Fundamenta Informaticae 84 (2008) 429–442
IOS Press

Change Detection of Remote Sensing Images with Semi-supervised Multilayer Perceptron

Swarnajyoti Patra, Susmita Ghosh
Department of Computer Science and Engineering
Jadavpur University, Kolkata 700 032, India
{patra_swarna,susmita_de}@rediffmail.com

Ashish Ghosh*
Machine Intelligence Unit and Center for Soft Computing Research
Indian Statistical Institute, B.T.Road, Kolkata 700 108, India
ash@isical.ac.in

Abstract. A context-sensitive change-detection technique based on semi-supervised learning with multilayer perceptron is proposed here. In order to take contextual information into account, input patterns are generated considering each pixel of the difference image along with its neighboring pixels. A heuristic technique is suggested to identify a few initial labeled patterns without using ground truth information. The network is initially trained using these labeled data. The unlabeled patterns are iteratively processed by the already trained perceptron to obtain a soft class label. Experimental results, carried out on two multispectral and multitemporal remote sensing images, confirm the effectiveness of the proposed approach.

Keywords: Semi-supervised learning, remote-sensing, change-detection, multitemporal images, neural network

1. Introduction

In remote sensing applications, change-detection is the process of identifying differences in the state of an object or phenomenon by analyzing a pair of images acquired on the same geographical area at

*Address for correspondence: Machine Intelligence Unit, Indian Statistical Institute, 203 B. T. Road, Kolkata 700 108, India

different times [26]. Such a problem plays an important role in many different domains like studies on land-use/land-cover dynamics [7], monitoring shifting cultivations [5], burned area assessment [3], analysis of deforestation processes [14], identification of vegetation changes [6], monitoring of urban growth [19] *etc*. Since all these applications usually require analysis of large areas, development of automatic change-detection techniques became of high relevance in order to reduce the effort required by manual image analysis.

In the literature [1, 2, 5, 10, 12, 16, 22], several supervised and unsupervised techniques for detecting changes in remote-sensing images have been proposed. The supervised methods require the availability of a "ground truth" from which a training set, containing information about the spectral signatures of the changes that occurred in the considered area between the two dates, is generated. The unsupervised approaches perform change-detection without using any additional information, besides the raw images considered. Therefore, from an operational point of view, use of unsupervised techniques becomes mandatory in many remote-sensing applications, as suitable ground truth information is not always available. Besides these two discrete methods of learning, one using a training set (if labeled patterns are available) and the other one without using training set (if labeled patterns are not available), another situation may arise where only a few training patterns are available. The semi-supervised learning comes into play in such a situation. For a problem like change-detection this approach appears to be more promising as we do not have sufficient number of properly labeled patterns but we may be able to identify some labeled patterns which definitely belong to either changed and/or unchanged regions [3, 4].

The change-detection techniques follow three sequential steps [26]: i) pre-processing, ii) image comparison, and iii) image analysis. During pre-processing step two raw images are taken as input and are made compatible using operations like co-registration, radiometric and geometric corrections, and noise reduction [24, 26]. In the next step, two pre-processed images are compared pixel by pixel to generate a third image, called the difference image, where differences between the two acquisitions (images) are highlighted. Once image comparison is performed, the image analysis (change-detection) process can be carried out adopting either context-insensitive or context-sensitive procedures. The most widely used context-insensitive analysis techniques are based on histogram thresholding [3, 18, 23]. Thresholding procedures do not take into account the spatial correlation between neighboring pixels in the decision process. To overcome this limitation, different context-sensitive change-detection procedures are in use [11].

In this article we propose a context-sensitive semi-supervised change-detection technique based on multilayer perceptron (MLP) that automatically discriminates the changed and unchanged pixels of the difference image without using ground truth information. In order to take care of the contextual information, the input patterns are generated considering each pixel of the difference image along with its neighboring pixels. The number of neurons in the input layer is equal to the dimension of the input pattern. As the network discriminates the changed and unchanged pixels of the difference image, the number of neurons in the output layer is two, one for the changed class and the other for the unchanged class. Initially the network is trained using a small set of labeled data. We suggest a technique to identify some labeled patterns automatically with the assumption that a pixel of the difference image belongs to the changed area if the grey values of that pixel and its neighbors pixels are very high and belongs to the unchanged area if their grey values are very low. The unlabeled patterns are iteratively processed by the perceptron to obtain the soft class label.

In order to assess the effectiveness of the proposed technique, we considered two multitemporal data sets corresponding to the geographical areas on the Mexico and the Island of Sardinia, Italy, respectively and compared the results provided by the proposed technique with other context-insensitive and context-sensitive techniques.

The article is organized into six sections. Section 2 provides a brief description of the MLP based semi-supervised technique. Section 3 gives a detailed description of our proposed technique to solve change-detection problems. The data sets used in the experiments are described in Section 4. Experimental results are discussed in Section 5. Finally, in Section 6, conclusions are drawn.

