FUZZY CLUSTERING APPROACH USING DATA FUSION THEORY AND ITS APPLICATION TO AUTOMATIC ISOLATED WORD RECOGNITION

B. Moshiri

Department of Electrical and Computer Engineering, Faculty of Engineering
Control & Intelligent Processing, Center of Excellence, ECE Department, University of Tehran,
Tehran, Iran, moshiri@ut.ac.ir

P. Eslambolchi

Department of Electrical and Computer Engineering, Faculty of Engineering, University of Tehran
Tehran, Iran, eslambol_par@engineer.com

R. HoseinNezhad

School of Engineering and Science, Swinburne University of Technology
Victoria 3122, Australia, rhoseinnezhad@swin.edu.au

(Received: August 25, 2002 – Accepted in Revised Form: November 5, 2003)

Abstract In this paper, utilization of clustering algorithms for data fusion in decision level is proposed. The results of automatic isolated word recognition, which are derived from speech spectrograph and Linear Predictive Coding (LPC) analysis, are combined with each other by using fuzzy clustering algorithms, especially fuzzy k-means and fuzzy vector quantization. Experimental results show that the proposed algorithms have better performance, compared to classical clustering.

Key Word Data Fusion Theory, K-Means Clustering, Fuzzy K-Means, Fuzzy Vector Quantization

چکیده در این مقاله، کاربرد الگوریتمهای خوشه‌بندی برای ترکیب اطلاعات در سطح تصمیم‌گیری ارایه می‌شود. نتایج تشخیص خودکار کلمات مجزا، که از روشهای مختلف (مانند اسپکتروگراف گفتار و الگوهای زمانی گفتار) بدست می‌آیند ، با استفاده از الگوریتمهای خوشه‌بندی فازی، بخصوص k-means فازی و چندی کننده برداری فازی با یکدیگر ترکیب می‌شوند. نتایج پیاده سازی نشان می‌دهد الگوریتمهای ارایه شده کارایی بهتری در مقایسه با روشهای کلاسیک خوشه‌بندی دارد.

1. INTRODUCTION

Data fusion theory has been widely applied in object identification problems during the recent two decades [1-3]. In these applications a decision is made about an object, based on the information about it, which are provided by multiple sensors or sources.

The problem of processing and combination of the information that are provided by different knowledge sources is usually referred as *multisensor data fusion*. Decision level fusion takes place when the output of the fusion process is a decision e.g. about the class of an object in an object identification application. Examples of decision level fusion methods include weighted decision methods, classical inference, Bayesian inference and Dempster-Shafer method [4,5].

In this paper, utilization of fuzzy clustering algorithms for decision level fusion is proposed. Classical clustering methods refer to a wide variety of methods that attempt to subdivide a data set into subsets (clusters). Fuzzy clustering algorithms, such as the *fuzzy k-means* (FKM) and the *fuzzy vector quantization* (FVQ) [6,7], consider each cluster as a fuzzy set, while a membership function measures the possibility that each training vector belongs to a cluster.

The FKM and FVQ methods are usually applied to combine results which are deduced from various single speech recognition algorithms, e.g. from speech spectrograph and time domain features (spectral coefficients) [8,9]. The results that are provided by these methods are accompanied with a degree of quality. The quality measure is then applied for data fuzzification.

There are various methods for word recognition e.g. Hidden Markov Model [10], Neural Networks [11,12] and Hybrid HMM-NN [13,14] methods. Basically in voice processing, a single voice model scarcely models the voice perfectly. Even if there were such a perfect voice model, it would not be useful due to its complexity.

In order to overcome the baffle, it is usually recommended to exploit some different models and estimate the result with data fusion methods to gain the benefits of each speech recognition method.

This paper has the following structure. In Section 2 the word recognition problem is described. In Section 3 the clustering methods are explained. Experimental results are represented in Section 4 and conclusions are drawn in Section 5.

