Decision Fusion of GA Self-Organizing Neuro-Fuzzy Multilayered Classifiers for Land Cover Classification Using Textural and Spectral Features

Nikolaos E. Mitrakis, *Student Member, IEEE*, Charalampos A. Topaloglou, Thomas K. Alexandridis, John B. Theocharis, *Member, IEEE*, and George C. Zalidis

***Abstract*—A novel Self-Organizing Neuro-Fuzzy Multilayered Classifier, the GA-SONeFMUC model, is proposed in this paper for land cover classification of multispectral images. The model is composed of generic fuzzy neuron classifiers (FNCs) arranged in layers, which are implemented by fuzzy rule-based systems. At each layer, parent FNCs are combined to generate a descendant FNC at the next layer with higher classification accuracy. To exploit the information acquired by the parent FNCs, their decision supports are combined using a fusion operator. As a result, a data splitting is devised within each FNC, distinguishing those pixels that are currently correctly classified to a high certainty grade from the ambiguous ones. The former are handled by the fuser, while the ambiguous pixels are further processed to enhance their classification confidence. The GA-SONeFMUC structure is determined in a self-constructing way via a structure-learning algorithm with feature selection capabilities. The parameters of the models obtained after structure learning are optimized using a real-coded genetic algorithm. For effective classification, we formulated three input sets containing spectral and textural feature types. To explore information coming from different feature sources, we apply a classifier fusion approach at the final stage. The outputs of individual classifiers constructed from each input set are combined to provide the final assignments. Our approach is tested on a lake-wetland ecosystem of international importance using an IKONOS image. A high-classification performance of 92.02% and of 75.55% for the wetland zone and the surrounding agricultural zone is achieved, respectively.**

***Index Terms*—Classifier fusion, fusion operators, genetic algorithms (GAs), image classification, neuro-fuzzy classifiers, remote sensing.**

I. Introduction

OVER the past decades, various types of classifiers based on computational intelligence have been reported in the literature for land cover image classification. Traditionally, neural network (NN) classifiers are suggested [1], [2], relying mainly on their mapping and adaptation capabilities. Considerable research has been directed toward devising fuzzy classifiers [3], [4] that provide soft decisions regarding the class to which a pixel belongs. Fuzzy theory offers an efficient framework for handling the uncertainty encountered in mixed pixels and the diverse types of textures in remote sensing problems. Recently, genetic algorithms (GAs) have been applied to multispectral classification with promising results [5]. GAs are also used for parameter optimization of NN classifiers [6] for classification improvement. Additionally, hybrid architectures have been developed [7]–[9], exploiting the merits of fuzzy logic and NN domains.

Combining classifiers is an important and promising research field [10]. In the literature, it has been given several names, including classifier fusion, multiple classifier systems, mixture of experts, classifier ensembles, and voting pool of classifiers. The basic idea behind this methodology is that better classification results can possibly be obtained by combining different information sources. In this context, there are generally two main approaches: classifier selection and classifier fusion. In classifier selection, each classifier is regarded as an expert in some local area of the feature space, thus working complementarily with regard to the others.

In classifier fusion, though, all classifiers are trained over the whole feature space; hence, they are considered on a competitive basis. The individual classifiers are arranged in a single layer, with their outputs combined via a fusion operator. Within this framework, several varieties can be considered, such as the classifier type, the model structure, the learning algorithm used for classifier training, the multispectral source acquired, and the feature type extracted. Along this line, multisensor data were used in [11] and [12], with the fusion scheme operating on the outputs of statistical and/or NN classifiers; whereas, in [13] a conjugate gradient NN classifier and a fuzzy classifier were combined for urban classification using IKONOS images. For the classification of multisensor data in [14], support vector machine (SVM) classifiers were used to classify each data source, and another SVM was trained on the *a priori* outputs of the SVMs in order to perform the decision fusion. In [15], multiple classifiers were developed incorporating multisensor

Manuscript received July 19, 2007; revised October 23, 2007. This work was funded by Pythagoras II, a research grant awarded by the Managing Authority of the Operational Program "Education and Initial Vocational Training" of Greece, supported in part by the European Social Fund—European Commission.

N. E. Mitrakis and J. B. Theocharis are with the Department of Electrical and Computer Engineering, Aristotle University of Thessaloniki, 541 24 Thessaloniki, Greece (e-mail: theochar@eng.auth.gr).

C. A. Topaloglou, T. K. Alexandridis, and G. C. Zalidis are with the Faculty of Agronomy, Aristotle University of Thessaloniki, 541 24 Thessaloniki, Greece.

Color versions of one or more of the figures in this paper are available online athttp://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TGRS.2008.916481

and geographical data. The same classifier type (i.e., NN) was applied to multispectral data in [16], leading to encouraging results. Recently, classifier fusion methods have been developed that use different feature types, such as spectral and textural features [17].

In [18], a voting/rejection approach is proposed, being a two-stage decision structure. At stage 1, a maximum-likelihood classifier (MLC) in parallel with an NN classifier is combined. Assuming that both classifiers agree on the class label of a sample, the sample is classified to that class. Otherwise, the sample is rejected. At stage 2, an independent NN classifier is trained to classify the ambiguous pixels of the first stage, providing the final class decision for these pixels. A modification of this method is proposed in [19] by using a multisource classifier based on consensus theory instead of an MLC.

In this paper, a multilayered neuro-fuzzy classifier is suggested, the GA-SONeFMUC model, for classification of multispectral images. The model is composed of interconnected elementary fuzzy neuron classifiers (FNC) organized in a multilayered feed-forward structure. Each FNC performs three major tasks: 1) decision fusion of parent FNCs at the previous layer; 2) data splitting that distinguishes between well-classified patterns and ambiguous patterns; and 3) classification enhancement of the ambiguous patterns through a small-scale neuro-fuzzy classifier. The GA-SONeFMUC structure is expanded in a self-constructing fashion by using a structure-learning algorithm with feature selection capabilities, based on the principles of Group Method of Data Handling [20]. After structure learning, parameter optimization is conducted via a real-coded genetic algorithm to improve further classification accuracy.

The generic FNCs incorporate the notions of parent classifier combination and subsequent classification of the voting/rejection scheme [19]. Our classifier, though, extends the above ideas with regard to the following aspects. At the FNC level, we use a family of fusion operators to combine the soft decision supports of the parent FNCs; whereas, in [19], simple aggregation rules are used operating on the hard decisions of the combined classifiers. At the structural level, instead of confining to a two-stage structure of voting/rejection, our classifier expands to a multilayered architecture, whereby FNCs are combined both in parallel and in sequential forms. Furthermore, GA-SONeFMUC extends the classifier fusion approach [10]. Instead of combining large-scale classifiers arranged in a single layer, generic FNC combination is expanded here for multiple layers.

Our model is applied to a lake-wetland ecosystem in Greece of particular ecological importance. Performance comparisons are given for the wetland zone and the surrounding agricultural zone based on an IKONOS image. Different feature sets are used, including the gray values of the four bands, features from a co-occurrence matrix, and wavelet and transformed spectral features. A systematic procedure is suggested for selecting the proper window size and angle required in feature calculations.

Apart from the internal use of fusers within each FNC, we also apply a classifier fusion method at the final stage. Independent classifiers are developed and trained from different feature sets. The outputs of the individual classifiers are then combined through a set of fusion operators, further improving the classification result.

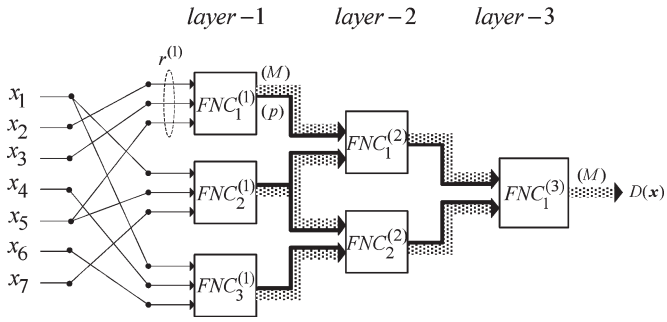


Fig. 1. GA-SONeFMUC architecture with $m = 7$, $L = 3$ layers.

II. GA-SONeFMUC Architecture

The suggested neuro-fuzzy classifier is a multilayered structure, comprising a number of $\ell = 0, \ldots, L$ layers with the ℓth layer including N_ℓ neurons. Neurons are defined as fuzzy neuron classifiers, denoted by $FNC_j^{(\ell)}$, $j = 0, \ldots, N_\ell$, $\ell = 0, \ldots, L$. The input layer $\ell = 0$ includes the feature components $x_1, \ldots, x_m$, while the output layer $\ell = L$ comprises the output node $FNC_1^{(L)}$. Fig. 1 shows an example of a three-layered $(L = 3)$ GA-SONeFMUC architecture composed of six FNCs $(N_1 = 3, N_2 = 2, N_3 = 1)$.

The classifier's design is accomplished by supervised learning based on a training data set of N labelled pairs: $D_N = \{(\boldsymbol{x}[q], C[q]), q = 1, \ldots, N\}$, where $\boldsymbol{x}[q] \in \Re^m$ denotes the feature vector, $C[q]$ is the class label, and $C = \{C_1, \ldots, C_M\}$ is the set of M classes. For convenience, the features are normalized in the range [0, 1], forming an overall feature space $F = [0, 1]^m$.

Parent FNCs are combined to generate a descendant FNC at the next layer, with better classification capabilities. Each FNC handles all patterns in D_N and provides two outputs: a decision support vector of length M (dotted thick line) and a transformed feature vector of length p (solid thick line). (The value of p will be determined in Section III-A of this paper.) In this respect, the connective links can be regarded as data buses transferring information from one layer to the succeeding one. The ending $FNC_1^{(L)}$ produces only a decision vector $D(\boldsymbol{x}) = [d_1(\boldsymbol{x}), \ldots, d_M(\boldsymbol{x})]^T$ that represents the overall output of GA-SONeFMUC, where $d_j(\boldsymbol{x}) \in [0, 1]$, $j = 1, \ldots, M$, denotes the certainty grade of $\boldsymbol{x}$ in class C_j. Furthermore, the FNCs at higher layers $(\ell = 2, \ldots, L)$ take as inputs the aforementioned two output vectors of the parent FNCs at the previous layer; hence, they receive a total of $2 \times (M + p)$ components. Especially, the FNCs at layer 1 take $r^{(1)}$ inputs, which are derived through combinations of the original feature components (single lines).

Unlike conventional classifiers where the class association is performed by a single model, the classification task is sequentially achieved here through repeated decision fusion, feature transformation, and decision making along the layers of GA-SONeFMUC.

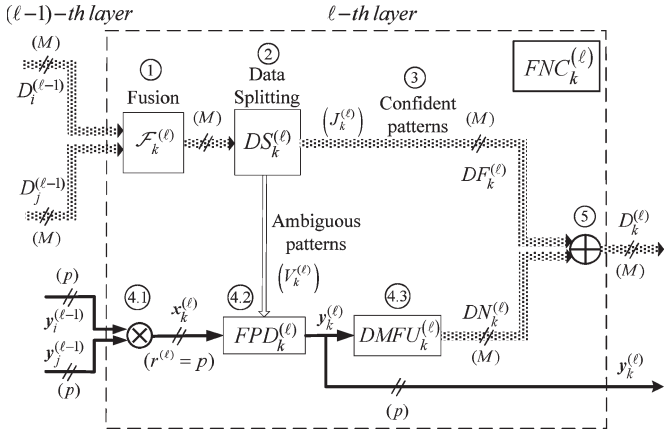


Fig. 2. General structure of the kth FNC at layer ℓ.

