Classification of Noisy Signals Using Fuzzy ARTMAP Neural Networks

Dimitrios Charalampidis*, Michael Georgiopoulos#, Takis Kasparis+

*dim@bruce.engr.ucf.edu, #michaelg@pegasus.cc.ucf.edu, +kasparis@pegasus.cc.ucf.edu
School of Electrical Engineering and Computer Science
University of Central Florida, Orlando, FL 32816, USA

Abstract - This paper describes an approach to classification of noisy signals using a technique based on the Fuzzy ARTMAP neural network (FAM). A variation of the testing phase of Fuzzy ARTMAP is introduced, that exhibited superior generalization performance than the standard Fuzzy ARTMAP in the presence of noise. We present an application of our technique for textured grayscale images. We perform a large number of experiments to verify the superiority of the modified over the standard Fuzzy ARTMAP. More specifically, the modified and the standard FAM were evaluated on two different sets of features (fractal-based and energy-based), for three different types of noise (Gaussian, uniform, exponential) and for two different texture sets (Brodatz, aerial). Furthermore, the classification performance of the standard and modified Fuzzy ARTMAP was compared for different network sizes.

Keywords – Fuzzy ARTMAP, Noise, Fractal Dimension, Energy, Texture, Classification, Segmentation

1. Introduction

During the past few years, neural networks (NN) have been extensively used for classification tasks. The main body of the work in this area has been concentrated in the use of feedforward NN models. Another family of NN architectures called ART architectures has been developed that can address successfully classification problems. This family includes Fuzzy ARTMAP NN (FAM) [1], Gaussian ARTMAP [2] etc. Like other members of the ARTMAP family, FAM has certain advantages over traditional feedforward NN models. One advantage of FAM is that it is faster than other neural networks, due to the small number of training epochs required by the network to "learn" the input data. FAM is even faster than other ARTMAP techniques due to the computationally "cheap" input/output mapping. Also, the classification results of FAM are easily interpretable. Compared to nearest neighbor techniques, which are also commonly used, FAMMN requires less memory and classification time since it uses a compressed representation of the data.

In this work, we introduce a variation of FAM that exhibits superior generalization performance than the standard FAMMN when the signals, are corrupted by additive noise. The variation of FAM, which is named FAM-m, uses prior knowledge of the characteristics of the feature set to adjust the regions of dominance of each class accordingly. We show that FAM-m does not introduce significant additional computational complexity to the standard FAM. In this paper we consider classification of textured images which are a case of 2-D signals, and our approach can be easily extended to other types of signals (e.g., 1-D, 2-D, etc.). Gaussian ARTMAP [2] has been used before for classification of natural textures [3].

The proposed modification of FAM is especially suited to applications where it is required that the feature set captures only the shape characteristics of the signal and not its actual amplitude or average value. Electrocardiographs used for medical diagnosis, speech signals, textured images and satellite images are examples of these type signals. For example, it is important to be able to classify correctly a texture that has been obtained in different illumination environments, or a speech signal independently of how loud it is. Fractal dimension (FD) is suitable for the aforementioned type of applications because it is independent of linear transformations applied to the signal [4][5][6]. Also, the normalized energy features (NE) that are used in this work are insensitive to linear transformations. Furthermore, in order to make the application more general, we use features that do not contain rotational information.

The paper is organized as follows. In Section 2, an overview of the Fuzzy ARTMAP architecture is presented. In section 3, the feature sets are presented. In Section 4, the proposed classification technique (FAM-m) is discussed. In Section 5 we present the experimental results and finally, in section 6 we conclude with some closing remarks.

2. Fuzzy ARTMAP Neural Network

The FAM is described in [2] and only a short description of it will be presented here. For the **training phase** of FAM, a list of *MP* training input/label pairs, such as $\{\mathbf{I}^1, O^1\}$, $\{\mathbf{I}^2, O^2\}$, . . . ,$\{\mathbf{I}^{MP}, O^{MP}\}$, is presented