2. ISOLATED WORD RECOGNITION

The complex problem of real-time speech recognition has been widely studied in recent decades. Even commercial products have already been appeared with different languages, dictionary sizes, platforms, etc. For example Natural Speech Communication (nsc.co.il) on DSP-based PCI boards and rank mounted boxes. Also Rubidium Ltd.'s Dialog Engine (www.rubidium.com) supports speech recognition, TTS, and dialog management. It is available as a System On a Chip or a software only solution. As another example, Advanced Recognition Technologies, Inc. (www.artcomp.com) develops handwriting and speech recognition solutions that are appropriate for embedding in consumer devices such as cell phones and toys. Despite, cost-effective and fast algorithms are still current subjects of research.

This paper focuses only on one part of speech recognition systems, which is “word identification”. Pre-processing and pattern recognition modules are the major two components of the proposed method for word identification. They are described in the following subsections.

2.1. Data Sets and Preprocessing

The audio data, which are used in the experiments, are 10 isolated words each of which have been pronounced 50 times by a person in different conditions and daytimes. The words are Persian numbers from 1 to 10. There are 500 voice samples, which 300 of them are used for training and the rest for test. The voice samples have been recorded in a noisy environment but for the sake of acoustic noise reduction, headset has been used. Voice was sampled at 8000 Hz with 16 bits per sample.

It is required to perform some preprocessing on the recording voice for feature extraction. This function is performed by one of the components of the speech recognizer that is called pre-processor. In this component, the silent detection [15] and spectral subtraction [16] are applied to reduce the additive noise. Then pre-emphasis and hamming window filters are applied. After these processes, the speech spectrum can be extracted.

The produced spectrogram is fed into a filter to choose some important frequencies according to the importance of frequency selection for comparison. They must be carefully selected. In [9] it is shown that dominant part of the spectrum has concentrated in the frequency intervals (200,800) and (1600,2000) and the rest has less importance. Hence we have used more frequency points or filters in these regions and less filters in the less important regions. In this research work, 50 filters have been used for evaluation of the voice spectrum.

2.2. Spectral Analysis and Specification

The speech features, i.e. spectrogram and LPC (Linear Predictive Coding) coefficients must be extracted after preprocessing and signal preparation for comparison purposes. In order to modify the recorded voice so as to be comparable with the trained data, each recorded voice must be standardized. Therefore, after start and end point detection, we divide the recorded voice samples into 50 sections and add zeroes or delete the samples till reaching 250 samples in each section. For this purpose we add zero samples to the voice or delete samples monotonically [9].

As it has been said before, we have used 50 filters to produce the spectrograph of the voice spectrum. We measure FFT amplitudes at 50 predefined frequencies for the 250 samples in each of the 50 sections and enter them as the columns of a matrix, called Sp. Thus the Sp matrix is filled with the voice spectrograph and can be used for comparing with the saved or trained data or can be used in training phase.

In training phase, Sp matrix is produced for N=50 voice sections, each containing 250 samples, then according to Equations 1 and 2 the Mean and Variance matrices are determined respectively and saved for comparison in recognition phase.

$$\text{Mean} = \frac{\sum_{k=1}^{N} sp_k}{N} \qquad (1)$$

$$\text{Var} = \sqrt{\frac{\sum_{k=1}^{N} (sp_k - \text{Mean})^2}{N^2}} \qquad (2)$$

In recognition phase the produced data Sp matrix must be compared with trained data, which have been saved in the form of Mean and Variance matrices.

A Gaussian similarly function performs this comparison as a classifier. Similar membership function is also defined in such a way that it can evaluate the similarity of a point (i,j) in Sp Matrix with corresponding point in trained data. One of the appropriate possible definitions for such a function is expressed as follows:

$$f_{i,j,k} = \frac{\exp\left[-\frac{1}{2}\left(\frac{(M_R(i) - M_T(j,k))^2}{K_M^2} + \frac{(V_R(i) - V_T(j,k))^2}{K_V^2}\right)\right]}{2\pi\, K_M\, K_V}$$

where $M_R(i)$ and $V_R(i)$ are the mean and the variance of the i^{th} frequency components (from the 50 frequencies) of the recorded samples, respectively. Similarly $M_T(j,k)$ and $V_T(j,k)$ are the mean and variance of the j^{th} frequency components of the k^{th} trained voice samples, respectively. K_M and K_V are parameters that their appropriate values are chosen by trial and error.