2. Semi-supervised Learning

The task of supervised learning of a classification problem requires correctly labeled training data. However, there are many practical domains in which unlabeled data are abundant but labeled data are expensive, difficult, or computationally hard to generate. The difficulty in obtaining class labels may arise due to incomplete knowledge or limited resources. The use of both labeled and unlabeled data for learning in classification problems [13, 20, 21, 25, 28] has recently been recognized. Semi-supervised learning technique addresses this problem by using a large amount of unlabeled data together with a small number of labeled data to design classifiers. Since semi-supervised learning requires less human effort and gives higher accuracy, it is of great interest both in theory and in practice.

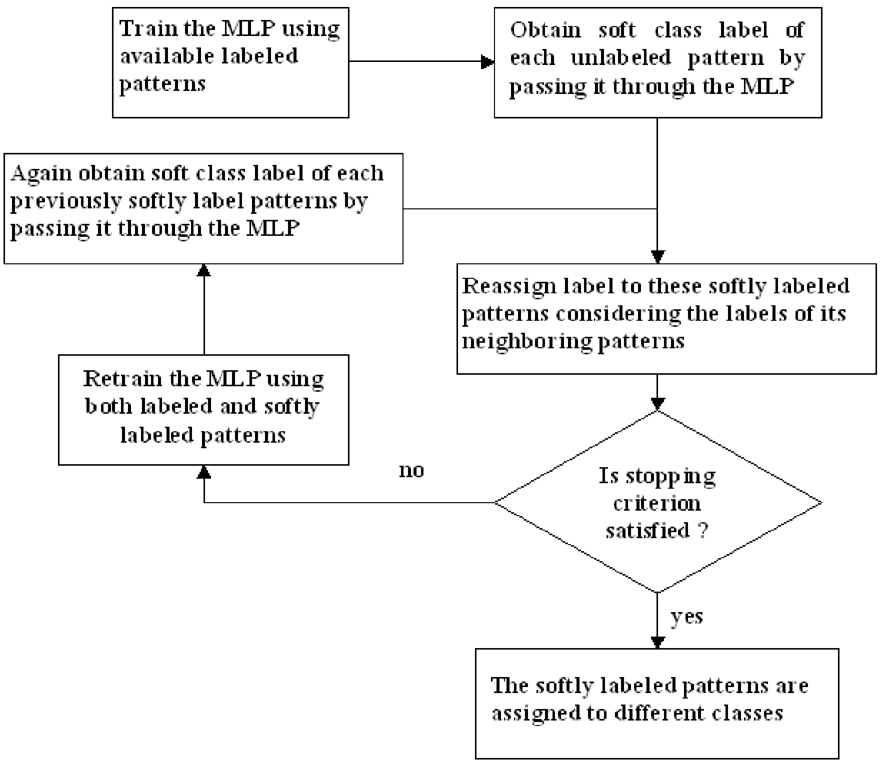


Figure 1. MLP based semi-supervised learning.

As mentioned earlier, in the present article we have used a semi-supervised technique based on MLP for automatic change detection using a large amount of unlabeled patterns and a few labeled patterns.

The labeled patterns may be available or can be generated by exploiting the intrinsic characteristics of the difference image. The semi-supervised technique used here is inspired by the principle used in [27] and is presented in Figure 1.

3. Proposed Change Detection Technique

In this section we propose a context-sensitive change-detection technique that automatically discriminates the changed and unchanged pixels of the difference image. Although we have used a semi-supervised technique, there is no need of human intervention to identify the labeled patterns. The labeled patterns are identified by applying a heuristic technique. In the following subsections we describe the steps involved in the proposed change-detection technique.

3.1. Generation of Input Patterns

To generate the input patterns, we first produce the "difference image" by considering the multitemporal images in which difference between the two considered acquisitions are highlighted. Different mathematical operations can be used to perform image comparison. The most widely used operator is difference. The difference can be applied to i) a single spectral band (Univariate Image Differencing) [26], ii) multiple spectral bands (Change Vector Analysis) [3, 26], iii) vegetation indices (Vegetation Index Differencing) [26] or to other linear (e.g., Tasselled Cap Transformation [9]) or non-linear combinations of spectral bands. Each choice gives rise to a different technique. Among these, the most popular Change Vector Analysis (CVA) technique is used here to generate the difference image. For each pair of corresponding pixels, spectral change vector is computed as the difference in the feature vectors at the two (acquisition) times. Then the pixel values in the difference image are associated with the magnitudes of the spectral change vectors. In some cases, depending on the specific type of changes to be identified, the comparison is made on a subset of the spectral channels.

Let us consider two co-registered and radiometrically corrected γ-spectral band images X_1 and X_2, of size $p \times q$, acquired over the same area at different times T_1 and T_2, and let $D = \{l_{mn}, 1 \leq m \leq p, 1 \leq n \leq q\}$ be the difference image obtained by applying the CVA technique to X_1 and X_2. Then

$$l_{mn} = (int)\sqrt{\sum_{\alpha=1}^{\gamma} (l_{mn}^{\alpha}(X_1) - l_{mn}^{\alpha}(X_2))^2}.$$

Here $l_{mn}^{\alpha}(X_1)$ and $l_{mn}^{\alpha}(X_2)$ are the gray values of the pixels at the spatial position (m, n) in α^{th} band of images X_1 and X_2, respectively.