repeatedly to the FAM until the desired mapping is established for all pairs. During the training phase a number N_a of nodes is created. Each node is represented by a weight vector which is designated by $\mathbf{w}_j^a = \{w_{j1}^a, \ldots, w_{j2M_a}^a\}$ and it is called template. Each template defines a M_a – dimensional hyper-box that includes within its boundaries all the training input patterns that were coded by the template. The first M_a elements of the template define a vector, which is the lower endpoint of the hyperbox and the last M_a elements define the complement, $1-\mathbf{v}_j^a$,of a vector $\mathbf{v}_j^a$, which is the upper endpoint. Consider the r-th input/label pair (i.e., $\{\mathbf{I}^r, \mathbf{O}^r\}$) from the training list. The bottom-up input of pattern $\mathbf{I}^r$ to nodes j is calculated according to:

$$T_j^a(\mathbf{I}^r) = \frac{|\mathbf{I}^r \wedge \mathbf{w}_j^a|}{\beta_a + |\mathbf{w}_j^a|} \tag{1}$$

where β_a is called the ART_a choice parameter. From the set of nodes that satisfy the vigilance criterion (see [2]), we choose the one that receives the maximum bottom-up input. If due to prior learning node j_{max} is mapped to a label different than $\mathbf{O}^r$ then the mapping is incorrect and the node is disqualified. The input pair is presented repeatedly, until a node is finally selected and the corresponding template is updated. After all patterns have been presented and no template changes occurred, the learning process is considered complete, otherwise the patterns are presented again. For the **test phase**, each input pattern **I** from the test list is assigned the label of the node that receives the maximum bottom-up input and satisfies the vigilance criterion. If no node satisfies the vigilance criterion, the label of the input pattern is designated as "unknown". We must note that the fuzzy min operator ($\wedge$) of two vectors $\mathbf{w}_1$ and $\mathbf{w}_2$ is a vector whose components are equal to the minimum of the corresponding components of $\mathbf{w}_1$ and $\mathbf{w}_2$, and the "size" ($|\cdot|$) of a vector **w** is defined to be equal to the sum of its components.

3. Feature sets

The first feature set is based on normalized energy (NE). First, the original image is filtered using Gabor filters [7] of standard deviation σ and central frequency u_o. Then, the energy in a window W of size R x R that is centered at the pixel with coordinates (x,y) can be defined as:

$$E_{u_o\sigma}^W(x,y) = \sum_{x'=x-R/2}^{x+R/2} \sum_{y'=y-R/2}^{y+R/2} |Z_{u_o}^{\sigma}(x',y') - m_{u_o\sigma}^W| \tag{2}$$

where $m_{u_o\sigma}^W$ is the mean grayscale value in the window W of the filtered image. In order to make energy dependent only on the texture and not on the magnitude of the image, we normalize it by dividing with the energy in the corresponding window W of the original image $E^W(x,y)$. Therefore, we define the normalized energy (NE) as the ratio $En_{u_o\sigma}^W(x,y) = E_{u_o\sigma}^W(x,y) / E^W(x,y)$. The first feature set consists of the NE of twelve filter versions of the original image as it is defined by (5), for twelve different central frequencies $u_o = 0.04k$, $k = 1, 2..12$, and constant standard deviation $\sigma = 23$.

The second feature set consists of six FD-based features that are computed using the variation method [8]. We use the FD, the FD of the higher grayscale values and the FD of the lower grayscale values, where the slope of the line that passes through two points $(\log(1/\delta), \log\{(R/\delta)^3 V_{\delta/R}\})$, $\delta = (1, 2)$ is considered (see [8]). We also use the FD, the FD of the higher grayscale values and the FD of the lower grayscale values, where the slope of the line that passes through two points $(\log(1/\delta), \log\{(R/\delta)^3 V_{\delta/R}\})$, $\delta = (2, 3)$ is considered. The FD of higher and lower grayscale values is actually the FD computed on the images defined as

$$Z^h(x,y) = \begin{cases} Z(x,y), \text{ if } Z(x,y) > m^W \\ FLAG, \text{ if } Z(x,y) < m^W \end{cases} \qquad Z^l(x,y) = \begin{cases} Z(x,y), \text{ if } Z(x,y) < m^W \\ FLAG, \text{ if } Z(x,y) > m^W \end{cases} \tag{3}$$

respectively, where m^W is the average grayscale value in a window W of size R x R around (x,y). A *FLAG* means that the corresponding pixel value is not included in the calculation of FD.

