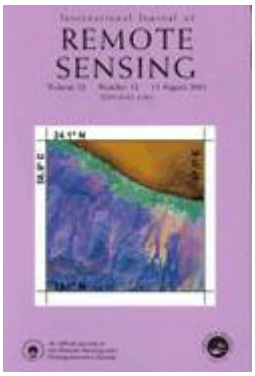


A Novel Self-Organizing Neuro-Fuzzy Multilayered Classifier for Land Cover Classification of a VHR Image

Journal:	*International Journal of Remote Sensing*
Manuscript ID:	TRES-PAP-2007-0075
Manuscript Type:	Research Paper
Date Submitted by the Author:	07-Feb-2007
Complete List of Authors:	Mitrakis, Nikolaos; Aristotle University of Thessaloniki, Electrical and Comp. Engineering Topaloglou, Charalampos; Aristotle University of Thessaloniki, Agronomy Alexandridis, Thomas; Aristotle University of Thessaloniki, Agronomy Theocharis, John; Aristotle University of Thessaloniki, Electrical and Comp. Engineering Zalidis, George; Aristotle University of Thessaloniki, Agronomy
Keywords:	FUZZY CLASSIFICATION, IKONOS, MULTISPECTRAL CLASS, THEMATIC MAPPING
Keywords (user defined):	LAND COVER CLASSIFICATION, HABITAT CLASSIFICATION, CROP CLASSIFICATION

A novel self-organizing neuro-fuzzy multilayered classifier for land cover classification of a VHR image

N. E. MITRAKIS†, C. A. TOPALOGLOU‡, T. K. ALEXANDRIDIS‡, J. B. THEOCHARIS*† and G. C. ZALIDIS§

†Department of Electrical and Computer Engineering, Faculty of Engineering, Aristotle University of Thessaloniki, Thessaloniki, Greece

‡ Lab of Remote Sensing and GIS, Faculty of Agronomy, Aristotle University of Thessaloniki, Thessaloniki, Greece

§Lab of Applied Soil Science, Faculty of Agronomy, Aristotle University of Thessaloniki, Thessaloniki, Greece

A novel self-organizing neuro-fuzzy multilayered classifier (SONeFMUC) is suggested in this paper, with feature selection capabilities, for the classification of an IKONOS image. The structure of the proposed network is developed in a sequential fashion using the group method of data handling (GMDH) algorithm. The node models, regarded as generic classifiers, are represented by fuzzy rule-based systems, combined with a fusion scheme. A data splitting mechanism is incorporated to discriminate between correctly classified and ambiguous pixels. The classifier was tested on the wetland of international importance of Lake Koronia, Greece, and the surrounding agricultural area. To achieve higher classification accuracy, the image was decomposed into two zones, namely, the wetland and the agricultural zone. Apart from the initial bands, additional input features are considered: textural features, IHS and tasseled cap transformation. To assess the quality of the suggested model, SONeFMUC is compared with a maximum likelihood classifier (MLC). The experimental results show that the SONeFMUC exhibited superior performance than MLC, providing lower confusion of the dominant classes in both zones. Especially in the wetland zone, an overall accuracy of 89.5% was attained.

Keywords: Fuzzy classifier; Neural networks; Classifiers fusion; Land cover classification; IKONOS

1. Introduction

Land cover classification of remotely sensed images has attracted considerable research interest over the past decades. Along the numerous applications in the field, several problems have been reported which reduce the accuracy and reliability of the resulting thematic maps. The presence of mixed pixels, the resolution of the acquired images, the reliability of training data, the number of classes and the high degree of spectral overlapping between the classes are key reasons for achieving low classification accuracies.

In order to tackle the problems encountered in land cover image classification, the research community has turned primarily to two major areas of interest. The former issue involves the enhancement of features used by the classification algorithm, as applied to the satellite image. Although original bands of satellite sensors remain the basic source of information in multispectral image classification, advanced features such as topographic information (Richards *et al.* 1982), tasseled cap features (Oetter *et al.* 2001), textural analysis (Haralick *et al.* 1973, Haralick and Shapiro 1992) and wavelet decomposition (Dekker 2003) have been developed in order to reduce the overlapping of the classes in the original feature space. However, the use of new features results in complex models due to the large dimension of the feature space. In order to arrive to the appropriate feature set, several techniques make use of pre-processing methods such as PCA (Li and Yeh 1998) and the time consuming method of trial-and-error (Shackelford and Davis 2003).

Secondly, more sophisticated classifiers are proven to be a major factor of improving the classification results. The first type of classifiers used mainly statistical parameters (Thomas *et al.* 1987), considered as 'hard' classifiers, such as the maximum likelihood. However, the assumption of the normal distribution of the data is a major drawback of such classifiers.

The remarkable achievements in the development of fuzzy classifiers, regarded as 'soft' classifiers, provide a fruitful approach (Wang 1990, Bárdossy and Samaniego 2002). These classification techniques were found to be more appropriate in tackling the mixed pixels problem (Bastin 1997, Tso and Mather 2001), since they take into consideration the ambiguities concerning the correct class to which a pixel belongs. However, Foody (1999) proposed that in order to resolve the mixed pixel problem, a continuum of classification fuzziness should be defined, i.e. not only the classifier but also the training and testing stages should be fuzzy.

Another promising type of classifiers is derived from the theory of neural networks (Benediktsson *et al.* 1993, Foody 1995, Kavzoglu and Mather 2003). Neural networks are capable

of dealing with complex problems, with high degree of overlapping between the classes by conducting a non-linear transformation of the original feature space through the layers of the network. Along this direction, a great variety of neural network architectures and training algorithms have been reported in the literature, with productive results (Atkinson and Tatnall 1997, Keramitsoglou *et al.* 2005). Nevertheless, most of the classifiers of this type require the definition of a large number of parameters, such as the number of hidden layers, the number of nodes, the range of the initial weights and the number of training data, which is difficult to be decided. To cope with this problem, Kavzoglu and Mather (2003) proposed a set of guidelines by conducting an extensive research of suggestions reported in the literature. Furthermore, during the past years considerable research is focused on the development of classifiers combining fuzzy logic and neural networks, thus resulting in superior classification methods (Carpenter *et al.* 1997, Lin *et al.* 2000) on remote sensing problems. An alternative approach is to combine different classifier types (Kuncheva *et al.* 2001), by exploiting the best attributes of every classifier. Particularly, in land cover classification, Giacinto and Roli (1997) proposed a method of combining the classification results of statistical and neural network classifiers using a modified K-nearest-neighbour rule as a metaclassifier. Furthermore, Briem *et al.* 2002 studied the application of boosting, bagging and consensus theory to derive multiple classifiers which outperformed the single classifiers on multisource remote sensing data.

The potential, though, of using a cascaded architecture or a sequential combination of individual classifiers has not been exploited enough, since most of these architectures are based on a hierarchical combination of classifiers (Giacinto and Roli 1997, Kumar *et al.* 1997, Briem *et al.* 2002). Wilkinson *et al.* (1995) proposed a voting/rejection approach, where a maximum likelihood classifier was used in parallel with a neural network in the first stage. In the second stage, an independent neural network was trained to classify the ambiguous pixels of the first stage. Benediktsson and Kanellopoulos (1999) modified the voting/rejection of Wilkinson *et al.* 1995 by using a multisource classifier based on consensus theory instead of a maximum likelihood classifier, and used the decision boundary feature extraction method (DBFE) as a tool of reducing the large dimensionality of the input space for hyperspectral data. Although the results showed that the logarithmic opinion pool (LOGP) classifier achieved the best performance, the authors stressed out the promising attributes and results of the voting/rejection approach. However, the neural network in the second stage of the algorithm should be supplied with a large number of training data in order to achieve satisfying classification performance. Hence, the efficiency of the method

is highly dependent on the degree of agreement of the classifiers in the first stage of the classification algorithm (Petrakos *et al.* 2001). Moreover, the use of a feature extraction algorithm as a pre-processing tool increased the computational cost.

The aim of this paper is to develop a novel self-organizing neuro-fuzzy multilayered classifier (SONeFMUC) with feature selection attributes, gradually evolved in a self-organizing manner, using the group method of data handling (GMDH) algorithm (Ivakhnenko 1968). Specific objectives include (i) the application of the SONeFMUC to a very high resolution IKONOS image using the initial bands and advanced features for land cover classification of a protected wetland and its surrounding agricultural area, and (ii) the validation of the SONeFMUC through its comparison with a maximum likelihood classifier.

2. Materials and methods

2.1 Study area

Lake Koronia is located in a tectonic depression in northern Greece (40' 41" N, 23' 09" E). Its watershed covers an area of 780 km^2, and formerly it drained eastward into Lake Volvi, then the sea. Climate of the region is transitional between Mediterranean and temperate. Mean annual precipitation is 455 mm, with a seasonal peak in December and a minimum in August (Mitraki *et al.* 2004).

The lake-wetland ecosystem is surrounded by an intensively cultivated agricultural area (figure 1). The dominant agricultural crops are maize, alfalfa, and cereals. There is no exploitation of surface water, and the only source of fresh water for irrigation, industrial and urban use is through groundwater resources. Irrigated agriculture is an important economic activity in the area, but recent development of numerous pump wells has resulted in the depletion of the aquifer, and a subsequent decrease in the lake's water level. The industrial sector has also increased in the last decade, discharging untreated effluents in the lake from fabric dying, food and dairy processing activities (Zalidis *et al.* 2004).

Due to the above mentioned pressures, Lake Koronia became progressively more eutrophic, especially after the early 1990's, and currently is hypertrophic (Mitraki *et al.* 2004). Along with the drastic alteration in the water level that reached a decrease of 80%, the natural ecosystem has suffered a severe degradation. There has been a significant loss of volume and habitat

heterogeneity in the lake and wetland. Emergent macrophyte community (dominated by *Phragmites australis*) has shifted lakewards and expanded on recently exposed lake bed (Alexandridis *et al.* 2006).

In recognition of its ecological importance, and to prevent further degradation, the lake-wetland system of Lakes Koronia - Volvi is protected by a number of legal and binding actions: it is a Wetland of International Importance according to Ramsar Convention (site code 57, area 163.88 km^2), a Special Protected Area designated by the implementation of European Directive 79/409/EEC (site code GR1220009, area 156.71 km^2), and a Site of Community Importance following the implementation of European Habitat Directive 92/43/EEC (site code GR1220001, area 269.47 km^2). The national and local relevant authorities have responded with the identification and mapping of habitats (Hellenic Ministry of Environment 2001), the compilation of the Master Plan for the restoration of Lake Koronia, and the Revised Restoration Plan for Lake Koronia (Zalidis *et al.* 2004). However, continuous monitoring of natural and agricultural environment is required according to the proposed management actions, and to fulfil the obligations to the international and European legislation.

Figure 1.

2.2 Dataset used

An IKONOS bundle image with 1 m spatial resolution in panchromatic and 4 m in multispectral (three visible and one near-infrared), covered 134 km^2 of the study area was used. The image was acquired on 7 August 2005, was clear from clouds and was acquired at nadir view angle in order to minimize noise reflectance from topographic effects.

The digital elevation model (DEM) of the area was used for the orthocorrection of the satellite image. It was produced at 5 m pixel size by contours with height interval 4 m which were digitized from topographic maps of scale 1:5000 using the ANUDEM interpolation method (Hutchinson 1991).

An extensive field survey was conducted at the first days of September 2005 to identify land cover classes which referred mainly to the agricultural and wetland area, and collect training and testing samples for the image classification and its accuracy respectively. Using a GPS receiver embedded in a palm top, 3920 locations were selected at regular intervals along the agricultural road network. The land cover on these locations was identified by visual inspection and was

afterwards labelled in 13 classes. The classification scheme included six crop types, five wetland habitats (following Annex I of Habitats Directive 92/43/EEC) and two ancillary land cover types (following the CORINE Land Cover nomenclature). The sampled points were separated into two different sets using the random stratified method; the training set (70%) and the test set (30%).

2.3 Manipulation of the dataset

IKONOS image was orthorectified in the Greek Geodetic Reference System (GGRS '87) with the use of DEM and ground control points which were collected from black and white photomaps of 1 m pixel size. The RMS error was 0.92 m and 2.41 m for panchromatic and multispectral image, respectively. Afterwards, the pan-sharpen image was produced using the forward-reverse principal components transforms (Chavez *et al.* 1991) with the panchromatic image, replacing the first principal component. This new image was useful in order to evaluate and minimize the bias in the samples which was selected with the field work. Atmospheric correction wasn't applied since it has little effect on classification accuracy when single date of remotely sensed data are going to be classified, as long as the training data from the image to be classified have the same relative scale (corrected, uncorrected) (Kawata *et al.*1990, Song *et al.* 2001).

Moreover, advanced features from the multispectral image were calculated using the initial four bands of the image. These features can be categorized in two groups, namely, textural and spectral features.

Textural analysis using the grey level co-occurrence matrices (GLCM) (Haralick and Shapiro 1992) comprise a common practice in land cover image classification (Lin *et al.* 2000), in order to decrease the degree of overlapping between the various types of classes. The image is raster scanned with sliding windows of $M \times M$ dimensions. A grey level co-occurrence matrix for each window is calculated indicating how often different grey levels, i, j, occur with a specific direction, $\theta = 0^0, 45^0, 90^0, 135^0$, and distance d between the pixel centres. Assuming that G grey levels occur within the image, a $G \times G$ matrix is computed, with the $(i, j) - th$ element of the matrix given as

$$p(i, j) = \frac{f_{ij}^{d,\theta}}{\sum_{i}^{N}\sum_{j}^{N} f_{ij}^{d,\theta}} \quad (1)$$

where i and j refer to matrix's rows and columns, $f_{ij}^{d,\theta}$ is the frequency of occurrence of grey levels (i, j) separated by a distance d and a direction θ, and N stands for the total number of pixels in the window, for a particular value of d.

