Extended Methods for Classification of Remotely Sensed Images Based on ARTMAP Neural Networks

Norbert Kopčo[1 2], Peter Sinčák[2], and Howard Veregin[3]

[1] Department of Cognitive and Neural Systems, Boston University
677 Beacon St., Boston, MA 02215, USA
[2] Laboratory of Artificial Intelligence, Department of Cybernetics and AI
Faculty of Electrical Engineering and Informatics, Technical University Košice
Letná 9, 040 01 Košice, Slovak Republic
[3] Department of Geography, University of Minnesota
267-19th Ave S., Minneapolis, MN 55455, USA
e-mail: cig@neuron-ai.tuke.sk

Abstract. This paper deals with two aspects of the application of ARTMAP neural networks for classification of satellite images obtained by remote sensing of the Earth. The first part contains an analysis of the influence of data representation and cluster determination method on classification accuracy. Three types of representation/determination are analyzed. Best results are obtained for Gaussian ARTMAP, an ARTMAP neural network using gaussian distributions for identification of clusters in feature space. In the second part, a method for evaluation of the classification quality is described. This method introduces a confidence index which is assigned to each individual pixel of the image, thus allowing generation of a confidence map for the classified image. The confidence index is computed conveniently exploiting features of the Gaussian ARTMAP learning algorithm. Using a threshold determining the minimal required confidence, this method allows one to generate a map which shows only pixels with prescribed minimal confidence, effectively creating an additional class containing pixels classified with subthreshold confidence.

Introduction

Satellite remote sensing is a progressive way of collecting data used for generation of various kinds of maps (vegetation maps, land-use maps, etc.). Usually, the most important part of processing of these data is classification of the multispectral imagery obtained by remote sensing. The standard approach to this task is the application of statistical classification methods, e.g., Maximum Likelihood classifier [13], or methods based on crude expert-defined rules [11]. These methods have the problem of high complexity and large data volume. It would also be desirable to develop a more autonomously functioning classifier so that subjective human participation in the task could be minimized.

A relatively new and potentionally useful method of classification of remotely sensed data is the method of classification using neural networks. In the last years, several types of neural networks have been used for classification of these data, most of them using the Backpropagation algorithm [15]. Another class of neural networks used in this domain is *Adaptive Resonance Theory* (ART) neural networks (see, e.g., [1],[14]). For an extensive discussion of the neural network versus statistical approaches to image processing see [12]. An interesting approach is described also in [10].

This study concentrates on two aspects of applying ARTMAP neural networks for classification of remotely sensed data. First, the study gives a brief intuitive description of the ARTMAP neural networks (Section 1). In Section 2, performacne of three types of ARTMAP networks, differing in the cluster determination method, is compared. Section 3 then describes a method for evaluation of classification quality for Gaussian ARTMAP neural networks.

The data for this study consists of an Landsat Thematic Mapper (TM) image of the city of Košice (located in Eastern Slovakia) and its environs. The whole image consists of 368,125 7-dimensional pixels, out of which 6,331 pixels were classified by an expert into seven thematic categories. Figure 1 shows the original image along with the seven categories identified by expert. The following classes

Fig. 1. Original image. Highlighted areas were classified by expert (see text).

are defined in the figure: A - urban area, B - barren fields, C - bushes, D - agricultural fields, E - meadows, F - woods, G - water.

1 ARTMAP neural networks

ARTMAP neural networks belong to a group of neural networks based on Adaptive Resonance Theory (ART), a theory of cognitive information processing in human brain [9]. The ART neural networks, as this group of neural networks is collectively known, are especially suitable for pattern recognition and classification applications using supervised as well as unsupervised learning. The first neural network model of this category, ART 1 [2], was a system for unsupervised classification of binary input data. This model was later extended to ART 2 [4] and fuzzy ART [5], ART systems for unsupervised classification of analog data. Also other modifications of the ART 1 model were introduced, for example Gaussian ART [16], which uses Gaussian distributions to define individual categories.