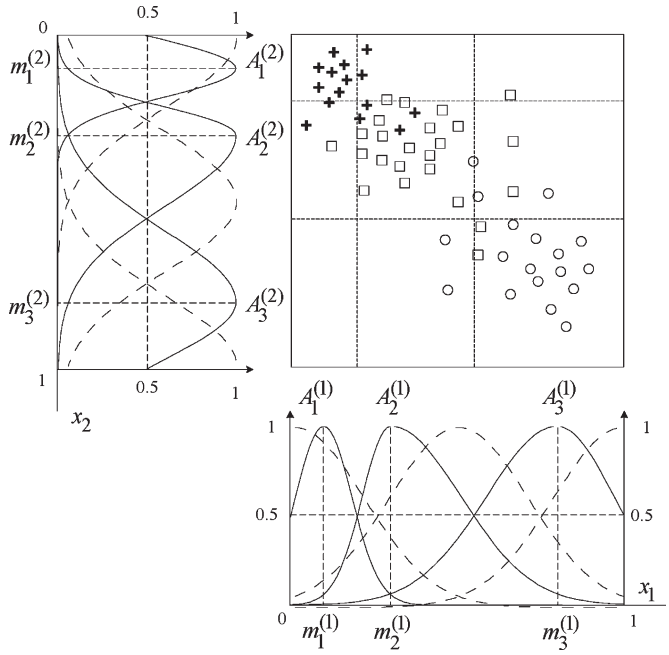


Fig. 3. Premise partition on a 2-D input space with $K_i = 3$ and $M = 3$ classes. Dotted lines represent initial membership functions. Solid lines represent the final shape after K-means clustering.

III. Neuron Model Description

The structure of an elementary neuron model classifier $FNC_k^{(\ell)}$ is shown in Fig. 2. Each FNC comprises four modules: $FNC_k^{(\ell)} = \{\mathcal{F}_k^{(\ell)}, DS_k^{(\ell)}, FPD_k^{(\ell)}, DMFU_k^{(\ell)}\}$. The fuser $(\mathcal{F}_k^{(\ell)})$ combines the decision outputs of the parent FNCs, while the associated unit $DS_k^{(\ell)}$ performs a data splitting into well-separated patterns and ambiguous patterns. The pair of modules $\{FPD_k^{(\ell)}, DMFU_k^{(\ell)}\}$ implements a neuro-fuzzy fuzzy classifier within each $FNC_k^{(\ell)}$, which is used to improve the accuracy of ambiguous patterns. The $FPD_k^{(\ell)}$ part performs a feature transformation through supervised learning; whereas, $DMFU_k^{(\ell)}$ produces soft decision supports for the pixels.

A. Fuzzy Partial Description (FPD)

The $FPD_k^{(\ell)}$ units $\ell = 1, \ldots, L$ are represented by a Takagi-Sugeno (TS) fuzzy system [21] with $r^{(\ell)}$ inputs, $\boldsymbol{x}_k^{(\ell)} \in \Re^{r^{(\ell)}}$, and p outputs, $\boldsymbol{y}_k^{(\ell)} \in \Re^p$, defined on normalized ranges, $x_{k,i}^{(\ell)} \in \aleph_{k,i} = [0,1]$, $i = 1, \ldots, r^{(\ell)}$ and $y_{k,j}^{(\ell)} \in Y_{k,j} = [0,1]$, $j = 1, \ldots, p$. The input-output components form the input space $\mathcal{I}_k^{(\ell)} = \aleph_{k,1} \times \cdots \times \aleph_{k,r^{(\ell)}} = [0,1]^{r^{(\ell)}}$ and output space $\mathcal{S}_k^{(\ell)} = Y_{k,1} \times \cdots \times Y_{k,p} = [0,1]^p$ of the $FPD_k^{(\ell)}$, respectively. The input vector $\boldsymbol{x}_k^{(\ell)}$ is formed by aggregating the outputs of the parent $FNC_i^{(\ell-1)}$ and $FNC_j^{(\ell-1)}$ from the previous layer: $\boldsymbol{x}_k^{(\ell)} = \boldsymbol{x}_i^{(\ell-1)} \otimes \boldsymbol{x}_j^{(\ell-1)}$ (see Section III-D).

The output space $\mathcal{S}_k^{(\ell)}$ is a p-dimensional hypercube that represents the class space (i.e., the space where the classes are defined). The value of p(and, hence, the number of FPD outputs) is determined as a function of the number of classes:

$$p = ceil(\log_2 M). \tag{1}$$

Definition of classes entails that for each class a target value is assigned for all output variables: $\boldsymbol{y}_d^{(C_j)} = [y_{d,1}^{(C_j)}, \ldots, y_{d,p}^{(C_j)}]$, $j = 1, \ldots, M$. The class targets are located at the 2^p vertices of the hypercube $\mathcal{S}_k^{(\ell)}$. When $2^p > M$, some classes share a number of neighboring vertices. Instead of using M separate FPD outputs (one for each class), we form, by using (1), a class space of considerably lower dimensionality $(p < M)$, especially for large M. For instance, for a 16-class problem, only $p = 4$ FPD outputs are required. The small space size reduces the complexity of FPD models and their computational load during learning.

Omitting for simplicity the layer index ℓ, let the input/output vectors of FPDs be denoted as $\boldsymbol{x} = [x_1, \ldots, x_r]^T$ and $\boldsymbol{y} = [y_1, \ldots, y_p]^T$. Each premise variable x_i, $i = 1, \ldots, r$ is described in terms of K_i fuzzy sets, forming the term set $TS_i = \{A_1^{(i)}, A_2^{(i)}, \ldots, A_{K_i}^{(i)}\}$. The fuzzy sets are represented by two-sided Gaussian membership functions (MF) located at a center value $m_j^{(i)}$, $j = 1, \ldots, K_i$

$$\mu_{j,t}^{(i)}(x_i) = \exp\left(-\left(x_i - m_j^{(i)}\right)^2 \Big/ \left(\sigma_{j,t}^{(i)}\right)^2\right), \; t = \{L, R\} \tag{2}$$

where $\sigma_{j,L}^{(i)}$, $\sigma_{j,R}^{(i)}$ are the widths of the left/right parts.

Considering a grid-type partition of the premise space, a total number of $R = \prod_{i=1}^{r} K_i$ fuzzy rectangular subspaces are formed, $\mathcal{A}^{(i)} = A_{i_1}^{(i)} \times \cdots \times A_{i_r}^{(i)}$, $A_{i_j}^{(i)} \in TS(x_i)$ (Fig. 3). Defining a fuzzy rule for each $\mathcal{A}^{(i)}$, the FPDs are described by R T-S rules of the form

$$R_m^{(i)} : \text{IF } x_1 \text{ is } A_{i_1}^{(1)} \text{AND} \cdots \text{AND } x_r \text{ is } A_{i_r}^{(r)}$$
$$\text{THEN } y_1^{(i)} = g_1^{(i)}(\boldsymbol{x}) \text{ AND} \cdots \text{AND } y_p^{(i)} = g_p^{(i)}(\boldsymbol{x}). \tag{3}$$

Traditionally, the consequent functions of the T-S fuzzy model are described by linear polynomials of the input variables

$$g_j^{(i)}(\boldsymbol{x}) = w_{0,j}^{(i)} + w_{1,j}^{(i)} x_1 + \ldots + w_{r,j}^{(i)} x_r, \quad j = 1, \ldots, p. \tag{4}$$

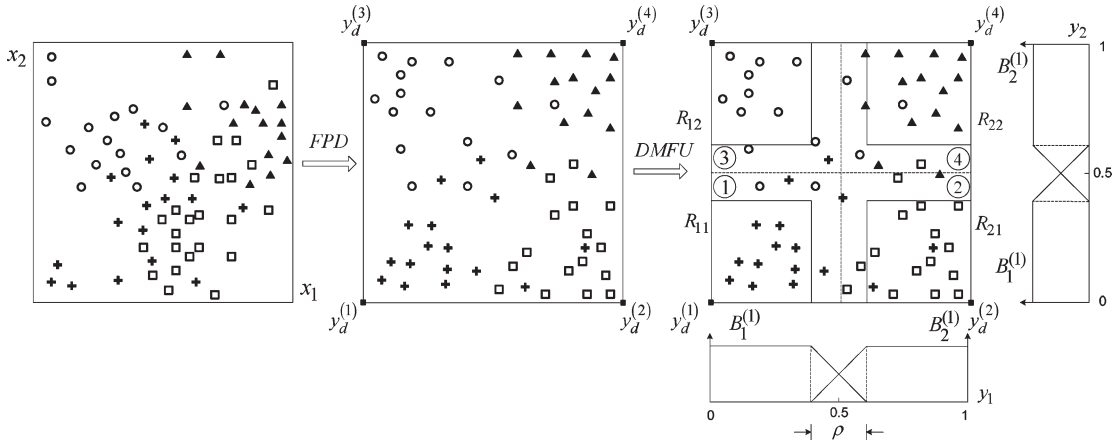


Fig. 4. Feature transformation of an FPD and the premise partition of a DMFU.

A simplified rule form with crisp consequents can be obtained from (4) by retaining only the constant term: $g_j^{(i)}(\boldsymbol{x}) = w_{0,j}^{(i)} j = 1, \ldots, p$.

The outputs of the FPD model are given as

$$y_j = \sum_{i=1}^{R} \bar{\mu}_i(\boldsymbol{x}) \cdot g_j^{(i)}(\boldsymbol{x}), \qquad j = 1, \ldots, p \tag{5}$$

where $\bar{\mu}_i(\boldsymbol{x})$ are the normalized firing strengths of the rules derived by

$$\bar{\mu}_i(\boldsymbol{x}) = \frac{\mu_i(\boldsymbol{x})}{\sum_{i=1}^{R} \mu_i(\boldsymbol{x})} \qquad \mu_i(\boldsymbol{x}) = \prod_{j=1}^{r} \mu_{i_j}^{(i)}(x_j). \tag{6}$$

The structure and parameters of $FPD_k^{(\ell)}$ are determined using the following FPD learning procedure.

F.1) *Rule Base Formulation:* Given $K_i,\ i = 1, \ldots, r^{(\ell)}$, formulate R fuzzy rules of the form (3), (4). Initially, the MF centers are uniformly placed within $\aleph_i$, while the membership widths are determined so that consecutive fuzzy sets exhibit a 0.5 degree of overlapping.

F.2) *Premise Partition:* To improve feature transformation of FPDs, we apply the K-means clustering method [22] on the MF centers with the goal of locating MFs at those regions exhibiting high-pattern concentrations. The initial MFs and their final shapes after data clustering are shown in Fig. 3.

F.3) *Rule Base Simplification:* To reduce the number of consequent parameters and the complexity of the FPD model, a rule reduction method is applied. For each rule, calculate the number of patterns included in the antecedent part of the rule, with a degree of firing fulfilling $\mu_i(\boldsymbol{x}) \geq 0.5$. Those rules that cover a number of patterns greater than a prescribed threshold (i.e., 5% of the training patterns) are retained; the rest ones are discarded.

F.4) *Consequent Weights Estimation:* Since the FPD outputs are linear with respect to the polynomial coefficients [see (4)–(6)], we calculate the optimal estimates of the consequent weights $w_{k,j}^{(i)}$ by using the recursive least squares estimation (RLSE) algorithm [23].

F.5) *FPD Outputs:* Compute the FPD outputs $\boldsymbol{y}_k^{(\ell)}$ by using (5).