This similarity is evaluated for each of the 50 sections in each of the 50 frequencies. Thus, 2500 values are resulted. Equation 3 determines the total similarity. S_k is the similarity of the recorded voice with the k^{th} trained voice by the spectrogram method.

$$S_k = \sum_{i=1}^{50}\sum_{j=1}^{50} f_{i,j,k} \qquad (2)$$

LPC (Linear Predictive Coding) time domain analysis is applied to the voice after pre-emphasis, hamming windowing and autocorrelation. LPC coefficients are determined by Durbin-Levinson method and then autocorrelation coefficients will be converted to cepstral coefficients by LPC analysis. According to the higher accuracy and more robustness of the cepstral coefficients with respect to the LPC coefficients, we adopted cepstral coefficients. In cepstral analysis 12 coefficients have been used.

After determining the cepstral coefficients, a reduction filter weighs the coefficients and their derivatives [8,9]. Cp is defined as the matrix that consists of the cepstral and their derivative coefficients in different time durations. Similar to the previous section, there are training and recognition phases. In training phase Cp matrix is produced for N = 50 voice sections then according to Equations 1 and 2 the Mean and Variance matrices are determined and saved to be compared in the next phase. In recognition phase the saved Cp matrix is compared to the previously saved matrix, which is associated with the trained data. As a result of this comparison, T_k values are obtained as below:

$$T_k = \sum_{i=1}^{n}\sum_{j=1}^{50} g_{i,j,k} \qquad (4)$$

where n is the number of coefficients in Cp and

$g_{i,j,k}$ is a similarly measure that is defined with the same formulation of $f_{i,j,k}$ for evaluation of the similarity of the ith cepstral coefficient of the saved matrix with respect to the jth cepstral coefficient of the kth trained voice sample.

Actually each T_k value is represented as the similarity of the recorded voice sample with the kth trained voice sample by spectrogram method.

3. FUZZY CLUSTERING

The well known classical k-means algorithms classifies each training vector in such a way that distance measure value is minimized. A set of M training vectors is clustered into a set of k codebook vectors by the following steps:

1. *Initialization*: k vectors are arbitrarily chosen as the initial set of code words in the codebook.
2. *Nearest-Neighbor Search*: For each training vector, find the closest code word in the current codebook (in terms of spectral distance) and assign that vector to the corresponding cell (associated with the closest code word).
3. *Centroid Update*: Update the code word in each cell using the centroid of the training vectors assigned to that cell.
4. *Iteration*: Repeat steps 2 and 3 until the average distance falls below the preset threshold.

The performance of the algorithm strongly depends on the initialization of the codebook vectors. Since the codebook is designed, any data is classified into a cluster based on a classical distance criterion.

These traditional clustering approaches generate partitions and every pattern is associated with one and only one cluster. Hence, the clusters are disjoint in such a hard clustering approach. Fuzzy clustering extends this notation to associate each pattern with every cluster using a membership function. The output of such an algorithm is a clustering but not a partition. The fuzzy k-means algorithm (FKM) classifies each vector to all clusters with different values of membership in [0, 1]. This membership value indicates association of a vector with each of the *k* clusters. Notice that the fuzzy k-means algorithm does not classify fuzzy data, but crisp data into fuzzy clusters [17,18]. The algorithm is derived from the constrained minimization of the following objective function:

$$J_m = \sum_{j=1}^{k} \sum_{i=1}^{M} u_j(x_i)^m \left\| x_i - y_j \right\|^2 \tag{5}$$

$1 < m < \infty$

where x_i is a training vector, y_j is a codebook vector and is considered as cluster center and $u_j(x_i)$ is the membership function of the jth cluster and is defined as follow:

$$u_j(x_i) \in [0,1] \qquad \forall i, j$$

$$\sum_{j=1}^{k} u_j(x_i) = 1 \tag{6-1}$$

$$u_j(x_i) = \frac{1}{\sum_{\ell=1}^{k} \left(\frac{\left\| x_i - y_j \right\|^2}{\left\| x_i - y_\ell \right\|^2} \right)^{\frac{1}{m-1}}} \tag{6-2}$$

The parameter m controls the fuzziness of clustering procedure and is always greater than one. When m tends to one, the clustering tends to the crisp clustering approach provided by the classical k-means algorithm. When a vector x, is an outliner (i.e. it is far from all cluster centers) their membership functions take very small values and that vector does not practically modify the cluster centers [19].