The next step in the development of ART models was the introduction of a new ART architecture, called ARTMAP [6], which was designed for supervised learning of arbitrary mappings of input vectors to output vectors. The ARTMAP neural network, which is the basic model of ARTMAP architecture, is in its basic version a modular system consisting of two ART 1 modules and a controlling module called Map-field. If the ART 1 modules in the ARTMAP architecture are replaced by fuzzy ART modules or Gaussian ART modules, a new ARTMAP model is obtained, called fuzzy ARTMAP [3] or Gaussian ARTMAP [16], respectively. The three above mentioned ARTMAP models (standard, fuzzy, and Gaussian ARTMAP) represent basic models of the ARTMAP architecture. The main difference among them is in the way they identify clusters in feature space. And the first part of the present paper deals with analysis of this difference in identification method, and its influence on computational properties of the three systems.

1.1 ARTMAP systems dynamics

A detailed description of the dynamics of the ART and ARTMAP systems, relevant to our study, can be found, e.g., in [6], [3], and [16], which contain a description of the standard ARTMAP architecture. A simplified version of the ARTMAP architecture was implemented for the present study. A detailed description of it can be found in [14]. In this section, an outline of the learning process in an ARTMAP system is presented with the goal of developing an intuitive understanding of the learning process. The basic structure of an ARTMAP system is shown in Figure 2. Individual blocks in the figure have the following function:

Input Layer (F0) At the beginning of each trial, a pattern is presented into this layer. The size of this layer (number of neurons) is equal to the dimensionality of input patterns (N).

Comparison Layer (F1) In the second step of each trial (see description of the algorithm below), the pattern from the Input Layer is copied into this layer. In the fourth step, the Input pattern and the optimal pattern represented

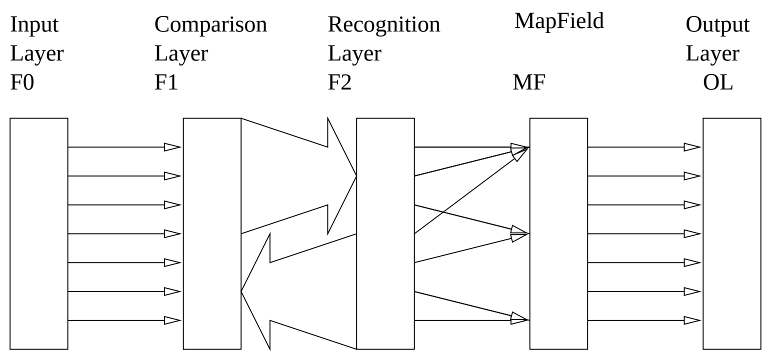

Fig. 2. Topology of ARTMAP neural networks

by the winning neuron in the Recognition Layer are compared here. If they are significantly different, a search for a new winning F2 neuron is initiated. The size of this layer is N.

Recognition Layer (F2) Each neuron in this layer defines one cluster in feature space (the identification method differs for different types of ARTMAP models). The size of this layer is variable. It grows during training as new clusters in feature space are identified.

MapField (MF) Each neuron in this layer represents one class and it receives input by a non-zero link from all the F2 neurons (i.e., all the clusters) which belong to the same class represented by this neuron. During training, the content of the MF layer is identical with the content of the Output Layer.

Output Layer (OL) Each neuron in this layer represents one class defined in the data set. At the beginning of each trial, a code of the class into which the input pattern should be classified is copied into this layer. Each class is represented by one neuron. The size of this layer is equal to the number of classes (M).

The ARTMAP neural network training procedure can be described as follows:

1. The input pattern is presented to the Input Layer. The corresponding class code is presented to the Output Layer.
2. The pattern from the Input Layer is identically copied into the Comparison Layer.
3. If there are no neurons in the Recognition Layer (F2) or if all the F2 neurons are reset, a new neuron is added to this layer. The connections from the F1 layer to the new neuron, which define the cluster identified by that neuron, are initialized to values identical with the input pattern. The new F2 neuron is also connected with the MapField neuron representing the class to which the input pattern belongs. The algorithm then continues by Step 1 (i.e., by a new training cycle).
4. If there are non-reset neurons in the F2 layer, the neuron in the Recognition Layer (F2) defining a cluster most suitable to include the input pattern is chosen as the winner.