Originally, Haralick *et al.* (1973) proposed 16 measures which derived from each co-occurrence matrix. Among them, four are considered to be the most important: *Contrast*, *Angular Second Moment*, *Correlation* and *Homogeneity* (Haralick and Shapiro 1992). They are calculated as follows:

$$Contrast = \sum_{i=1}^{G}\sum_{j=1}^{G}|i-j|^2\, p(i,j) \tag{2}$$

which is a measure of the local variations between a pixel and its neighbours,

$$Angular\ Second\ Moment = \sum_{i=1}^{G}\sum_{j=1}^{G} p(i,j)^2 \tag{3}$$

which is a measure of uniformity,

$$Correlation = \sum_{i=1}^{G}\sum_{j=1}^{G}\frac{(i-\mu)(j-\mu)\,p(i,j)}{\sigma^2} \tag{4}$$

which is a measure of how correlated is a pixel to its neighbour, and

$$Homogeneity = \sum_{i=1}^{G}\sum_{j=1}^{G}\frac{p(i,j)}{1+|i-j|} \tag{5}$$

which measures the closeness of the distribution of elements in the GLCM to its diagonal. The above textural analysis was applied to the four bands of IKONOS image providing us with a total of 16 features.

Two different colour spaces were produced from the initial bands, and served as input data in the image classification. The first uses intensity (I), hue (H), and saturation (S) as the three positioned parameters (in lieu of R, G, and B). This is an advantageous system, since it presents colours more closely to the human's perception. The intensity represents the total amount of the light in a colour, the hue is the property of the colour determined by its wavelength, and the saturation is the purity of the colour (Zhang and Hong 2005). For the calculation of IHS transformation only three bands are needed, so a pseudo-colour RGB composite of IKONOS using channels four, three and two respectively was used.

In addition, a second colour space was applied, which is a linear transformation of the four multispectral bands of IKONOS image, that offers a means to optimize information extraction for vegetation studies, related with the classes of interest. This transformation is called tasseled cap (Kauth and Thomas, 1976) and has produced three data structure axes which define the vegetation information: *brightness*, *greenness* and *wetness*. These new features are sensor-dependent, and until recently they were only available for Landsat 5 and 7 images. Horne (2003), using approximately 200 different scenes of IKONOS images globally, managed to derive tasseled cap coefficients for IKONOS images.

2.4 Statistical classifier

Using four initial bands, 16 co-occurrence features and six spectral features, the image was classified using the training data with the maximum likelihood (ML), a spectral probabilistic 'hard' classifier. This is a common classifier used in many studies, and was used to compare the performance of the proposed SONeFMUC network, which is developed and analysed in the following section.

3. Development of SONeFMUC

3.1 The SONeFMUC architecture

The suggested neuro-fuzzy classifier is a multilayered structure, as depicted in figure 2. The network consists of a number of $\ell = 0,...,M$ layers, with the $\ell - th$ layer including N_ℓ neurons. The neurons are defined as *fuzzy neuron classifiers*, denoted as $FNC_j^{(\ell)}$, where $j = 0,...,N_\ell$ and $\ell = 0,...,M$. The input layer $\ell = 0$ includes the m networks inputs, $x_1, x_2, \ldots, x_m$, representing the feature components, while the output layer $\ell = M$ comprises the output node $FNC_1^{(M)}$, providing the overall network's decision.

Figure 2.

The neuron models in each layer are regarded here as generic local classifiers, working in a sub-region of the feature space and represented by fuzzy rule-based systems. Parent $FNC's$ at each layer are combined to generate a descendant FNC at the next layer, with better classification capabilities. The generic node classifiers perform successive feature transformations and decision

makings, using the distributed fuzzy rule bases and the fuzzy reasoning approach. Hence, the classification task is sequentially achieved by the $FNC's$ arranged along the layered structure of the SONeFMUC.

In a supervised learning scheme, the design of the classifier is based on a set of classified examples, used to establish the association between the pattern attributes and the class labels. Consider that the patterns are distributed over a set of M disjoint classes. Let us assume a training data set comprising N input-output observation pairs: $D_N = \{(\boldsymbol{x}[q], C[q]), q = 1, \ldots, N\}$ where $\boldsymbol{x}[q] = [x_1[q], \ldots, x_n[q]]^T$ denote the feature components, $C[q]$ is the class label for the q-th observation, and $C = \{C_1, ..., C_M\}$ is the set of classes. For convenience, the features are normalized in the range $[0,1]$, forming the feature space $F = [0,1]^n$.

Unlike the conventional classifiers which assume a given pattern to a single class, the suggested SONeFMUC model is a fuzzy classifier performing a map $F \to [0,1]^M$. Accordingly, they produce a decision output vector $D(\boldsymbol{x}) = [d_1(\boldsymbol{x}), ..., d_M(\boldsymbol{x})]^T$ embracing all classes, where $d_j(\boldsymbol{x}) \in [0,1]$, $j = 1, ..., M$, represents the grade of certainty in the assertion that pattern $\boldsymbol{x}$ belongs to class C_j. Based on the classification data, the structure of SONeFMUC is progressively expanded in layers using a structure learning algorithm.

3.2 FNC models at the first layer

The structure of the $FNC_j^{(1)}$ at the layer 1 is shown in figure 3. The generic model consists of two fuzzy rule based-systems: the so called fuzzy partial description (FPD) and a decision making fuzzy unit ($DMFU$). Therefore, the $FNC's$ at the first layer are represented as a pair of modules: $FPD_j^{(1)} = \{FPD_j^{(k)}, DMFU_j^{(1)}\}$, $j = 1, ..., N_1$. In the following, we describe the function of each one of the constituent units.

3.2.1 Fuzzy partial description (FPD). The $FPD's$ are represented by fuzzy rule-based Takagi Sugeno Kang (TSK) systems (Takagi and Sugeno 1985). Instead of receiving the entire attribute set $\{x_1, x_2, ..., x_m\}$, the input vector of $FNC_j^{(1)}$ denoted by $\boldsymbol{x}_j^{(1)}$, contains a small subset of p

features taken from the above set: $\boldsymbol{x}_j^{(1)} = \left[x_{j_1}, x_{j_2}, ..., x_{j_p} \right]^T \in \Re^p$. The feature subsets associated with each *FPD* are derived through recombinations of the original features, a task achieved during the structure learning process. The number of the inputs (p) exciting the *FPD's* at the first layer is a design parameter specified by the user. In the example case shown in figure 3, a generic neuron classifier is depicted with two *FPD* inputs.

The TSK systems considered in this paper provide two outputs. The output vector is denoted as $\boldsymbol{y}_j^{(1)} = \left[y_{j,1}^{(1)}, y_{j,2}^{(1)} \right]^T$, where the output variables are normalized in the range $y_{j,k}^{(1)} \in [0,1], k = 1,2$. The output space of $FNC_j^{(1)}$ formed by the components $y_{j,1}^{(1)}$, $y_{j,2}^{(1)}$, coincides with the class space, that is, the space where the classes are defined. Definition of the classes relates to the class targeting issue. To this end, a target value for both output variables is assigned for each class: $\boldsymbol{y}_d^{(C_j)} = \left[y_{d,1}^{(C_j)}, y_{d,2}^{(C_j)} \right]^T$, C_j, $j = 1,...,M$. The class ordering is accomplished by following a heuristic scheme determined by the designer.

The purpose of introducing a second output for the *FPD's* is explained as follows. The two-dimensional output space provides more flexibility for arranging the classes, based on their relative distance within the feature space. In addition, an increased number of classes can be considered, allowing SONeFMUC models to handle multi-class problems effectively. Furthermore, the inclusion of an additional output improves the representation capabilities of the models, thus facilitating the feature transformation task performed by the *FPD's*.

Figure 3.

For simplicity sake, in the remainder of this section we assume that the input and output vector of the *FPD's* are denoted by $\boldsymbol{x} = \left[x_1, x_2, ..., x_p \right]^T$, and $\boldsymbol{y} = \left[y_1, y_2 \right]^T$, respectively. Each premise variable x_i, $i = 1,..., p$ is partitioned into K_i fuzzy sets, $\left\{ A_1^{(i)}, \ldots, A_{K_i}^{(i)} \right\}$, described by two-sided Gaussian fuzzy sets. The membership functions are described by

$$\mu_j^{(i)}(x_i)=\begin{cases}\exp\left(-\frac{1}{2}\left(\frac{x_i-m_j^{(i)}}{\sigma_{j,L}^{(i)}}\right)^2\right), & x_i \le m_j^{(i)} \\ \exp\left(-\frac{1}{2}\left(\frac{x_i-m_j^{(i)}}{\sigma_{j,R}^{(i)}}\right)^2\right), & x_i > m_j^{(i)}\end{cases} \qquad i=1,\ldots,p, \quad j=1,\ldots,K_i \tag{6}$$

where $m_j^{(i)}$ is the mean and $\sigma_{j,L}^{(i)}$, $\sigma_{j,R}^{(i)}$ are the widths of the left and right hand parts of $\mu_j^{(i)}(x_i)$. Initially, the fuzzy sets $A_j^{(i)}$ are evenly distributed along the x_i axis (see figure 4(a)):

$$m_j^{(i)}=\frac{(j-1)}{(K_i-1)}, \qquad i=1,\ldots,p, \qquad j=1,\ldots,K_i \tag{7}$$

The number of fuzzy sets K_i is a user defined parameter, being the same for all input variables, in every *FPD*, throughout the SONeFMUC structure. In order to improve the feature transformation of the *FPD's*, a *K*-means clustering algorithm (Lee *et al.* 2001) is applied on the membership centres along each input. The goal is to locate the fuzzy sets in a way that the resulting fuzzy partitions are focused on regions with large data collections. Upon termination of data clustering, the fuzzy sets are tuned as indicated in figure 4(b). The values of $\sigma_{j,L}^{(i)}$ and $\sigma_{j,R}^{(i)}$ are calculated such as the consecutive fuzzy sets exhibit a degree of overlapping of 0.5.

Figure 4(a),(b).

Following a grid-type partition approach, we create a total number of $R=\prod_{i=1}^{r}K_i$ rectangular fuzzy subspaces, $\mathcal{A}^i$, determined by the Cartesian product: $\mathcal{A}^i=A_{i_1}^{(1)}\times A_{i_2}^{(2)}\times\cdots\times A_{i_r}^{(r)}, i=1,...,R$. Figure 5 shows an indicative two-dimensional fuzzy partition after data clustering. Each subspace forms a fuzzy region within the premise space where a fuzzy rule is defined.

The *FPD's* are described by *R* TSK-type fuzzy modeling rules of the form:

$$R_m^{(i)}: IF\ x_1\ is\ A_{i_1}^{(1)}\ AND\cdots AND\ x_p\ is\ A_{i_p}^{(1)}\ THEN\ y_1=g_1^{(i)}(\boldsymbol{x})\ \ AND\ y_2=g_2^{(i)}(\boldsymbol{x}) \tag{8}$$

The rule functions are given either as

$$g_r^{(i)}(\boldsymbol{x})=w_{0,r}^{(i)},\ r=1,2 \tag{9}$$

implementing the crisp consequent, or as a polynomial of the *FPD* inputs

$$g_r^{(i)}(\boldsymbol{x})=w_{0,r}^{(i)}+w_{1,r}^{(i)}x_1+\ldots+w_{p,r}^{(i)}x_p,\ r=1,2 \tag{10}$$

implementing the traditional TSK consequent function.

Figure 5.

For each pattern submitted to the *FPD*, the degree of firing of the rules are determined by integrating the antecedent fuzzy sets associated with the partition $\mathcal{A}^i$, through an AND operator:

$$\mu_i(\boldsymbol{x}) = \prod_{j=1}^{R} \mu_{i_j}^{(i)}(\boldsymbol{x}) \tag{11}$$

The outputs of the *FPD* are then derived by

$$y_r = \sum_{i=1}^{R} \frac{\mu_i(\boldsymbol{x}) g_r^{(i)}(\boldsymbol{x})}{\sum_{s=1}^{R} \mu_s(\boldsymbol{x})}, r = 1,2 \tag{12}$$

Based on the set of rules obtained above, we proceed to a rule base simplification procedure with the scope to reduce the number of rules. Along this direction, each modelling rule $R_m^{(i)}$ is evaluated by computing the percentage concentrations of patterns, $\tilde{n}_i$, over the entire training set: $\tilde{n}_i = \frac{n_i}{N} \times 100\ (\%), i = 1,...,R$. n_i stands for the number of patterns that are included in the antecedent part $A_j^{(i)}$ with a degree of firing fulfilling $\mu_i(x) \geq 0.5$. The values of $\tilde{n}_i$ are then arranged in descending order and compared to a prescribed threshold ξ set up by the user (i.e. $\xi = 5\%$). Fuzzy rules exhibiting $\tilde{n}_i \geq \xi$ are retained while the rest of them are discarded. At the end of this process we obtain a simplified fuzzy model (*FPD*), including a reduced number of rules $\tilde{R} < R$. Patterns that are located at fuzzy cells corresponding to removed rules are covered by the neighbouring strong ones. This is made possible by noticing that the fuzzy regions are described by Gaussian rather than triangular memberships, of appropriate centres and widths. Accordingly, every pattern takes a stronger or weaker firing, assisting its manipulation by the reduced rule base.