After producing the difference image, the input patterns are generated corresponding to each pixel in the difference image D, considering its spatial neighborhood system of order d. In the present case 2^{nd} order neighborhood $(d = 2)$ is considered, and the input vectors contain nine components considering the gray value of the pixel and the gray values of its eight neighboring pixels. So the pattern set $U = \{\vec{u}(1), \vec{u}(2), ..., \vec{u}(N)\}$ contains N $(N = p \times q)$ pattern vectors in nine-dimension feature space. The patterns generated by the above technique have two main advantages: (i) each pixel of the difference image is represented in high dimensional feature space (as a pattern) that may reduce the overlapping between changed and unchanged pixels in D and (ii) each pattern takes some contextual information which helps to reduce the effect of noisy pixels in the decision process.

3.2. Description of the MLP

The multilayer perceptron [15] has one input, one output and one or more hidden layers. Neurons/nodes in one layer of the network are connected to all the neurons in next layer. The system has no feedback connection. The network receives a vector input and produces a vector output. Let S be the number of layers in the network and $y_j^r(n)$ denote the output signal of the j^{th} neuron in the r^{th} layer for an input pattern $\vec{u}(n)$, where $n = 1, 2, ..., N$ and w_{ij}^r be the connection strength between the i^{th} neuron in the $(r-1)^{th}$ layer and j^{th} neuron in the r^{th} layer. For an input pattern vector $\vec{u}(n)$, the output value of neuron j in the input layer is defined as $y_j^0(n) = u_j(n)$, which is sent to the first hidden layer as an input signal. A neuron j in the r^{th} $(r \geq 1)$ layer takes input signal $v_j^r(n) = \sum_i y_i^{r-1}(n).w_{ij}^r + w_{0j}^r$, where w_{0j}^r is the connection strength between a fixed unit bias to neuron j, and produces an output $y_j^r(n) = f(v_j^r(n))$. The activation function $f(.)$ is defined as $f(v_j^r(n)) = \frac{1}{1+exp(-v_j^r(n))}$ which scales the activation sigmoidally between 0 to 1. The network is trained using backpropagation algorithm [15] that iteratively adjusts coupling strengths (weights) in the network to minimize the error between the desired pattern and the predicted pattern i.e., minimizing the sum-square error $\sum_{n=1}^{N} \sum_{j=1}^{C} \left(y_j^{S-1}(n) - t_j(n) \right)^2$, where $y_j^{S-1}(n)$ and $t_j(n)$ are the predicted and desired value of the output layer neuron j for input pattern $\vec{u}(n)$, respectively. In the present paper we try to discriminate changed and unchanged pixels of the difference image using MLP. For this purpose, the input patterns are generated corresponding to each pixels of the difference image D considering spatial contextual information as described in the previous section. As the generated patterns have nine features and belong to either changed class or unchanged class, the architecture of the MLP used here has nine input neurons in the input layer and two neurons in the output layer (one for changed class and another for unchanged class). We considered only single hidden layer and the corresponding network architecture is depicted in Figure 2.

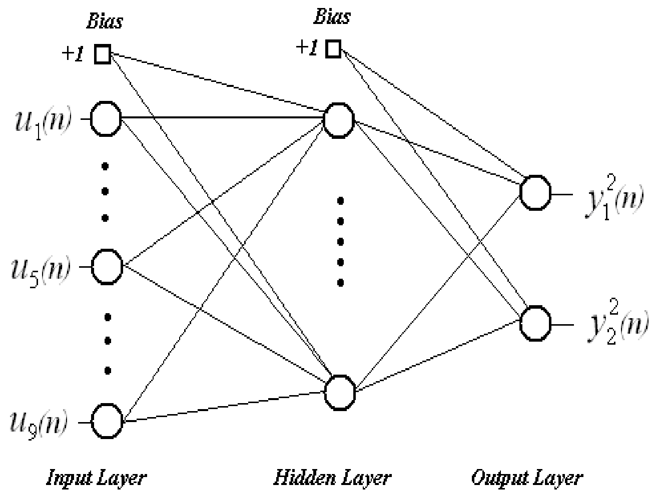


Figure 2. Architecture of the MLP used to solve change-detection problem.

3.3. Labeled Pattern Generation

MLP needs labeled patterns for learning. In this section we suggest a technique to automatically identify some patterns which either belong to changed or unchanged class without using ground truth information of the difference image.

As described in Section 3.1 each pixel in the difference image is found out using the magnitude of the difference between the corresponding feature vectors of the original images. If we consider the properties of the difference image, a reasonable assumption could be as follows: a pixel with small gray value exhibits a high probability of being in unchanged class, whereas a pixel with large gray value has a high probability of being in changed class. As component values of the generated pattern vectors contain gray values of pixels in the difference image, the patterns whose component values are very low belong to unchanged class and the patterns whose component values are very high belong to changed class. To identify these patterns automatically, K-means (K=2) clustering algorithm [8] is applied. Let lc and uc be the two centroids obtained by K-means algorithm in nine-dimensional feature space. We also considered two other points $lb\,(0, ..., 0)$, the possible minimum component values of the patterns which is near to lc and $ub\,(255, ..., 255)$, the possible maximum component values of the patterns which is near to uc in the same feature space (see Figure 3, patterns are presented in two-dimensional feature space). A pattern can be assigned to the unchanged class if it is inside the hypersphere whose center is at lb and radius is the distance between lb and lc or it can be assigned to changed class if it is inside the hypersphere whose center is at ub and radius is the distance between ub and uc **else** it is considered as unlabeled.