5. The cluster associated with the winning F2 neuron (represented as a pattern of connection activations) is compared to the input pattern. If the F2 pattern does not represent the input pattern sufficiently (i.e., if these two patterns are not sufficiently **similar**), the winning F2 neuron is reset and blocked from winning for the remainder of the current training cycle. The algorithm then returns to Step 3.
6. The winning F2 neuron identifies a cluster which belongs to a certain class (category). If this class differs from the class to which the input pattern belongs, the winning F2 neuron is again frozen and the similarity criterion (see Step 5) is temporarily made stricter so that the same winning F2 neuron can not be accepted. The algorithm then continues by Step 3.
7. If the class represented by the winning F2 neuron is identical to the class to which the current pattern belongs, the winner is accepted and the weights of links connecting the winning F2 neuron with the F1 layer are updated so that the cluster associated with the winning F2 neuron is more similar to the current input pattern.

This procedure is repeated until the network classifies all the input patterns correctly.

In the testing/application phase, the procedure is as follows:

1. The unknown input pattern is presented into the Input Layer and copied into the Comparison Layer.
2. A winner in the Recognition Layer is found. If the pattern represented by the winner is not sufficiently similar to the input pattern, the system is not able to classify the input pattern. Otherwise, the input pattern is classified into the MapField class associated with the winning F2 neuron.

There are two adjustable parameters in the ARTMAP networks. The first of them is the baseline similarity threshold (or so-called vigilance parameter) used in the Comparison Layer computations. This parameter can range from zero to one (one meaning that identity of the compared patterns is required). The second parameter is the learning rate and again it can range from zero to one.

It has been shown before that results obtained by ARTMAP networks are dependant on the order in which the input patterns are presented during training. To suppress this presentation-order dependence, a method of *voting* is usually applied. In this method, several (usually five) independent ARTMAP networks are trained on the same data-set presented in different order. In the testing phase a new pattern is classified into the class voted for by majority of the networks.

2 Influence of data representation and cluster identification method on classification accuracy

The description given in the previous section holds, up to some minor details, for all three modifications of the ARTMAP model compared in the present study (standard, fuzzy, and Gaussian ARTMAP). The main difference among the algorithms is in the way the Recognition Layer (F2) neurons define the clusters they represent. This is closely related to the way in which the input patterns have to be pre-processed before training or testing. Individual properties of the examined models can be summarized as follows.

ARTMAP This model requires the data to be in a binary format. So, to be able to apply it on the RS data used in this study the data has to be transformed. We chose to use a mapping which transforms data into binary code by direct conversion of a decimal value into a binary value. Each digit of this value is then processed individually. This transformation has several consequences. First, the dimensionality of the patterns (and consequently the size of the network and the computation time) is considerably larger (eight-fold increase for decimal numbers ranging from 0 to 255). Second, this transformation dramatically changes relations among patterns. For example, values of 0 (00000000 binary) and 128 (10000000 binary) have in binary space the same Euclidian distance as values of 0 (00000000 binary) and 1 (00000001 binary). In ARTMAP, each cluster is defined by a hyperrectangle in feature space (e.g., a cube in the three-dimensional space). In our case it means that each cluster will be represented by a 56-dimensional hyperrectangle. It follows that this transformation will lead to creation of different clusters for binary data compared to analog data.

fuzzy ARTMAP This model accepts analog (continuous) data in the range [0, 1]. So the only transformation of data that needs to be done is rescaling. Each cluster in the Recognition Layer is again defined by a hyper-rectangle of the same dimensionality as is the dimensionality of the input data (in the present study, seven dimensions).

Gaussian ARTMAP In this model, each cluster in the Recognition Layer is defined as a Gaussian probability distribution with a mean and variance in each dimension, and *a priori* probability of a given cluster. This method does not require any transformation of data. But, to obtain better comparability with other methods, we rescaled the data in the same way as in the fuzzy ARTMAP experiments (range [0, 1]).

2.1 Definition of task

Although all three analyzed methods are very similar, as can be seen in the above discussion, the difference in the way they determine individual clusters should lead to significantly different results when used for the same classification task. The goal of this experiment is to compare the three classification methods in terms of classification accuracy achieved. The results should suggest the most suitable method of cluster identification for the remote sensing data used here. Also, the goal is to present Gaussian ARTMAP as a new method for classification of data in remote sensing. The expected results are as follows.