For a particular input $\boldsymbol{x}_k^{(\ell)}$, the $FPD_k^{(\ell)}$ model performs a nonlinear mapping: $\mathcal{I}_k^{(\ell)} \overset{FPD_k^{(\ell)}}{\rightarrow} \mathcal{S}_k^{(1)},\ k = 1, \ldots, N_\ell$. The outputs $\boldsymbol{y}_k^{(\ell)} = FPD_k^{(\ell)}(\boldsymbol{x}_k^{(\ell)})$ can be regarded as nonlinear transformations of the original features $x_k^{(\ell)}$, defined on the output space $\mathcal{S}_k^{(\ell)}$. Given the class targets, a supervised learning task is accomplished with the following objective: patterns $\boldsymbol{x}_k^{(\ell)}[q]$ belonging to class C_j should produce an output $\boldsymbol{y}_k^{(\ell)}[q]$ located in a neighborhood of the respective class target $y_d^{(C_j)}$, $j = 1, \ldots, M$. Feature transformation facilitates discrimination of the patterns along the classes, thus leading to more accurate classifications results by $DMFU_k^{(\ell)}$. Fig. 4 shows an illustrative case of a $M = 4$ problem. Using (1), two FPD outputs $(p = 2)$ are required, and the class targets are set to $\boldsymbol{y}_d^{(C_i)} = \{(0,0), (1,0), (0,1), (1,1), i = 1, \ldots, 4\}$ at the vertices of a 2-D hypercube. The FPD transforms the original features to the output space where the patterns are closer to their target values, thus exhibiting a higher degree of class discrimination.

B. Decision Making Fuzzy Unit (DMFU)

The decision making fuzzy unit $DMFU_k^{(\ell)}$ is a fuzzy rule-based system operating on the space $\mathcal{S}_k^{(\ell)}$ of the transformed features. The outputs $\boldsymbol{y}_k^{(\ell)}$ of $FPD_k^{(\ell)}$ serve as inputs to $DMFU_k^{(\ell)}$ (Fig. 2). For the patterns $\boldsymbol{x}$ handled by $FPD_k^{(\ell)}$, $DMFU_k^{(\ell)}$ produces a soft decision vector, including the degrees of support in each class: $DN_k^{(1)}(\boldsymbol{x}) = [dn_{k,1}^{(1)}(\boldsymbol{x}), dn_{k,2}^{(1)}(\boldsymbol{x}), \ldots, dn_{k,M}^{(1)}(\boldsymbol{x})]^T$. Hence, it performs a soft classification mapping: $class : \mathcal{S}_k^{(\ell)} \rightarrow [0,1]^M$.

Each DMFU input $y_j, j = 1, \ldots, p$ is partitioned into two fuzzy sets $\{B_1^{(j)}, B_2^{(j)}\}$ that are represented by trapezoidal membership function centered at the target values of the classes (Fig. 4). Along each axis, we allow a transition between adjacent fuzzy sets, controlled by an overlapping threshold ρ.

This threshold is defined as a small portion of the universe of discourse, taking values in the range [0.20, 0.40]. Regions with overlapping fuzzy sets are ambiguous regions between the classes, providing low degrees of certainty for the pixels. On the contrary, pixels belonging to the rectangular regions surrounding the class targets are well classified to a high degree of confidence.

Combining fuzzy sets along the inputs of DMFU, we form a partition comprising rectangular fuzzy spaces: $\mathcal{B}^{(i)} = B_{i_1}^{(1)} \times \ldots \times B_{i_p}^{(p)}, i = 1, \ldots, 2^p$. For each $\mathcal{B}^{(i)}$, a fuzzy classification rule is applied of the form

$$R_c^{(i)}: \text{ IF } y_1 \text{ is } B_{i_1}^{(1)} \text{ AND} \ldots \text{AND } y_p \text{ is } B_{i_p}^{(p)} \text{ THEN } (y_1[q], \ldots, y_p[q]) \text{ is } C_i, \quad C_i \in C. \quad (7)$$

The fuzzy inference of $DMFU_k^{(\ell)}$ proceeds as follows.

D.1) Calculate the degree of firing of the classes $\beta_{C_i}, i = 1, \ldots, M$

$$\beta_{C_i} = \mu_{i_1}^{(1)}(y_1(q)) \wedge \mu_{i_2}^{(2)}(y_2(q)) \wedge \ldots \wedge \mu_{i_p}^{(2)}(y_p(q)). \quad (8)$$

D.2) Compute the normalized firings as follows:

$$\bar{\beta}_{C_i} = \beta_{C_i} \Big/ \sum_{r=1}^{M} \beta_{C_r}. \quad (9)$$

D.3) Calculate the soft decision output vector of DMFU:

$$DN = [dn_1, dn_2, \ldots, dn_M]^T = [\bar{\beta}_{C_1}, \bar{\beta}_{C_2}, \ldots, \bar{\beta}_{C_M}]^T. \quad (10)$$

The elements $dn_j \in [0, 1]$ denote the degree of support given by the FNC that pattern $\boldsymbol{y}[q]$, a transformed version of the pattern $\boldsymbol{x}[q]$, belongs to class C_j.

C. Aggregation of Previous FPD Outputs

FNCs at layer 1 of the GA-SONeFMUC operate on the original feature set, $\{x_1, \ldots, x_m\}$ (Fig. 1). Instead of receiving the whole attribute set, the input vector $\boldsymbol{x}_k^{(1)}$ of $FPD_k^{(1)}$ contains a small subset of $r^{(1)} = 2, 3$ features taken from the above set: $\boldsymbol{x}_k^{(1)} = [x_{k_1}, \ldots, x_{k_{r^{(1)}}}]^T \in \Re^{r^{(1)}}$. The particular features serving as inputs to each FPD are determined via a structure learning algorithm (described in Section V). Further, since the FNCs in layer 1 obtain no previous decision evidence, fusion is not applied at this layer. The entire training data set is passed and processed by the pair of modules $\{FPD_k^{(1)}, DMFU_k^{(1)}\}$, $k = 1, \ldots, N_1$.

The inputs of the FNCs at higher layers ($\ell \geq 2$) are determined by combining the outputs of two parent FNCs at the previous layer. In order to reduce the complexity of the FPD models and to relax their parameter learning demands, the input vector components $\boldsymbol{x}_k^{(\ell)} = [x_{k_1}, \ldots, x_{k_p}]^T$ of $FPD_k^{(\ell)}$ at layers $\ell \geq 2$ are formed as a weighted average of the outputs $\boldsymbol{y}_i^{(\ell-1)}$, $\boldsymbol{y}_j^{(\ell-1)}$ associated with the parent $FPD_i^{(\ell-1)}$ and $FPD_j^{(\ell-1)}$, as follows:

$$\boldsymbol{x}_k^{(\ell)} = \boldsymbol{y}_i^{(\ell-1)} \otimes \boldsymbol{y}_j^{(\ell-1)} = \frac{w_1}{w_1 + w_2} \boldsymbol{y}_i^{(\ell-1)} + \frac{w_2}{w_1 + w_2} \boldsymbol{y}_j^{(\ell-1)}. \quad (11)$$

The weights represent the performance of the parent FNCs on a validation data set: $w_1 = E_{i,vl}^{(\ell-1)}$, $w_2 = E_{j,vl}^{(\ell-1)}$. The aggregation rule (11) confines the input size to $r^{(\ell)} = p$ for $\ell \geq 2$. It suggests that $\boldsymbol{x}_k^{(\ell)}$ takes values on a line connecting the points $\boldsymbol{y}_i^{(\ell-1)}$ and $\boldsymbol{y}_j^{(\ell-1)}$, being closer to output of the more qualifying among the parent FNCs.

D. FNC-Level Classification

In order to exploit prior decision evidence, a fusion unit is introduced in each FNC that integrates the soft-decision outputs of its parent FNC at the previous layer. Based on the fuser outcomes, a data-splitting mechanism is developed, which discriminates pixels into correctly classified and ambiguous ones. Data splitting allows focusing on those patterns for which adequate degree of confidence is not achieved yet, and improve their accuracy. This provides an effective data control, leading to computational savings for large data sets.

The data flow within each $FNC_k^{(\ell)}$ can be in five stages (Fig. 2).

Stage 1) *Fusion:* The soft decision outputs derived from parent FNCs at layer $(\ell - 1)$, $D_i^{(\ell-1)}$, and $D_j^{(\ell-1)}$ are fused as

$$DF_k^{(\ell)}(\boldsymbol{x}) = \mathcal{F}_k^{(\ell)} \left\{ D_i^{(\ell-1)}(\boldsymbol{x}), D_j^{(\ell-1)}(\boldsymbol{x}) \right\} \quad (12)$$

where $\mathcal{F}_k^{(\ell)}$ denotes a fusion operator. The resulting decision vector contains the grades of certainty for every pattern to all classes

$$DF_k^{(\ell)}(\boldsymbol{x}) = \left[df_{k,1}^{(\ell)}(\boldsymbol{x}), df_{k,2}^{(\ell)}(\boldsymbol{x}), \ldots, df_{k,M}^{(\ell)}(\boldsymbol{x}) \right]^T \quad (13)$$

with $df_{k,r}^{(\ell)} \in [0, 1]$, $r = 1, \ldots, M$. (The type of fusers used in this paper will be discussed in Section IV.) The quality of parent classifiers is ascertained by comparing the $df_{k,r}^{(\ell)}$ values to a user-defined threshold. Assuming that

$$df_{k,r}^{(\ell)}(\boldsymbol{x}) \geq \vartheta \quad (14)$$

pattern $\boldsymbol{x}$ is classified to a class with a degree of certainty higher than ϑ (i.e. $\vartheta = 0.8$).

Stage 2) *Data Splitting:* Based on (14), data splitting $(DS_k^{(\ell)})$ is applied locally at each $FNC_k^{(\ell)}$, decomposing the data set D_N into two subsets: $J_k^{(\ell)}$, $V_k^{(\ell)}$ with $D_N = J_k^{(\ell)} \cup V_k^{(\ell)}$. $J_k^{(\ell)}$ includes patterns fulfilling (14), which are currently well classified with high grade of certainty; whereas, $V_k^{(\ell)}$ contains patterns that are either misclassified or correctly classified with an inadequate level of confidence.

Stage 3) *Confident Patterns Decisions:* Patterns $\boldsymbol{x} \in J_k^{(\ell)}$ are handled by the fuser $\mathcal{F}_k^{(\ell)}$, with their decision supports given by $DF_k^{(\ell)}(\boldsymbol{x})$.

Stage 4) *Ambiguous Patterns Decisions:* Patterns $\boldsymbol{x} \in V_k^{(\ell)}$ are submitted to the local neuro-fuzzy classifier $\{FPD_k^{(\ell)}, DMFU_k^{(\ell)}\}$ to improve their discrimination quality.

a) Aggregate the parent FNC outputs through (11) to compute the input vector $\boldsymbol{x}_k^{(\ell)}$. If $DS_i^{(\ell-1)}$ of the parent $FNC_i^{(\ell-1)}$ has assigned $\boldsymbol{x} \in V_i^{(\ell-1)}$, then $\boldsymbol{y}_i^{(\ell-1)}$ is taken as the output of $FPD_i^{(\ell-1)}$. Further, when $\boldsymbol{x} \in J_i^{(\ell-1)}$, then we set $\boldsymbol{y}_i^{(\ell-1)} = \boldsymbol{y}_d^{(C_i)}$: the target value of the class decided by the fuser $\mathcal{F}_i^{(\ell-1)}$ for that pixel.

b) Enter $\boldsymbol{x}_k^{(\ell)}$ to $FPD_k^{(\ell)}$ and perform steps F1–F5 of the FPD learning (Section III-A) to obtain the transformed features $\boldsymbol{y}_k^{(\ell)} = FPD_k^{(\ell)}\{\boldsymbol{x}\}$.

c) Input $\boldsymbol{y}_k^{(\ell)}$ to $DMFU_k^{(\ell)}$ and perform steps D1–D3 to compute the decision output $DN_k^{(\ell)}(\boldsymbol{x})$ (Section III-B).