4. The FAM modification

In this section, we present a modification of the FAM, named FAM-m that performs well in the presence of noise. Prior to classification, the features are extracted from the textures. The set of feature vectors extracted from the textures is divided into training and test set. The training of the FAM-m is exactly the same as the training phase

of the standard FAM. The bottom-up input of node j as it is defined by (1), includes the calculation of the denominator $D_j = \beta_a + |\mathbf{w}_j|$. We notice that after training is over, D_j remains unchanged. Even if the training is on-line, this quantity remains unchanged when the test phase takes place. The quantities D_j for every node j, are stored in memory along with the templates $\mathbf{w}_j$, so that they are not recalculated in the test phase. The bottom-up input is expressed as:

$$T_j^a(\mathbf{I}^r) = \frac{|\mathbf{I}^r \wedge \mathbf{w}_j^a|}{D_j} \tag{4}$$

The advantage that this implementation offers to the test phase of FAM-m will be apparent after the description of the test phase of FAM-m.

Assume that we are dealing with an M_a-dimensional hyperspace. Each node can then be represented as an M_a-dimensional hyperbox. When a pattern is tested, a hyperbox competes with other hyperboxes to associate this pattern with the class that it represents. Generally, each hyperbox is dominant in the region that it occupies and in some region around it. We have shown experimentally that patterns that are extracted from a noisy texture will move, at least in most of the cases, towards the direction of the pattern that is extracted from a pure noise surface. We name this pattern *the noise pattern* $\mathbf{I}^N$. Therefore, if in the test phase of the FAM we modify the region of influence of each hyperbox so that it gives more emphasis to coordinates that are further away from the noise pattern than the ones that are closer, then we have a better chance of correctly classifying the noisy textures. The modification of the test phase of the FAM needs to be such that it also allows the correct classification of noiseless textures. We have implemented a variation of the test phase of FAM (i.e., FAM-m) to achieve the aforementioned objectives. First, we define the distance between a template $\mathbf{w}$ and a pattern $\mathbf{I}$ as:

$$\text{dist}(\mathbf{w},\mathbf{I}) = |\mathbf{w}| - |\mathbf{w} \wedge \mathbf{I}| \tag{5}$$

In the special case where the template $\mathbf{w}$ is a pattern $\mathbf{I}'$, the distance between $\mathbf{I}$ and $\mathbf{I}'$ as it is defined in (5) is simply the distance between the patterns in terms of the L_1 norm:

$$\text{dist}(\mathbf{I}',\mathbf{I}) = \|\mathbf{I}-\mathbf{I}'\|_1 = \sum_{i=1}^{M_a} |I_i - I_i'| \tag{6}$$

Assume that the hyperbox defined by $\mathbf{w}_2$ is closer to $\mathbf{I}^N$ than the one defined by $\mathbf{w}_1$: $\text{dist}(\mathbf{w}_1,\mathbf{I}^N) > \text{dist}(\mathbf{w}_1,\mathbf{I}^N)$. Then, node 2 benefits from this movement over node 1 because it tends to "capture" patterns that belong to node 1 and move towards $\mathbf{I}^N$. In our approach we modify the boundary between the regions of dominance by dividing the bottom-up input of a node with a term that depends on the corresponding template $\mathbf{w}$. We denote this term as $N(\mathbf{w})$. The function $N(\mathbf{w}) = M_a - \gamma\, \text{dist}(\mathbf{w}, \mathbf{I}^N)$, where γ is a small constant associated with the amount of noise that has affected the textures, is a good choice. First, if the distance of two nodes from $\mathbf{I}^N$ is the same: $\text{dist}(\mathbf{w}_1, \mathbf{I}^N) = \text{dist}(\mathbf{w}_2, \mathbf{I}^N)$ we do not want to favor one node versus the other. Second, if $\text{dist}(\mathbf{w}_1, \mathbf{I}^N) < \text{dist}(\mathbf{w}_2, \mathbf{I}^N)$ this modification favors node 1 by moving the boundary between regions of dominance towards node 2. The bottom-up input for node j of FAM-m for the test phase is equal to:

$$Tm_j^a(\mathbf{I}^r) = \frac{1}{M_a - \gamma\, \text{dist}(\mathbf{w}_j^a,\mathbf{I}^N)} \cdot \frac{|\mathbf{I}^r \wedge \mathbf{w}_j^a|}{\beta_a + |\mathbf{w}_j^a|} \tag{7}$$

where $\mathbf{I}^r$ is the r-th input pattern from the list of test patterns. The value of γ can be the largest possible so that the modification does not introduce more than a specified percentage of extra misclassification on the training set.