– *ARTMAP:* Because of the high dimensionality which resulting from the transformation required by this algorithm, it is expected that the algorithm will be very sensitive to small changes in data and its ability to generalize will be poor.

– *fuzzy ARTMAP:* Based on previous studies and authors' experiences it is expected that fuzzy ARTMAP will perform very well in this task.
– *Gaussian ARTMAP:* This neural network has been never before used for RS data classification. But the experiments performed by the authors previously on benchmark classification tasks suggest that it should be very powerful.

2.2 Methods

The data set consists of 6331 seven-dimensional patterns. The set was randomly divided into a training set (3166 patterns) and a test set (3165 patterns). Five copies of the training set were created, each containing the training patterns in a different random order. The training and testing sets were then transformed for each individual method (binarization for ARTMAP, scaling for fuzzy and Gaussian ARTMAP). For each method, optimal values of parameters were estimated using a validation technique in which a sequence of trainings and tests was repeated for different combinations of parameter values, training on a subset containing 90% of the training set and testing on the remaining 10% of the training set. After optimal values of parameters were found, five independent networks of each type were trained on the training sets. All the networks were then tested on the test set using the *voting* method. The parameters used in training of each network were as follows (ρ-baseline vigilance/similarity, β-learning rate, and γ-initial variance in Gaussian ARTMAP): ARTMAP ($\rho = 0.3, \beta = 1$), fuzzy ARTMAP ($\rho = 0.8, \beta = 1$), Gaussian ARTMAP ($\rho = 0.0, \beta = 1, \gamma = 0.5$).

2.3 Results and discussion

The performance of the methods was evaluated in terms of per cent of correctly classified test set patterns weighted by the size of each class (weighted PCC). Also, a confusion matrix for each of the methods was computed. Table 1 gives

Table 1. Performance (in weighted % of correctly classified test patterns) of the three methods on individual training sets and for voting

	Set #1	Set #2	Set #3	Set #4	Set #5	Voting
ARTMAP	90.76	88.95	90.27	88.47	89.51	92.20
fuzzy ARTMAP	93.72	91.48	91.66	90.82	92..16	93.95
Gaussian ARTMAP	93.90	93.57	93.49	94.24	93.09	94.04

the performance of the three methods expressed in weighted PCC. For each network type, individual performance (after training on a single training set) as well as overall performance (obtained using *voting*) is shown. The results show that the highest classification accuracy is obtained by the Gaussian ARTMAP neural network, both on individual training sets and with voting. The classification accuracy is slightly worse for fuzzy ARTMAP, and considerably worse for ARTMAP algorithm. The poor performance of the ARTMAP network can be assigned, as expected, to the poor ability of the network to generalize. This is suggested also by the fact that this network used in each training approximately 200 F2 neurons (clusters) whereas the other two methods needed only 80 to 90 F2 neurons. Another important result shown in the table is that Gaussian ARTMAP, compared to fuzzy ARTMAP, is much less sensitive to the training set ordering. This observation is supported by the smaller variance in PCC obtained when Gaussian ARTMAP was trained on individual training sets, as well as by the fact that the improvement obtained by application of *voting* on this method is much smaller than that for fuzzy ARTMAP. The difference in the performance is even stronger if non-weighted PCC is used. In this case (data not shown), Gaussian ARTMAP without voting performed better than fuzzy ARTMAP with voting.

Tables 2, 3, and 4 show the confusion matrices for each neural network model with voting. The tables show that, although there are significant differences in

Table 2. Confusion matrix for ARTMAP network with voting (weighted PCC = 92.20). Each item in the table gives the per cent of pixels from a given Actual Class (column) classified into given Predicted Class (row). The Total for each Actual Class (bottom row) gives per cent of patterns in the test set belonging to the corresponding Actual Class. The Total for each Predicted Class has analogous meaning