Fuzzy Vector Quantization (FVQ) is a soft decision making algorithm. In its initialization level, each training vector can be assigned to be a codebook vector, being concentrated at a cluster center. Each vector x_i is likely to belong to the jth cluster with a measure of $v_j(x_i)$. This function is defined in such a way that it is equal to one when $\left\| x_i - y_j \right\|^2$ is zero and it becomes zero when $\left\| x_i - y_j \right\|^2$ is $d_{max}(x_i)$ or more. One appropriate definition is given as below:

If $\left\| x_i - y_j \right\|^2 \le d_{max}(x_i)$ then:

$$v_j(x_i) = \left(1 - \frac{\left\| x_i - y_j \right\|^2}{d_{max}(x_i)} \right)^{\mu} \tag{6}$$

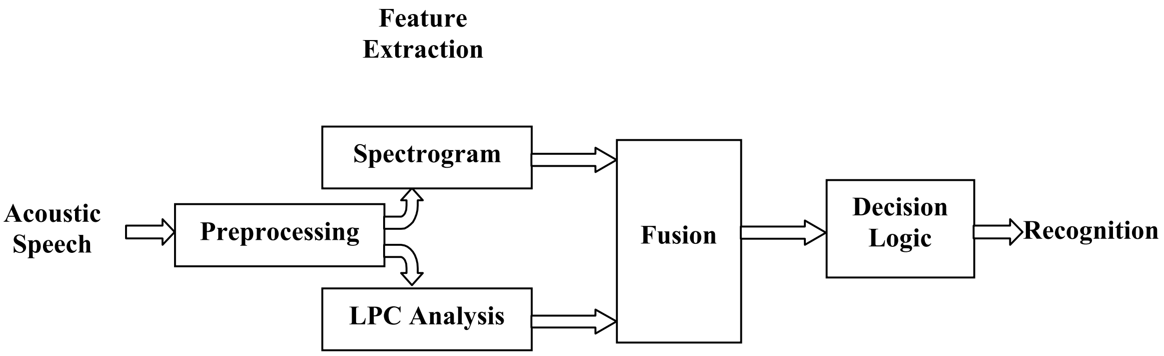

Figure 1. Structure of the proposed word recognition method.

otherwise:

$$v_j(x_i) = 0$$

where μ is a positive integer. Each of the training vectors is assigned to one cluster. Notice that, similar to the FKM algorithm, the FVQ algorithm does not classify fuzzy data [5,17]. The definition difference between the two functions $v_j(x_i)$ and $u_j(x_i)$ arises from the fact that in FVQ, for the sake of quantization purposes, $v_j(x_i)$ becomes zero when x_i and y_j are far enough apart from each other. Meanwhile $v_j(x_i)$ is not required to satisfy the condition:

$$\sum_{j=1}^{k} v_j(x_i) = 1$$

because, there is no need for it to behave like a fuzzy membership function.

The advantage of fuzzy vector quantization versus fuzzy k-means is elimination of the effect of the initial codebook selection on the quality of the clustering and avoiding the priori assumptions about the level of fuzziness required for a clustering task. Similarly to the fuzzy k-means algorithm, the fuzzy vector quantization algorithm does not classify fuzzy data. Detailed steps of FVQ algorithm comes in the following section:

1. Select an initial fuzzy partition of the *N* objects into K clusters by selecting the N*K membership matrix V, an element $v_j(x_i)$ of this matrix represents the likelihood of membership of an object x_i in the j^{th} cluster.
2. Using V, find the value of a fuzzy criterion function, e.g., a weighted squared error criterion function, associated with the corresponding partition.
3. Repeat step 2 until entries in V do not change significantly.