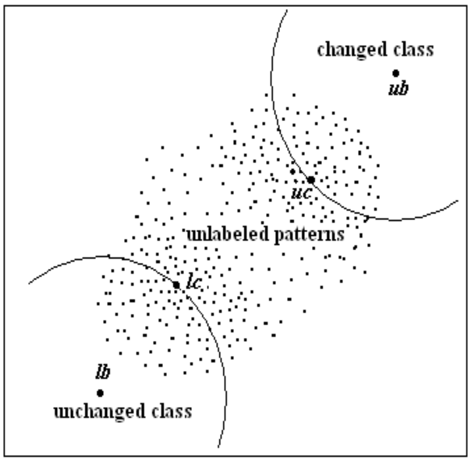


Figure 3. Labeled and unlabeled patterns presented in two-dimensional feature space.

The pattern set U is represented as $U = \{(\vec{u}(n), \vec{t}(n)),\ n = 1, 2, ..., N\}$, where $\vec{u}(n)$ is the n^{th} input pattern vector and $\vec{t}(n)$ is the target vector of the corresponding input pattern. The target vector $\vec{t}(n)$ where $\vec{t}(n) = \{[t_1(n), t_2(n)] \mid\ t_i(n) \in (0, 1),\ \forall n\}$ represents the changed class (when $\vec{t}(n) = [1, 0]$), unchanged class (when $\vec{t}(n) = [0, 1]$) and unlabeled pattern (when $\vec{t}(n) = [0, 0]$).

3.4. Assignment of Label Value to Unlabeled Patterns

Let us first train the network using the labeled patterns that are automatically identified by a method described in Section 3.3. The decision classes are considered as fuzzy sets [17] and we assume that the network's output values provide degree of membership to the fuzzy sets. Let us suppose the n^{th} input pattern is presented to the network. The membership value $\mu_j(\vec{u}(n))$ of the n^{th} input pattern $\vec{u}(n)$ to the j^{th} fuzzy set is then given by the output $y_j^2(\vec{u}(n))$ of the j^{th} $(j = 1, 2)$ output node of the network. The contrast within the set of membership values $\mu_j(\vec{u}(n))$ is then increased [27] by using the following formula.

$$\mu_j(\vec{u}(n)) \;= \begin{cases} 2[\mu_j(\vec{u}(n))]^2, \; 0 \leq \mu_j(\vec{u}(n)) \leq 0.5 \\ 1 - 2[1 - \mu_j(\vec{u}(n))]^2, \; 0.5 < \mu_j(\vec{u}(n)) \leq 1.0. \end{cases} \tag{1}$$

Now we find out the k_{nn} nearest neighbors for each unlabeled pattern. Finding out the k_{nn} nearest neighbors considering all patterns is a time consuming task. To reduce the time complexity, instead of considering all the patterns we find out k_{nn} nearest neighbors for an unlabeled pattern corresponding to a pixel of the difference image by considering only the patterns generated by its surrounding pixels. Let M^n be the set of indices of the k_{nn} nearest neighbors of the unlabeled pattern $\vec{u}(n)$. Then the target vector $\vec{t}(n)$ for the unlabeled pattern $\vec{u}(n)$ is computed by

$$\vec{t}(n) = \left[\frac{\sum_{i \in M^n} \mu_1(\vec{u}(i))}{k_{nn}}, \frac{\sum_{i \in M^n} \mu_2(\vec{u}(i))}{k_{nn}} \right]. \tag{2}$$

For labeled patterns $\vec{t_j}(n) = 0 \; or \; 1$, for $j = 1, 2$.

3.5. Learning Algorithm

The regions of lower pattern density usually separate the classes. Therefore, decision boundary between the classes should be located in such lower pattern density regions. Initially, the network is trained using labeled patterns only. The output value to each unlabeled pattern is then obtained by passing it through the network. Then the target values (soft class label) of these patterns are estimated using equations (1) and (2) consecutively. Next, the network is retrained using both the labeled and the softly labeled patterns and again the target values of the softly labeled patterns are re-estimated using equations (1) and (2) consecutively. To observe the stabilization of network output, after completion of each epoch, we calculate the sum of square error by using the following formula

$$\sum_{n=1}^{N} \sum_{j=1}^{2} \left(y_j^2(n) - t_j(n)\right)^2. \tag{3}$$

The re-estimation and re-training steps are iterated until the sum of square error obtained using equation (3) does not change much in consecutive epoch (becomes stable) or the number of epochs exceeds some given number. The corresponding learning algorithm is given in Table 1.

Table 1. Learning algorithm of the network

Step 1:	Train the network using labeled patterns only.
Step 2:	Assign soft class labels to each unlabeled patterns by using equations (1) and (2) consecutively.
Step 3:	Train the network using both the labeled and the softly labeled patterns.
Step 4:	Re-estimate the target values of the softly labeled patterns by using equations (1) and (2) consecutively.
Step 5:	If the sum of square error obtained from equation (3) becomes stable or the number of epochs exceeds some given number then goto Step 6; else goto Step 3.
Step 6:	Stop.