	Actual Class							
Predicted Class	A	B	C	D	E	F	G	Total (%)
A'	**84.82**	0.35	0.52	0.00	0.46	0.00	2.64	2.21
B'	4.02	**99.40**	1.57	0.00	0.00	0.00	0.00	36.84
C'	2.68	0.08	**75.17**	0.46	0.46	4.55	3.25	5.40
D'	0.00	0.00	0.52	**94.11**	0.00	13.13	0.61	28.78
E'	1.34	0.16	2.80	0.00	**98.92**	0.00	0.00	6.67
F'	5.80	0.00	17.13	5.46	0.00	**82.13**	1.22	15.36
G'	1.34	0.00	2.27	0.00	0.00	0.19	**92.29**	4.74
Total (%)	2.24	36.87	5.72	28.37	6.48	15.39	4.93	100.00

size of individual classes, all three methods classify almost evenly well patterns from all the categories. Also, there is no pair of categories which would be systematically confused by any of the networks. This suggests that there is no significant overlap in the data set and that the obtained performance reflects almost exclusively each network's capability to classify the data and to generalize information contained in the training data set. The color-encoded classification map obtained by the Gaussian ARTMAP network is shown in Figure 3.

All these results show that Gaussian ARTMAP is the best method for the chosen task. There is only one exception to this observation. The classification accuracy (PCC) of this network on the training set after the training was finished

Table 3. Confusion matrix for fuzzy ARTMAP network with voting (PCC = 93.95). Format as described in Table 2

	Actual Class							
Predicted Class	A	B	C	D	E	F	G	Total (%)
A'	**88.84**	0.87	1.57	0.00	0.00	0.00	2.64	2.53
B'	2.68	**98.97**	0.00	0.00	0.00	0.00	0.00	36.56
C'	5.80	0.16	**79.55**	0.11	0.00	2.47	3.25	5.31
D'	0.00	0.00	0.52	**96.33**	0.00	8.45	0.00	28.66
E'	0.00	0.00	2.27	0.00	**100.00**	0.00	0.00	6.60
F'	1.34	0.00	12.76	3.56	0.00	**88.30**	0.61	15.39
G'	1.34	0.00	3.32	0.00	0.00	0.84	**93.51**	4.96
Total (%)	2.24	36.87	5.72	28.37	6.48	15.39	4.93	100.00

Table 4. Confusion matrix for Gaussian ARTMAP network with voting (PCC = 94.04). Format as described in Table 2

	Actual Class							
Predicted Class	A	B	C	D	E	F	G	Total (%)
A'	**91.52**	0.35	1.05	0.11	0.00	0.00	1.22	2.34
B'	0.00	**99.48**	0.00	0.00	0.00	0.00	0.00	36.68
C'	5.80	0.00	**87.24**	0.11	0.00	4.55	4.46	6.07
D'	1.34	0.00	0.00	**97.22**	0.00	7.21	0.00	28.72
E'	0.00	0.16	0.00	0.00	**100.00**	0.00	0.00	6.54
F'	0.00	0.00	8.92	2.57	0.00	**88.04**	1.22	14.85
G'	1.34	0.00	2.80	0.00	0.00	0.19	**92.90**	4.80
Total(%)	2.24	36.87	5.72	28.37	6.48	15.39	4.93	100.00