4. FUSION PROCESS

In order to explain the data fusion in a sense, firstly the voice samples are processed by two methods: Spectrogram and LPC analysis. As the result, 10 values for S_k and 10 values for T_k are derived. These 20 values include some uncertainty, which makes them inappropriate to be directly applied for decision-making. Therefore, we suggest that they are fused together and fed into one of the clustering methods (Classical K-Means, Fuzzy K-Means or Fuzzy Vector Quantization). Indeed, the clustering method is interpreted as a fusion method in decision level, which results uncertainty reduction like any other fusion method. Actually the x_i vectors in each of these methods contain S_k and T_k values. In a sense, these two sets of values are fused together by the clustering method. The result of the clustering process will be a decision for the voice sample, about the cluster (phoneme) that it belongs to. The whole structure of the proposed word identification method is depicted in Figure 1.

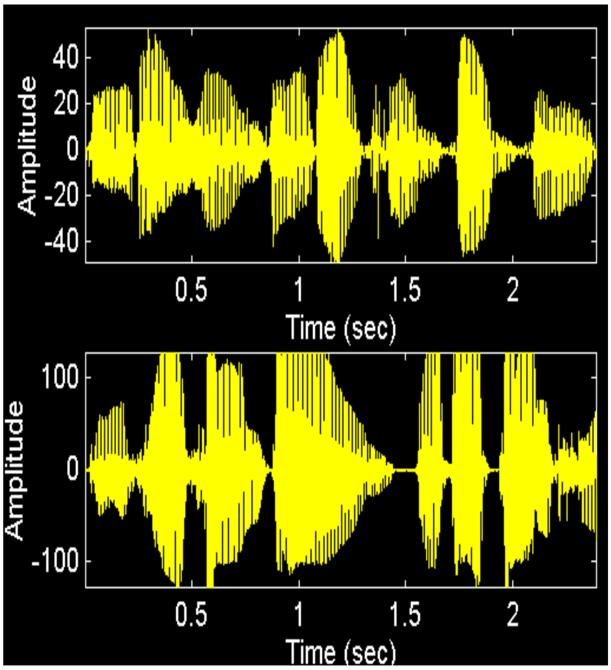

Figure 2. Sampled voice data, pronounced by the two speakers in one of their trials.

5. EXPERIMENTAL RESULTS

Speech data, which are used for this experiment, are 10 isolated words each pronounced 50 times by a person in different conditions and daytimes. The words are Persian numbers from 1 to 10. We have two female speakers and collect 500 voice samples for each speaker, which 300 of the 500 samples of one of the speakers were randomly selected and used for training and the rest for test. The voice samples have been recorded in a noisy environment but for the sake of acoustic noise reduction, headset has been used. The recording format was 16 bit with 8 KHz sampling rate. Figure 2 shows the recorded voice samples of the two speakers in one of their trials.

Both spectrogram and LPC analysis were carried out on all of the 1000 voice samples that were generated in 1000 pronouncing trials (both on the training and the test samples). The results are presented in Figures 3 and 4, for one of the trials for the two speakers.

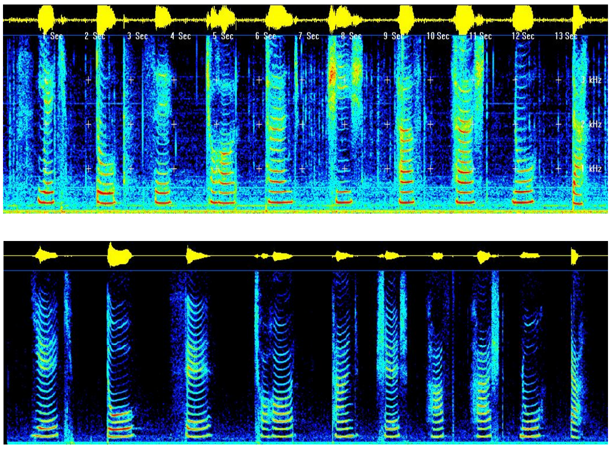

Figure 3. Spectrogram of the voice samples in one of the trials for the two speakers.

