TWO NONEXCLUSIVE NEURO-FUZZY CLASSIFIERS FOR RECOGNITION OF MUSICAL INSTRUMENTS

G. Costantini[1], F.M. Frattale Mascioli[2], P.Antici[1]

[1]Department of Electronics Engineering, University of Rome "Tor Vergata"
Via di Tor Vergata 110, 00133 Roma, Italy, e-mail: costantini@uniroma2.it
[2]INFO-COM Department, University of Rome "La Sapienza"
Via Eudossiana 18, 00184 Roma, Italy, e-mail: mascioli@infocom.ing.uniroma1.it

ABSTRACT

The classification of single musical sources is an essential step in order to obtain the source separation and the automatic transcription of polyphonic music. In this paper, we present a first experience of recognition of five different musical instruments (clarinet, flute, oboe, saxophone and violin). For such task, a nonexclusive classifier capable of fuzzy decisions is especially suitable, due to the inevitable overlaps among data. We used two different neuro-fuzzy classifier for recognition of musical instruments and we compared the obtained results.

1 INTRODUCTION

In acoustic and musical context, the source separation and recognition problem is very relevant to processing single source sound independently of background and to automatically transcribing polyphonic music. Several attempts in this direction have been recently made [1-3].
As musical instruments can be played in very different ways and sounds of different instruments can have similar characteristics, the classification of single sound sources, by means of their characteristic parameters, is the first step to be made for the source separation task. As characteristic parameter we limit our attention only to the harmonics of the signal. Hence, the success of a recognition method requires the use of appropriate pre-processing algorithms, capable of extracting all the distinctive features of the musical signals, and the application of classification methods operating in a nonexclusive environment. Regarding the musical notes, we mainly analyze the sustain phase of the sound, which is close to a periodic signal for traditional instruments, as those considered in the present case. The pre-processing method is described in section 2. It constitutes a preliminary attempt to modeling the musical signals.
A consequence of the fact that musical instruments can be played in various manners is that the points representing the musical signals in a feature space are contained in regions, which overlap. The correspondence instrument/region is therefore confusing: i.e., there are not sharp boundaries among the regions associated with the instruments. Due to this fact, the use of a nonexclusive neuro-fuzzy classifier can be desirable, thanks to its characteristics of robustness and accuracy, as will be better explained in section 3.
With regard to classification, the nonexclusive context calls for the use of a fuzzy approach. As will be discussed in section 3, the proposed solution is characterized by two steps. Firstly, the signals produced by a single instrument are stored in a sufficient number of exemplars. Then, the amplitudes of their principal harmonics are reported in vectors, which are clustered in a suitable space. In the second step, all the clusters regarding the instruments of interest are inserted into a neuro-fuzzy network, which carries out the classification.

2 MUSICAL SIGNAL PRE-PROCESSING

A first problem to be solved regards the number NS of temporal samples of the signal to be considered in relation to the sampling frequency. If NS is too small, the frequency resolution of the spectral lines is not sufficient to distinguish between two adjacent musical notes.