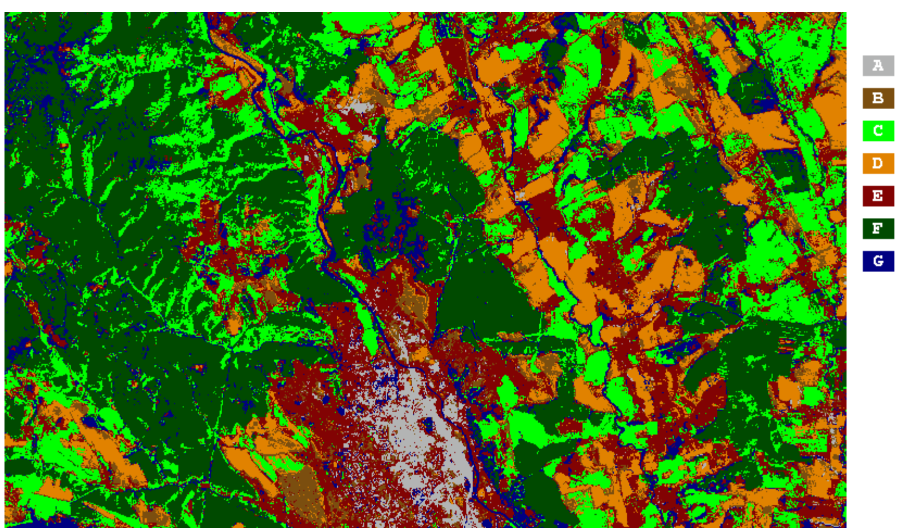

Fig. 3. Classification map obtained by the Gaussian ARTMAP classifier

was always less than 100% (usually 97.7%), whereas for ARTMAP and fuzzy ARTMAP this parameter always reached 100%. This result can be interpreted in two ways. On the one hand it means that there may be tasks for which fuzzy ARTMAP or ARTMAP are more suitable, e.g., in case when all the data are available during the training. On the other hand this result underscores the generalization capabilities of Gaussian ARTMAP because it means that the network is able to identify and ignore noisy data in the training set.

3 Classification quality evaluation using confidence index

It is often very useful to have a measure of confidence of classification, especially in the remote sensing area where the differences in spectral content of patterns belonging to the same class can be quite considerable. There are several statistical approaches to this problem (see, e.g., [8]). But these methods are usually very computationally intensive. Therefore it would be very useful to find a method for assigning confidence to each classified pixel exploiting computations done as part of the classification process itself. In this section, a method is developed for evaluation of the classification confidence in Gaussian ARTMAP, the system which showed the best performance in classification of the remotely sensed data (see previous section).

3.1 Description of the method and results

In Gaussian ARTMAP, as mentioned in Section 2, each category comprises of a set of clusters. Each of these clusters contains patterns from a Gaussian probability distribution defined by a mean and variance in each dimension of the input space. When a new pattern is presented, the probability of that pattern belonging to each of the existing clusters is computed using a Bayes discrimination function (see [7], p. 24)

$$g_j(I) = \log\left(p(I|\omega_j)\right) + \log P(\omega_j) \tag{1}$$

where I is the input pattern, $p(I|\omega_j)$ is the conditional density of I given cluster j, and $P(\omega_j)$ is the *a priori* probability of cluster j. In Gaussian ARTMAP the pattern is classified into the most probable category. The probability measure of the new pattern belonging to a given category is defined as a sum of probabilities of the new pattern belonging to any of the clusters associated with that category, i.e.,

$$R_k(I) = \sum_{j \in \Omega(k)} \exp(g_j(I)) \tag{2}$$

where $\Omega(k)$ represents the set of clusters associated with category k. The new pattern is then assigned to the category k with the highest probability measure $R_k(I)$. If the *voting* method is used to suppress the influence of training pattern ordering (see Section 1.1), the probability measure defined by equation 2 is extended over all the networks used in *voting*

$$R_k(I) = \sum_{l=1}^{V} \left(\sum_{j \in \Omega(k)} \exp(g_{j,l}(I)) \right) \tag{3}$$

where V is the number of networks used for *voting*.

The probability measure of input I belonging to category k, $R_k(I)$, is evaluated for every category. And this fact offers a straightforward way to define a confidence of the decision made by the system when category K was chosen for input I. We define this confidence index as

$$c(I) = \frac{R_K(I)}{\sum_{k=1}^{M} R_k(I)}. \tag{4}$$

This index is evaluated for each individual pixel of the image and its value can range from 0 to 1. Using a simple encoding which assigns a shade of gray to each value of index *c(I)* a confidence map for the classified image can be obtained. As an example, Figure 4 shows the confidence map for the image classified by