4. Description of Data Sets

In order to carry out an experimental analysis aimed at assessing the effectiveness of the proposed approach, we considered two multitemporal data sets corresponding to geographical areas of Mexico and Island of Sardinia, Italy. A detailed description of each data set is given below.

4.1. Data Set of Mexico Area

The first data set used for the experiment is made up of two multispectral images acquired by the Landsat Enhanced Thematic Mapper Plus (ETM+) sensor of the Landsat-7 satellite over an area of Mexico on 18th April 2000 and 20th May 2002. From the entire available Landsat scene, a section of 512×512 pixels has been selected as test site. Between the two aforementioned acquisition dates a fire destroyed a large portion of the vegetation in the considered region. Figures 4(a) and 4(b) show channel 4 of the 2000 and 2002 images, respectively. In order to be able to make a quantitative evaluation of the effectiveness of the proposed approach, a reference map was manually defined (see Figure 4(d)) according to a detailed visual analysis of both the available multitemporal images and the difference image (see Figure 4(c)). Different color composites of the above mentioned images were used to highlight all the portions of the changed area in the best possible way. Experiments were carried out to produce, in an automatic way, a change-detection map as similar as possible to reference map that represents the best result obtainable with a time consuming procedure.

Analysis of the behavior of the histograms of multitemporal images did not reveal any significant difference due to light and atmospheric conditions at the acquisition dates. Therefore, no radiometric correction algorithm was applied. The 2002 image was registered with the 2000 one using 12 ground control points. The procedure led to a residual average misregistration error on ground control points of about 0.3 pixels.

4.2. Data Set of Sardinia Island, Italy

The second data set used in the experiment is composed two multispectral images acquired by the Landsat Thematic Mapper (TM) sensor of the Landsat-5 satellite in September 1995 and July 1996. The test site is a section of 412×300 pixels of a scene including lake Mulargia on the Island of Sardinia (Italy). Between

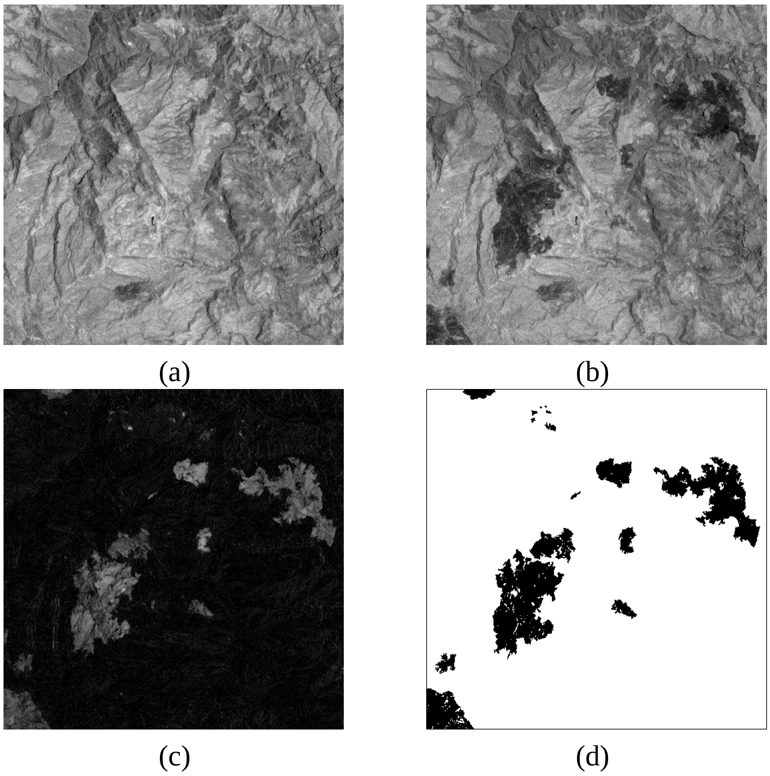


Figure 4. Image of Mexico area. (a) Band 4 of the Landsat ETM+ image acquired in April 2000, (b) band 4 of the Landsat ETM+ image acquired in May 2002, (c) corresponding difference image generated by CVA technique, and (d) reference map of the changed area.

the two aforementioned acquisition dates the water level in the lake increased (see the lower central part of the image). Figures 5(a) and 5(b) show channel 4 of the 1995 and 1996 images. As done for the Mexico data set, in this case also a reference map was manually defined (see Figure 5(d)) according to a detailed visual analysis of both the available multitemporal images and the difference image (see Figure 5(c)). Like the earlier image, in this case also the histograms did not show any significant difference and therefore, no radiometric correction algorithms were applied on the multitemporal images. The images were co-registered with 12 ground control points resulting in an average residual misregistration error of about 0.2 pixels.