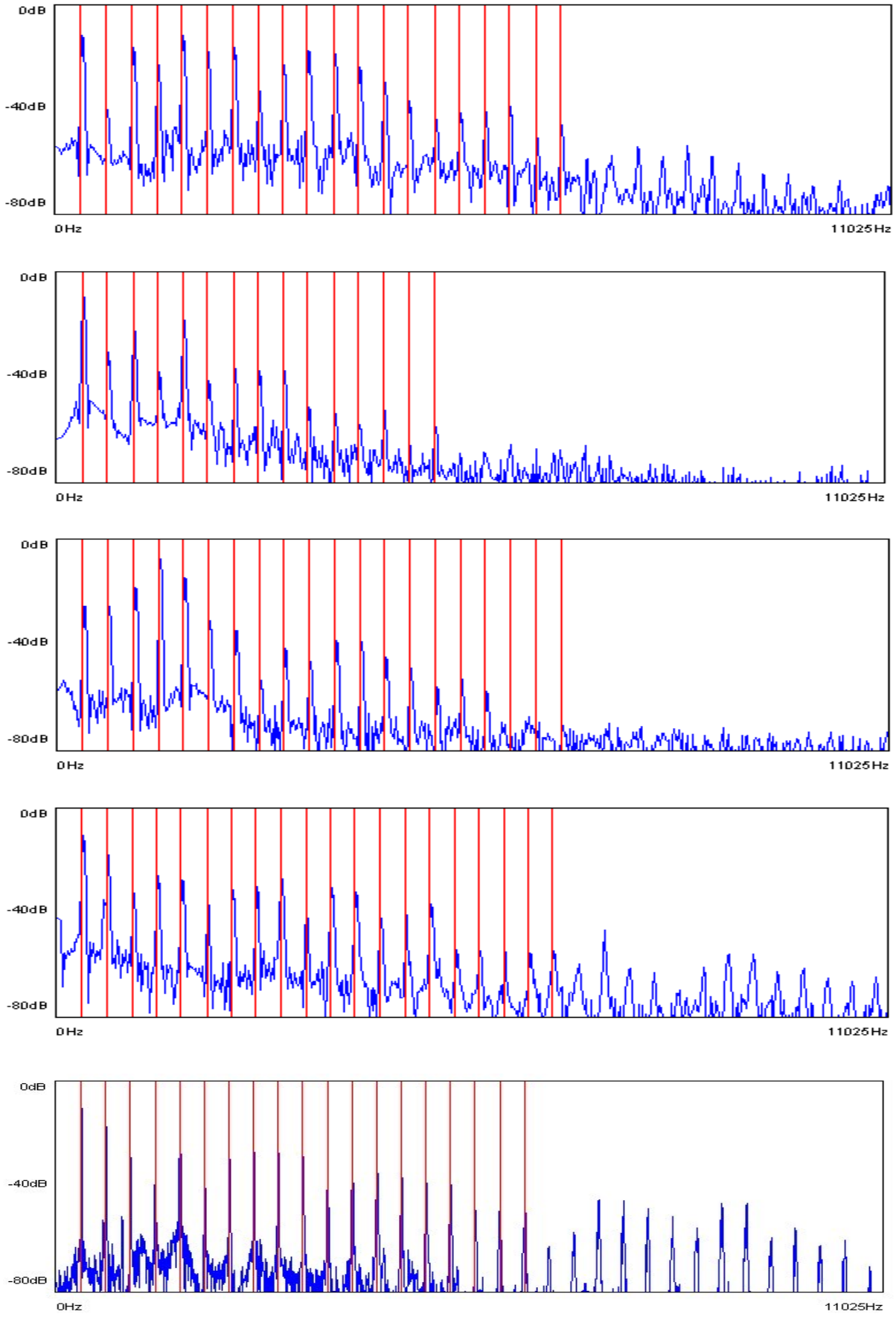

Figure 1. FFT spectra of E4 note for clarinet, flute, oboe, violin and saxophone.

As an example, in the case of commonly used sampling frequency of 44.1 kHz, a value NS=512 yields a frequency resolution equal to 86 Hz, while the frequency difference between the musical note A4 at 440 Hz and the adjacent A4# is only 26.2 Hz. On the other hand, the number of FFT spectral lines with NS=512 is

equal to 256. They are too many in relation to the available information and to the necessity of a reasonable computational burden. The previous dilemma is solved by using a value of NS sufficiently large for guaranteeing the required frequency resolution and by retaining only the principal spectral lines in a number M chosen independently of NS. As we considered the notes C4 to C5 (261 Hz – 522 Hz) we used NS=4096 and M=20.
The determination of the harmonics is undertaken as proposed in [4]. A line of the spectrum with frequency f_i is retained if its amplitude L_i, expressed in dB, satisfies the following relations with respect to the neighboring lines:

$$L_i > L_{i-1} \quad (1)$$
$$L_i \geq L_{i+1} \quad (2)$$
$$L_i - L_{i+j} \geq 7 \text{ dB}, \qquad \text{for } j = \pm 2, \pm 3 \quad (3)$$

The frequency f_c of the corresponding harmonic is:

$$f_c = f_i + 0.46(L_{i+1} - L_{i-1}) \quad (4)$$

The previous procedure is empirical. In particular, the default value of 7 dB in (3) could be replaced by a suitable value in the range 3÷10 dB in order to obtain a better tuning. Not always 20 harmonics are revealed, sometimes their amplitude is so small that they can not be distinguished from the noise. This harmonics has to be neglected because they do not add any information for the network. Considered the flute note in fig.1, only 15 harmonics are revealed.
As said in the introduction, the feature vectors associated with the musical signals, obtained by the previous pre-processing procedure, are not clearly differentiated. This is a consequence of the similarity of the sounds produced by the instruments. In order to clarify this point, we show in Fig. 1 the spectra of the same musical note (E4) produced by five traditional musical instruments: one stringed instrument (violin) and four wind instrument (flute, clarinet, oboe, saxophone). They will be considered successively in the experimental test described in section 4. The spectra are obtained by using 2048 samples, sampling frequency equal to 22050 Hz. A resolution of 4096 samples and sampling frequency of 44100 gives similar spectra. The vertical lines appearing in the figure indicate the harmonics determined by the pre-processing procedure application.
The examination of the five spectra evidences that:

- the spectrum of the oboe is different from the other three, since its initial harmonics are not the most important;
- in the case of clarinet, the second harmonic is nearly absent;
- for the flute, only the first harmonics are prominent;
- the spectra of violin and clarinet are very close each other.

D.Luce and M.Clark demonstrated [5], which these characteristics also hold in the case of other musical notes of the same instrument. Hence the frequency spectrum of a given instrument is independent from the note played (and thus from the fundamental frequency). However the spectrum is subject to large variations depending on the specific musician technique of playing.

3 NONEXCLUSIVE CLASSIFICATION

A nonexclusive approach to classification is hotly required in those problems in which the classes are heavily overlapping each other, as in the case of musical instruments recognition here presented. In fact, while an exclusive classification system (like the classical perceptron) attempts to eliminate overlaps among classes by means of critical boundaries, the nonexclusive approach considers overlapping regions containing a non negligible information about the problem domain. From this point of view, fuzzy logic seems a natural tool for nonexclusive classification, because a pattern can be assigned a degree of membership to each class in a partition. A way to solve a nonexclusive *k*-class problem can be based on the co-operation of *k* independent fuzzy clustering systems, one per each class, as presented in [6]. More precisely, the overall supervised problem is treated as *k* disjointed unsupervised ones, in order to preserve the information content of the overlapping regions. The nonexclusive classification strategy utilized in this work is characterized by the absence of critical a-priori choices and a reasonable computational load. The resulting network is mainly

controlled by a single parameter that defines number and extension of clusters, automatically tuned on the basis of a relative index of cluster validity.

3.1 Hyperbox based clustering procedure

Aim of the application of a clustering procedure on a single class is that of locating and sizing proper fuzzy membership functions into the input space, in relation to the distribution of the current data. For this goal, it is required a fuzzy clustering technique that makes no a-priori assumption on the number of clusters to be used, i.e.: a constructive technique. In [6] was proposed a clustering algorithm inspired by Simpson's Min-Max [7], chosen for its constructive nature, robustness and low computational cost. In the following, we remind briefly the main characteristics of the clustering procedure.
During the clusters individuation phase, hyperboxes parallel to the co-ordinate axes (hyper-parallelepipeds) are created and expanded in order to contain each class pattern. A hyperbox is completely defined by two extreme vertices: the 'min' and 'max' vertices. For each pattern in the training set, already existing hyperboxes are considered to be expanded in order to include it. The expansion process is constrained by the maximum hyperbox size parameter θ. Small values of θ yield a large number of hyperboxes, and vice versa. If no existing hyperbox can be expanded, then a new one is created (its min and max points coincide with current pattern). We remark that the membership functions utilized by Simpson do not fit the nonexclusive classification task (in fact, they have a "plateau" that corresponds to an output fixed to one). Consequently, at the end of pattern examination each hyperbox is substituted by a properly sized hyper-ellipsoid. The corresponding membership functions are "generalized bells" [8], whose parameters are determined on the basis of hyperbox extension and number of patterns inside it.
The clustering quality depends critically on the user-defined parameter θ. In order to determine automatically an optimal value of θ, a strategy is adopted based on a cluster validity index, measuring compactness and separation of the clusters. In particular, a modified version of Davies-Bouldin relative index [9] is chosen, but also other relative indices (both crisp and fuzzy [10]) can be applied for this purpose. The optimal partition is obtained in correspondence of the minimum value of the index. Nevertheless, it must be pointed out that relative indices have in general not a regular trend; rather, they are affected by several local minima. Therefore, in order to find the optimal value θ_{opt} without being trapped in local minima of the index, a proper sampling of the selected interval of θ is necessary. The value θ_{opt} which minimizes the index is determined by considering all the sampled values.