Having determined the premise parameters (means and widths), the outputs of the *FPD's* are linear with respect to the consequent weights. Therefore, optimal estimates of these parameters can be obtained using the recursive least square estimate (RLSE) method (Goodwin and Sin 1984). Given a specific class targeting, the RLSE method calculates the appropriate values of the consequent weights so that the distance between the *FPD* outputs and the class targets is

minimized. As a result, a supervised learning task is achieved with the following objective: patterns $\boldsymbol{x}_j^{(1)}(q)$ belonging to a particular class C_j, should produce an output $\boldsymbol{y}_k^{(1)}(q)$ located in a neighbourhood of the respective class target $\boldsymbol{y}_d^{(C_j)}$, $j=1,...,M$. From this point of view, each *FPD* realizes a non-linear mapping from the initial feature space to a transformed output space. The *FPD* outputs can be regarded as transformed versions, $\boldsymbol{y}_j^{(1)} = FPD_j^{(1)}\left(\boldsymbol{x}_j^{(1)}\right)$, of the input features, which are more separated compared to the original features. Feature transformation facilitates discrimination of the patterns along the classes, thus leading to more accurate classification results obtained by the following *DMFU* units.

In order to show the function undertaken by the feature transformation, an illustrative case is indicated in figure 6 for a four-class problem ($M=4$) with two input features. As can be seen, the class examples in the original feature space (x_1, x_2) are mixed with each other, making the discrimination among them a difficult task. On the other hand, the transformed patterns defined by the *FPD* outputs are more separated due to the fuzzy inference mechanism and the effectiveness of the parameter learning algorithm (RLSE).

Figure 6.

3.2.2 Decision making fuzzy unit (DMFU). As shown in figure 3, a decision making fuzzy unit (*DMFU*) is introduced in each *FNC*, following the associated *FPD* module, with the goal to determine the degree of support given by the local classifier in each class. The continuous outputs of the *FPD*, $\boldsymbol{y}=[y_1, y_2]^T$, serve as inputs of the *DMFU* while its output is a *M*-dimensional vector, denoted as *DN*, that provides the soft decision profile of the neuron classifier.

Figure 7.

Each $y_r \in [0,1]$, $r=1,2$, is divided into $L_r, r=1,2$ fuzzy sets, namely $\left\{B_1^{(r)},\ldots,B_{L_r}^{(r)}\right\}$. The values of L_r are decided based on the number of classes at hand, so that: $L_1 \times L_2 \geq M$. The fuzzy sets of *DMFU* are now represented by trapezoidal membership functions, centred at the target values of each class. The fuzzy partition of a *DMFU* input is given graphically in figure 7. The parameters of the intermediate membership functions are defined as

$$a_i^{(r)} = \frac{(i-1)}{(L_r-1)} - \rho,\ \beta_i^{(r)} = \frac{(i-1)}{(L_r-1)} + \rho,\ \gamma_i^{(r)} = \frac{i}{(L_r-1)} - \rho,\ \delta_i^{(r)} = \frac{i}{(L_r-1)} + \rho \qquad (13)$$

for $i = 2,\ldots,L_r - 1,\ r = 1,2$, where ρ is a small percentage of $1/L_r$ (i.e. $\rho = 20\%$) that controls the degree of overlapping between adjacent fuzzy sets. For the left-most and the right-most membership functions the relevant parameters are given by

$$a_1^{(r)} = \beta_1^{(r)} = 0,\ \gamma_1^{(r)} = a_2^{(r)},\ \delta_1^{(r)} = \beta_2^{(r)},\ a_{L_r}^{(r)} = \gamma_{L_r}^{(r)},\ \beta_{L_r}^{(r)} = \delta_{L_r}^{(r)},\ \gamma_{L_r}^{(r)} = \delta_{L_r}^{(r)} = 1 \quad (14)$$

Figure 8.

Using the grid partition, we obtain $L_1 \times L_2 \geq M$ fuzzy subspaces. An indicative partition of the *DMFU* for a four-class case ($M = 4$) is shown in figure 8. In order to represent $M = 4$ classes, we define $L_1 = 2$ and $L_2 = 2$ membership functions. For each rectangular subspace we define a classification rule of the form:

$$R_c^{(i)} : IF\ y_1\ is\ B_{i_1}^{(1)}\ AND\ y_2\ is\ B_{i_2}^{(2)}\ THEN\ \left(y_1[q], y_2[q]\right)\ is\ C_j \quad (15)$$

where $C_j \in C$ is the label for class *j*.

The fuzzy inference mechanism of the *DMFU*'*s* is realized in two steps:

Step 1. Calculate the degree of firing of the classes, β_{C_i}, $i = 1,...,M$

$$\beta_{C_i} = \mu_{i_1}^{(1)}\left(y_1(q)\right) \wedge \mu_{i_2}^{(2)}\left(y_2(q)\right) \quad (16)$$

Step 2. Calculate the normalized firings:

$$\bar{\beta}_{C_i} = \frac{\beta_{C_i}}{\sum_{r=1}^{M} \beta_{C_r}} \quad (17)$$

The output of the *DMFU* is a vector, *DN*, derived as follows:

$$DN = \left[dn_1,\ldots,dn_M\right]^T = \left[\bar{\beta}_{C_1},\ldots,\bar{\beta}_{C_M}\right]^T \quad (18)$$

The components $dn_i \in [0,1],\ i = 1,\ldots,M$, represent the degree of support given by the local classifier $FNC_j^{(1)}$, under the hypothesis that a particular pattern $y[q]$, belongs to class-i. The soft outputs of *DMFU* can be hardened in order to make a crisp decision upon the class each pattern belongs, such that

$$DN\left(y[q]\right) = C_x \Rightarrow dn_{c_x} = \max\left\{dn_v\right\},\quad v = 1,\ldots,M \quad (19)$$

With regard to the example considered in figure 6, the class labels can be easily assigned as shown in figure 9. The dotted lines split the class decision regions whereas the solid lines represent confident regions, where patterns are classified with a high grade of certainty.

Figure 9.

Each $FNC_j^{(1)}$ provides two output sources: a vector of continuous outputs $y_j = \left[y_{j,1}, y_{j,2} \right]^T$, and a M-dimensional vector $DN_j^{(1)}$, the output of the $DMFU_j^{(1)}$, that includes the soft decision supports for all classes (see figure 3). The outputs of the $FNC's$ at layer 1 serve as inputs to the $FNC's$ to be generated in the second layer.

3.3 FNC models at layer 2

The structure of the neuron classifiers, $FNC_k^{(\ell)}$, at layer 2 ($\ell = 2$) is demonstrated in figure 10. Two parent $FNC's$ from the preceding layer are combined to generate a descendant FNC at that layer. The neuron classifiers are now represented by the triple: $FNC_k^{(\ell)} = \left\{ FPD_k^{(\ell)}, DMFU_k^{(\ell)}, \mathcal{F}_k^{(\ell)} \right\}$. To exploit the information acquired by these parent classifiers we make use of a fusion scheme, being a common practice in *combining classifiers* (Kuncheva *et al.* 2001). To this end, a fusion operator ($\mathcal{F}_k^{(\ell)}$) is introduced within each $FNC_k^{(\ell)}$, to aggregate the outputs of the parent classifiers. The decision output of the fuser is attached to the descendant FNC. Additionally, the fusion algorithm serves as a means to discriminate between those patterns that are currently well classified by the parent classifiers, and the ones which need further investigation by the FPD unit of the offspring classifier. This procedure gives rise to a *data splitting* mechanism that offers efficient handling of the data flow and reduction of the computational cost.

Figure 10.

The derivation of the decision supports of the $FNC's$ at layer 2 are calculated through the following steps:

A.1. *Decision fusion:* For a given pattern $\boldsymbol{x}[q]$, the decision outputs of the antecedent classifiers $DN_i^{(\ell-1)}$ and $DN_j^{(\ell-1)}$ are fused as follows

$$DF_k^{(\ell)}\left(\boldsymbol{x}[q]\right) = \mathcal{F}_k^{(\ell)}\left\{ DN_i^{(\ell-1)}\left(\boldsymbol{x}[q]\right), DN_j^{(\ell-1)}\left(\boldsymbol{x}[q]\right) \right\} \quad (20)$$

where $\mathcal{F}$ denotes a fusion operator. The resulting decision output, $DF_k^{(\ell)}$, includes the certainty grades given by the fuser for all decision supports for all classes

$$DF_k^{(\ell)}\left(\boldsymbol{x}[q]\right)=\left[df_{k,1}^{(\ell)}\left(\boldsymbol{x}[q]\right),\ldots,df_{k,M}^{(\ell)}\left(\boldsymbol{x}[q]\right)\right] \quad (21)$$

There are several types of fusion operators reported in the literature. In general, they are distinguished into two major categories: the *class–conscious* and the *class–indifferent* fusion methods, depending on the way they handle the decision information carried out by the combined classifiers. In this paper, four different types of fusion operators are used, namely *min*, *weighted average*, *fuzzy integral*, and *decision templates* (Kuncheva *et al.* 2001). For instance, the *min* operator is a simple and a rather conservative aggregation operator, belonging to the class–conscious methods:

$$df_{k,v}^{(\ell)}=\min\left\{dn_{i,v}^{(\ell)},dn_{j,v}^{(\ell)}\right\},\qquad v=1,\ldots,M \quad (22)$$

A.2. Data *splitting (DS)*: The decision profile obtained by the fuser, $DF_k^{(\ell)}$, provides the degree of support that a pattern belongs in each class by combining the outputs of the parent classifiers $FNC_i^{(\ell-1)}$ and $FNC_j^{(\ell-1)}$. To ascertain the classification level of the patterns, the values $df_{k,r}^{(2)}$ are compared to a user's defined threshold:

$$df_{k,r}^{(2)}\geq\vartheta,\qquad \vartheta\in[0.5,1] \quad (23)$$

where ϑ represents the degree of confidence (i.e. $\vartheta=0.8$) that a pattern belongs to a certain class, $\boldsymbol{x}[q]\in C_r$. The above condition is called here the *maximum classification level criterion* (MCLC).

Based on the MCLC criterion, the entire data set D_N is divided into two disjunctive subsets $J_k^{(\ell)}$ and $V_k^{(\ell)}$, so that $D_N=J_k^{(\ell)}\cup V_k^{(\ell)}$. The subset $J_k^{(\ell)}$ includes those patterns that fulfil the MCLC criterion, that is, they are currently well classified with high grade of certainty. Furthermore, the subset $V_k^{(\ell)}$ contains the rest of data patterns which are either misclassified by the fuser or correctly classified with a low degree of support.

A.3. *Handling of patterns in* $J_k^{(\ell)}$: In this case, the fuser produces high grade of certainties for a class, implying that both parent classifiers agree strongly on the same class. Therefore, well classified patterns contained in $J_k^{(\ell)}$ are handled by the fuser, i.e. their decision outputs are derived

by $DF_k^{(\ell)}\left(\boldsymbol{x}[q]\right), \forall\, \boldsymbol{x}[q] \in J_k^{(\ell)}$.

A.4. Handling *of patterns in* $V_k^{(\ell)}$: For these patterns, the fuser produces low certainty grades for all classes (below confidence threshold ϑ), which means that either a conflict occurs between the parent $FNC's$, or patterns are correctly classified by the parent classifiers with weak degree of supports. Hence, ambiguous patterns included in $V_k^{(\ell)}$ are fed to $FPD_k^{(\ell)}$ for further processing. They are subject to an additional feature transformation to improve their discrimination.

Since two parent classifiers are combined at a time, the input vector of the $FPD_k^{(\ell)}$ is a four-dimensional vector defined as $x_k^{(\ell)} = \left[\left(x_i^{(\ell-1)}\right)^T, \left(x_j^{(\ell-1)}\right)^T\right]^T$. In the output space formed by FPD outputs $y_{k,j}^{(\ell)}$, $j = 1,2$, the classes are ordered following exactly the same class targeting scheme as the one defined for the $FNC's$ at the first layer. The construction of the $FPD_k^{(\ell)}$ proceeds along the steps described in section 3.2. The model building process includes: the data clustering through *K*-means algorithm, the formulation of the fuzzy rule base, the rule base simplification, and determination of the consequent weights. Having determined the locations of the new transformed features $y_k^{(\ell)}$ ($FPD_k^{(\ell)}$ outputs) of the patterns contained in $V_k^{(\ell)}$, decision making is then applied by means of the corresponding $DMFU_k^{(\ell)}$ module, to obtain the soft decision output vector:

$$DN_k^{(\ell)}\left(\boldsymbol{x}[q]\right) = DMFU_k^{(\ell)}\left\{y_k^{(\ell)}[q]\right\} = \left[dn_{k,1}^{(\ell)}(\boldsymbol{x}), dn_{k,2}^{(\ell)}(\boldsymbol{x}), ..., dn_{k,M}^{(\ell)}(\boldsymbol{x})\right]^T \quad (24)$$

where $dn_{k,j}^{(\ell)} \in [0,1]$ denotes the degree of support for a pattern belonging in class C_j.