Stage 5) *Decision Aggregation:* Due to data splitting, the overall decision output of $FNC_k^{(\ell)}$, denoted as $D_k^{(\ell)}(\boldsymbol{x}) = [D_{k,1}^{(\ell)}(\boldsymbol{x}), \ldots, D_{k,M}^{(\ell)}(\boldsymbol{x})]^T$, is formulated by integrating the above two sources of evidence, as follows:

$$D_k^{(\ell)}(\boldsymbol{x}) = DF_k^{(\ell)}(\boldsymbol{x}) \oplus DN_k^{(\ell)}(\boldsymbol{x}). \tag{15}$$

For every $\boldsymbol{x}$, the crisp decision as to the class each pattern belongs can be obtained by

$$class(\boldsymbol{x}) = C_r \Rightarrow d_{k,C_r}^{(\ell)}(\boldsymbol{x}) = \max_{j=1,\ldots,M} \left\{ d_{k,j}^{(\ell)}(\boldsymbol{x}) \right\} \tag{16}$$

where $d_{k,j}^{(\ell)}(\boldsymbol{x}) = df_{k,j}^{(\ell)}(\boldsymbol{x})$ if $\boldsymbol{x} \in J_k^{(\ell)}$ and $d_{k,j}^{(\ell)}(\boldsymbol{x}) = dn_{k,j}^{(\ell)}(\boldsymbol{x})$ if $\boldsymbol{x} \in V_k^{(\ell)}$.

Each $FNC_k^{(\ell)}$ provides two types of information (Fig. 2): the "continuous" outputs $\boldsymbol{y}_k^{(\ell)}$ and the decision supports $D_k^{(\ell)}$ given either by $DMFU_k^{(\ell)}(DN_k^{(\ell)})$ or by $\mathcal{F}_k^{(\ell)}(DF_k^{(\ell)})$. Additionally, hard decisions for the patterns are obtained, which are used for evaluating the classification performance of $FNC_k^{(\ell)}$.

IV. Fusion Schemes

The fusion unit combines a set of parent classifiers to construct a descendant classifier. Suppose G parent classifiers $\{FNC_1, \ldots, FNC_G\}$ are available. The decision vector DF is calculated by fusing the classifier outputs

$$DF = \mathcal{F}\{D_1, \ldots, D_G\} \tag{17}$$

where $DF = [df_1, \ldots, df_M]^T$, $D_i = [d_{i,1}, \ldots, d_{i,M}]^T$, $i = 1, \ldots, G$.

The classifier outputs can be organized in a decision profile (DP), given by a $G \times M$ matrix

$$DP(\boldsymbol{x}[q]) = \begin{bmatrix} d_{1,1} & \ldots & d_{1,M} \\ \vdots & & \vdots \\ d_{G,1} & \ldots & d_{G,M} \end{bmatrix}. \tag{18}$$

Fusion methods using only the information of column j to derive the decision support for class j are called class-conscious methods; whereas, methods that handle the whole content of the DP matrix are called class-indifferent methods. In this paper, we employ three class-conscious fusion operators: the minimum, the weighted average, and the fuzzy integral. Furthermore, the decision templates are also used belonging to the class-indifferent fusers.

A. Fuzzy Integral

A fuzzy integral (FI) provides a means for combining objective evidence in the form of decision supports offered by the FNCs and subjective evaluation of the importance of the parent classifiers [16]. To calculate the decision support df_j for class C_j, we associate a vector of fuzzy measures $g = [g(1), \ldots, g(M)]$ with column j of DP and proceed with the following six steps.

Step 1) Sort the components of the jth column of $DP(\boldsymbol{x})$ in descending order: $d_{i_1,j}, \ldots, d_{i_G,j}$, with $d_{i_1,j}$ denoting the highest degree of certainty.

Step 2) The fuzzy densities $g^i, i = 1, \ldots, G$ represent the degree of importance of the parent FNCs toward the final evaluation. Assuming that p_i denotes the performance attained by each FNC using the class C_j patterns on a validation set, the densities are calculated by

$$g^i = p_i \Big/ \sum_{j=1}^{G} p_j, \qquad i = 1, \ldots, G, \sum_{i=1}^{G} g^i = 1. \tag{19}$$

Step 3) Sort fuzzy densities $g^{i_1}, \ldots, g^{i_G}$, following the order selected in the first step.

Step 4) Calculate the unique root $\lambda \geq -1$ of the polynomial

$$\lambda + 1 = \prod_{i=1}^{G} (1 + \lambda g^i). \tag{20}$$

Step 5) Set $g(1) = g^1$, and calculate the rest $(G-1)$ fuzzy measures using the following recursive equation:

$$g(k) = g^k + g(k-1) + \lambda g^k g(k-1), 2 \leq k \leq G. \tag{21}$$

Step 6) The degree of support of class C_j is given by

$$df_j(x) = \max_{k=1}^{G} \{\min\{d_{k,j}(x), g(k)\}\}. \tag{22}$$

B. Minimum (MIN) and Weighted Average (wAVG) Fusers

The minimum (MIN) fuser is the simplest and yet the most conservative fusion method; the degree of support for class C_j is given by

$$df_j(\boldsymbol{x}) = MIN\{d_{1,j}, \ldots, d_{G,j}\}. \tag{23}$$

The wAVG fuser uses the degree of importance of the parent classifiers

$$df_j(\boldsymbol{x}) = w_1 d_{1,j} + \cdots + w_g d_{G,j} \tag{24}$$

where $w_i = g^i, i = 1, \ldots, G$, as determined in (19).

C. Decision Templates (DT)

In the decision templates (DT) approach [10], the multiple classifier outputs are compared to a characteristic template for each class. Assuming a labeled training set D_N, the decision template $DT_j(\boldsymbol{X})$ for class C_j is defined as a $G \times M$ matrix, with its elements calculated as

$$dt_j(k,s)(\boldsymbol{D}_N) = \frac{\sum_{k=1}^{N} Ind\left(\boldsymbol{x}[k], j\right) d_{k,s}\left(\boldsymbol{x}[k]\right)}{\sum_{k=1}^{N} Ind\left(\boldsymbol{x}[k], j\right)} \tag{25}$$

where $k = 1, \ldots, G$ and $s = 1, \ldots, M$. $Ind(\boldsymbol{x}[k], j)$ take a value of 1 if $\boldsymbol{x}[k]$ has a crisp label j and is set to 0 otherwise. $DP(\boldsymbol{x}[q])$ and $DT_j(\boldsymbol{X})$ can be regarded as 2-D fuzzy sets with $G * M$ elements. Given $\boldsymbol{x}[k]$ to be classified, *DT* computes the decision support for class C_j by matching $DP(\boldsymbol{x}[q])$ with the $DT_j(\boldsymbol{X})$ associated to that class, as follows:

$$df_j\left(\boldsymbol{x}[q]\right) = \mathcal{G}\left\{DP(\boldsymbol{x}), DT_j(\boldsymbol{x})\right\} \tag{26}$$

where $\mathcal{G}$ denotes a similarity measure. The higher the degree of similarity between these entries, the higher the certainty grade for that class $df_j(\boldsymbol{x}[q])$. The matching rule is implemented here using the inclusion operator of A in B: $I_1(A, B) = \|A \cap B\|/\|A\|$, where A and B are fuzzy sets, $\|.\|$ is the relative cardinality of a fuzzy set, and $\cap$ stands for the intersection (realized by min).

V. GA-SONeFMUC Building Method

A. Structure-Learning Algorithm

The suggested model is generated in a self-organizing way by means of the group method of data handling (GMDH) algorithm [20]. Particularly, the structure of GA-SONeFMUC is not fixed in advance. Starting from the original system inputs (features), new layers are sequentially developed, until a final topology is obtained, satisfying the performance requirements. Initially, we decompose the data set into a training D_{tr} set, a validation D_{vl} set, and a testing D_{test} set, with $n_{tr} + n_{vl} + n_{test} = D_N$. The structural parameters of the model are selected, as shown in Table I.

The GMDH algorithm proceeds along six steps.

Step 1) *Construct the FNCs of the First Layer.* They operate on the original feature set $\boldsymbol{x} = \{x_1, \ldots, x_m\}$. Features at this layer are combined by $r^{(1)}$, creating $Q^{(1)} = \binom{n}{r^{(1)}} = n!/(n - r^{(1)})!r^{(1)}!$ possible FNCs. A feature combination forms the input space of the corresponding FPD. Construction of the FNCs involves FPD learning (F1–F6), and decision making through DMFU (D1–D3).

TABLE I
DESIGN PARAMETERS OF GA-SONeFMUC

Parameter	Range
$r^{(1)}$	[2, 3]
$r^{(\ell)}$, $\ell \geq 2$	p
K_i, $i = 1, \ldots, r^{(\ell)}$	[2,3]
Rule type	(Crisp, Polynomial)
$F_k^{(\ell)}$	(MIN, wAVG, FI, DT)
G	2
ϑ	[0.7, 0.9]
ρ	[0.2, 0.4]
W	30

Step 2) *Evaluate the Performance of the FNCs at the Current Layer.* To assess the quality of the generated FNCs, each node is evaluated on the validation set, using an error measure:

$$E_{k,vl} = \frac{1}{n_{vl}} \left\{ \sum_{j=1}^{p} \sum_{q=1}^{n_{vl}} \left\{ y_{d,j}^{C}[q] - y_{k,j}^{(\ell)} \right\}^2 + \sum_{q=1}^{n_{vl}} \left\{ C_d[q] \neq C_j[q] \right\} \right\}. \tag{27}$$

The first term in (27) is the mean squared error (MSE) function that measures the deviation between the actual FPD outputs and the class targets. The second term corresponds to the number of misclassifications.

Step 3) *Formulate the Best Set at the Current Layer.* Sort the individual error measures in ascending order and retain a number of W FNCs with the best performance; whereas, the rest are discarded. The design parameter W is set to 30 in our simulations. Outputs of these FNCs serve as inputs to the FNCs at the next layer.

Step 4) *Construct the New FNCs.* Determine the structure of $FNC_k^{(\ell)}$, $k = 1, \ldots, Q^{(\ell)}$ by combining its parent classifiers $FNC_i^{(\ell-1)}$ and $FNC_j^{(\ell-1)}$. FNCs at the previous layer are combined by two ($G = 2$), leading to a total number of $Q^{(\ell)} = \binom{W}{2} = W!/(W - 2)!2!$ nodes in the current layer. Combining parents FNCs means that we make use of both types of outputs being offered: the continuous outputs $\boldsymbol{y}_i^{(\ell-1)}$ and $\boldsymbol{y}_j^{(\ell-1)}$ (transformed feature values) and the decision vectors $D_i^{(\ell-1)}$ and $D_j^{(\ell-1)}$. The former are first aggregated using (11) and fed as inputs to the $FPD_k^{(\ell)}$; the latter are submitted to the fuser $F_k^{(\ell)}$. Finally, FPD learning (F1–F6), and decision making through DMFU (D1–D3) is conducted.

Step 5) The algorithm proceeds by repeating steps 2) through 4), until a termination criterion is fulfilled

$E_*^{(\ell)} \geq E_*^{(\ell-1)}$, i.e., the best node performance attained at the current layer is inferior to the one at the previous layer or when the number of layers created is larger than a predefined maximum number of layers, $L \geq L_{\max}$.