5. Description of the Experiments

In order to establish the effectiveness of the proposed technique, the present experiment compares the change-detection result provided by the proposed method with a context-insensitive Manual Trial and Error Thresholding (MTET) technique [2], the K-means clustering [8] technique and a context-sensitive techniques presented in [3] based on the combined use of the EM algorithm and Markov Random Fields (MRF) (we refer to it as EM+MRF technique). The MTET technique generates a minimum error change-detection map under the hypothesis of spatial independence among pixels by finding a minimum error

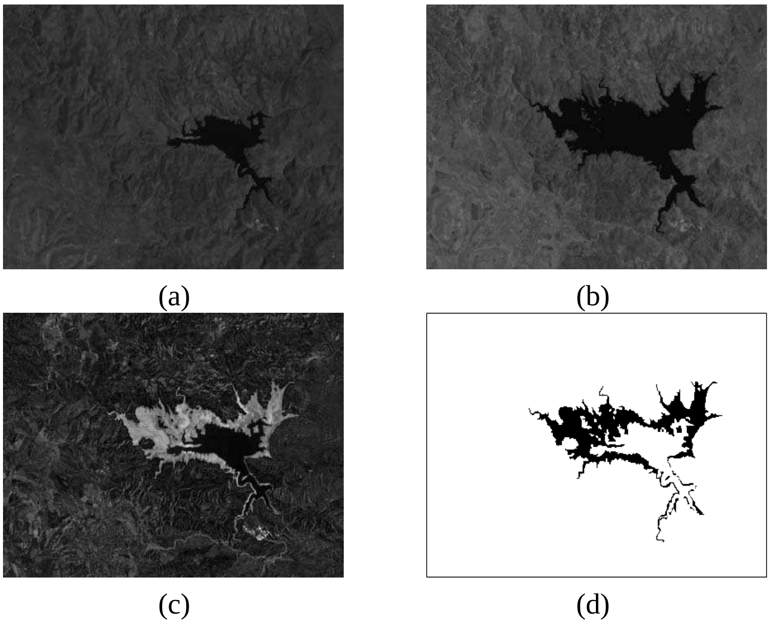


Figure 5. Image of Sardinia Island, Italy. (a) Band 4 of the Landsat TM image acquired in September 1995, (b) band 4 of the Landsat TM image acquired in July 1996, (c) difference image generated by CVA technique using bands 1, 2, 4, & 5; and (d) reference map of the changed area.

decision threshold for the difference image. The minimum error decision threshold is obtained by computing change-detection errors (with the help of the reference map) for all values of the decision threshold. K-means clustering algorithm is applied on the generated patterns with $K = 2$. Comparisons were carried out in terms of both overall change-detection error and number of false alarms (i.e., unchanged pixels identified as changed ones) and missed alarms (i.e., changed pixels categorized as unchanged ones).

In the present experiment the architecture of MLP is $9 : 8 : 2$ i.e., the network has 9 input neurons, 8 hidden neurons in a single hidden layer and 2 output neurons. To find out k_{nn} nearest neighbors for each input pattern we have taken 50×50 window and the value of k_{nn} is taken as 8. The network is assumed to converge when the sum-square error becomes constant with increasing number of epochs.

5.1. Result Analysis: Mexico Data Set

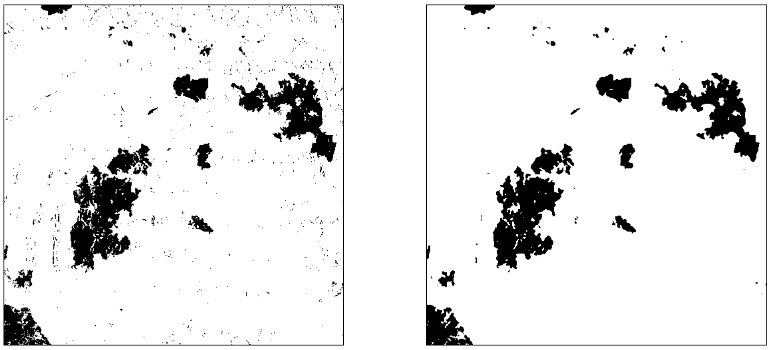
First of all we performed some trials in order to determine the most effective spectral bands for detecting the burned area in the considered data set. On the basis of the results of these trials, we found that band 4 is more effective to locate the burned area. Hence we used the difference image generated by spectral band 4 only.

Table 2 shows that the overall error obtained by the proposed techniques based on MLP is much smaller than that incurred by the context-insensitive MTET technique. Concerning the error typology, the proposed technique resulted 3305 pixels as overall error (2602 missed alarms and 703 false alarms), whereas the MTET procedure involved 4591 pixels as overall error (2404 missed alarms and 2187 false alarms). Figure 6 depicts the change-detection maps. A visual comparison points out that the proposed

Table 2. Overall error, missed alarms and false alarms resulting by MTET, K-means, EM+MRF, and proposed techniques (Band 4, Mexico data set).

Techniques	**Missed alarms**	**False alarms**	**Overall error**
MTET	2404	2187	4591
K-mean	3108	665	3773
EM+MRF ($\beta = 1.5$)	946	2257	3203
Proposed	2602	703	3305

approach generates a more smooth change-detection map compared to the MTET procedure. From Table 2 one can also see that the proposed MLP based technique generates better change-detection results (overall error 3305 pixels) than the result (overall error 3773 pixels) produced by K-means technique. The best result (overall error 3202 pixels) obtained by existing EM+MRF technique when the parameter β of MRF [3] was set to 1.5 (this value was defined manually and corresponds to the minimum possible error) is also close to the result obtained by the proposed technique.