A.5. *Overall decision output*: The overall decision profile of $FNC_k^{(\ell)}$ is formulated as a composition of two parts:

$$D_k^{(\ell)} = DF_k^{(\ell)} \oplus DN_k^{(\ell)} = \left[d_{k,1}^{(\ell)}, d_{k,2}^{(\ell)}, ..., d_{k,M}^{(\ell)}\right]^T \quad (25)$$

where the $DF_k^{(\ell)}$ and $DN_k^{(\ell)}$ come from the fuser and the modules pair $\left\{FPD_k^{(\ell)}, DMFU_k^{(\ell)}\right\}$, respectively, functioning on the $J_k^{(\ell)}$ and $V_k^{(\ell)}$ data subsets: $D_k^{(\ell)}\left(\boldsymbol{x}[q]\right) = DF_k^{(\ell)}\left(\boldsymbol{x}[q]\right)$, $\forall\, \boldsymbol{x}[q] \in J_k^{(\ell)}$ and $D_k^{(\ell)}\left(\boldsymbol{x}[q]\right) = DN_k^{(\ell)}\left(\boldsymbol{x}[q]\right)$, $\forall\, \boldsymbol{x}[q] \in V_k^{(\ell)}$.

When a hard decision is to be made, the patterns are assigned to a class C_r fulfilling the maximum argument principle:

$$D_k^{(\ell)}\left(\boldsymbol{x}[q]\right) = C_r \Rightarrow d_{k,C_r}^{(\ell)} = \max_{j=1,\ldots,M}\left\{d_{k,j}^{(\ell)}\right\} \tag{26}$$

The above fuzzy-to-crisp transformation operates either on $df_{k,j}^{(\ell)}$ or $dn_{k,j}^{(\ell)}$, depending on whether the pattern of interest belongs to $J_k^{(\ell)}$ or $V_k^{(\ell)}$.

3.4 FNC models at higher layers

The general structure of the *FNC's* at higher layers ($\ell \geq 3$) is shown in figure 11. The derivation of the decision supports for this type of *FNC's* is accomplished using a modified version of the procedure for the *FNC's* at layer 2 described in section 3.2. The modifications take place at steps A.1 and A.4 and are described as follows:

$A'.1$ The outputs of the parent *FNC's*, namely, the $FNC_i^{(\ell-1)}$ and the $FNC_j^{(\ell-1)}$ are now given by equation (25). They are composed of two parts: the $DF_{i,j}^{(\ell-1)}$ coming from the fusers $\mathcal{F}_{i,j}^{(\ell-1)}$ and the $DN_{i,j}^{(\ell-1)}$ produced by the $DMFU_{i,j}^{(\ell-1)}$ modules. Accordingly, the decision vector determined by the fuser $\mathcal{F}_k^{(\ell)}$, associated with the $FNC_k^{(\ell)}$, to be generated at layer ℓ is given by

$$DF_k^{(\ell)} = \mathcal{F}\left\{D_i^{(\ell-1)}\left(\boldsymbol{x}\right), D_j^{(\ell-1)}\left(\boldsymbol{x}\right)\right\} = \left[df_{k,1}^{(\ell)}\left(\boldsymbol{x}\right), \ldots, df_{k,M}^{(\ell)}\left(\boldsymbol{x}\right)\right] \tag{27}$$

$A'.4$ *Handling of patterns in* $V_k^{(\ell)}$: Assume that the antecedent classifiers $FNC_i^{(\ell-1)}$ and $FNC_j^{(\ell-1)}$ assign a pattern $\boldsymbol{x}_k^{(\ell-1)} \in V_k^{(\ell-1)}$ to the classes C_i^* and C_j^*, respectively. We check whether patterns $\boldsymbol{x}_i^{(\ell-1)}$ and $\boldsymbol{x}_j^{(\ell-1)}$ which are associated with $\boldsymbol{x}_k^{(\ell)}$ are included in the sets $J_i^{(\ell-1)}$, $J_j^{(\ell-1)}$ or to the sets $V_i^{(\ell-1)}$, $V_j^{(\ell-1)}$, respectively. The input to the $FPD_k^{(\ell)}$ is then formed as a four-dimensional vector $\boldsymbol{x}_k^{(\ell)} = \left[\left(\boldsymbol{x}_{k,1}^{(\ell)}\right)^T, \left(\boldsymbol{x}_{k,2}^{(\ell)}\right)^T\right]^T \in \Re^4$, where the parts $\boldsymbol{x}_{k,1}^{(\ell)}, \boldsymbol{x}_{k,2}^{(\ell)} \in \Re^2$ are defined by

Formulate FPD inputs

{

If $\boldsymbol{x}_i^{(\ell-1)} \in V_i^{(\ell-1)}$ Then $\boldsymbol{x}_{k,1}^{(\ell)} = \boldsymbol{y}_i^{(\ell-1)}$

Else If $\boldsymbol{x}_i^{(\ell-1)} \in J_i^{(\ell-1)}$ Then $\boldsymbol{x}_{k,1}^{(\ell)} = \boldsymbol{y}_d^{(C_i^*)}$

If $\boldsymbol{x}_j^{(\ell-1)} \in V_j^{(\ell-1)}$ Then $\boldsymbol{x}_{k,2}^{(\ell)} = \boldsymbol{y}_j^{(\ell-1)}$

Else If $\boldsymbol{x}_j^{(\ell-1)} \in J_j^{(\ell-1)}$ Then $\boldsymbol{x}_{k,2}^{(\ell)} = \boldsymbol{y}_d^{(C_j^*)}$

}

Figure 11.

The mechanism described above exhibits some remarkable merits discussed in the following: i) First, the fuser exploits the decision supports given by the antecedent classifiers. Patterns which are well classified from their parents have a large support in the fuser's output, resulting in a confident decision for the class they belong. Provided that a certain confidence level is exceeded, they are excluded from the fuzzy model construction. ii) Data splitting leads to significant computational savings, a convenient merit, especially when difficult problems are examined with large data sets and large number of classes. iii) Exclusion of well classified data allows the FPD's to focus on those patterns where adequate classification accuracy is not yet achieved. iv) The antecedent classifiers in each layer are developed following different paths, starting from the original feature space and proceeding through the layered structure of the SONeFMUC. Hence, among the family of feature subspaces investigated, an optimal path is decided, leading to enhanced classification rates.

3.5 Structure learning

The proposed model is generated in a self-organizing manner, by means of the GMDH algorithm (Ivakhnenko 1968), described below. Particularly, the structure of the SONeFMUC is not predetermined in advance. Starting from the original system inputs (features), new layers are sequentially developed, until a final topology is obtained, satisfying the performance requirements.

The structure learning of SONeFMUC proceeds along the following steps:

Step 1 Determine the system's inputs. The decided set of input variables (features), $x_i = 1, \ldots, m$, must exhibit a reasonable degree of separability between the classes.

Step 2. Formulate data sets: The data set D_N is divided into a training data set D_{trn}, a validation data set D_{val}, and a testing data set D_{chk}, comprising n_{trn}, n_{val} and n_{chk}, respectively, with

$n_{trn} + n_{val} + n_{chk} = N$. The training and validation data sets are used for determining the structure of the individual $FNC's$, at each layer, while the checking data are employed for evaluating the SONeFMUC model obtained at the end of structure learning.

Step 3. Select the model parameters: In this step, we choose the values of the structural parameters involved in the $FPD's$, the $DMFU's$ and the fusion mechanism:

- Select the number of inputs, p, of the $FPD's$. For the $FNC_k^{(1)}$ at layer 1, p takes values in the range $p \in [2,4]$, with the inputs chosen among the set of original features $\{x_1, x_2, ..., x_m\}$. We consider that all $FNC_k^{(\ell)}$ appearing at higher layers, $\ell \geq 2$, are developed by combining two parent $FNC's$ of the previous layer. Accordingly, the $FPD_k^{(\ell)}$ receives a four-dimensional input vector ($p = 4$), formed by concatenating the outputs $\boldsymbol{y}_i^{(\ell-1)}$ and $\boldsymbol{y}_i^{(\ell-1)}$ of the parent $FPD_i^{(\ell-1)}$ and $FPD_j^{(\ell-1)}$, respectively.
- Select the number of membership functions K_i, $i = 1,...,r$, used to partition the premise space of $FPD_k^{(\ell)}, \ell = 1,...,M$. In our simulations, we keep the same values of K_i for all $FNC's$ throughout the SONeFMUC model.
- Select the number of membership functions L_i, $i = 1,2$, used to partition the space of the $DMFU_k^{(\ell)}$. The choice of these values depends upon the number of classes of the problem at hand. Once the appropriate targeting scheme is decided, it is applied uniformly for all $DMFU's$ at all layers.
- Select the type of the consequent functions of the rules (8) describing the $FPD_k^{(\ell)}$. We consider here two traditional forms of the TSK rules:

$$\begin{aligned} &Crisp: && w_0 \\ &Linear: && w_0 + w_1 x_1 + \ldots + w_p x_p \end{aligned} \qquad (28)$$

- Select the fusion operator used to implement the fuser $\mathcal{F}_k^{(\ell)}$, among the three class-conscious aggregation rules (min, weighted average, and fuzzy integral), and the class-indifferent alternative, the DT method. Once a specific fuser type is chosen, it is applied uniformly for all $FNC's$ during the structure learning of a particular SONeFMUC model.
- Select the threshold value $\vartheta \in [0.5, 0.8]$ which defines the confidence level that controls the data splitting within each $FNC_k^{(\ell)}, \ell = 2,...,M$. The greater the value of ϑ, the smaller the amount of

patterns whose decision is derived by the fuser $\mathcal{F}_k^{(\ell)}$ ($DF_k^{(\ell)}$), and the greater the data portion being handled by the pair of modules $\left\{FPD_k^{(\ell)}, DMFU_k^{(\ell)}\right\}$ ($DN_k^{(\ell)}$).

Step 4. Consider the so called best set, $BS^{(\ell-1)} = \left\{FNC_i^{(\ell-1)}, i = 1,...,W\right\}$, formed at layer $(\ell-1)$. The $BS^{(\ell-1)}$ comprises a number of W individuals that represent the most qualifying $FNC's$ retained at that layer. The outputs of these individuals form the candidate input set used for the construction of the individuals $FNC_k^{(\ell)}$ at layer ℓ. Especially, the best set $BS^{(0)}$ includes the m original feature components: $BS^{(0)} = \left\{x_1, x_2, ..., x_m\right\}$. As for the $BS^{(\ell)}$ at the succeeding layers, it is determined as described in step 8.

Step 5. *Create the population $P^{(\ell)}$ of candidate FNC's at layer ℓ*: The population $P^{(\ell)}$ is formulated by recombining the individuals in $BS^{(\ell-1)}$ obtained at layer $(\ell-1)$. The new individuals to be generated are obtained by combining parent $FNC's$ from $BS^{(\ell-1)}$. Considering all possible combinations ${}^{W}C_2$, we conclude with a total number of $Q^{(\ell)} = \binom{W}{2} = W!/(W-2)!2!$ new $FNC's$ that form the population at the current layer: $P^{(\ell)} = \left\{FNC_k^{(\ell)}, \ell = 1, ..., Q^{(\ell)}\right\}$.

Step 6. Construct the $FNC_k^{(\ell)}$ models of $P^{(\ell)}$: In this step we determine the structure of the $FNC_k^{(\ell)}$, $\ell = 1,...,Q^{(\ell)}$, by combining its parent modules $FNC_i^{(\ell-1)}$ and $FNC_j^{(\ell-1)}$. Combining the parents $FNC's$ means that we make use of both types of outputs being offered, that is, the continuous outputs $\boldsymbol{y}_i^{(\ell-1)}$ and $\boldsymbol{y}_j^{(\ell-1)}$ (transformed feature values) and the soft decision vectors, $D_i^{(\ell-1)}$ and $D_j^{(\ell-1)}$. The building of $FNC_k^{(\ell)}$ is carried out by following the procedures described in sections 3.2-3.4.

Step 7. Evaluate the FNC's of $P^{(\ell)}$: Each $FNC_k^{(\ell)}$ is evaluated to assess its approximation and predictive capabilities. For the training data set, calculate the following error measure:

$$E_{trn,k} = \frac{1}{n_{trn}} \sum_{j=1}^{2} \sum_{q=1}^{n_{trn}} \left\{ y_{d,j}^{C}[q] - y_{k,j}^{(\ell)} \right\}^2 + \sum_{q=1}^{n_{trn}} \left\{ C_d[q] \neq C_j[q] \right\}, k = 1,...,Q^{(\ell)} \qquad (29)$$

The first term in equation (29) is the mean squared error function, computing the proximity of the transformed outputs $y_{k,j}^{(\ell)}$ to the respective class targets. The second term determines the total number of misclassifications occurring over the training data set. Assuming that two individuals

produce the same number of misclassifications, the one will be selected exhibiting the lower mean squared error, that is, providing better placement of the transformed features. The $E_{val,k}$ associated with the validation data set, D_{val}, is calculated in a similar way. The $FNC's$ are evaluated using a weighted average metric:

$$E_k = (1-a)\cdot E_{trn,k} + a\,E_{val,k} \tag{30}$$

where $a \in [0,1]$ is a weight parameter specified by the user that controls the balance between $E_{trn,k}$ and $E_{val,k}$.

Step 8. Formulate the best set $BS^{(\ell)}$ at layer ℓ: The values of E_k are ordered in ascending order. The first individual corresponds to the most qualifying FNC, denoted as $FNC_*^{(\ell)}$, having the lowest classification error. A collection of W $FNC's$ are retained that exhibit the lower classification errors. These $FNC's$ form the best set $BS^{(\ell)}$ at the current layer, including the highly qualified individuals of the population, $P^{(\ell)}$. The outputs of the $FNC's$ contained in $BS^{(\ell)}$ serve as candidate inputs for the next layer while the remaining ones are discarded.