Step 6) Once the stopping criteria are satisfied for some $L \leq L_{\max}$, the node classifier with the best performance $FNC_*^{(L)}$ is considered as the ending node of GA-SONeFMUC, providing the decision outputs of the model. The remaining FNCs at the output layer are discarded. In the following, we perform a reverse flow tracing through the network's structure, moving from the output layer to the input layer. All nodes at the intermediate layers (the input layer included), having no contribution to the $FNC_*^{(L)}$, are removed from the network. As regards the model's inputs, a subset is retained from the original features including the most distinguishing features (feature selection).

B. Parameter Learning Algorithm

Having obtained the structure of GA-SONeFMUC, parameter learning is performed to improve further the classification performance. The MF centers and the consequent weights involved in the FPDs, as well as the threshold ρ controlling the MF overlapping at the premise part of the DMFUs, are the parameter set tuned by means of a GA.

A steady state GA with overlapping populations is devised with real chromosome encoding. Parent selection is performed using the roulette wheel method. The genetic operators applied are uniform initialization and crossover and Gaussian mutation. Linear scaling in raw scores is also used. The chromosome of the GA is represented by $S_i = (\tilde{m}_1, \ldots, \tilde{m}_q, \tilde{w}_1, \ldots, \tilde{w}_q, \tilde{\rho}_1, \ldots, \tilde{\rho}_q)$. The parameters to be tuned by the GA are subject to several constraints. Constrained search spaces are defined, following the approach in [24], whereby tuning of the centers of premise membership functions are restricted to the intervals: $m_j^{(i)} \in [c_{j,L}^{(i)}, c_{j,R}^{(i)}]$. As regards the consequent weights, the respective search spaces are formulated by $w_{j,k}^{(i)} \in [-\gamma * |w_{j,k}^{(i)}| + w_{j,k}^{(i)}, \gamma * |w_{j,k}^{(i)}| + w_{j,k}^{(i)}]$, where $\pm\gamma * |w_{j,k}^{(i)}|$ denotes a predefined small percentage of the initial parameter values. Similarly, the search space of the overlapping thresholds are formulated as $\rho_j^{(i)} \in [-\gamma * (1/(L_j - 1)) + a_j^{(i)}, \gamma * (1/(L_j - 1)) + a_j^{(i)}]$. To obtain individuals with good predictive capabilities, the fitness function is defined as $fitness = 1/E_{k,vl}$, with $E_{k,vl}$ calculated as in (27).

VI. Feature Sets

Gray level values of bands are usually employed to classify ground cover types from multispectral spaces. However, due to mixed pixels and diverse texture types, prior research reveals that the use of advanced features can improve the discrimination level among the classes. Therefore, we consider four feature sets: spectral features from the bands, features from a gray level co-occurrence matrix (GLCM), wavelet features, and transformed spectral features.

A. Spectral Band Features

Classification of our study area is achieved using an IKONOS bundle image with four bands (three visible and one near-infrared) that exhibit 1-m spatial resolution in panchromatic and 4 m in multispectral. We chose four features from the spatial domain, each one corresponding to the gray values of the bands.

B. Features From GLCM

Textural analysis using GLCM [25] is a common practice in land cover image classification [8]. The image is raster scanned with sliding windows of $M \times M$ dimensions. A GLCM for each window is calculated, indicating how often different gray levels i, j occur with a specific direction $\theta = 0^\circ, 45^\circ, 90^\circ, 135^\circ$ and distance d between the pixel centers. Assuming G gray levels within the image, a $G \times G$ matrix is computed, with the (i, j)th element given as

$$p(i,j) = f_{ij}^{d,\theta} \Big/ \sum_{i}^{N} \sum_{j}^{N} f_{ij}^{d,\theta} \tag{28}$$

where i and j refer to matrix's rows and columns, $f_{ij}^{d,\theta}$ is the frequency of occurrence of gray 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. Among the 16 possible measures, four are considered to be the most important: contrast (CON), angular second moment (ASM), correlation (COR), and homogeneity (HOM). As a result, we obtain a total of 16 textural features for the four bands.

C. Transformed Spectral Features (TSF)

Two alternative color spaces are also produced from the initial bands. The first is the IHS transformation using intensity (I), hue (H), and saturation (S) as the three positioned parameters (in lieu of R, G, and B). Intensity represents the total amount of the light in a color, hue is a property of the color determined by its wavelength, and saturation is the purity of the color [26].

The second color space is the Tasseled Cap transformation [27], producing three data structure axes that define vegetation information: Brightness, Greenness, and Wetness.

Overall, six TSF features (three IHS and three Tasseled Cap) are computed for the entire image.

D. Wavelet Features (WF)

Fast wavelet transform (FWT) [28] is based on multiresolution analysis of images. Using a 2-D FWT filter bank, we perform a two-level wavelet decomposition of the original image (Fig. 5). At the first level, four subimages are formed containing the approximation (LL_1), horizontal detail (LH_1), vertical detail (HL_1), and diagonal detail (HH_1) coefficients, respectively. The LL_1 subimage at level 1

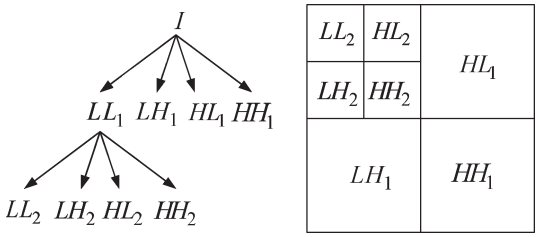


Fig. 5. Two-level wavelet image decomposition.

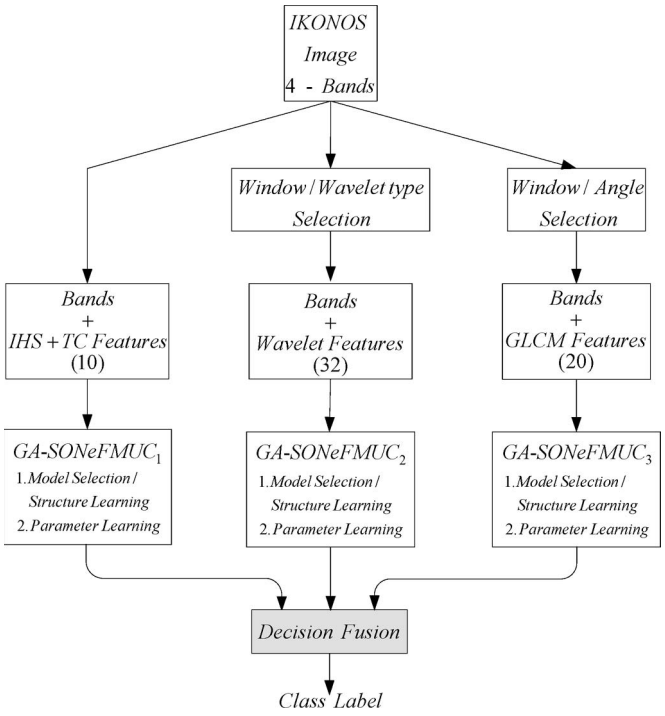


Fig. 6. Multifeature fusion architecture.

is further decomposed to four subimages at the second level. For each spectral band, we consider seven subimages, $\{LL_2, LH_2, HL_2, HH_2, LH_1, HL_1, HH_1\}$ and compute the total energy measure from the wavelet coefficients associated with each subimage

$$Energy_i = \sum_j \sum_k SubI_i(j,k)^2. \tag{29}$$

Energy distribution from FWT provides detailed description of the frequency content of an image. The WF set comprises a total of 28 features (seven energy features times four bands).

E. Window Size-Angle Selection

FWT and GLCM require a proper selection of the window size for the calculation of features. In addition, a suitable angle value should be decided for the GLCM method. Usually, a time-consuming trial-and-error approach is adopted for selecting these parameters, based on the classifier's performance. A more systematic procedure is proposed in this paper. The window size/angle for FWT and GLCM are chosen in terms of a modified within-class scatter criterion [29] used in Fisher linear discriminant analysis.

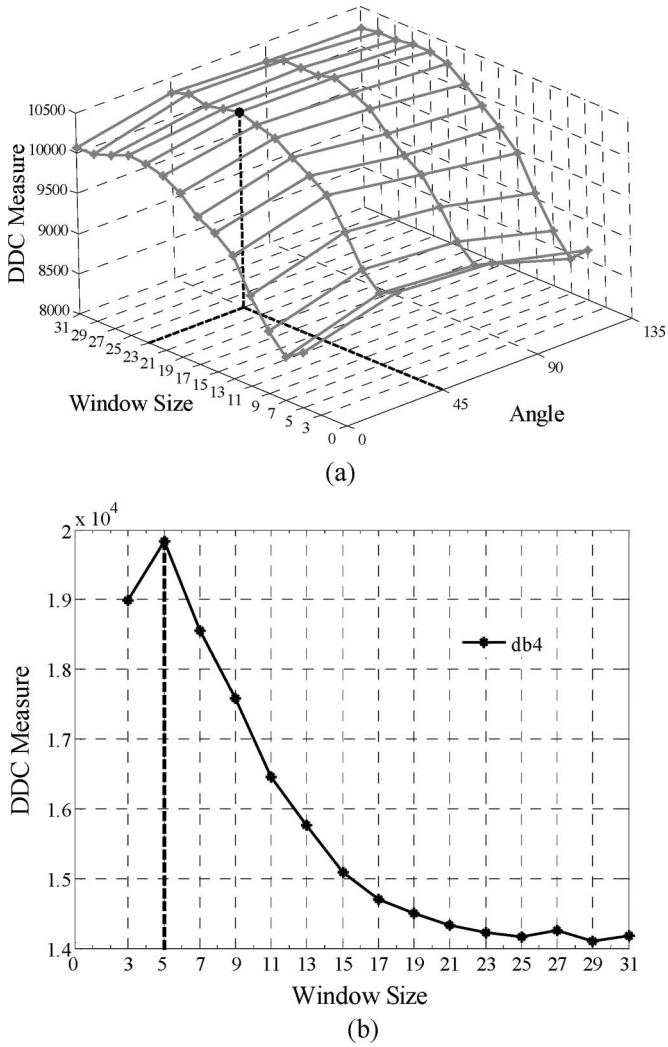


Fig. 7. (a) Window size-angle selection for the GLCM and (b) window size selection for the FWT using db4 in the agricultural zone.

For an M-class problem, let D_j denote a subset with n_j pixels that are assigned to class-j, and n is the number of features. For each feature-i and class-j, calculate the mean

$$m_{i,j} = \frac{1}{n_j} \sum_{x_i \in D_j} x_i, \qquad i=1,\ldots,n, \quad j=1,\ldots,M. \tag{30}$$

The scatter is given by

$$\tilde{s}_{i,j}^2 = \sum_{x_i \in D_j} (x_i - m_{i,j})^2, \qquad i=1,\ldots,n, \quad j=1,\ldots,M. \tag{31}$$

The total within-class scatter for feature-i is then computed as

$$J_i = \sum_{j=1}^{M} \sum_{k>j}^{M} \frac{|m_{i,j} - m_{i,k}|^2}{\tilde{s}_{i,j}^2 + \tilde{s}_{i,k}^2}, \qquad i=1,\ldots,d. \tag{32}$$

Adding J_i for all features (textural features for all bands), we obtain a measure of the degree of the discrimination (DDC)

TABLE II
STRUCTURE CHARACTERISTICS AND OVERALL ACCURACIES OF GA-SONeFMUC IN WETLAND ZONE CLASSIFICATION USING DIFFERENT FEATURE TYPES. COMPARISON WITH MLC IS GIVEN

GA-SONeFMUC	Wetland Map Classification							
	Feat.	Layers	C. T.	F. S.	F. T.	S. L. Test Perf (%)	P. L. Test Perf (%)	MLC Test Perf (%)
Bands+WF	11	4	TSK	(3,2)	MIN	86.50	89.37*	74.44
Bands+GLCM	10	4	TSK	(3,2)	MIN	84.87	89.37*	84.87*
Bands+TSF	6	3	Crisp	(3,3)	FI	77.91	82.21	73.21
Bands+WF+GLCM+TSF	15	5	TSK	(3,2)	MIN	84.25	86.91	71.58

C. T. ≡ Consequent Type, F. S. ≡ Fuzzy Sets ($\ell = 1, \ell \geq 2$), F. T. ≡ Fusion Type, S./P. L. ≡ Structure/Parameter Learning, *indicates best attained performance.

