unlabelled neurons including only dead units; (c) a $50 \times 50$ feature map with unlabelled neurons including both dead units and disconnected units; and (d) a $50 \times 50$ feature map with unlabelled neurons including only dead units. This figure appears in colour in the electronic version of this article and in the plate section at the back of the printed journal.
we used all pixels from a Maximum Likelihood classification as training sites. In this case, all input vectors will be associated with a class and thus it is likely that any unlabelled neurons will be the result of dead units. Figure 2b shows the feature map labelled by these training data. Interestingly, only two neurons were unlabelled and all others were assigned to their associated classes. This suggests that only two neurons were dead units (accounting for 0.69% of the total neurons) and all other unlabelled neurons (41.9% of all neurons) were disconnected units. When the feature map size is increased to $50 \times 50$ (2,500 neurons) and the above procedure is repeated, more unlabelled neurons (80.3%) were found (Figure 2c). In addition, the proportions of dead and disconnected units changed to 25.3% and 55.0%, respectively. Figure 2d shows the feature map with only dead units. This result does make sense in that tuning a larger sized SOM is more likely to increase both dead and disconnected units. This interesting phenomenon confirmed our early reasoning. It is important to note that the above procedure is merely an illustration of distinguishing dead units and disconnected units, because we would never have such ideal training sites that can cover every pixel of the entire study area.

3 The Auxiliary Labelling Algorithm

In unsupervised classification, competitive layer neurons are organized into classes through the use of a clustering procedure such as an agglomerative clustering or $k$-means procedure. In supervised classification, this clustering is based on association with training site data. We therefore approached the classification of unlabelled neurons by adding an auxiliary algorithm using the logic of agglomerative clustering to assign unlabelled neurons to the clusters already formed from the supervised stage.

To establish a class for pixels that trigger the second form of disconnected units in the SOM feature map during classification, it was decided to evaluate the mean distance ($\bar{D}_i$) between the input pattern and all neurons associated with each class: i.e.

$$\bar{D}_i = \frac{1}{N_i} \sum_{j=1}^{N_i} ||x - w|| \quad \forall \text{Label}(j) = i$$

where $x$ is the input vector, $w$ the organized reference vector (weight vector), and $N_i$ is the number of the output layer neurons that are labelled by class $i$ ($i = 1, 2, \ldots, m$).

With this algorithm, all pixels that trigger the disconnected neurons will be assigned a class that has a minimum $\bar{D}_i$. This algorithm can serve as a tool to detect pixels with uncertainty and provide useful information especially when one has very limited information about the underlying study area.

We also considered evaluating the minimum distance ($D_{\text{mini}}$) between unclassified pixels and the neurons associated with each class rather than the means, i.e.

$$D_{\text{mini}} = \min_{j} ||x - w|| \quad \forall \text{Label}(j) = i$$

With this algorithm, problem pixels are assigned to a class with the minimum $D_{\text{mini}}$. However, initial tests did not yield noticeable differences with the above approach. Thus we have decided to look at this in a separate testing program.
The above two methods aim to assign classes to problem pixels directly without interaction with those unlabelled neurons (disconnected units). We have also considered an intermediate procedure, namely, labelling those unlabelled neurons by evaluating the mean distance $D_{\text{Class,unlabelled}}$ between the unlabelled neuron reference weights and all neurons associated with each class. This procedure is similar to the calculation of a U-matrix (Ultsch et al. 1989, Kohonen 2001). The unlabelled neuron is then assigned to the class with the minimum $D_{\text{Class,unlabelled}}$. Once those previously unlabelled neurons are labelled, classification will not be a problem. This algorithm, however, is under construction at this moment and therefore has not been applied in this experiment.

4 A Case Study Evaluation

4.1 Data Description

To evaluate the implications of this procedure for handling unlabelled neurons we undertook a classification of a SPOT HRV image (with 3-bands: G, R, and IR) from 1991. A sub-image of $565 \times 453$ pixels covering the region ($11.3 \text{ km} \times 9.0 \text{ km}$) around Westborough, Massachusetts was extracted as the study site (Figure 4a). Training and validation sites for 12 land use/cover classes were digitized and extracted from the imagery. A total of 4,597 training samples were selected for training the SOM and a total of 3,085 test samples were used to validate the accuracy of the results (Table 1).