Step 9. Check for the termination criterion: In this step we check whether the expansion of the SONeFMUC model is going to be terminated at the current layer or will keep growing by introducing an additional layer. Assume that the best FNC in $BS^{(\ell)}$, $FNC_*^{(\ell)}$, exhibits a classification score denoted by $E_*^{(\ell)}$. The best node $FNC_*^{(\ell)}$ is temporarily regarded as the output of the SONeFMUC model. The model expansion stops when either of the following conditions is fulfilled:

- After the layer ℓ reaches a maximum number of layers, $M_{\max}$, determined by the user.
- The best performance attained at the current layer exceeds the one obtained at the previous layer: $E_*^{(\ell)} \geq E_*^{(\ell-1)}$.

Upon termination of the evolution, proceed to step 11.

Step 10. Determine the inputs to the next layer: Assuming that $E_*^{(\ell)} < E_*^{(\ell-1)}$ and the maximum number of layers is not yet reached, the SONeFMUC models is allowed to expand by including a new layer. The retained individuals $FNC_k^{(\ell)}$ in $BS^{(\ell)}$ are recombined again to generate the $FNC's$ at layer $(\ell+1)$. Accordingly, their outputs $\boldsymbol{y}_k^{(\ell)}$ and $D_k^{(\ell)}$ are submitted as inputs to the descendant $FNC's$ at the next layer. To this end, go to step 5 for the generation of the new population.

Step 11. Recover the network's architecture: Once the stopping criteria are satisfied for some $M \le M_{\max}$, the node classifier with the best performance, $FNC_*^{(M)}$, is considered as the ending node of SONeFMUC, providing the decision outputs of the model. The remaining $FNC's$ at the output layer are discarded. In the following, we perform a reverse flow tracing through the network's structure, moving from the output to the input layer. All nodes at the intermediate layers (the input layer included), having no contribution to the FNC selected at the final layer are removed from the network. As regards the model's inputs, among the original features $\{x_1, ..., x_m\}$, a subset of significant features is retained whereas the rest of them are discarded.

Figure 12.

Figure 12 demonstrates the network expansion via the GMDH method. The dotted lines indicate the recovered structure of the SONeFMUC. The resulting model is a three-layered network, with a total number of six $FNC's$. Four features are selected by GMDH, x_1, x_2, x_3 and x_5, from a total of m original features.

The SONeFMUC classifier exhibits some remarkable attributes, distinguishing it from other classification model of the literature:

- The SONeFMUC are self-organizing multilayered networks whose structure is developed sequentially in a layer-by-layer basis, following a systematic expansion procedure, the GMDH method. Depending on the complexity of the particular classification problem, the network depth is properly controlled, so that the performance of the obtained model fulfils the design requirements.
- The GMDH methodology implements inherently the so called feature selection task. In the final model recovered at the end of structure learning, only the most important features are retained having a significant contribution to the classification mapping, while the unnecessary ones are discarded. In that respect, the GMDH performs two tasks simultaneously, namely, model building and feature selection.
- The FPD modules of the $FNC's$ perform successive feature transformations through the layers. Starting from the input layer, the original features are repeatedly transformed between the intermediate layer spaces. For higher layers, each new FPD improves the class discrimination, allowing the corresponding $DMFU's$ to draw more accurate classification assignments.

4. Experimental results

4.1 Application of the SONeFMUC to the study area

The SONeFMUC network was applied to a multispectral IKONOS image using the set of training samples recognized in the field. Owing to the large number of classes and the spectral overlapping of the feature signatures, we were confronted with misclassification problems, especially in classes that represent vegetation cover. Therefore, based on the pan-sharpen image and after careful photo-interpretation, the image was segmented into two zones: the wetland zone which includes the lake and its surrounding wetland vegetation, and the agricultural zone. In the wetland zone five classes were recognized (water bodies, phragmites, tamarix, wet meadows, and trees). Furthermore, we consider eight classes in the agricultural zone, six of them referring to different crop types (maize, alfalfa, cereals, orchards, vegetables, and fallow) while the remaining two in other land cover types (urban areas and shrubs). The SONeFMUC classifier was applied to both of these zones.

Since the structure learning process uses a validation data set to obtain appropriate networks with higher generalization capabilities, the training set was split further in training and validation using 60% and 40% of the original training set, respectively. The MLC classifier used the original training set for the training stage. The testing set was the same for both methods. To initiate the structure learning of the SONeFMUC classifier for each zone, a number of structural parameters should be decided: the number of fuzzy sets in each input, the form of the rules and the type of the fuser. This task is accomplished through the following thee-step procedure:

Step 1. At this stage we consider the *min* fusion operator, and develop via GMDH different network combinations using three or five fuzzy sets along each *FPD* input and crisp or linear rules.

Step 2. The best network based on the checking data performance was selected as the most appropriate.

Step 3. The remaining three types of fusion, namely the weighted average, the fuzzy integral, and the decision templates were used to the network decided in step 2. The network architecture exhibiting the higher classification accuracy on the checking data was selected as the final model.

In all simulations, the grey level co-occurrence matrices were calculated using a window size of

7 x 7, direction $\theta = 0^0$, and distance $d = 1$ for the wetland zone, and a window size of 23 x 23, direction $\theta = 45^0$, and distance $d = 1$ for the agricultural zone. The performance of the obtained model was evaluated in terms of four parameters: the confusion error matrix, the overall accuracy, the Khat statistic, and the Z-score with 95% confidence level (Congalton and Green 1999).

4.1.1 Application of the SONeFMUC to the wetland zone. Following the above three-step procedure, a network was selected using three fuzzy sets for the FPD inputs, TSK type rules, and the fuzzy integral as the fusion scheme. The resulting network is a five-layered structure with an overall classification accuracy of 89.5% on the testing data. Table 1 shows the confusion matrix along with the producer's and user's accuracy in percentage (PA%, UA% respectively). Additionally, the statistical parameters Khat and Z–score were calculated, and used to assess the quality of classifiers.

The SONeFMUC model exhibits a strong agreement between the remotely sensed classification and the reference data due to the high value of the Khat parameter, which is greater than 0.8 (Congalton and Green 1999). In addition, the Z-test was performed, leading to a value of 33.14. Since the Z-score is higher than the value $Z_c = 1.96$, the classification provided by SONeFMUC was significantly better than random, at a confidence level of 95%.

As depicted in table 1, the SONeFMUC classified better the classes of phragmites, wet meadows, and water bodies. The high accuracy in these classes is attributed to the small spectral overlapping and within class variance. On the other hand, poor classification results were achieved for the tamarix and trees classes. Specifically, tamarix was strongly confused with phragmites since their spectral signatures are very similar. Moreover, a similar situation was confronted in class trees, which was confused with tamarix and phragmites; these classes represent vegetation cover with similar spectral characteristics.

Table 1.

4.1.2 Application of the SONeFMUC to the agricultural zone. The SONeFMUC model selected for that case is a six-layer network, using five fuzzy sets for each *FPD* input, crisp type rules, and the min fuser. The model provided an overall accuracy of 74.21% on the testing data set. Table 2 hosts the confusion matrix for the agricultural zone, including the corresponding statistical parameters.

The SONeFMUC classifier exhibits a moderate agreement between the remotely sensed

classification and the reference data since the Khat value falls in the range $[0.4, 0.8]$. As expected, the value of Z-score is large enough to support that the method is significantly better than random selection with confidence level 95%. The overall classification accuracy was found smaller than the one attained in wetland zone. A possible reason may be the larger number of classes, as compared to the wetland zone. Nevertheless, SONeFMUC exhibits an average producer's accuracy of 82.6% on three major classes (maize, alfalfa and cereals) implying that they are classified more accurately than others, although the dominant classes are overestimated, as shown from the user's accuracy percentages. On the contrary, orchards, vegetables, and shrubs are underestimated with poor producer's accuracy. The reason is that these classes contain different subclasses. For instance, vegetables may be tomatoes, watermelon or eggplants, a situation recognized with field work, thus leading to a large number of misclassifications. A mosaic of land cover map of both zones is illustrated in figure 13(a) which depicts the result of the SONeFMUC.

Table 2.

4.2 Comparison with MLC

In order to validate the classification results of the SONeFMUC algorithm, the IKONOS image was classified using MLC, a traditional spectral probabilistic 'hard' classifier.

4.2.1 MLC classification of the wetland zone. Table 3 includes the confusion matrix provided by MLC on the testing data. A high overall performance of 84.51% is obtained, being though 5% lower than the one of SONeFMUC. The Khat statistic was found 0.74, suggesting a moderate agreement between the remotely sensed classification and the reference data, as opposed to the SONeFMUC where the agreement was found very strong. Additionally, the Z-test was calculated as 25.48 which is significantly greater than Z_c showing that classification is better than random.

MLC exhibits a good performance on classifying wet meadows and water. However, compared to the SONeFMUC, phragmites classification gave poor results; most omission errors could be found with tamarix, although phragmites cover a large area of the wetland. A low level of accuracy is also achieved in trees and tamarix as in the SONeFMUC algorithm, which shows the inefficiency of any of these two algorithms in classifying these classes.

Table 3.

4.2.2 MLC classification of the agricultural zone. The MLC was also applied in the agricultural zone, with the error confusion matrix shown in table 4. A moderate overall accuracy of 71% is obtained, slightly lower than the one of SONeFMUC (approximately 74%). Khat is 0.65 and Z-score 33.84, indicating a fair agreement but suggesting that the classification result is not random.

Based on the statistical parameters, the MLC algorithm performed similarly to the SONeFMUC model in the agricultural zone. However, useful information about the performance of the two classification methods could be derived by analysing further the error confusion matrix. For example, in the first three classes (maize, cereals, and alfalfa), the dominant classes in the zone, the mean producer's accuracy of the MLC is lower (72.6%) than SONeFMUC's (82.6%), indicating an underestimation of MLC in these classes. On the other hand, fallow is classified better in MLC than SONeFMUC, although this class is confused in both classifiers with cereals. The reason is that they are spectrally similar due to the fact that the satellite image was acquired on August when cereals are harvested and fallows represent low vegetation cover. A mosaic of land cover map of both zones is illustrated in figure 13(b), showing the results obtained by MLC.

Figure 13(a),(b).

Table 4.

4.3 Discussion of the results

The classification algorithms (SONeFMUC and MLC) were applied to the wetland and agricultural zones of the satellite image, providing differing results. In the wetland zone, the SONeFMUC was found most suitable because the overall accuracy was higher and Khat showed a higher reliability of the results. Furthermore, the Z-test between the two methods demonstrates that their performance was significantly different in the wetland zone (see table 5).

The results followed a similar trend in the agricultural zone, where SONeFMUC offered a higher overall accuracy and Khat. However, there was no sufficient statistical evidence that SONeFMUC performed significantly different. This may be due to the larger number of classes in the agricultural zone compared to the wetland zone (eight and five, respectively), and the similar spectral characteristics of the crops. However, as regards the dominant classes, the SONeFMUC was found more suitable than MLC which had overestimated them.

Useful information was derived using a visual assessment of land cover maps produced by both

classifiers. Figure 14 illustrates a region of thematic map in the agricultural zone, showing the superior performance of the suggested approach. SONeFMUC yields less pixel misclassifications, thus producing more homogeneous fields. Moreover, MLC exhibits a large confusion between alfalfa and maize (blue and red colour respectively) in many fields, a situation not confronted in SONeFMUC.

A similar visual assessment can be carried out in the wetland zone, as illustrated in figure 15. The overestimation of tamarix against phragmites by MLC can be observed, a result appearing also in the confusion matrix hosted in table 3. SONeFMUC classified uniformly the stretch phragmites which is the correct land cover type. Additionally, in the lake's shore where the depth of the water is low, a large amount of pixels was classified errroneously by MLC as phragmites and wet meadows; these pixels though belong to water, as correctly labelled by SONeFMUC.

Owing to the structure learning (GMDH), the proposed method accomplished the feature selection task. The network generated for the wetland zone used as inputs only eight out of 26 original features. The subset of significant features selected by GMDH includes: bands 1-3, the ASM co-occurrence feature from the third band, greenness and brightness from the tasseled cap features, and intensity and hue from the IHS features. In the agricultural zone, the resulting network used only 12 out of the 26 initial features. These features consist of the four bands, correlation from the first band, homogeneity from the third band, ASM from the fourth band, the three tasseled cap features, and intensity and hue from the IHS features. On the contrary, the maximum likelihood classifier doesn't even provide this capability; therefore feature selection is based on the designer's experience or on a time consuming trial-and-error process.

Table 5.

Figure 14.

Figure 15.

5. Conclusions

In this paper, the SONeFMUC is proposed and applied to land cover classification of a Very High Resolution image in a protected area of high ecological interest. To improve classification accuracy, the image was divided into two zones: the wetland zone, where land cover corresponded to five habitat classes, and the agriculture zone with eight crop classes. The performance of

SONeFMUC is contrasted to the one the MLC classifier.

The structure of SONeFMUC is expanded in a self-organizing manner via the GMDH algorithm. At the end of structure learning, the appropriate network is recovered including the most informative feature as inputs (feature selection). The node models of the network are regarded as generic fuzzy rule-based classifiers, performing successive feature transformations and decision making. The decision outputs of the fuzzy neuron classifiers are combined using a fusion scheme. The fuser serves also as a means to discriminate those patterns that are well classified by the parent FNC's and the ones which require further investigation by the neurons in the following layer.