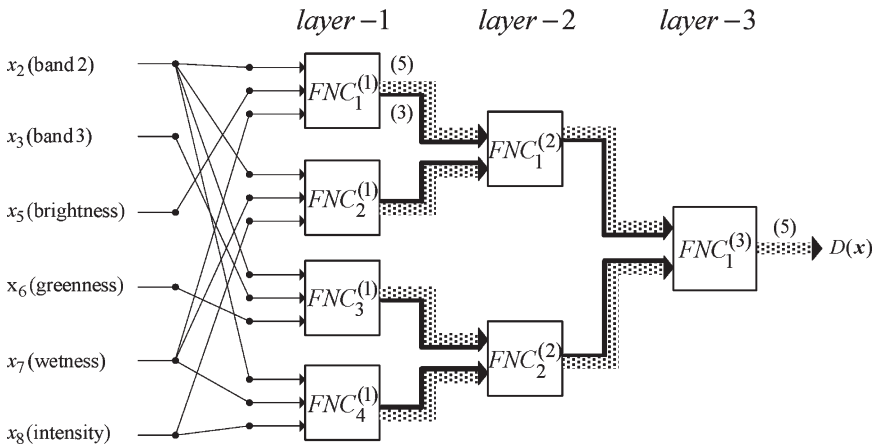


Fig. 8. GA-SONeFMUC structure for the Band+Spectral feature set in the wetland zone.

TABLE III
OVERALL ACCURACIES OBTAINED BY GA-SONeFMUC CLASSIFIER FUSION IN THE WETLAND ZONE. COMPARISON WITH MLC IS GIVEN

Fusion Method	GA-SONeFMUC Test Perf. (%)	MLC Test Perf (%)
MV	87.53	82.62
NB	90.18	85.28*
BKS	92.02*	83.44
MIN	91.21	-
FI	89.57	-
DT	90.18	-

*indicates best attained performance.

among the classes, for a specific window size-angle pair

$$DDC = \sum_{i=1}^{n} J_i. \tag{33}$$

The higher the DDC, the more discriminated the classes. The window size axis is adequately discretized while the angle takes the values $\theta = 0°, 45°, 90°, 135°$, using $d = 1$. For the GLCM, we consider the different size-angle combinations and compute the corresponding DDC values. From the generated 3-D plot, we then choose the optimal combination that shows the maximum DDC value.

As regards the wavelets features, only the window size space is considered. We examined nine different wavelet basis functions for the image decomposition: $\{\mathrm{db1}, \mathrm{db4}, \mathrm{db8}, \mathrm{sym1}, \mathrm{sym4}, \mathrm{sym8}, \mathrm{coif1}, \mathrm{coif4}, \mathrm{haar}\}$. For each wavelet type,

TABLE IV
ERROR CONFUSION MATRIX IN WETLAND ZONE USING THE BKS FUSION METHOD OVER THE GA-SONeFMUC CLASSIFIERS

GA-SONeFMUC	Wetland map classification – Decision Fusion (BKS)						
	Phragmites	Tamarix	Wet meadows	Trees	Water	PA (%)	UA (%)
Phragmites	103	5	7	2	0	91.15	88.03
Tamarix	8	50	3	6	0	84.75	74.63
Wet meadows	2	3	197	2	0	95.17	96.57
Trees	0	1	0	46	0	82.14	97.87
Water	0	0	0	0	54	100.0	100.0
Reference	113	59	207	56	54		

Overall Accuracy = 92.02%: Khat = 0.89.

TABLE V
ERROR CONFUSION MATRIX IN WETLAND ZONE USING THE NB FUSION METHOD OVER THE MLC CLASSIFIERS

MLC	Wetland map classification – Decision Fusion (NB)						
	Phragmites	Tamarix	Wet meadows	Trees	Water	PA (%)	UA (%)
Phragmites	98	16	16	14	0	86.73	68.06
Tamarix	11	35	1	0	0	59.32	74.47
Wet meadows	3	3	190	2	0	91.79	95.96
Trees	1	5	0	40	0	71.43	86.96
Water	0	0	0	0	54	100.0	100.0
Reference	113	59	207	56	54		

Overall Accuracy = 85.28%: Khat = 0.80.

TABLE VI
STRUCTURE CHARACTERISTICS AND OVERALL ACCURACIES OF GA-SONeFMUC IN AGRICULTURAL ZONE CLASSIFICATION USING DIFFERENT FEATURE TYPES. COMPARISON WITH MLC IS GIVEN

GA-SONeFMUC	Agriculture Map Classification							
	Feat.	Layers	C. T.	F. S.	F. T.	S. L. Test Perf (%)	P. L. Test Perf (%)	MLC Test Perf (%)
Bands+WF	6	3	TSK	(2,2)	FI	72.32	73.47*	62.86
Bands+GLCM	9	4	TSK	(3,2)	MIN	71.28	73.01	68.28*
Bands+TSF	6	3	TSK	(3,2)	DT	70.13	71.05	59.05
Bands+WF+GLCM+TSF	10	5	TSK	(2,2)	MIN	70.82	71.74	53.17

C. T. ≡ Consequent Type, F. S. ≡ Fuzzy Sets ($\ell = 1, \ell \geq 2$), F. T. ≡ Fusion Type, S./P. L. ≡ Structure/Parameter Learning, *indicates best attained performance.

a plot is created showing DDC versus window size and the size value is selected corresponding to the peak DDC. Among the local optima, the one is chosen giving the maximum DDC, and this window size is used for the extraction of wavelet features.

VII. MULTIFEATURE DECISION FUSION

The feature sets described in Section VI are organized into four broader groups, regarded as input sets to the classifiers. The first three input sets are Bands+WF (32 features), Bands+GLCM (20 features), and Bands+TSF (ten features). The gray levels of the bands are combined with textural and spectral features to enhance the distinguishing capabilities of the input sets.

For each of the above input sets, individual GA-SONeFMUC classifiers are developed and trained. To exploit the decision supports obtained from independent feature families, classifiers fusion is applied at the final stage. The final class assignments are derived by combining the individual classifiers outputs through a fusion operator (Fig. 6). Prior research, also supported by our simulation results, indicates that decision fusion improves classification accuracy, as compared to the performance attained by individual classifiers. As fusion operators, we employ the MIN, FI, and DT (Section IV). Since the above fusers operate on soft labels, they are unable to validate the crisp assignments of a maximum-likelihood classifier (MLC) used in our comparison tests. Therefore, three additional fusion methods are considered: majority voting (MV), "Naïve"-Bayes combination (NB), and behavior-knowledge space (BKS) [9]. These fusers are also applied to GA-SONeFMUC models by first hardening their outputs.

Finally, the fusion results obtained by combining the above three input sets are compared to a classifier developed using a fourth input set that includes all features: Bands+GLCM+WF+TSF (54 features). Since all features are incorporated, this classifier does not require fusion.

TABLE VII
OVERALL ACCURACIES OBTAINED BY GA-SONeFMUC CLASSIFIER FUSION IN THE AGRICULTURAL ZONE. COMPARISON WITH MLC IS GIVEN

Fusion Method	GA-SONeFMUC Test Perf. (%)	MLC Test Perf (%)
MV	74.05	70.36
NB	74.74	71.86*
BKS	75.32	71.51
MIN	75.55*	-
FI	73.47	-
DT	74.16	-

*indicates best attained performance.

VIII. EXPERIMENTAL RESULTS

A. Study Area

Lake Koronia is located in a tectonic depression in northern Greece (40′ 41″ °N, 23′ 09″ °E). The lake-wetland ecosystem is surrounded by an intensively cultivated agricultural area. The industrial sector has recently increased, discharging untreated effluents in the lake from fabric dying and food and dairy processing activities. 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. For classification purposes, an IKONOS bundle image is acquired covering the 134 km^2 of the study area.

An extensive field survey was conducted to identify land cover classes, referred mainly to the agricultural and wetland area. A total number of 4100 locations were selected at regular intervals along the agricultural road network, using the stratified random sampling method. The land cover on these locations was identified by visual inspection and was afterward labeled in thirteen 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). Patterns were separated into two different sets: the training set (70%) and the testing set (30%).

The GA-SONeFMUC network was applied to the 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 vegetation cover classes. Therefore, based on the pan-sharpened image and after careful photointerpretation, the image was segmented into two zones: the wetland zone, including the lake and its surrounding wetland vegetation, and the agricultural zone. In the wetland zone five classes were recognized: water, phragmites, tamarix, wet meadows, and trees. Furthermore, we consider eight classes in the agricultural zone. Six of them refer to different crop

TABLE VIII
ERROR CONFUSION MATRIX IN AGRICULTURAL ZONE USING THE MIN FUSION METHOD OVER THE GA-SONeFMUC CLASSIFIERS

GA-SONeFMUC	Agricultural Map Classification – Decision Fusion (MIN)									
	Alfalfa	Cereals	Maize	Orchards	Vegetables	Fallow	Shrubs	Urban	PA (%)	UA (%)
Alfalfa	161	7	21	4	24	4	4	0	80.10	71.56
Cereals	25	194	2	6	6	28	14	0	92.38	70.55
Maize	13	0	151	11	7	0	0	0	85.31	82.97
Orchards	1	4	3	28	2	1	5	2	53.85	60.87
Vegetables	1	0	0	3	13	0	1	0	24.07	72.22
Fallow	0	5	0	0	2	27	5	1	45.00	67.50
Shrubs	0	0	0	0	0	0	22	0	43.14	100.00
Urban	0	0	0	0	0	0	0	59	95.16	100.00
Reference	201	210	177	52	54	60	51	62		

Overall Accuracy=75.55%: Khat=0.70.

TABLE IX
ERROR CONFUSION MATRIX IN THE AGRICULTURAL ZONE USING THE NB FUSION METHOD OVER THE MLC CLASSIFIERS

MLC	Agricultural Map Classification – Decision Fusion (NB)									
	Alfalfa	Cereals	Maize	Orchards	Vegetables	Fallow	Shrubs	Urban	PA (%)	UA (%)
Alfalfa	135	11	14	7	16	1	0	0	67.16	73.77
Cereals	29	186	2	6	12	37	20	2	88.57	65.72
Maize	32	2	160	13	12	1	1	0	90.40	74.07
Orchards	3	1	1	25	2	2	3	0	47.62	64.52
Vegetables	0	0	0	0	11	0	0	0	15.22	100.0
Fallow	2	3	0	0	1	19	0	0	31.67	76.00
Shrubs	0	5	0	1	0	0	27	0	44.44	66.67
Urban	0	2	0	0	0	0	0	60	93.75	93.75
Reference	201	210	177	52	54	60	51	62		

Overall Accuracy=71.86%: Khat=0.64.

types (maize, alfalfa, cereals, orchards, vegetables, and fallow), while the remaining two in other land cover types (urban areas and shrubs).

B. Experiment Organization

The GA-SONeFMUC classifier was applied to both of the aforementioned zones. The experimental setup is designed as shown in Fig. 6.