3.2 PCA based clustering procedure

The classical approach of probability driven clustering is to model the natural cluster made with the training set using a mixture of random data sources: the training set $X = \{\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_N\}$ is constituted by a group of N patterns, every pattern being a random variable extracted from one (and only one) of the K "sources" that make up the mixture. Every "source" represents a cluster which is characterized by a specific probability density $p(x|i;\theta_i)$, where θ_i is the vector of the unknown parameters that define the probability and is characterized by the probability a priori π of the cluster i, where $\pi_i \geq 0$ and $\sum_{i=1}^{K} \pi_i = 1$. The complete model of the mixture consists in a probability density, which is made of the weighed sum of the single probabilities π_i:

$$p(\mathbf{x}\ ;\boldsymbol{\theta}) = \sum_{i=1}^{K} \pi_i\, p(\mathbf{x} \mid i\ ;\boldsymbol{\theta}_i) \qquad (3.1)$$

The problem of clustering would be assigning every pattern to his correct cluster in a crisp way, as every pattern is generated by one and only one cluster. If through the patterns at disposition it is possible to estimate the whole group $\boldsymbol{\theta} = \{\boldsymbol{\theta}_1, \boldsymbol{\theta}_2, \ldots, \boldsymbol{\theta}_K; \pi_1, \pi_2, \ldots, \pi_K\}$ of parameter of the model, the assignment may be made considering the probability density that, in this case, would be known. In fact, using the Bayes's rule [11] it is possible to obtain the probability a posteriori R_{ni} that, given a pattern $\mathbf{x}_n$, this pattern is generated by an aleatory source, the cluster i:

$$R_{ni} = \pi(i \mid \mathbf{x}_n) = \frac{\pi_i p(\mathbf{x}_n \mid i)}{p(\mathbf{x}_n)} \qquad (3.2).$$

It is obvious that one can assign the pattern $\mathbf{x}_n$ to a cluster maximizing the probability, that is

$$\pi(h \mid \mathbf{x}_n) = \max_{i=1,\ldots,K} R_{ni} \qquad (3.3).$$

In this way we have formalized the problem of crisp clustering using a statistical approach of estimation of the parameters $\boldsymbol{\theta}$.
However, considering the probabilities a posteriori of a group of independent, identical distributed variables $X = \{\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_N\}$ extracted from the mixture (3.1) we have formalized also the problem of probabilistic clustering where we associate to every pattern a label $[R_{n1}\ R_{n2}\ \ldots\ R_{nK}]$, the sum of the elements being unitary; hence the procedure of "decision" expressed by the (3.3) constitute the phase of deprobabilization**.**
There are several solving methods that allows to estimate the parameters $\boldsymbol{\theta} = \{\boldsymbol{\theta}_1, \boldsymbol{\theta}_2, \ldots, \boldsymbol{\theta}_K; \pi_1, \pi_2, \ldots, \pi_K\}$ for resolving univocally this kind of clustering problems. We used one of the most established methods, the method of Maximum Likelihood Estimation, MLE), proposed in 1973 by Duda and Hart [12]. Consequently a precise but unknown mixture characterizes the pattern space and observing the data-set of independent, identical distributed variables $X = \{\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_N\}$ we need to determine the constant and variable parameters θ.
For the experiments we used a nonexclusive PCA-based classifier. The PCA (Principal Components Algorithm) works as follows: the classifier firstly analyses the data set and determines a direction where the variance and consequently also the information are maximized. Secondly it chooses, between the remaining components, a second direction, orthogonal to the directions already determined, that has maximum variance. This procedure is iterated up to the maximum subspace dimension, parameter that can be set during the training phase. Our experiments were made with subspace dimension 5, 10 and 20. Thus we reduce the original subspace dimension N (equal to the number of harmonics we give as input) to a dimension M which is smaller. Moreover, we guarantee that during the mapping from an N dimensional to an M dimensional space we loose the less information possible. As probability density the classifier uses the gaussian density. Summarizing the classifier models the training set with a mixture of gaussian curves.

3.3 Classifier architecture

The application of the previously described clustering procedure on each class of the training set leads to have *k* optimised partitions. In Fig. 2 it is shown the resulting three-layer network.
Each neuron in the hidden layer is associated with a fuzzy cluster and yields a membership value according with it. These neurons are grouped in *k* sets (dashed rectangles in figure), each set being the result of a clustering procedure. Neurons of third layer are in number of *k*, one per class. Each of them computes the membership value of the fuzzy union among clusters of its own class. After the normalizing block (labeled in figure with "N"), pure fuzzy outputs [13] are available. If a hard decision is needed, the latter block (labeled "WTA" in figure) performs the Winner Takes All defuzzification method, and gives as output the label of the class having maximum membership value among all. Weights of connections between second and third layer can be tuned by a supervised procedure in order to minimize overall crisp error on the whole training set.