A high classification accuracy of 89.5% is obtained by SONeFMUC in the wetland zone (Khat = 0.83). The suggested classifier was able to discriminate the dominant habitat classes: phragmites, wet meadows, and water bodies. In the agricultural zone, the performance of SONeFMUC was lower, showing an overall accuracy of 74.21% (Khat = 0.67). The basic reason for this was the large spectral overlapping between the crop classes together with larger number of classes as compared to the wetland zone. The dominant classes of the agricultural zone (alfalfa, maize and cereals) were discriminated in a satisfying degree, since the average producer's accuracy was 82.6%. Finally, it should be noticed that the proposed method was able to select the appropriate input features for each zone, leading to higher classification accuracy.

Performance comparisons with MLC verified the efficiency of the proposed SONeFMUC. In both zones, SONeFMUC achieved higher overall accuracy and Khat than MLC. Particularly, in the wetland zone the overall accuracy of MLC was 5% lower and the Khat revealed poorer classification quality. In addition, the Z–test between the two methods proved that their performance was significantly different. In the agricultural zone, the quality statistics of MLC was lower but comparable to the SONeFMUC. Nevertheless, MLC underestimated the dominant crops, contrary to SONeFMUC, making the use of the suggested method more appropriate.

Apart from the initial bands, more informative features are extracted from the multispectral image, namely, textural and spectral features. Future work will explore the capabilities of our network using wavelet transformation as additional input.

Acknowledgements

This study was funded by 'Pythagoras II', a research grant awarded by the Managing Authority of

the Operational Programme 'Education and Initial Vocational Training' of Greece, which is partially funded by the European Social Fund – European Commission.

References

ALEXANDRIDIS, T., TAKAVAKOGLOU, V., ZALIDIS, G. and CRISMAN, L. T., 2006, Remote sensing and GIS techniques for selecting a self-sustainable scenario for Lake Koronia. *Environmental Management*, (in press).

ATKINSON, M. P. and TATNALL, L. R. A., 1997, Introduction: Neural networks ion remote sensing. *International Journal of Remote Sensing*, **18**, 699–709.

BÁRDOSSY, A. and SAMANIEGO, L., 2002, Fuzzy rule-based classification of remotely sensed imagery. *IEEE Transactions on Geosience and Remote Sensing*, **40**, 362–374.

BASTIN, L., 1997, Comparison of fuzzy c-means classification, linear mixture modeling and MLC probabilities as tools for unmixing coarse pixels. *International Journal of Remote Sensing*, **18**, 3629–3648.

BENEDIKTSSON, J. A., SWAIN, P. H. and ESROY, O. K., 1993, Conjugate-gradient neural networks in classification of multisource and very-high-dimensional remote sensing data. *International Journal of Remote Sensing*, **14**, 2883–2903.

BENEDIKTSSON, J. A. and KANELLOPOULOS, I., 1999, Classification of multisource and hyperspectral data based on decision fusion. *IEEE Transactions on Geosience and Remote Sensing*, **37**, 1367–1377.

BRIEM, J. G., BENEDIKTSSON, J. A. and SVEINSSON, R. J., 2002, Multiple classifiers applied to multisource remote sensing data. *IEEE Transactions on Geosience and Remote Sensing*, **40**, 2291–2299.

CARPENTER, A. G., GJAJA, N. M., GOPAL, S. and WOODCOCK, E. C., 1997, ART neural networks for remote sensing: vegetation classification from Landsat TM and terrain data. *IEEE Transactions on Geosience and Remote Sensing*, **35**, 308–325.

CHAVEZ, P. S. Jr., SIDES, S. C. and ANDERSON, J. A., 1991, Comparison of Three Different Methods to Merge Multiresolution and Multispectral Data: Landsat TM and SPOT Panchromatic. *Photogrammetric Engineering & Remote Sensing*, **57**, 295–303.

CONGALTON, R. G. and GREEN, K., 1999, *Assesing the Accuracy of Remotely Sensed Data: Principles and Practices.* (Boca Raton: Lweis Publishers).

DEKKER, J. R., 2000, Texture analysis and classification of ERS SAR images for map updating of urban areas in the Netherlands. *IEEE Transactions on Geosience and Remote Sensing*, **41**, 1950–1958.

FOODY, G. M., 1995, Land cover classification using an artificial neural network with ancillary information. *International Journal of Geographical Information Systems*, **9**, 527–542.

GIACINTO, G. and ROLI, F., 1997, Ensembles of neural networks for soft classification of remote sensing images. In *Proceedings of the European Symposium on Intelligent Techniques*, 19–21 March 1997, Bari, Italy, pp. 166–170.

GOODWIN, C. G. and SIN, S. K., 1984, *Adaptive filtering and control* (New Jersey: Prentice–Hall).

HARALICK, M. R., SHANMUGAN, K. and DINSTEIN, I., 1973, Textural features for image classification. *IEEE Transactions on Systems, Man and Cybernetics*, **3**, 610–621.

HARALICK, M. R. and SHAPIRO, G. L., 1992, *Robot and computer vision: vol. 1* (Boston: Addison-Wesley), pp. 457–462.

Hellenic Ministry of Environment, Physical Planning and Public Works: Environmental Planning Division, 2001. Identification and description of habitat types in areas of interest for nature conservation. Final Report, Athens, Greece.

HORNE, J. F., 2003, A tasseled cap transformation for IKONOS images. In *Proceedings of ASPRS Annual Conference*, 5–9 May 2003, Anchorage, Alaska.

HUTCHINSON, M. F., 1988, Calculation of hydrologically sound digital elevation models. In *Third International Symposium on Spatial Data Handling*, 17-19 August, Sydney, (Ohio: International Geographical Union), pp. 117–133.

IVAKHNENKO, G. A., 1968, The group method of data handling; a rival of method of stochastic approximation. *Soviet Automatic Control*, **1-3**, 43–45.

KAUTH, R. J. and THOMAS, G. S., 1976, The Tasseled Cap – A graphical description of the spectral-temporal development of agricultural crops as seen by Landsat. In *Proceedings of the Symposium on Machine Processing of Remotely Sensed Data*, 29 June–1 July 1976, West Lafayette, Indiana, (New York: Institute of Electrical and Electronics Engineering), pp. 4B41–4B51.

KAWATA, Y., ONTANI, A., KUSAKA, T. and UENO, S., 1990, Classification accuracy for the MOS – 1 MESSR data before and after the atmospheric correction. *IEEE Transactions on Geoscience Remote Sensing*, **28**, 755–760.

KERAMITSOGLOU, I., SARIMVEIS, H., KIRANOUDIS, T. C. and SIFAKIS, N., 2005, Radial basis function neural networks classification using very high spatial resolution satellite imagery: an application to the habitat area of Lake Kerkini (Greece). *International Journal of Remote Sensing*, **26**, 1861–1880.

KUMAR, S. A., BASU, K. S. and MAJUMDAR, L. K., 1997, Robust classification of multispectral data using multiple neural networks and fuzzy integral. *IEEE Transactions on Geosience and Remote Sensing*, **35**, 787–790.

KUNCHEVA, I. L., BEZDEK, C. J. and DUIN, W. P. R., 2001, Decision templates for multiple classifier fusion: an experimental comparison. *Pattern Recognition*, **34**, 299–314.

LEE, M. H., CHEN, M. C., CHEN, M. J. and JOU, L. Y., 2001, An efficient fuzzy classifier with feature selection based on fuzzy entropy. *IEEE Transactions on Systems, Man and Cybernetics*, **31**, 426–432.

LI, X. and YEH, A. G. O., 1998, Principal component analysis of stacked multi-temporal images for the monitoring of rapid urban expansion in the Pearl River Delta. *International Journal of Remote Sensing*, **19**, 1479–1500.

LIN, T.-C., LEE, C.-Y. and PU, C.-H., 2000, Satellite sensor image classification using cascade architecture of neural fuzzy network. *IEEE Transactions on Geosience and Remote Sensing*, **38**, 1033–1043.

MITRAKI, C., CRISMAN, L. T. and ZALIDIS, G., 2004, Lake Koronia: Shift from autotrophy to heterotrophy with cultural eutrophication and progressive water-level reduction. *Limnologica*, **34**, 110–116.

OETTER, D. R., COHEN, W. B., BERTERRETCHE, M. K., MAIERSPERGER, T. K. and KENNEDY, R. E., 2001, Land cover mapping in an agricultural setting using multiseasonal Thematic Mapper data. *Remote Sensing of Environment*, **76,** 139–155.

PETRAKOS, M., BENEDIKTSSON, J. A. and KANELLOPOULOS, I., 1999, The effect of classifier agreement on the accuracy of the combined classifier in decision level fusion. *IEEE Transactions on Geosience and Remote Sensing*, **39**, 2539–2546.

RICHARDS, J. A., LANDGREBE, D. A. and SWAIN, P. H., 1982, A means for utilizing ancillary information in multispectral classification. *Remote Sensing of Environment*, **12**, 463–477.

SHACKELFORD, K. A. and DAVIS, H. C., 2003, A hierarchical fuzzy classification approach for high-resolution multispectral data over urban areas. *IEEE Transactions on Geosience and Remote Sensing*, **41**, 1920–1932.

SONG, C., WOODCOCK, C. E., SOTO, K. C., LENNEY, M. P. and MACOMBER, A. S., 2001, Classification and change detection using Landsat TM data: When and how to correct atmospheric effects. *Remote sensing of environment*, **75**, 230–244.

TAKAGI, T. and SUGENO, M., 1985, Fuzzy identifications and its application to modeling and control. *IEEE Transactions on Systems, Man and Cybernetics*, **15**, 116–132.

THOMAS, I. L., BENNING, V. M. and CHING, N. P., 1987, *Classification of Remotely Sensed Images* (Bristol: Adam Hilger).

TSO, B. and MATHER, M. P., 2001, *Classification Methods for Remotely Sensed Data* (London: Taylor & Francis).

WANG, F., 1990, Fuzzy supervised classification of remote sensing images. *IEEE Transactions on Geosience and Remote Sensing*, **28**, 194–201.

WILKINSON, G. G., FIERENS, F. and KANELLOPOULOS, I., 1995, Integration of neural and statistical approaches in spatial data classification. *Geographical Systems*, **32**, 1–20.

ZALIDIS, C. G., TAKAVAKOGLOU, V., and ALEXANDRIDIS, T., 2004, Revised restoration plan of Lake Koronia. Aristotle University of Thessaloniki, Department of Agronomy, Laboratory of Applied Soil Science, Thessaloniki (in Greek, English summary).

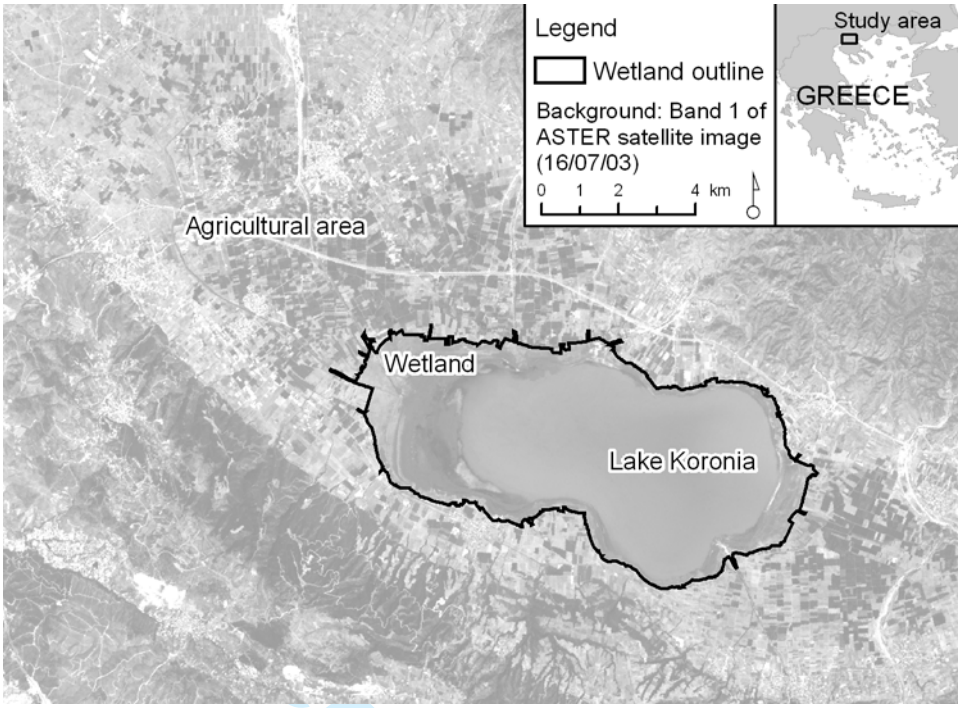


Figure 1. Location map and main land cover of the study area.

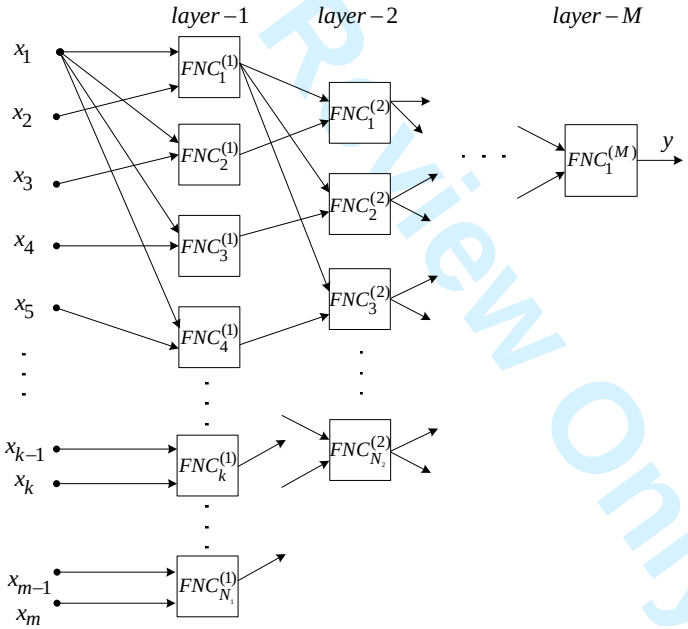


Figure 2. A general SONeFMUC architecture.