A two-dimensional example that shows the effect of the modification is shown in Figure 1. The input patterns in this example belong to two classes denoted as class 1 and class 2 and they are represented by "x" and "+"respectively. These patterns are NE–based feature vectors that are actually extracted from two aerial textures respectively. The "light gray" area represents the region of dominance of the hyperboxes that belong to class 1, and the "white" area the region of dominance of the hyperboxes that belong to class 2. The "dark gray" area indicates the region in which the modification has effect. More specifically, it is the region that belongs to class 1 for the standard FAM but it belongs to class 2 for the FAM-m. We can see that the boundary between the regions of dominance of the two classes has been shifted towards class 1, which is the one closer to the noise pattern. For this example the value of γ was selected equal to 0.45. The effect of the modification is the following: for textures that have not been affected by noise FAM may have slightly better classification performance over FAM-m but when the textures are affected by noise FAM-m gives significantly better results. Experimental results show that the modification is effective.

As a reminder, the denominator D_j of the bottom-up input was stored and it remains constant in the test phase. In the case of the FAM-m the denominator will be:

$$D_j^m = [M_a - \gamma\, dist(\mathbf{w}_j^a, \mathbf{I}^N)]\,[\beta_\alpha + |\mathbf{w}_j|] \quad (8)$$

The new denominator D_j^m can be stored in memory as well, since it depends only on the template $\mathbf{w}_j$ that remains constant in the test phase. As a result the bottom-up input in the test phase will be equivalent to the bottom-up input as it is defined in (4), and does not introduce any more calculations than (4).

5. Experimental Results

We performed a large number of experiments in order to illustrate the effectiveness of the proposed FAM-m for classification of signals affected by additive noise. We considered the application of texture classification. Two different texture sets were used. The first texture set consisted of 20 textures obtained from the Brodatz album [9]. The second texture set consisted of 20 textures obtained from aerial images. A large number of patterns was extracted from both texture sets as it will be described later. We considered these two texture sets because we wanted to examine the classification performance of FAM-m on Brodatz textures which are obtained in ideal environmental conditions, and aerial textures that represent a more realistic situation.

Two feature sets with different characteristics were extracted from each texture set. The first one consisted of 12 NE-based features and the second one consisted of 6 FD-based features as they were described in section 3. The NE-based feature set is more robust to noise than the FD-based feature set, but the classification results that were obtained with the FD-based feature set are better when noise is not present. The training and testing were performed separately for each feature set. Let us consider the NE-based feature set. For *Brodatz textures*, a total of 2560 either NE or FD feature vectors (or equivalently patterns) that were extracted from non-overlapping windows of size 16x16 were used for training. A total of 1280 feature vectors were used for testing in the case where noise is not present. For *aerial textures*, a total of 1280 either NE or FD feature vectors that were extracted from non-overlapping windows of size 16x16 were used for training. A total of 640 feature vectors were used for testing in the case where noise is not present. The size of the windows was selected to be relatively small (16x16) so that segmentation of more that one texture in the same image is possible if desired. We considered 3 different types of additive noise, namely Gaussian noise, uniform noise and exponential noise. For each type of noise we considered different values of standard deviation. Approximately 82,000 NE-based and 82,000 FD-based feature vectors were extracted from the 20 Brodatz textures for *each* type of noise and for *each* value of standard deviation. Approximately 20,000 NE-based and 20,000 FD-based feature vectors were extracted from the 20 aerial textures for *each* type of noise and for *each* value of standard deviation. The scope of this paper is to illustrate the superior performance of FAM-m over FAM in the presence of noise independently of the size of the network. For this reason, we compared the FAM and the FAM-m for different number of nodes.