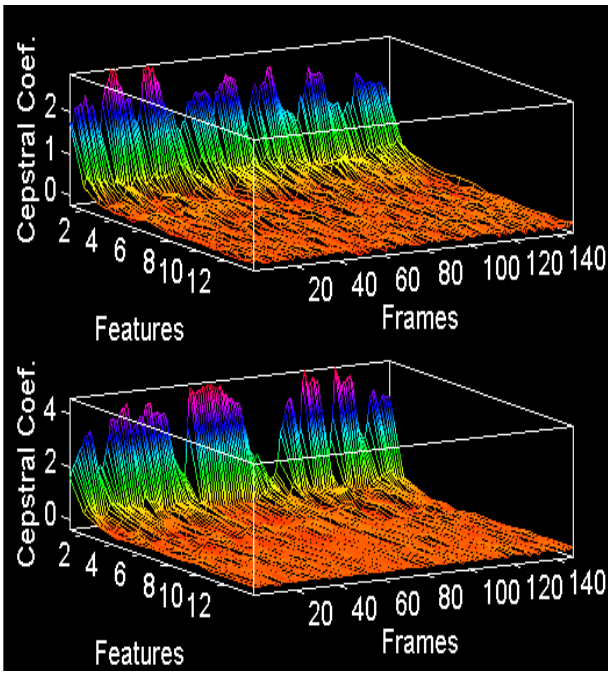

Figure 4. LPC analysis results as the cepstral coefficients in one of the trials for the two speakers.

The three algorithms (Classical K-Means, Fuzzy K-Means and Fuzzy Vector Quantization) that were described in section 3, are utilized to fuse results coming from the two different methods of speech recognition, as explained in section 4. The experimental results have been shown in

TABLE 1. Experimental Results for Speaker A.

SNR (db)	KM		FKM		FVQ	
30	**FR**	**FA**	**FR**	**FA**	**FR**	**FA**
	1%	0%	1%	0%	5%	0%
20	4%	2%	2%	1%	10%	1%
10	8%	2%	3%	1%	11%	3%

TABLE 2. Experimental Results for Speaker B.

SNR(db)	KM		FKM		FVQ	
30	**FR**	**FA**	**FR**	**FA**	**FR**	**FA**
	1%	2%	2%	0%	4%	1%
20	5%	4%	5%	6%	12%	3%
10	8%	4%	5%	7%	15%	5%

Tables 1 and 2. In these tables FR shows the False Rejection rate and FA shows the False Acceptance rate and effectiveness of system is measured according to these parameters.

The results express that FVQ (Fuzzy Vector Quantization) does not have an appropriate performance in such an application and the FR and FA rates are unacceptably high. But the more important achievement is about the FKM (Fuzzy K-Means) method. While FKM performance is comparable with the Classical K-Means in lower SNR (Signal to Noise Ratio), it is more efficient in higher levels of SNR.

6. CONCLUSIONS

Applying fuzzy clustering algorithms for decision level data fusion in an automatic isolated word recognition system was proposed in this paper. In the proposed method, results coming from two speech recognition methods (spectrograph and LPC Analysis) are fused either by fuzzy k-means or by fuzzy vector quantization. These clustering methods are considered as the fusion tool for integrating the results of the two voice processing methods. They cause that the uncertainty and noise level in the results coming from the two voice processing methods; fall down and decision-making becomes possible. Some experiments were carried out for isolated word identification. Experimental results showed that the fuzzy clustering methods (as fusion methods) give rise to a low level of false acceptance and false rejection rates, expressing reliable decisions for word identification. Moreover, the fuzzy K-means clustering algorithm shows a better performance compared to the classical k-means in higher levels of S/N ratios. Meanwhile the experimental results show the generalization power of the proposed algorithm because learning takes place based only on the samples, which have been pronounced by one speaker, and is successful in isolated word identification for the samples, pronounced by another person.