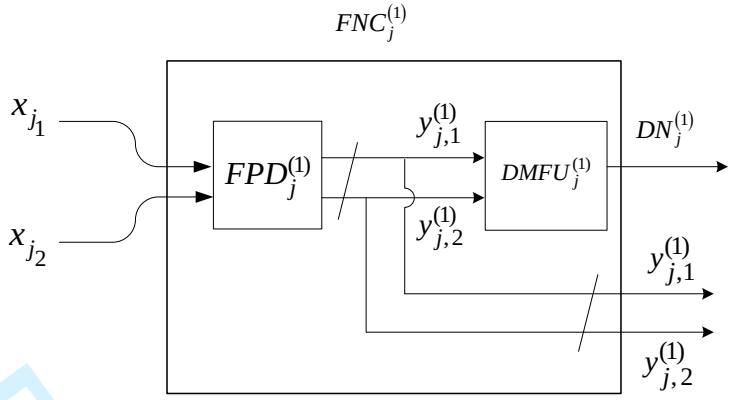


Figure 3. An example of the structure of a node in the first layer of the network. The node incorporates two inputs, two continuous outputs, a FPD unit and a DMFU unit.

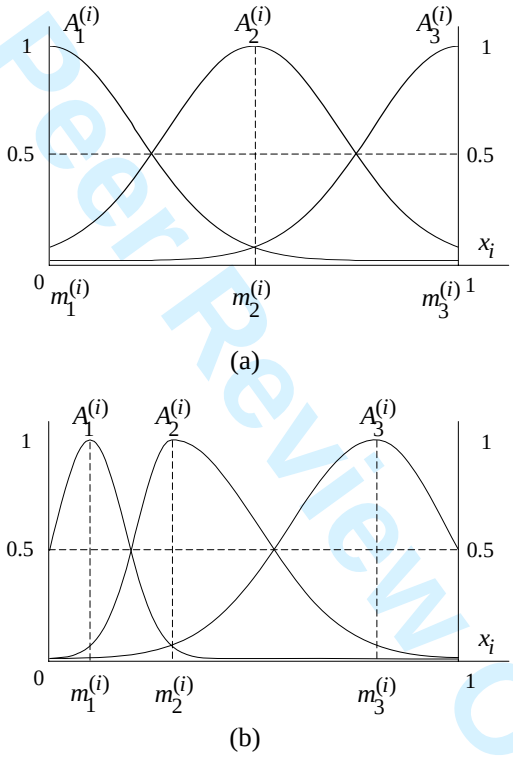


Figure 4. (a) Initial arrangement of membership functions with $K_i = 3$, (b) membership functions with tuned centres using *K*-means clustering.

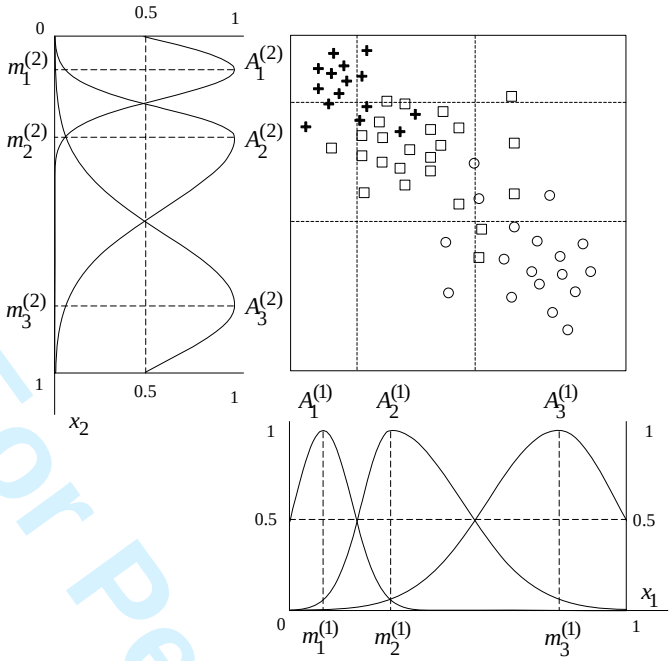


Figure 5. Premise partition after tuning the membership functions using the K-means clustering algorithm.

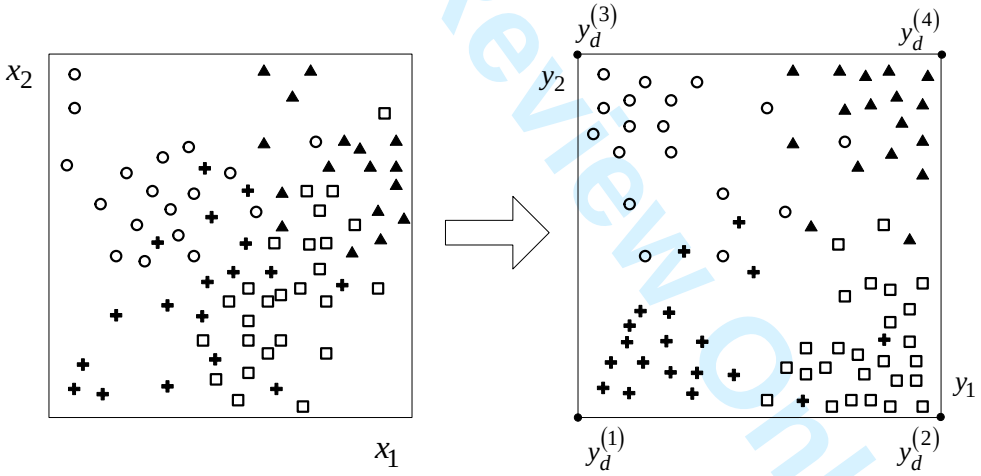


Figure 6. An illustrative graphical representation of the non-linear mapping implemented by the FPD unit for an artificial problem of $M = 4$ classes. Class one patterns are represented by a cross, class two by a square, class three by a circle and class four by a triangle.

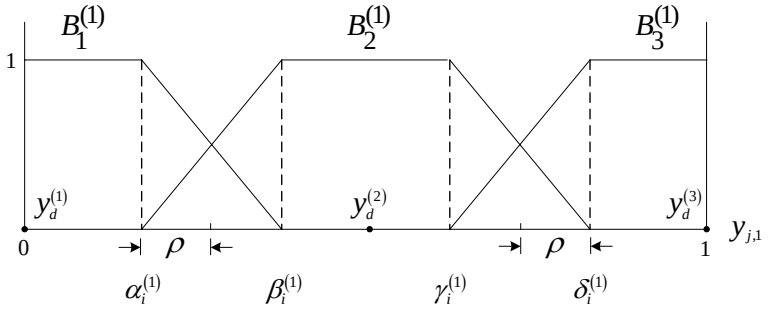


Figure 7. Shape of the membership functions in the premise part of the DMFU.

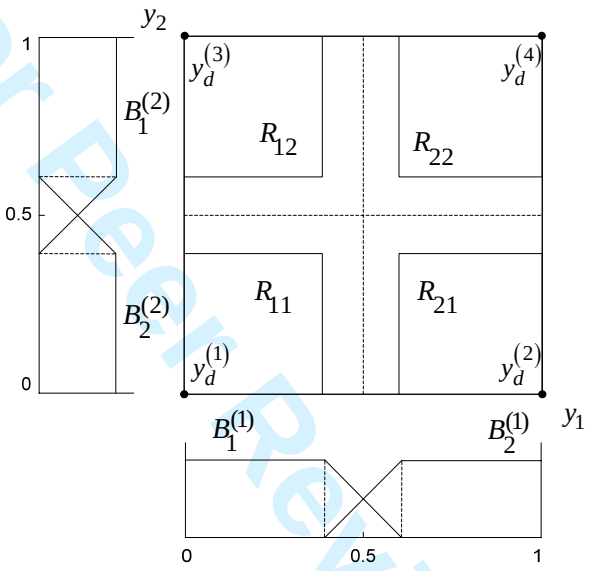


Figure 8. Partition of the DMFU input space for $M = 4$ classes.

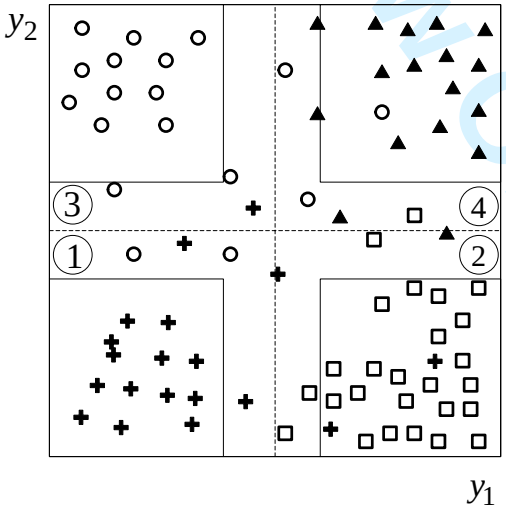


Figure 9. Graphical representation of the class decision regions designed by the DMFU unit, for the artificial problem of 4 classes.

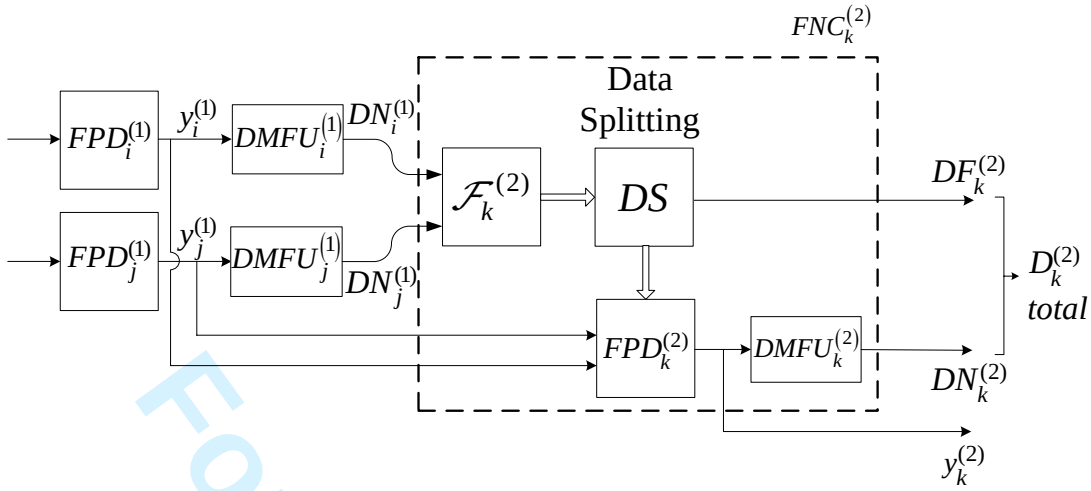


Figure 10. Neuron structure of nodes in the $\ell = 2$ layer of the network.

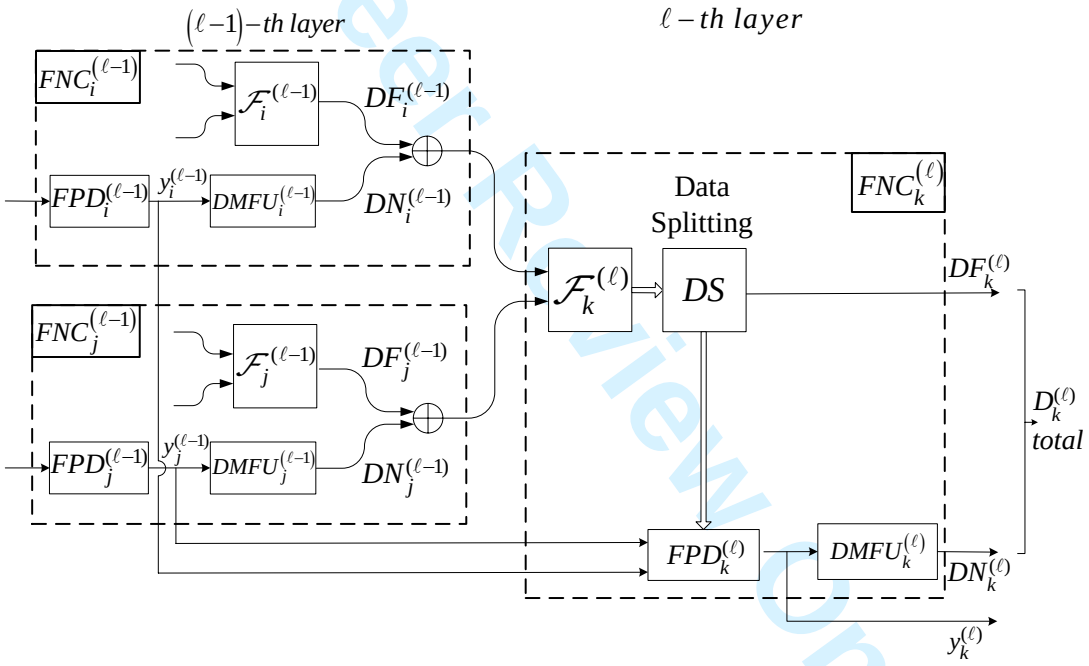


Figure 11. Neuron structure of nodes in layers $\ell \geq 3$.