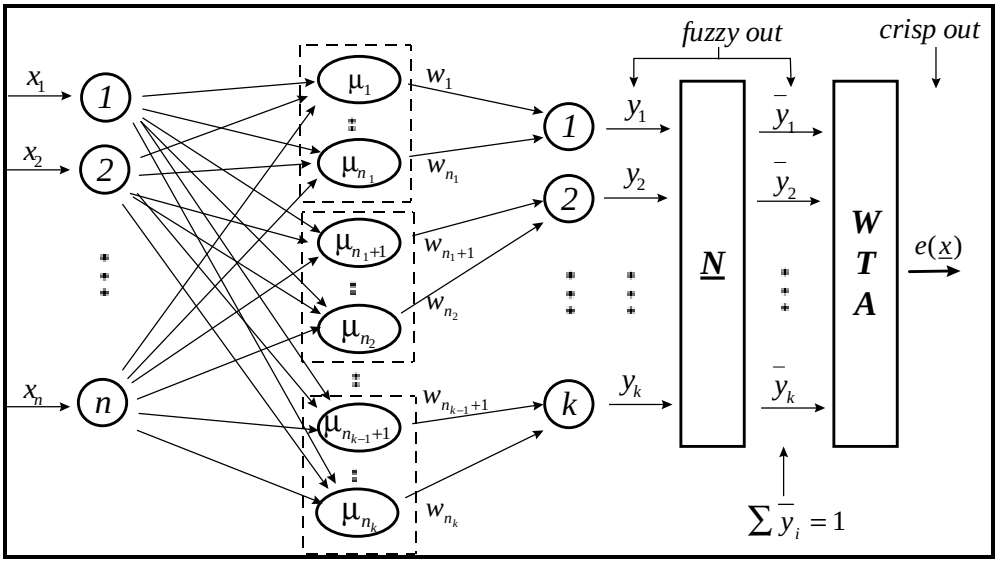

Figure 2. Nonexclusive neuro-fuzzy classifier architecture.

4 RESULTS OF RECOGNITION EXPERIMENT AND CONCLUSIONS

A recognition experiment was carried out for evaluating the performances of a neuro-fuzzy classifier based on PCA and another neuro-fuzzy classifier based on hyperboxes.

The training set is constituted by 390 patterns, 78 for each instrument (Clarinet, Flute, Oboe, Saxophone and Violin). Two experiments were made, the first considering for each pattern a vector of 20 components, the second considering a vector of 10 components. The components correspond to the amplitude of the first harmonics, obtained with the procedure described in section 2. The considered spectra are determined by applying a FFT to a sequence of 4096 samples with a sampling frequency of 44100 Hz. For each instrument, 6 segments of each of the 13 musical notes included in the range from c4 to c5 are considered. The number of bits for A/D conversion is 16.

The test set is constituted by 80 pattern, with the same characteristics of the training set, concerning musical notes of the 4th octave (261 Hz – 522 Hz) randomly extracted from various musical executions.

The classifier based on PCA allows establishing the maximal subspace dimension of the network. In this case, the classifier determines the components which are more characteristic for the given instrument and generates consequently the network. Thus, choosing a small subspace dimension, the computational burden of the network is reduced during the validation of the test set.

Experiments were made with a subspace dimension 20, 10 and 5 and the performances of the two classifiers are compared in Tab. 1.

The examination of Tab.1 evidences the superiority of the PCA based classifier in terms of generalization error and structural complexity.

Firstly, the PCA classifier generates fewer clusters and obtains, nevertheless, better results as for the training set, as for the test set. Secondly, the subspace dimension does not influence notably the error rate. Moreover, even using only 10 components for the pattern – that is to say the first 10 harmonics – the error rate does not increase considerably (about 2.5 %).

Consequently, the first 10 harmonics contain great part of the essential information the network needs for classifying the different musical instruments. In the contrary, using a subspace dimension of 5 but considering the first 10 harmonics the classifier produces an error of about 9 %. One can deduce that the information the network needs for classifying the different musical instruments is mostly contained in 5 components, which are not the first 5 harmonics but are chosen individually by the classifier. In fact, it should be noted that choosing a subspace dimension of 5 and using the first 5 harmonics the error increases strongly, to more than 20 % for the validation and to 30 % for the test set.