7. REFERENCES

1. Teissier, P., Robert-Ribes, J., Schwartz, J. L. and Guerin-Dugue, A., “Comparing Models for Audiovisual Fusion in a Noisy-Vowel Recognition Task”, ***IEEE Transactions on Speech and Audio Processing***, Vol. 7, Issue 6, (November 1999), 629-642.
2. Chen, B., Wang, H. and Lee, L. S., “Discriminating Capabilities of Syllable-Based Features and Approaches of Utilizing Them for Voice Retrieval of Speech Information in Mandarin Chinese”, ***IEEE Trans. on Speech and Audio Processing***, Vol. 10, Issue 5, (July 2002), 303–314.
3. Tanyer, S. G. and Ozer, H., “Voice Activity Detection in Nonstationary Noise”, ***IEEE Trans. on Speech and Audio Processing***, Vol. 8, Issue 4, (July 2000), 478–482.
4. Farrell, K. R., Ramachandran, R. P. and Mammone, R. J., “An Analysis of Data Fusion Methods for Speaker Verification”, ***ICASSP '98***, Vol. 2, Seattle, WA, USA, (May 1998), 1129-1132.
5. Chibelushi, C. C., Mason, J. S. D. and Deravi, F., “Feature Level Data Fusion for Bimodal Person Recognition”, ***Proc. of the IEE Sixth International Conference on Image Processing and its Applications***, Vol. 1, Dublin, Ireland, (July 14-17, 1997), 399-403.
6. Keller, J. M., Gader, P. D., Tahani, H., Jung-Hsien, C. and Magdi, M., “Advances in Fuzzy Integration for Pattern Recognition”, ***Fuzzy Sets and Systems***, Vol. 65, (1994), 273-283.
7. Dasarathy, B. V., “Decision Fusion”, IEEE Computer Society Press, Portland, Orlando, USA, (July 1994).
8. Jankowski, C. R., “A Comparison of Signal Processing Front Ends for Automatic Word Recognition”, ***IEEE Trans. On Speech and Audio Processing***, Vol. 3, Issue

4, (July 1995), 286-293.

9. Rabiner, L. and Juang, B. H., "Fundamentals of speech recognition", Prentice Hall, Upper Saddle River, Englewood Cliffs, New Jersey, USA, (1993).

10. Dermatas, E. and Kokkinakis, G., "Algorithm for Clustering Continuous Density HMM by Recognition Error", ***IEEE Trans. on Speech and Audio Processing***, Vol. 4, Issue 3, (May 1996), 231-234.

11. Yuan-Fu, L. and Sin-Horng, C., "A Modular RNN-Based Method for Continuous Mandarin Speech Recognition", ***IEEE Trans. on Speech and Audio Processing***, Vol. 9, Issue 3, (March 2001), 252-263.

12. Tan, L. and Ching, P. C., "Cantonese Syllable Recognition Using Neural Networks", ***IEEE Trans. On Speech and Audio Processing***, Vol. 7, Issue 4, (July 1999), 466-472.

13. Zavaliagkos, G., Zhao, Y., Schwartz, R. and Makhoul, J., "A Hybrid Segmental Neural Net/Hidden Markov Model System for Continuous Speech Recognition", ***IEEE Trans. on Speech and Audio Processing***, Vol. 2, Issue 1, (January 1994), 151-160.

14. Neukirchen, C., Rottland, J., Willett, D. and Rigoll, G., "A Continuous Density Interpretation of Discrete HMM Systems and MMI-Neural Networks", ***IEEE Trans. on Speech and Audio Processing***, Vol. 9, Issue 4, (May 2001), 367-377,

15. Penack, J. and Nelson, D., "The NP Speech Activity Detection Algorithm", ***ICASSP-95***, Vol. 1, Detroit, USA, (May 1995), 381-384.

16. Boll, S. F., "Suppression of Acoustic Noise in Speech Using Spectral Subtraction", ***IEEE Trans. on ASSP***, Vol. ASSP-27, No.2, (April 1979), 113-120.

17. Chatzis, V., Bors, A. G. and Pitas, I., "Multimodal Decision-Level Fusion for Person Authentication", ***IEEE Trans. On Systems***, ***Man and Cybernetics***, Vol. 29, Issue 6, (November 1999), 674-680.

18. Karayiannis, C. and Nicolaos, B., "An Axiomatic Approach to Soft Learning Vector Quantization and Clustering", ***IEEE Trans. on Neural Networks***, Vol. 10, Issue 5, (September 1999), 1153-1165.

19. Garjdos, S. and Lorincz, A., "Fuzzy-Based Clustering of Speech Recognition Database", ***Proc. of IEEE International Conference on Systems***, ***Man and Cybernetics***, Vol. 3, (October 1998), 2744-2749.

20. Strope, B. and Alwan, A., **"**A Model of Dynamic Auditory Perception and Its Application to Robust Word Recognition", ***IEEE Trans. on Speech and Audio Processing***, Vol. 5, Issue 5, (September 1997), 451-464.