(a) (b)

Figure 6. Change-detection maps obtained for the data set related to the Mexico area using (a) MTET technique, and (b) proposed technique.

5.2. Result Analysis: Sardinia Island Data Set

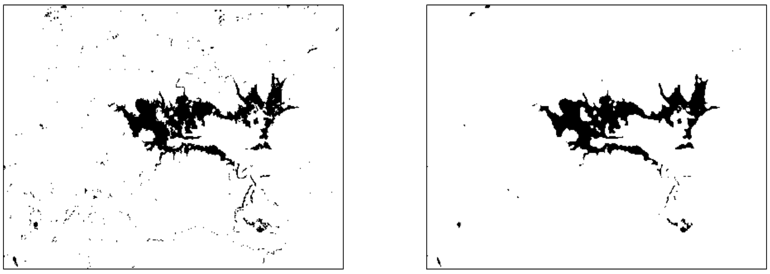
We applied the CVA technique on spectral bands 1, 2, 4, and 5 of the two multispectral images, as preliminary experiments show that the above channels contain useful information on the changes of water body. Note that for this image the water level in the lake increased during the two dates of acquisition.

The change-detection results obtained using different techniques are shown in Table 3. The results obtained by the proposed context-sensitive technique is better than the results produced by the MTET procedure. In greater detail, the overall error produced by the proposed technique is 1597 pixels whereas the overall error produced by the MTET technique is 1890 pixels. For visual comparison Figure 7 shows the change-detection maps produced by the two techniques. From the table we see that the proposed

Table 3. Overall error, missed alarms and false alarms resulting by MTET, K-means, EM+MRF, and proposed techniques (Sardinia Island data set).

Techniques	**Missed alarms**	**False alarms**	**Overall error**
MTET	1015	875	1890
K-mean	637	1881	2518
EM+MRF ($\beta = 2.2$)	592	1108	1700
Proposed	1294	303	1597

technique produced much better result than the K-means technique. Using the same input domain, the proposed technique generates overall error of 1597 pixels (1294 missed alarms and 303 false alarms) whereas the K-means technique produced overall error of 2518 pixels (637 missed alarms and 1881 false alarms). It is also seen that the proposed context-sensitive MLP based technique provides better accuracy than the best result (overall error 1700 pixels) yielded by the existing context-sensitive EM+MRF technique with $\beta = 2.2$.



(a) (b)

Figure 7. Change-detection maps obtained for the data set related to the Island of Sardinia, Italy by using (a) MTET technique, and (b) proposed technique.

6. Discussion and Conclusions

In this paper, a semi-supervised and automatic context-sensitive technique for change-detection in multitemporal images is proposed. The technique discriminates the changed and unchanged pixels in the difference image by using a multilayer perceptron. The number of neurons in the input layer is equal to the dimension of the input patterns and the number of neurons in the output layer is two. The input patterns are generated considering each pixel in the difference image along with its neighboring pixels, in order to take into account the spatial contextual information. On the basis of the characteristics of these input patterns, a heuristic technique is suggested to automatically identify a few input patterns that have very high probability to belong either to changed or to unchanged class. Depending on these labeled patterns and assuming that the lower pattern density region can act as separator between the classes, a semi-supervised learning algorithm based on MLP is used to assign the soft class label for each unlabeled pattern.

The presented technique shows the following advantages: (i) it is distribution free, i.e., like EM+MRF model presented in [3] it does not require any explicit assumption on the statistical model of the distributions of classes of changed and unchanged pixels, (ii) it does not require human efforts to identify the labeled patterns, i.e., the labeled patterns are identified heuristically without using any ground truth information. Like other semi-supervised cases, the time requirement of this technique is little more. It is worth noting that for the considered kind of application it is not fundamental to produce results in real time.

Experimental results obtained on different real multitemporal data sets confirm the effectiveness of the proposed approach. The presented technique significantly outperforms the standard optimal-manual MTET technique and K-means technique. The proposed technique also provides comparable overall change-detection error to the one achieved with the context-sensitive EM+MRF technique.

Acknowledgements

Authors would like to thank the Department of Science and Technology, Government of India and University of Trento, Italy, the sponsors of the ITPAR program and Prof. L. Bruzzone, the Italian collaborator of this project, for providing the data.

References

[1] Alberti, M., Weeks, R., Coe, S.: Urban land cover change analysis in central puget sound, *Photogram. Eng. Remote Sensing*, **70**(9), 2004, 1043–1052.

[2] Bazi, Y., Bruzzone, L., Melgani, F.: An unsupervised approach based on the generalized Gaussian model to automatic change detection in multitemporal SAR images, *IEEE Trans. Geosci. Remote Sensing*, **43**(4), 2005, 874–887.

[3] Bruzzone, L., Prieto, D. F.: Automatic analysis of the difference image for unsupervised change detection, *IEEE Trans. Geosci. Remote Sensing*, **38**(3), 2000, 1171–1182.