1) Window Size/Angle Selection: Initially, we determine the proper window size and angle required by GLMC and window size for WF following the approach suggested in Section VI. To highlight the procedure, Fig. 7(a) shows the DDC values for different size-angle pairs in the agricultural zone. The optimal combination (23, 45°) is selected exhibiting the maximum DDC value, and these values are used in the calculations of the GLCM features. For the WF, Fig. 7(b) depicts the DDC values versus window size. For simplicity, among the nine wavelet functions considered, only the db4 is shown, which includes the global optimum choice. We chose a size of 5 for the computation of wavelet features on this zone. It can be seen that larger windows lead to a gradually diminishing discrimination among the classes. Regarding the wetland zone, we selected a size-angle of (7, 0°) for the GLCM and a size of 5 using coif1 wavelet function for the derivation of the WF.
2) In each zone, independent GA-SONeFMUC classifiers are developed coming from the different input sets. Structure learning is performed first for deciding the model's architecture.
   a) Since structure learning uses a validation data set to obtain appropriate networks with higher generalization capabilities, the original training set was further divided into training and validation sets (60%–40% partition). The MLC used the original training set during the training stage. The testing set was the same for both methods.
   b) Because five and eight classes are considered in the wetland and agricultural zones, respectively, we set $p = 3$; in other words, we consider three inputs $(r^{(\ell)} = p)$ and three outputs for the higher layer FPDs $(\ell \geq 2)$ for all GA-SONeFMUC models. Additionally, we consider three inputs for the FPDs at layer 1 $(r^{(1)} = 3)$ (i.e., the original features are combined by three). The confidence threshold and the initial value of the overlapping was set to $\vartheta = 0.8$ and $\rho = 0.2$, respectively.
   c) To initiate GA-SONeFMUC structure learning, the following parameters should also be decided: the number of fuzzy sets in each input, the rule form, and the fuser type. Model selection is accomplished through a thee-step procedure. At step 1), we consider the MIN fuser and develop, via GMDH, different models using

two or three fuzzy sets along each FPD input, and rules with polynomial or crisp consequents. During step 2), the network with the best performance on the validation data is selected as the most appropriate. Then, for step 3), based on the model decided in step 2), three additional models are evaluated using the remaining three fusion types (wAVG, FI, DT). The network exhibiting the higher classification accuracy on the validation data is selected as the final model.

3) Real-coded GA is applied next for parameter learning of the model obtained after structure. We consider a population size of 40 individuals, Gaussian mutation, and uniform crossover operators with probabilities 0.01 and 0.9, respectively. At each generation, 50% of the population is replaced while evolution lasts for 1000 generations.
4) As a final step, the decision results obtained from Bands+WF, Bands+GLMC, and Bands+TSF are combined using the classifier fusion (Section VII).

C. GA-SONeFMUC Classification of the Wetland Zone

The results for the wetland zone are summarized in Table II. The models based on Bands+GLCM and Bands+WF features exhibit the same classification accuracy, achieving an 89.37% performance; whereas, Bands+TSF features show the worst score, 82.21%. It can be seen that Bands+WF+GLMC+TSF does not offer better performance compared to the other partial input sets, with the exception of Bands+TSF. This is attributed to the fact that the inclusion of all features complicates considerably the feature space, which causes difficulties for distinguishing the most discriminating features to be considered as model inputs. On the contrary, more homogeneous input sets comprising wavelet and textural features facilitate feature selection. Only 11 out of 32 wavelet features and ten out of 20 co-occurrence features are retained for classification of the wetland zone, demonstrating the feature selection capabilities of GA-SONeFMUC.

Table II shows that the GA-SONeFMUC models obtained after structure learning alone exhibit good performance, being superior to the MLCs for all input sets. Assuming that this accuracy level is acceptable, the GA parameter learning stage can be omitted. All GA-SONeFMUC simulations were performed on an Intel Pentium IV 2.8-GHz processor with 512 MB of RAM. For wetland classification using the Bands+WF features, model building and evaluation of the testing data required 13.1 min and 0.3 s, respectively, while MLC needed 4.8 s and 0.1 s for the same experiment. Since the classification task is an off-line procedure, the computation load is not prohibitive, in view of the feature selection and the enhanced performance obtained by GA-SONeFMUC.

To illustrate the model structure, Fig. 8 shows a three-layered GA-SONeFMUC containing seven FNCs. The model is generated using Bands+TSF inputs. The FPD inputs for all layers are described by three MFs with the FPD rules having crisp consequents. From the ten original features, only the six features shown in Fig. 8 are selected as model inputs. The proposed model is compared with a traditional MLC classifier.

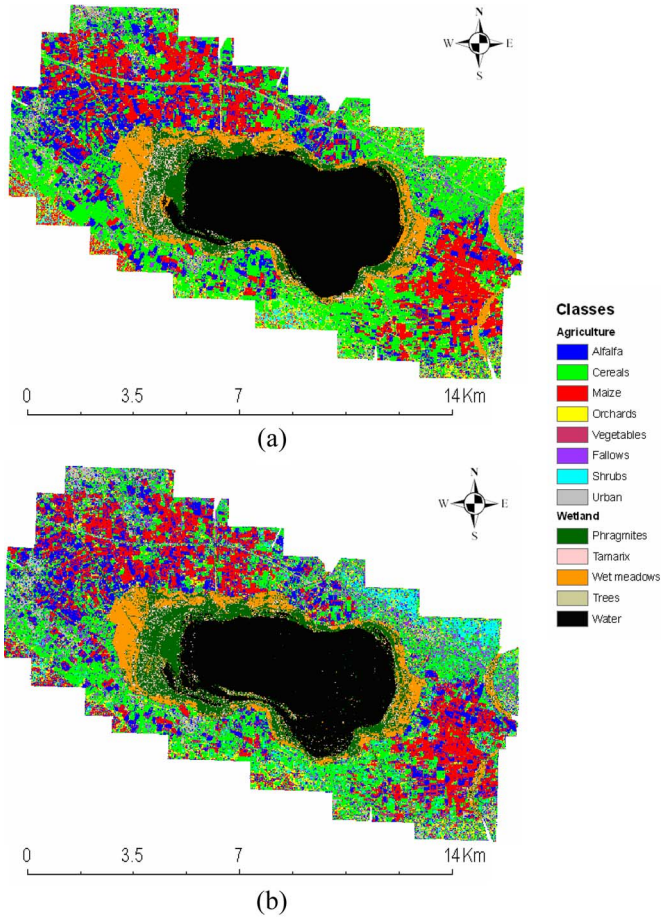


Fig. 9. Mosaic of IKONOS land cover classification of wetland and agricultural zones using (a) GA-SONeFMUC and (b) MLC.

The best MLC rate is obtained using the Band+GLCM feature set, exhibiting 84.47%. This is inferior to the performance attained by GA-SONeFMUC to a percentage of almost 5%.

As a final step, the outputs of the three independent classifiers are combined using the MIN, FI, and DT fusers. In addition, after hardening of the soft decision outputs, the crisp fusion methods of MV, NB, and BKS are also applied. The results presented in Table III demonstrate the capabilities of the classifier fusion approach for accuracy improvement. BKS fuser provided the best performance for the GA-SONeFMUC, achieving an overall accuracy of 92.02% and a Khat equal to 0.89 (see [30] for a definition of Khat). Furthermore, fusion of the MLCs through NB also improved classification, offering an accuracy of 85.28%, which is inferior to the one obtained by GA-SONeFMUC by a percentage of approximately 7%.

Tables IV and V host the error confusion matrices of GA-SONeFMUC and MLC, respectively. Comparative analysis illustrates the superior classification capabilities of the suggested model. Both classifications labeled correctly the water class (producer's and user's accuracy 100%). This was expected because the spectral signature of water is clearly separable from vegetation cover classes. As contrasted to MLC, GA-SONeFMUC achieved a better performance in the two basic classes of that zone (phragmites, wet meadows) in both user's and producer's accuracies. GA-SONeFMUC provided a higher user's accuracy in phragmites class, hence minimizing

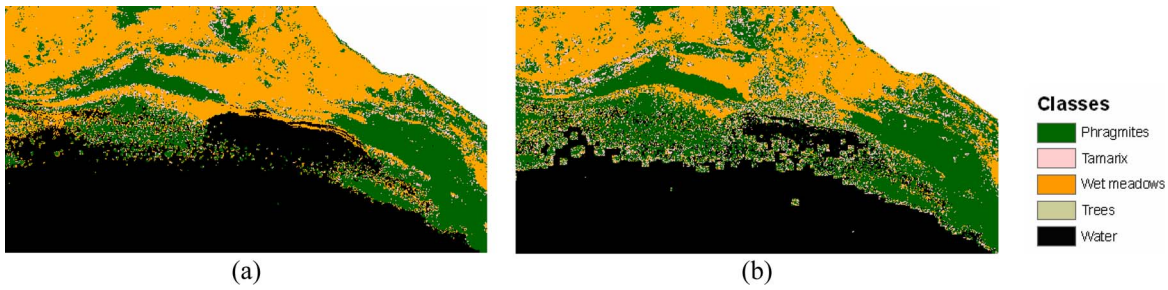


Fig. 10. Subset of the land cover map produced with (a) GA-SONeFMUC and (b) MLC in the wetland zone north of the lake.

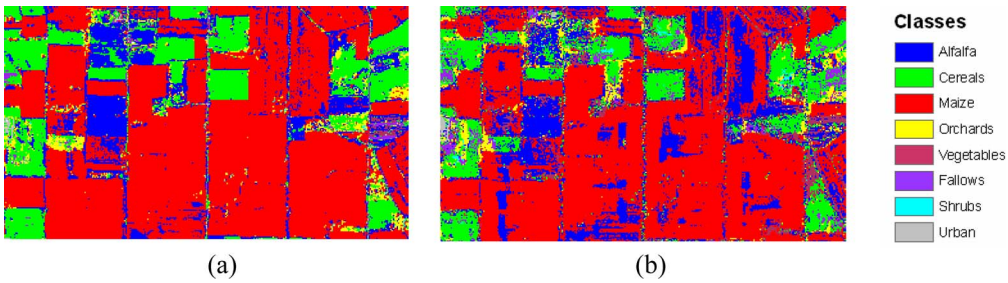


Fig. 11. Subset of the land cover map produced with (a) GA-SONeFMUC and (b) MLC in the agricultural zone east of the wetland.

overestimation in this class. A better performance is also observed in the other two classes, covering the smallest area in the zone.

D. GA-SONeFMUC Classification of the Agricultural Zone

Table VI shows the resulting GA-SONeFMUC models, for the agricultural zone classification. After parameter learning, the model with the best performance was the one operating on the Bands+WF set. The overall accuracy was 73.47%, being lower than the best performance achieved in the wetland zone. This degradation is attributed to the higher number of classes in this zone that exhibit a much larger spectral overlapping. Due to structure learning, only six out of 32 features are selected as the features with greater impact on class discrimination: two bands and four wavelet features. Compared to the best performance obtained by MLC (68.28%), GA-SONeFMUC outperformed by an amount of 5% on the testing data set.

Table VII hosts the classification results obtained after classifier fusion. Fusion of the GA-SONeFMUC using the MIN operator provided an overall accuracy of 75.55% and a Khat equal to 0.70. This is almost 4% better compared to the best result obtained by fusing MLCs through the NB crisp operator (71.86%). Tables VIII and IX show the error confusion matrices for the GA-SONeFMUCs and MLCs, respectively, corresponding to the best fuser choices for each model type. GA-SONeFMUC's fusion leads to higher accuracy in the three dominant classes (alfalfa, cereals, maize). Based on the user's accuracy percentages the proposed method exhibits smaller overestimation compared to MLC. The opposite occurs in the other three crop classes (orchards, vegetables, fallow), where a slightly larger overestimation is observed. Both classifiers achieved a high performance in classifying urban areas. GA-SONeFMUC underestimated class shrubs but showed a better accuracy contrarily to MLC, which exhibited low producer's and user's accuracy.