All combinations of the parameters described above were considered. More specifically, the classification performance of FAM and FAM-m was examined on the two texture sets, for the two feature sets, for three different types of noise, for different values of the standard deviation of noise, and for different sizes of the networks. The classification results are shown in Tables 1 and 2. Table 1 and table 2 present the classification results when the FD-based feature set and the NE-based feature set is used respectively, when noise is present.

We must note that the noise pattern $\mathbf{I}^N$ was extracted from a pure Gaussian noise texture for both FD-based and NE-based feature sets. The noise pattern could have been extracted from any other type of white noise. The reason is that the distance between the noise pattern extracted from a pure noise texture and the patterns extracted from other textures is relatively large, independently of the type of noise. The value of γ was found to be approximately equal to 0.1 for the NE-feature set, and approximately equal to 0.2 for the FD-feature set.

Noise:	**Gaussian**		**Uniform**		**Exponential**	
St. Devi.:	14.2	18.2	14.2	18.2	14.2	18.2
FAM 357 Nodes	69.3	55.4	66.2	54.4	73.3	65.4
FAM-m 357 Nod.	**72.6**	**62.3**	**69.4**	**58.1**	**73.8**	**68.5**
FAM 657 Nodes	69.6	55.6	71.4	57.0	74.0	64.7
FAM-m 657 Nod.	**73.3**	**62.5**	**72.8**	**60.1**	**75.3**	**68.8**
FAM 1280 Nodes	70.2	56.7	71.3	56.7	76.2	67.4
FAM-m1280 Nod.	**73.0**	**61.3**	**72.4**	**59.2**	**77.2**	**69.6**

(a)

Noise:	**Gaussian**		**Uniform**		**Exponential**	
St. Devi.:	14.2	18.2	14.2	18.2	14.2	18.2
FAM 268 N.	72.4	55.7	63.6	47.1	81.0	66.2
FAM-m 268 N.	**81.7**	**65.4**	**72.4**	**55.5**	**86.8**	**74.3**
FAM 577 N.	73.5	56.0	63.7	49.2	80.0	67.0
FAM-m 577 N.	**79.9**	**66.0**	**70.4**	**55.2**	**86.7**	**74.8**
FAM 1348 N.	74.2	56.6	66.3	50.9	81.3	66.7
FAM-m1348 N.	**79.8**	**63.4**	**72.9**	**56.4**	**87.1**	**74.3**
FAM 2560 N.	74.2	57.1	66.3	50.3	81.0	67.5
FAM-m2560 N.	**79.5**	**63.3**	**71.2**	**55.1**	**87.0**	**75.3**

(b)

Table 1: PCC of FAM and FAM-m for the FD feature and (a) aerial textures (b) Brodatz textures.

Noise:	Gaussian			Uniform			Exponential		
St. Dev.:	14.4	21.6	28.8	14.4	21.6	28.8	14.4	21.6	28.8
FAM-357 Nodes	77.8	68.4	62.7	79.1	67.3	64.1	80.8	72.7	64.5
FAM-m 357 Nod.	**82.0**	**73.3**	**70.1**	**83.0**	**73.8**	**70.0**	**82.9**	**77.9**	**76.4**
FAM-657 Nodes	84.0	74.1	69.4	83.6	72.3	70.1	85.2	79.1	74.0
FAM-m 657 Nod.	**84.6**	**78.0**	**74.1**	**84.0**	**76.3**	**74.7**	**84.9**	**82.2**	**79.2**
FAM-1280 Nodes	82.4	70.6	66.2	82.0	69.6	66.9	83.5	75.5	70.6
FAM-m 1280 N.	**84.5**	**72.8**	**68.6**	**82.5**	**71.0**	**68.8**	**83.9**	**77.9**	**73.6**

(a)