[4] Bruzzone, L., Prieto, D. F.: An adaptive semiparametric and context-based approach to unsupervised change detection in multitemporal remote-sensing images, *IEEE Trans. Image Processing*, **11**(4), 2002, 452–466.

[5] Bruzzone, L., Serpico, S. B.: An iterative technique for the detection of land-cover transitions in multitemporal remote-sensing images, *IEEE Trans. Geosci. Remote Sensing*, **35**(4), 1997, 858–867.

[6] Chavez, P. S., MacKinnon, D. J.: Automatic detection of vegetation changes in the southwestern United States using remotely sensed images, *Photogram. Eng. Remote Sensing*, **60**(5), 1994, 1285–1294.

[7] Cihlar, J., Pultz, T. J., Gray, A. L.: Change detection with synthetic aperture radar, *Int. J. Remote Sensing*, **13**(3), 1992, 401–414.

[8] Duda, R. O., Hart, P. E., Stork, D. G.: *Pattern Classification*, Singapore: John Wiley, 2001.

[9] Fung, T.: An assessment of TM imagery for land-cover change detection, *IEEE Trans. Geosci. Remote Sensing*, **28**(4), 1990, 681–684.

[10] Ghosh, S., Bruzzone, L., Patra, S., Bovolo, F., Ghosh, A.: A context-sensitive technique for unsupervised change detection based on Hopfield-type neural networks, *IEEE Trans. Geosci. Remote Sensing*, **45**(3), 2007, 778–789.

[11] Ghosh, S., Patra, S., Ghosh, A.: A neural approach to unsupervised change detection of remote-sensing images, in: *Speech, Audio, Image and Biomedical Signal Processing using Neural Networks* (B. Prasad, S. R. M. Prasanna, Eds.), Springer-Verleg, Berlin, 2008, 239–264.

[12] Gopal, S., Woodcock, C.: Remote sensing of forest change using artificial neural networks, *IEEE Trans. Geosci. Remote Sensing*, **34**(2), 1996, 398–404.

[13] Gosselin, P. H., Cord, M.: Feature-based approach to semi-supervised similarity learning, *Pattern Recognition*, **39**(10), 2006, 1839–1851.

[14] Hame, T., Heiler, I., Miguel-Ayanz, J. S.: An unsupervised change detection and recognition system for forestry, *Int. J. Remote Sensing*, **19**(6), 1998, 1079–1099.

[15] Haykin, S.: *Neural Networks: A Comprehensive Foundation*, Singapore: Pearson Education, Fourth Indian Reprint, 2003.

[16] Kasetkasem, T., Varshney, P. K.: An image change detection algorithm based on Markov random field models, *IEEE Trans. Geosci. Remote Sensing*, **40**(8), 2002, 1815–1823.

[17] Klir, G. S., Yuan, B.: *Fuzzy Sets and Fuzzy Logic - Theory and Applications*, Prentice-Hall, Englewood Cliffs, 1995.

[18] Melgani, F., Moser, G., Serpico, S. B.: Unsupervised change-detection methods for remote sensing data, *Opt. Eng.*, **41**, 2002, 3288–3297.

[19] Merril, K. R., Jiajun, L.: A comparison of four algorithms for change detection in an urban environment, *Remote Sens. Environ.*, **63**(2), 1998, 95–100.

[20] Miller, D. J., Uyar, H. S.: A mixture of experts classifier with learning based on both labeled and unlabeled data, in: *Advances in Neural Information Processing Systems* (M. C. Mozer, M. I. Jordan, T. Petsche, Eds.), vol. 9, MIT Press, 1997, 571–577.

[21] Nigam, K., McCallum, A. K., Thrun, S., Mitchell, T.: Text classification from labeled and unlabeled documents using EM, *Machine Learning*, **39**, 2000, 103–134.

[22] Patra, S., Ghosh, S., Ghosh, A.: Unsupervised change detection in remote-sensing images using modified self-organizing feature map neural network, *Int. Conf. on Computing: Theory and Applications (ICCTA-2007), Kolkata, India*, IEEE Computer Society Press, 2007, 716–720.

[23] Radke, R. J., Andra, S., Al-Kofahi, O., Roysam, B.: Image change detection algorithms: A systematic survey, *IEEE Trans. Image Processing*, **14**(3), 2005, 294–307.

[24] Richards, J. A., Jia, X.: *Remote Sensing Digital Image Analysis*, 4th ed. Berlin: Springer-Verlag, 2006.

[25] Seeger, M.: Learning with labeled and unlabeled data, *Technical report, University of Edinburgh*, 2001.

[26] Singh, A.: Digital change detection techniques using remotely sensed data, *Int. J. Remote Sensing*, **10**(6), 1989, 989–1003.

[27] Verikas, A., Gelzinis, A., Malmqvist, K.: Using unlabelled data to train a multilayer perceptron, *Neural Processing Letters*, **14**, 2001, 179–201.

[28] Yoon, S. Y., Lee, S. Y.: Training algorithm with incomplete data for feed-forward neural networks, *Neural Processing Letters*, **10**, 1999, 171–179.