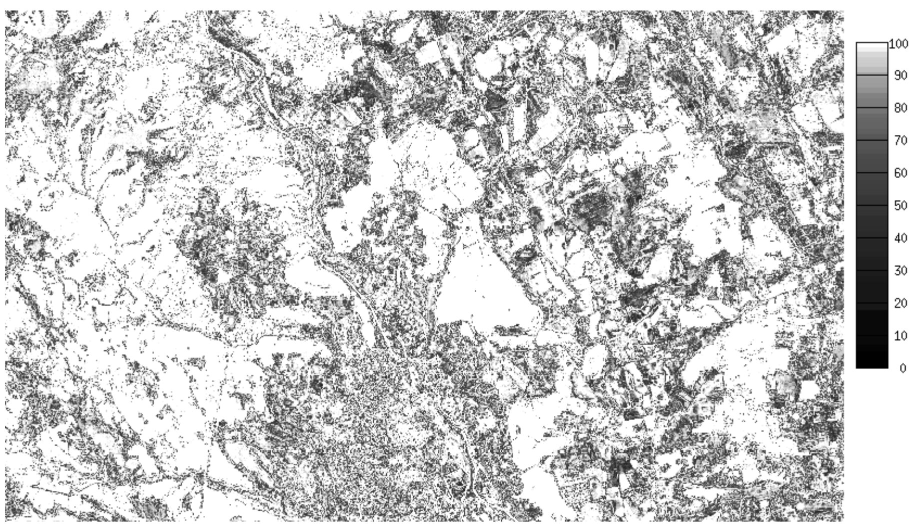

Fig. 4. Confidence map of the image classified by Gaussian ARTMAP using *voting* (confidence expressed in %)

the Gaussian ARTMAP model with *voting* (shown in Figure 3).

Next, a confidence threshold, Θ, can be introduced, which defines the minimal confidence required for a pixel to be assigned into the proposed category. Then, if the confidence for the pixel is subthreshold, the pixel can be classified as belonging to an unknown category. The choice of the confidence threshold value influences two counteracting aspects of the classification. First, the higher the value of Θ, the higher is the classification accuracy of the suprathreshold pixels. But, a higher value of Θ also means that fewer pixels will be classified into any of the known categories. To analyze these two aspects, a pair of graphs is shown in Figure 5. The dashed lines in the Figure show the threshold value ($\Theta = 0.921$)

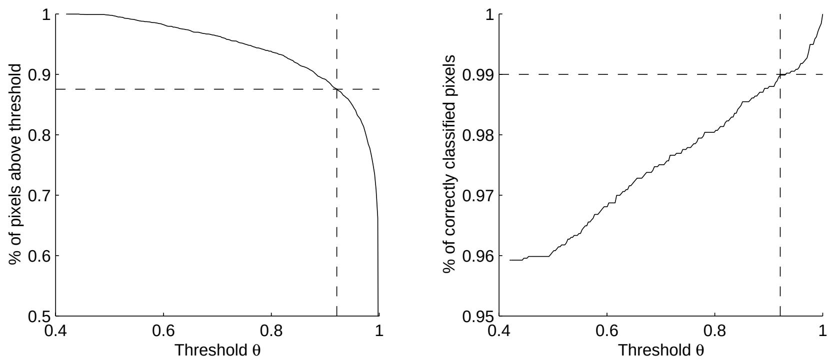

Fig. 5. Per cent of suprathreshold pixels of the image (on the left) and non-weighted per cent of correctly classified pixels (on the right) as a function of value of the confidence threshold Θ. See discussion in text.

which assures that 99% of the suprathreshold pixels will be classified correctly (non-weighted). With this threshold, the system will be able to classify 87.5% of the image. The resulting image is shown in Figure 6. In this image, the white color represents a new category of pixels unknown to the system.

4 Conclusion

This paper gives a description of the application of three different types of ARTMAP neural network for classification of images obtained from remote sensing. First, a description of the application of the ARTMAP methods for remote sensed data is given. Second, a comparison of performance of three ARTMAP neural networks differing in the input data representation method and the cluster identification method is presented. This analysis shows that Gaussian ARTMAP, which identifies clusters as data from Gaussian probability distributions, is the best classifier for this kind of data. Third, a method based on computational properties of the Gaussian ARTMAP neural network is described which allows assignment of a confidence measure, called confidence index, to the classification of each pixel. This makes it possible to create confidence maps and thresholded classification maps with prescribed classification accuracy. These maps allow a deeper insight into performance of the classifier.