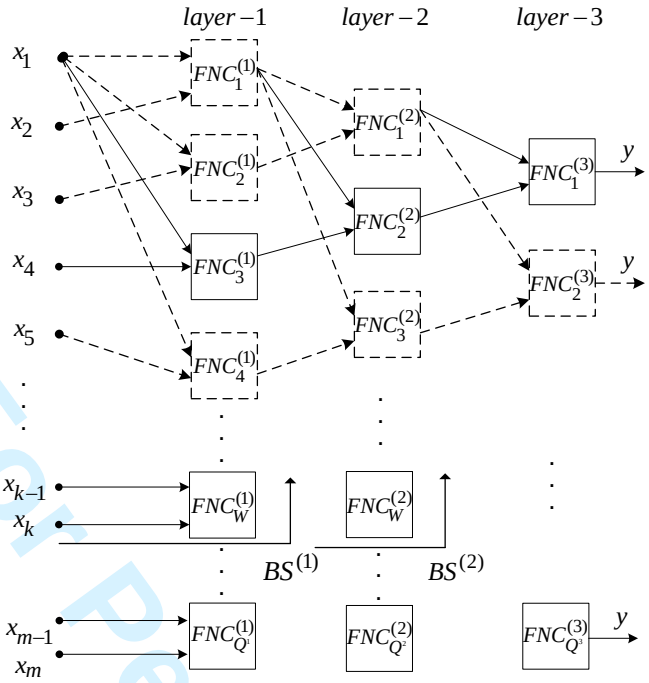
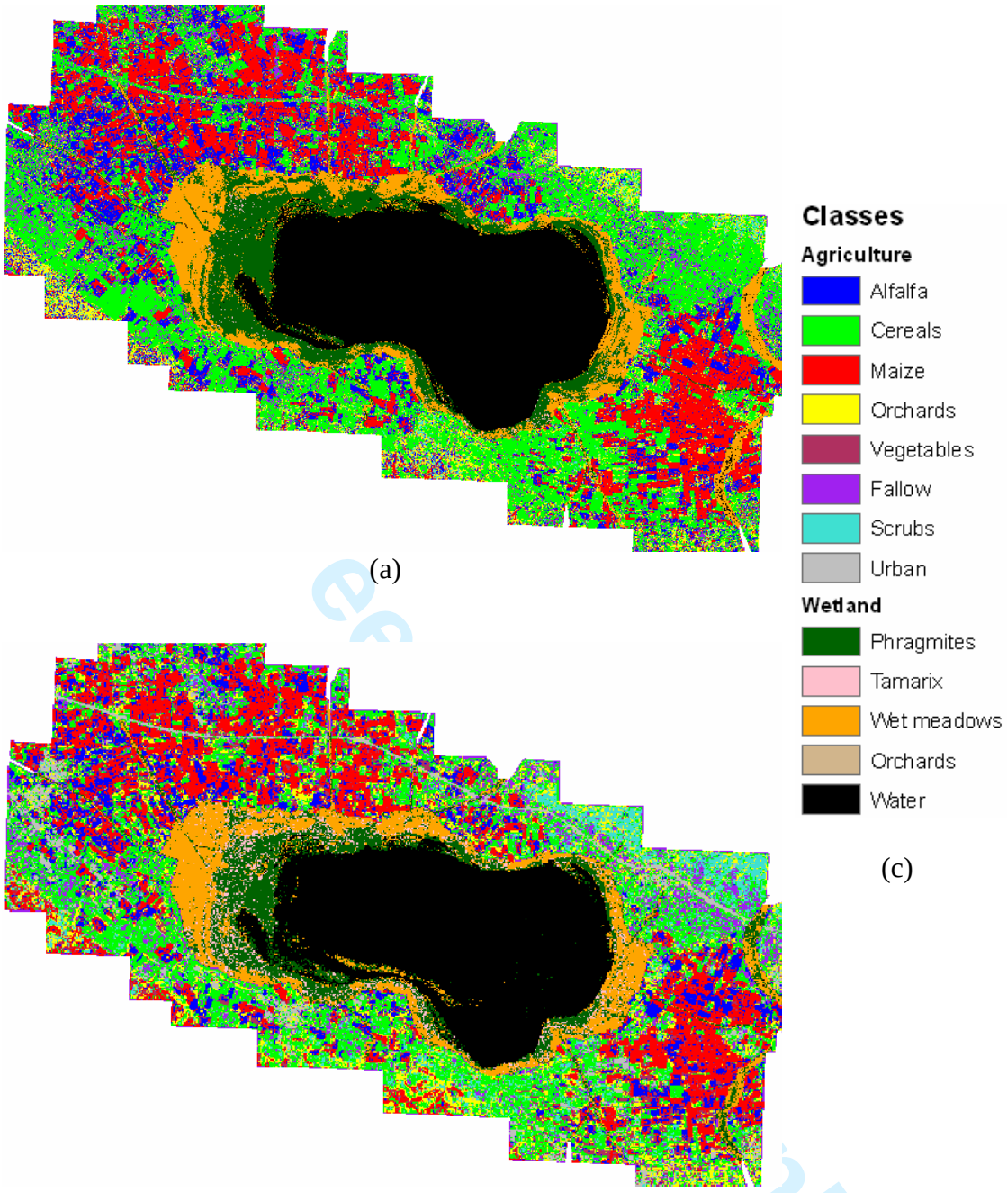


Figure 12. Example of the Structure Learning process by means of the GMDH algorithm.

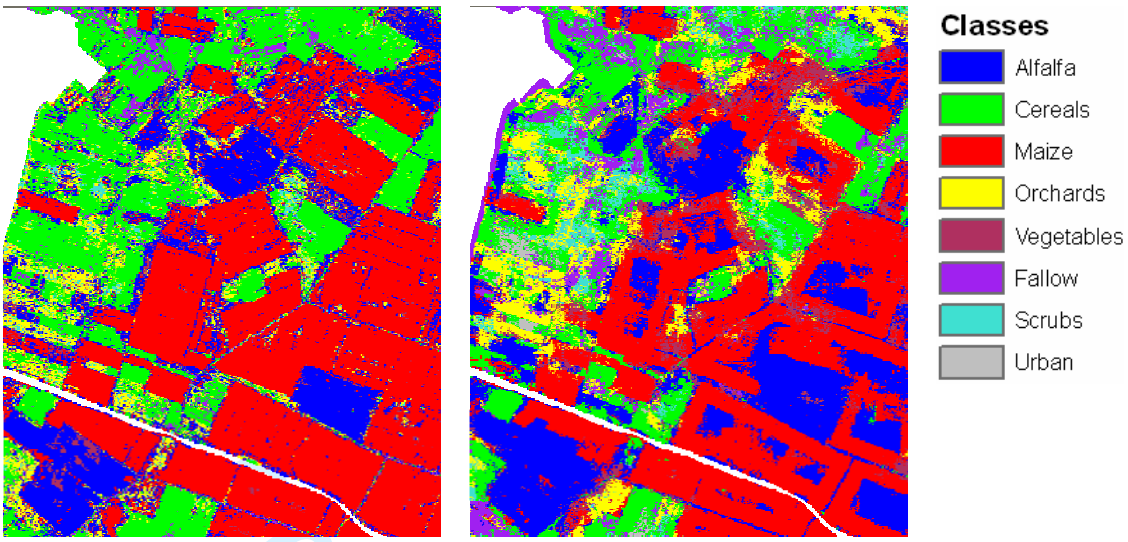


(a)

(b)

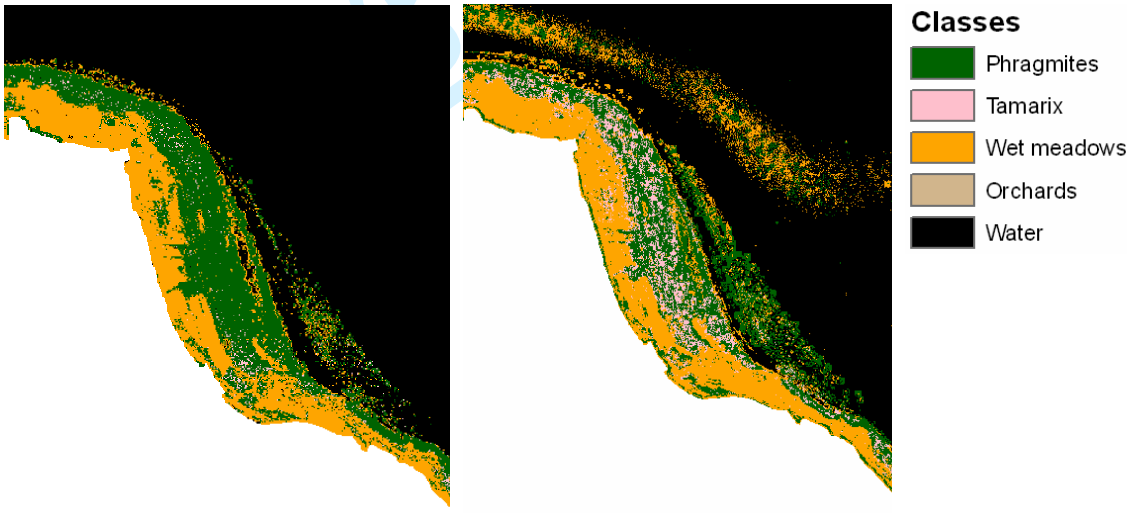
(c)

Figure 13. Mosaic of IKONOS land cover classification of wetland and agricultural zone using SONeFMUC (a) and MLC (b). The legend is presented in (c)



(a) (b) (c)

Figure 14. A subset of the land cover map produced with SONeFMUC (a), and MLC (b) in the agricultural zone east of the wetland. Legend is given in (c).



(a) (b) (c)

Figure 15. A subset of the land cover map produced with SONeFMUC (a), and MLC (b) in the wetland zone south of the lake. Legend is given in (c).

Table 1. Confusion matrix obtained by the application of the SONeFMUC to the testing data set of the wetland zone.

	Wetland map classification						
SONeFMUC	Phragmites	Tamarix	Wet meadows	Trees	Water Bodies	PA (%)	UA (%)
Phragmites	102	12	7	3	0	90.27	82.26
Tamarix	7	17	2	3	0	54.84	58.62
Wet meadows	4	1	198	0	0	95.65	97.54
Trees	0	1	0	3	0	33.33	75.00
Water Bodies	0	0	0	0	21	100.0	100.0
Reference	113	31	207	9	21		

Overall Acccuracy=89.5%: Khat=0.83: Z-score=33.14

Table 2. Confusion matrix obtained by the application of the SONeFMUC to the testing data set of the agricultural zone.

	Agriculture map classification									
SONeFMUC	Alfalfa	Cereals	Maize	Orchards	Vegetables	Fallow	Shrubs	Urban	PA (%)	UA (%)
Alfalfa	149	8	23	3	12	3	2	0	74.13	74.50
Cereals	18	190	2	5	4	23	14	2	90.48	73.64
Maize	21	1	147	8	11	0	1	0	83.05	77.78
Orchards	6	1	2	21	2	1	1	0	50.00	61.76
Vegetables	5	1	3	2	15	0	0	0	32.61	57.69
Fallow	2	8	0	1	2	33	2	2	55.00	66.00
Shrubs	0	1	0	2	0	0	7	0	25.93	70.00
Urban	0	0	0	0	0	0	0	28	87.50	100.0
Reference	201	210	177	42	46	60	27	32		

Overall Acccuracy=74.21%: Khat=0.67: Z-score=34.83

Table 3. Confusion matrix obtained by the application of the MLC to the testing data set of the wetland zone.

	Wetland map classification						
MLC	Phragmites	Tamarix	Wet meadows	Trees	Water Bodies	PA (%)	UA (%)
Phragmites	87	10	9	7	0	76.99	76.99
Tamarix	19	15	1	0	0	48.39	42.86
Wet meadows	7	6	197	0	0	95.17	93.81
Trees	0	0	0	2	0	22.22	100.00
Water Bodies	0	0	0	0	21	100.00	100.00
Reference	113	31	207	9	21		

Overall Acccuracy=84.51%: Khat=0.74: Z-score=25.48

Table 4. Confusion matrix obtained by the application of the MLC to the testing data set of the agricultural zone.

	Agriculture map classification									
MLC	Alfalfa	Cereals	Maize	Orchards	Vegetables	Fallow	Shrubs	Urban	PA (%)	UA (%)
Alfalfa	119	7	5	0	4	0	0	0	59.20	88.15
Cereals	25	162	3	6	4	19	4	0	77.14	72.65
Maize	27	0	144	2	5	0	0	0	81.36	80.90
Orchards	12	7	15	34	6	3	1	0	80.95	43.59
Vegetables	18	3	10	0	22	0	0	0	47.83	41.51
Fallow	0	23	0	0	5	38	3	0	63.33	55.07
Shrubs	0	3	0	0	0	0	19	0	70.37	86.36
Urban	0	5	0	0	0	0	0	32	100.0	86.49
Reference	201	210	177	42	46	60	27	32		

Overall Acccuracy=71.7%: Khat=0.65: Z-score=33.69

Table 5. Comparison of the classification performance of the SONeFMUC with the MLC in wetland and agricultural zone, using statistical parameters and Z-test performance.

	Wetland		Agricultural	
	SONeFMUC	MLC	SONeFMUC	MLC
Overall Accuracy	89.5%	85.04%	74.21%	71.7%
Khat	0.83	0.75	0.67	0.65
Z-score	33.14	26.21	34.84	3.69
Z-test	2.18*		0.72*	

*95% confidence level