FAM-268 Nodes	87.9	77.6	69.6	87.2	76.2	67.3	88.4	84.5	80.1
FAM-m 268 N.	**88.6**	**83.7**	**75.7**	**88.5**	**82.1**	**72.5**	**89.0**	**86.5**	**84.0**
FAM-577 Nodes	88.4	81.5	74.7	88.3	80.5	74.1	88.4	84.7	82.3
FAM-m-577 N.	**89.4**	**84.6**	**78.9**	**89.2**	**83.5**	**77.1**	**89.4**	**86.5**	**84.6**
FAM-1348 Nodes	90.0	86.4	80.4	89.7	85.1	76.0	89.7	88.2	86.3
FAM-m-1348 N.	**90.3**	**88.3**	**84.1**	**90.0**	**87.6**	**79.1**	**90.3**	**89.3**	**88.1**
FAM-2560 Nodes	89.4	84.8	78.7	88.8	82.8	74.6	89.7	87.1	83.8
FAM-m-2560 N.	**90.0**	**86.6**	**82.4**	**89.6**	**85.6**	**77.6**	**89.9**	**88.7**	**86.8**

(b)

Table 2: PCC of FAM and FAM-m for the NE feature and (a) aerial textures (b) Brodatz textures

In summary, all experiments show that the PCC is always larger for FAM-m when the standard deviation of noise is larger than 7.2, independently of the type of the noise, in the texture set and the feature set. The difference is larger for larger values of the standard deviation. When noise is not present and when the standard deviation of noise is equal to 7.2, the PCC is similar for the FAM and the FAM-m. The PCC is slightly better (about 0.5% in the average) for FAM if noise is not present, which is the price paid for the significantly improved performance of FAM-m over FAM when noise is present. When noise is not present, the average PCC of all networks is 85% and 90% for NE and FD respectively and for aerial textures, and 90% and 94% for NE and FD and the Brodatz textures. The rest of the results are shown in Tables 1 and 2. We can also notice that the PCC is larger for FAM-m when noise is present independently of the number of nodes. More specifically, in the presence of noise, the PCC of FAM-m for a specified number of nodes, is always larger than the PCC of FAM for the same number of nodes. We must note that the difference in PCC between FAM and FAM-m is larger for smaller networks. This suggests that FAM-m provides better compressed representation of the training data, and also that FAM-m performs better in the case of small training sets.

6. Conclusions

In this paper we introduced a variation of the testing phase of the FAM that we named FAM-m. We have demonstrated that FAM-m exhibits superior generalization performance than FAM in the classification of signals that are corrupted with noise, independently of the type of noise and the size of the network. Furthermore, we have shown that FAM-m does not require more classification time than FAM. The introduced modification of FAMMN was based on the fact that values of signal features which are distant from feature values that correspond to a pure noise signal, are affected more severely than values of signal features that are close. If the variance of the noise that contaminates the signals is estimated, then the FAM could adapt so that the classification results are further improved. In this paper we have considered classification of textured images which are a special case of 2-D signals.

7. References

[1] Carpenter, G.A., Grossberg, S., Markuzon, N., Reynolds, J.H., and Rosen, D.B. , "Fuzzy ARTMAP: A neural network architecture for incremental supervised learning of analog multidimensional maps", *IEEE Transactions on Neural Networks,* No. 3, pp. 698-713, 1992.

[2] Williamson, J.R., "Gaussian ARTMAP: A neural network for fast incremental learning of noisy multidimensional maps", *Neural Networks*, Vol. 9, No. 5, pp. 881-897, 1996.

[3] Grossberg, S., and Williamson, J., " A self-organizing system for classifying complex images: Natural textures and synthetic aperture radar", *Technical Report CAS/CNS-96-002*, 1996.

[4] Charalampidis, D., Kasparis, T., Rolland, J., "Segmentation of textured images based on multiple fractal feature combinations", *Proceedings of the SPIE, Visual Information Processing VII*, pp. 25-37, 1998.

[5] Garding, J., "Properties of fractal intensity surfaces", *Pattern Recognition Letters*, pp. 319-324, 1988.

[6]Pentland, A.P., "Fractal based description of natural scenes", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 6, No. 6, pp. 661-674, November 1984.
[7]Jain, A.K., and Farrokhinia, F., "Unsupervised texture segmentation using Gabor filters", *Pattern Recognition*, Vol. 24, No 12, pp. 1167-1185, 1991.
[8] Dubuc, B., Roques-Carmes, C., Tricot, C., and Zucker, S.W., "The variation method: a technique to estimate the fractal dimension of surfaces", *SPIE, Visual Communications and Image Processing II*, Vol. 845, pp. 241-248, 1987.
[9] Brodatz P., *Textures: A Photographic Album for Artists and Designers*, New York: Dover, 1966.