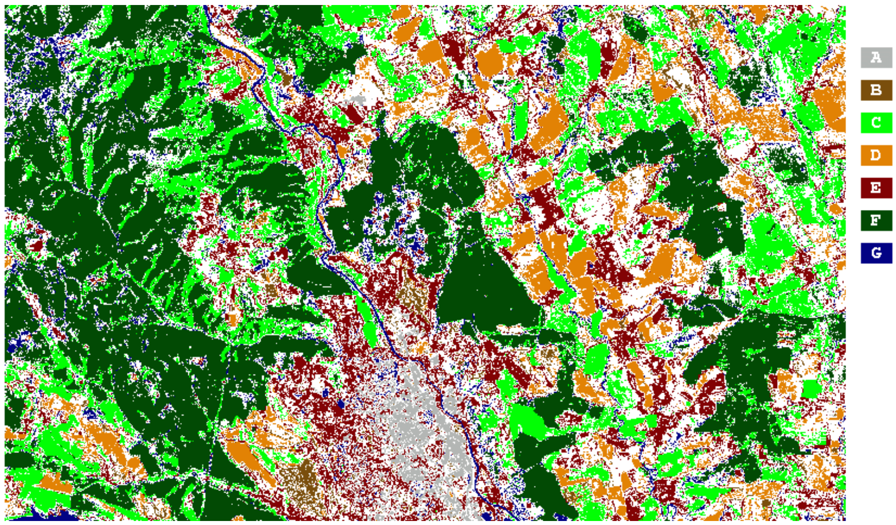

Fig. 6. Thresholded classification map obtained by the Gaussian ARTMAP classifier ($\Theta = 0.921$)

References

1. 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 Transactions on Geoscience and Remote Sensing*, 35(2):308–325, 1999.
2. G.A. Carpenter and S. Grossberg. A massively parallel architecture for a self-organizing neural pattern recognition machine. *Computer Vision, Graphics, and Image Processing*, 37:54–115, 1987.
3. G.A. Carpenter, S. Grossberg, N. Markuzon, J.H. Reynolds, and D.B. Rosen. Fuzzy ARTMAP: A neural network architecture for incremental supervised learning of analog multidimensional maps. *IEEE Transactions on Neural Networks*, 3(5):698–713, 1992.
4. Gail Carpenter and Stephen Grossberg. Art 2: Self-organization of stable category recognition codes for analog input patterns. *Applied Optics*, 1987. Special issue on neural networks.
5. Gail A. Carpenter, Stephen Grossberg, and B. Rosen David. FUZZY ART: an adaptive resonance algorithm for rapid, stable classification of analog patterns. Technical Report CAS/CNS-TR-91-006, Boston University, 1991.
6. Gail A. Carpenter, Stephen Grossberg, and John Reynolds. ARTMAP: Supervised real-time learning and classification of nonstationary data by a self-organizing neural network. *Neural Networks*, 4:565–588, 1991.
7. R.O. Duda and P.E. Hart. *Pattern Classification and Scene Analysis*. Wiley, New York, 1973.
8. K. Fukunaga. *Introduction to Statistical Pattern Recognition*. Academic Press, San Diego, CA, 1972.
9. Stephen Grossberg. Adaptive pattern classification and universal recoding, I: Feedback, expectation, olfaction, and illusions. *Biological Cybernetics*, 23:187–202, 1976.

10. International Neural Network Society. *Interactive Segmentation of Biomedical Images*, Brno, Czech Republic, May 1998. Vitium Press. ISSN 1211-421X, ISBN-80-241-1169-4.
11. B.G. Lees and K. Ritman. Decision-tree and rule-induction approach to integration of remotely sensed and gis data in mapping vegetation in disturbed or hilly environments. *Environmental Management*, 15:823–831, 1991.
12. Robert J. Mokken. Remote sensing, statistics and artificial neural networks. Some R and D perspectives., 1995.
13. J. Richards. *Remote sensing digital image analysis: An introduction.* Springer-Verlag: Berlin, 1993.
14. P. Sinčák, H. Veregin, and N. Kopčo. Conflation techniques in multispectral image processing. *Submitted to IJRS in February 1997*, 1997.
15. P. Werbos. *Beyond Regression: New Tools for Prediction and Analysis in the Behavioral Sciences.* PhD thesis, Harvard University, 1974.
16. James R. Williamson. Gaussian ARTMAP: A neural network for fast incremental learning of noisy multidimensional maps. *Neural Networks*, pages 881–897, 9 1996.