The fuzzy outputs available in the proposed classifier allow investigating the nature of the recognition errors, due to the overlap of the input space regions corresponding to the 5 instruments. In particular, the majority of the errors concern the confusion between clarinet, saxophone and violin. This result is in agreement with the remark that their spectra are very similar.

Classifier	Index	N. of components	Subspace dimension	Cluster generated	Absolute Crisp error (training set)	Crisp error in % (training set)	Absolute Crisp error (test set)	Crisp error in % (test set)
Hyperbox based	DW-Davies	20		25	59	15.36	24	30
		10		35	42	10.93	25	31.25
	Fuzzy comp	20		12	52	13.54	13	16.25
		10		12	88	22.65	21	26.25
PCA based	Life Time	20	20	5	29	7.55	8	10
		20	10	5	32	8.33	8	10
		20	5	5	42	10.94	10	12.5
		10	10	5	23	5.99	10	12.5
		10	5	5	34	8.85	12	15
		5	5	7	92	23.96	24	30

Table 1. Recognition results.

ACKNOWLEDGMENTS

The authors wish to thank A. Bellisario and D. Baldelli for experimental co-operation.

REFERENCES

[1] M. Marolt, “Feedforward Neural Networks for Piano Music Transcription”, *Proc. of XII Colloquium on Musical Informatics*, Gorizia (Italy), pp. 240-243, Sept. 1998.

[2] K.D. Martin, “A Blackboard System for Automatic Transcription of Simple Polyphonic Music”, MIT Media Laboratory, Perceptual Computing Section, Technical Report No. 385, 1996.

[3] D. Nunn, A. Purvis, and P. Manning, “Source Separation and Transcription of Polyphonic Music”, http://capella.dur.ac.uk/doug/icnmr.html, 1997.

[4] E. Terhardt, G. Stoll, and M. Seewann, “Algorithm for extraction of pitch and pitch salience from complex tonal signals”, *J. Acoust. Soc. Am.*, **71**(3), March 1982.

[5] D.Luce and M.Clark, Jr., “Physical Correlates of Brass-Instrument Tones”, *J. Acoust. Soc. Am.*, **42** (6), March 1967 pp. 1232-1243.

[6] F.M. Frattale Mascioli, G. Risi, A. Rizzi, and G. Martinelli, “A Nonexclusive Classification System Based on Co-operative Fuzzy Clustering”, *Proc. of EUSIPCO’98*, Rhodes (Greece), pp. 395-398, 8-11 Sept. 1998.

[7] P.K. Simpson, “Fuzzy Min-Max Neural Networks - Part 2: Clustering”, *IEEE Trans. on Fuzzy Systems*, **1**(1), pp. 32-45, 1993.

[8] J.S.-R. Jang and C.T. Sun, “Neuro-Fuzzy Modeling and Control”, *Proc. of the IEEE*, **83**(3), pp. 378-406, 1995.

[9] D.L. Davies and D.W. Bouldin, “A Cluster Separation Measure”, *IEEE Trans. on Pattern Analysis and Machine Intelligence*, **1**, pp. 224-227, 1979.

[10] J.C. Bezdek, W.Q. Li, Y. Attikiouzel, and M. Windham, "A Geometric Approach to Cluster Validity for Normal Mixtures”, *Soft Computing*, 1(4), pp. 166-179, 1997.

[11] A. Papoulis, "Probability, Random Variables, and Stochastic Processes", *McGraw-Hill, Inc., International Edition*, 1991

[12] R. O. Duda, P. E. Hart, *"Pattern Classification and Scene Analysis", John Wiley & Sons, New York.*, Sect. 3.2, 1993

[13] J.C. Bezdek, "A Review of Probabilistic, Fuzzy, and Neural Models for Pattern Recognition", *Journal of Intelligent and Fuzzy Systems*, **1**, pp. 1-25, 1993.

[14] P.K. Simpson, "Fuzzy Min-Max Neural Networks - Part 1: Classification", *IEEE Trans. on Neural Networks*, **3**(5), pp. 776-786, 1992.

[15] F.M. Frattale Mascioli and G. Martinelli, "A Constructive Approach to Neuro-Fuzzy Networks*",* *Signal Processing*, **64**(3), pp. 347-358, 1998.