4.2 Application of the SOM in Land Cover Classification

The software used for this research was a SOM classifier being developed by the authors as part of a suite of neural net classifiers for an upcoming release of the IDRISI GIS and Image Processing software system (Eastman 2003) The interface of SOM was designed as Figure 3. In this experiment, a series of SOMs with different output (competitive

<table>
<thead>
<tr>
<th>Class ID</th>
<th>Land use/cover</th>
<th>Training site pixels</th>
<th>Test site pixels</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>High density residential area</td>
<td>193</td>
<td>72</td>
</tr>
<tr>
<td>2</td>
<td>Low density residential area</td>
<td>220</td>
<td>132</td>
</tr>
<tr>
<td>3</td>
<td>Industrial and commercial area</td>
<td>189</td>
<td>158</td>
</tr>
<tr>
<td>4</td>
<td>Roads/Transportation</td>
<td>92</td>
<td>76</td>
</tr>
<tr>
<td>5</td>
<td>Deep water</td>
<td>808</td>
<td>379</td>
</tr>
<tr>
<td>6</td>
<td>Cropland</td>
<td>152</td>
<td>166</td>
</tr>
<tr>
<td>7</td>
<td>Deciduous forest</td>
<td>1,623</td>
<td>905</td>
</tr>
<tr>
<td>8</td>
<td>Wetland</td>
<td>420</td>
<td>488</td>
</tr>
<tr>
<td>9</td>
<td>Conifer forest</td>
<td>178</td>
<td>126</td>
</tr>
<tr>
<td>10</td>
<td>Grass</td>
<td>85</td>
<td>95</td>
</tr>
<tr>
<td>11</td>
<td>Shallow water</td>
<td>584</td>
<td>469</td>
</tr>
<tr>
<td>12</td>
<td>Reeds</td>
<td>53</td>
<td>19</td>
</tr>
<tr>
<td>Total</td>
<td></td>
<td>4,597</td>
<td>3,085</td>
</tr>
</tbody>
</table>
layer) neurons (from 4 × 4 to 40 × 40) was investigated to examine the effect of neuron density on classification accuracy. The number of iterations for coarse tuning is determined by the number of input samples from the raw waveband images and the number of fine tuning iterations is user-defined. The initial neighborhood radius for lateral interaction is initially set to be large enough to cover the entire output layer. It then decreases over iterations and reaches a minimum value of one at the last iteration.

Additional user-defined parameters included the learning rate for coarse tuning and the gain terms for fine tuning (using the LVQ2 algorithm). Experimentation indicated that generally better results can be obtained with a learning rate ranging from 0.5 to 1.0 for coarse tuning and an extremely small gain term ranging from 0.0001 to 0.0005 for fine tuning. Results also indicated that Kappa reached a maximum with a configuration of 17 × 17 (289) output neurons, with no additional improvement as more neurons were added. A total of 21,357 iterations for the coarse tuning resulted from the sampling rate chosen and 2,758,200 iterations for fine tuning resulted from using the 4,597 training site pixels over 700 epochs. All parameters used in this experiment are listed in Table 2.

Figure 3  The prototype interface for the SOM module in IDRISI. This figure appears in colour in the electronic version of this article and in the plate section at the back of the printed journal.
4.3 Results and Discussion

The classification result from a Maximum Likelihood classifier (MLC) using equal priors was also computed as a reference. Figure 4c shows the result from the SOM with pixels associated with unlabelled neurons left unclassified. In the result, 25.9% of all pixels were found to be associated with unlabelled neurons and thus would be left unclassified without the use of the auxiliary algorithm discussed. However, using the Maximum Likelihood map as a reference, it was determined that 91.8% of these unclassified pixels were at the boundaries of classes and thus potentially mixed. The CVN (Center versus Neighbor) pattern metric in IDRISI was used to determine whether pixels were at the boundaries between classes or were interior pixels. Of the 8.0% that were not at edges, visual examination showed that they were indeed unknown classes, usually occurring in geographically distinct patches. Several patches turned out to be forested wetlands which were not represented by the wetland training pixels and one turned out to be an orchard that was effectively a mixture of trees and grass at the spatial resolution (20 m × 20 m) of the SPOT HRV sensor. Figure 5 illustrates these two examples. From this we concluded that unlabelled neurons do indeed represent unknown classes and mixed pixels, with (in our case) mixed pixels dominating the occurrences.