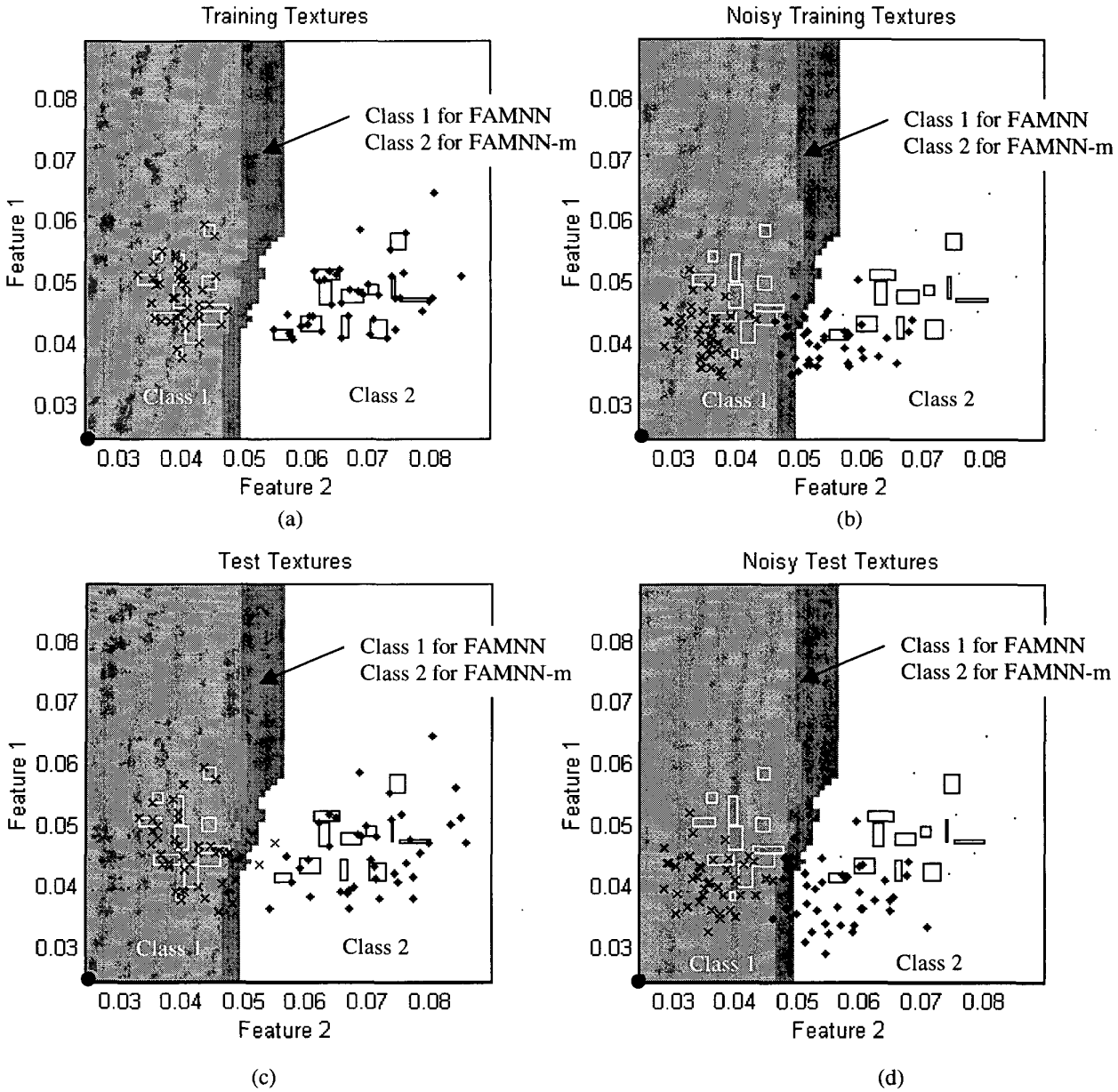

Figure 1: The effect of the modification: (a) Patterns extracted from the training texture set, (b) Patterns extracted from the training texture set when noise is added, (c) Patterns extracted from the "clean" test texture set, (d) Patterns extracted from the test texture set when noise is added.