In addition to statistical comparisons of the two classifications, a visual assessment of thematic land cover maps (Fig. 9) was carried out in many subareas of both zones, and useful information was derived. Firstly, in the wetland zone (Fig. 10), MLC overestimated phragmites in the north part, where the area is covered by wet meadows. Moreover, inside the lake, MLC overestimated phragmites against water, which is the correct class. GA-SONeFMUC classification produces a more clear result compared to MLC, which produced a blurred thematic map with a lot of misclassified pixels. In the agricultural zone (Fig. 11), GA-SONeFMUC produces distinguishable shapes and classes for each agriculture field. On the contrary, MLC exhibits a strong confusion especially between alfalfa and maize classes. Hence, a lot of parcels can be labeled using only the classified image obtained by the proposed model.

IX. Conclusion

The GA-SONeFMUC classifier is proposed for land cover classification of satellite images. The model exhibits a number of attractive attributes: hierarchical structure comprising interconnected small-scale FNCs arranged in layers, decision fusion of parent FNCs, data splitting and classification improvement of ambiguous pixels, self-organizing structure learning and GA optimization, feature selection capabilities, successive classification through simultaneous feature transformations, and decision makings along the layers. Each FNC at a higher layer improves class discrimination, locating a portion of the training examples closer to the respective class targets. GA-SONeFMUC extends the principles of the voting/rejection scheme and the classifier combination to multiple layers.

For efficient classification, we considered different input sets comprising the gray level band values and textural and spectral features. We also applied classifier fusion to exploit information acquired by different feature sources. The proposed method was tested in Lake Koronia and its surrounding agricultural area, producing fruitful results. An overall accuracy of 92.02% and 75.55% was attained in the wetland and the agricultural zones, respectively. Compared to MLC the proposed model achieved better performances by a 7% and 4% in each zone. The resulting thematic maps (Fig. 9) indicate the better classification quality provided by GA-SONeFMUC.

REFERENCES

[1] J. A. Benediktsson, P. H. Swain, and O. K. Esroy, "Conjugate-gradient neural networks in classification of multisource and very-high-dimensional remote sensing data," *Int. J. Remote Sens.*, vol. 14, no. 15, pp. 2883–2903, 1993.

[2] T. Kavzoglu and P. M. Mather, "The use of backpropagating artificial neural networks in land cover classification," *Int. J. Remote Sens.*, vol. 24, no. 23, pp. 4907–4938, 2003.

[3] A. Bárdossy and L. Samaniego, "Fuzzy rule-based classification of remotely sensed imagery," *IEEE Trans. Geosci. Remote Sens.*, vol. 40, no. 2, pp. 362–374, Feb. 2002.

[4] F. Wang, "Fuzzy supervised classification of remote sensing images," *IEEE Trans. Geosci. Remote Sens.*, vol. 28, no. 2, pp. 194–201, Mar. 1990.

[5] S. Bandyopadhyay, U. Maulick, and A. Mukhopadhyay, "Multiobjective genetic clustering for pixel classification in remote sensing imagery," *IEEE Trans. Geosci. Remote Sens.*, vol. 45, no. 5, pp. 1506–1511, May 2007.

[6] Z. Liu, A. Liu, C. Wang, and Z. Niu, "Evolving neural network using real coded genetic algorithm (GA) for multispectral image classification," *Future Gener. Comput. Syst.*, vol. 20, no. 7, pp. 1119–1129, Oct. 2004.

[7] G. A. Carpenter, M. N. Gjaja, S. Gopal, and C. E. Woodcock, "ART neural networks for remote sensing: Vegetation classification from Landsat TM and terrain data," *IEEE Trans. Geosci. Remote Sens.*, vol. 35, no. 2, pp. 308–325, Mar. 1997.

[8] C. T. Lin, Y. C. Lee, and H. C. Pu, "Satellite sensor image classification using cascaded architecture of neural fuzzy network," *IEEE Trans. Geosci. Remote Sens.*, vol. 38, no. 2, pp. 1033–1043, Mar. 2000.

[9] S. K. Meher, B. U. Shankar, and A. Ghosh, "Wavelet-feature-based classifiers for multispectral remote-sensing images," *IEEE Trans. Geosci. Remote Sens.*, vol. 45, no. 6, pp. 1881–1886, Jun. 2007.

[10] L. I. Kuncheva, J. C. Bezdek, and R. P. W. Duin, "Decision templates for multiple classifier fusion: An experimental comparison," *Pattern Recognit.*, vol. 34, no. 2, pp. 299–314, Feb. 2001.

[11] G. Giacinto and F. Roli, "Ensembles of neural networks for soft classification of remote sensing images," in *Proc. Eur. Symp. Intell. Techniques*, Bari, Italy, Mar. 1997, pp. 166–170.

[12] M. Petrakos, J. A. Benediktsson, and I. Kanellopoulos, "The effect of classifier agreement on the accuracy of the combined classifier in decision level fusion," *IEEE Trans. Geosci. Remote Sens.*, vol. 39, no. 11, pp. 2539–2546, Nov. 2001.

[13] M. Fauvel, J. Chanussot, and J. A. Benediktsson, "Decision fusion for the classification of urban remote sensing images," *IEEE Trans. Geosci. Remote Sens.*, vol. 44, no. 10, pp. 2828–2838, Oct. 2006.

[14] B. Waske and J. A. Benediktsson, "Fusion of support vector machines for classification of multisensor data," *IEEE Trans. Geosci. Remote Sens.*, vol. 45, no. 12, pp. 3858–3866, Dec. 2007.

[15] G. J. Briem, J. A. Benediktsson, and J. R. Sveinsson, "Multiple classifiers applied to multisource remote sensing data," *IEEE Trans. Geosci. Remote Sens.*, vol. 40, no. 10, pp. 2291–2299, Oct. 2002.

[16] A. S. Kumar, K. S. Basu, and K. L. Majumar, "Robust classification of multispectral data using multiple neural networks and fuzzy integral," *IEEE Trans. Geosci. Remote Sens.*, vol. 35, no. 3, pp. 787–790, May 1997.

[17] M. Han and X. Tang, "A classification framework of neural networks fusing spectrum and texture information," in *Proc. IGARSS*, Seoul, Korea, 2005, vol. 6, pp. 3814–3817.

[18] G. G. Wilkinson, F. Fierens, and I. Kanellopoulos, "Integration of neural and statistical approaches in spatial data classification," *Geograph. Syst.*, vol. 32, pp. 1–20, 1995.

[19] J. A. Benediktsson and I. Kanellopoulos, "Classification of multisource and hyperspectral data based on decision fusion," *IEEE Trans. Geosci. Remote Sens.*, vol. 37, no. 3, pp. 1367–1377, May 1999.

[20] A. G. Ivakhnenko, "The group method of data handling; a rival of method of stochastic approximation," *Sov. Autom. Control*, vol. 13, no. 3, pp. 43–45, 1968.

[21] T. Takagi and M. Sugeno, "Fuzzy identifications and its application to modeling and control," *IEEE Trans. Syst., Man, Cybern.*, vol. SMC-15, pp. 116–132, Jan./Feb. 1985.

[22] H. M. Lee, C. M. Chen, J. M. Chen, and Y. L. Jou, "An efficient fuzzy classifier with feature selection based on fuzzy entropy," *IEEE Trans. Syst., Man, Cybern. B, Cybern.*, vol. 31, no. 3, pp. 426–432, Jun. 2001.

[23] G. C. Goodwin and K. S. Sin, *Adaptive Filtering: Prediction and Control*. Englewood Cliffs, NJ: Prentice-Hall, 1984.

[24] O. Cordon, F. Herrera, F. Hoffmann, and L. Magdalena, *Genetic Fuzzy Systems: Evolutionary Tuning and Learning of Fuzzy Knowledge Bases*. Singapore: World Scientific, 2001.

[25] R. M. Haralick and L. G. Shapiro, *Robot and Computer Vision*, vol. 1. Reading, MA: Addison-Wesley, 1992.

[26] Y. Zhang and G. Hong, "An IHS and wavelet integrated approach to improve pan-sharpening visual quality of natural colour IKONOS and QuickBird images," *Inf. Fusion*, vol. 6, no. 3, pp. 225–234, Sep. 2005.

[27] J. H. Horne, "A tasseled cap transformation for IKONOS images," in *Proc. ASPRS Annu. Conf.*, Anchorage, AK, 2003.

[28] S. G. Mallat, "A theory for multiresolution signal decomposition: The wavelet representation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 11, no. 7, pp. 674–693, Jul. 1989.

[29] R. Duda, P. Hart, and D. Stork, *Pattern Classification*, 2nd ed. New York: Wiley, 2001.

[30] R. G. Congalton and K. Green, *Assessing the Accuracy of Remotely Sensed Data: Principles and Practices*. New York: Lewis, 1998.

Nikolaos E. Mitrakis (S'04) received the degree in electrical engineering from the Aristotle University of Thessaloniki, Thessaloniki, Greece, in 2003. He is currently working toward the Ph.D. degree at the Department of Electrical and Computer Engineering, Division of Electronics and Computer Engineering, Aristotle University of Thessaloniki.

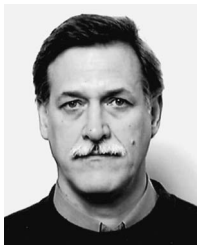
His research interests include fuzzy systems, neural networks, remote sensing, neuro-fuzzy classifiers, decision fusion, and evolutionary algorithms.

Charalampos A. Topaloglou received the degree in agriculture from the Aristotle University of Thessaloniki, Thessaloniki, Greece, in 2002, and the M.S. degree in GIS and remote sensing, in 2004. He is currently working toward the Ph.D. degree at the Faculty of Agronomy, Aristotle University of Thessaloniki.

His research interests include remote sensing, GIS analysis, modeling of environmental indices, landscape metrics, statistical and fuzzy classification, and fuzzy systems.

Thomas K. Alexandridis received the degree in agriculture and the Ph.D. degree from the Aristotle University of Thessaloniki, Thessaloniki, Greece and the M.S. degree in applied remote sensing from Cranfield University, Cranfield, U.K.

He is currently a researcher in the Faculty of Agronomy, Aristotle University of Thessaloniki. His research interests include environmental, agricultural, and water resources monitoring and modeling and multiple-scale issues in remote sensing and GIS.



John B. Theocharis (M'90) received the degree in electrical engineering and the Ph.D. degree from Aristotle University of Thessaloniki, Thessaloniki, Greece, in 1980 and 1985, respectively.

He is currently an Associate Professor in the Department of Electronic and Computer Engineering, Division of Electronics and Computer Engineering, Aristotle University of Thessaloniki. His research activities include fuzzy systems, neural networks, adaptive control, and modeling of complex nonlinear systems.

George C. Zalidis received the degree in agriculture from the Aristotle University of Thessaloniki, Thessaloniki, Greece, in 1980 and the Ph.D. degree from Michigan State University, East Lansing, in 1987.

He is currently a Professor of soil pollution and degradation in the Lab of Applied Soil Science, Faculty of Agronomy, Aristotle University of Thessaloniki. His research activities include soil quality and sustainability, bioremediation of degraded areas, restoration and rehabilitation of wetland ecosystems, and wetland inventory and mapping.