The accuracy of land cover classification with the above approaches was assessed using total accuracy and the overall Kappa statistic (Liu et al. 2004). The latter, which adjusts for chance agreement, indicates the degree of agreement between two maps. For calculation of both the two assessment methods, Richard and Jia (1998) provide useful guidelines. When conducting a comparison among only those cells that were labelled without use of the auxiliary algorithm by SOM (Figure 4c for SOM and Figure 4d for MLC), the SOM performed better than MLC in terms of total accuracy by 3.3% and overall Kappa by 0.04 (MLC: total accuracy = 85.9% and overall Kappa = 0.83; SOM: total accuracy = 89.2%, overall Kappa = 0.87). Note that in Figure 4d the same areas that were left unclassified by the SOM when the auxiliary algorithm was not used were forced to be unclassified in the Maximum Likelihood map so that the SOM and Maximum Likelihood procedures could be fairly evaluated without

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Input layer neuron number</td>
<td>3</td>
</tr>
<tr>
<td>Output layer neuron number</td>
<td>$17 \times 17$</td>
</tr>
<tr>
<td>Initial neighborhood radius</td>
<td>25.04</td>
</tr>
<tr>
<td>Minimum learning rate</td>
<td>0.5</td>
</tr>
<tr>
<td>Maximum learning rate</td>
<td>1.0</td>
</tr>
<tr>
<td>Minimum gain term</td>
<td>0.0001</td>
</tr>
<tr>
<td>Maximum gain term</td>
<td>0.0005</td>
</tr>
<tr>
<td>Coarse tuning iterations</td>
<td>21,357</td>
</tr>
<tr>
<td>Fine tuning iterations</td>
<td>2,758,200</td>
</tr>
<tr>
<td>Fine tuning algorithm</td>
<td>LVQ2</td>
</tr>
</tbody>
</table>

© 2006 The Authors. Journal compilation © 2006 Blackwell Publishing Ltd
Transactions in GIS, 2006, 10(4)
Figure 4  (a) Color-composite image made from the three multi-spectral images of SPOT HRV showing the study area. Classification result produced by (b) Maximum Likelihood Classifier (MLC); (c) Self-Organizing Feature Map (SOM) with unclassified pixels; (d) Maximum Likelihood with the pixels unclassified by SOM masked out; and (e) SOM using the auxiliary labelling algorithm. This figure appears in colour in the electronic version of this article and in the plate section at the back of the printed journal.
use of the modified SOM procedure. When all pixels were compared between the MLC (Figure 4b: total accuracy = 87.8%, overall Kappa = 0.86) and SOM with the auxiliary labelling procedure (Figure 4e: total accuracy = 91.1% and overall Kappa = 0.89), both classifiers improved by 1.9%, but the margin between them remained the same (Table 3). Thus while our results are consistent with reports of a generally higher classification with SOM over the Maximum Likelihood procedure (Ji 2000), it would appear that with the auxiliary algorithm discussed here, the SOM has only the same level of success (or lack thereof) as Maximum Likelihood in classifying mixed pixels.

Figure 5  (a) Forested wetland in the SPOT HRV 1991 color-composite image; (b) forested wetland in the MassGIS 2001 orthophoto image; (c) orchard in the SPOT HRV 1991 color-composite image; and (d) orchard in the MassGIS 2001 orthophoto image. This figure appears in colour in the electronic version of this article and in the plate section at the back of the printed journal
5 Conclusions

From this evaluation, several important conclusions can be drawn. First, the problem of unlabelled neurons only arises in the context of supervised classification using SOM, and is not associated with the more familiar concept of dead units that affects SOM in unsupervised applications. Second, the unlabelled neurons are associated with unknown and mixed pixels which can represent a substantial portion of the image being classified. Third, the auxiliary algorithm proposed here for assigning classes to unlabelled neurons performs with the same success as that experienced with Maximum Likelihood. In our implementation of SOM in the IDRISI system we allow the user to specify whether unlabelled neurons should be assigned class labels using the algorithm discussed here. In that manner, the user can focus attention on areas of uncertainty in the image. However, the algorithm discussed accommodates cases where a complete enumeration is desired.

Acknowledgments

The support of the University Consortium of Geographic Information Science (UCGIS) and Blackwell (the publishers of Transactions in GIS) are gratefully acknowledged. The authors are grateful to Dr. C.Y. Ji of the Institute of Remote Sensing and GIS Applications of Peking University for his helpful advice. The authors also would like to thank Dr. James Toledano, Mr. Hao Chen, Dr. Honglei Zhu, Mr. Stefano Crema and Mr. Ivan Lucena of Clark Labs for their insightful discussion and suggestions. The procedures covered in this paper have been implemented in the IDRISI GIS and Image Processing System.

References

Bian Z and Zhang X 2000 Pattern Recognition. Beijing, Tsinghua University Press
Tso B and Mather P M 2001 *Classification Methods for Remotely Sensed Data*. New York, Taylor and Francis
Villmann T, Merenyi E, and Hammer B 2003 Neural maps in remote sensing analysis. *Neural Networks* 16: 389–403