Automatic Speaker Identification Based on Fuzzy Theory and Neural network Using Genetic Algorithm

Ching-Tang Hsieh, Eugene Lai and You-Chuang Wang
Department of Electrical Engineering,
Tamkang University, Taipei , Taiwan,
Republic of China

Abstract

This paper proposes a two-stage speaker identification structure, using the average value of first formant (Vi) and zero crossing rate as the parameters and then preliminarily clustering the speech data by the distributed fuzzy rules to eliminate unnecessary silence and consonants. In order to fasten the operation speed to achieve the real time system, we also use the genetic algorithm to screen out unnecessary fuzzy rules. The results of the experiment show that the distributed fuzzy rules do effectively cluster the data and have the adaptability to independent speakers. Also, in screening the fuzzy rules, the genetic algorithm can greatly eliminate unnecessary fuzzy rules, and the difference of the recognition rate is under 1%. After preliminarily screening the speech data, we use the back-propagation neural network as the last speaker recognition structure. Since the system has eliminated silence and less stable consonants, we find that, according to the results of the experiment, the whole recognition rate can also get well-improved. Furthermore, this two-stage recognition structure proposed in the paper makes speaker identification automatic.

1. Introdution:

Generally, speaker recognition can be divided into two parts: speaker identification and speaker verification. Speaker verification refers to whether the speech samples belong to some specific speaker or not. Thus, the result can only be yes or no and it is caculated by the critical value. Of course, this critical value needs to be acquired from the experiment or set by the experts. The setting of the critical value affects the recognition rate of the whole system, and there are many studies in this aspect. Speaker identification system compares the speech samples with referential samples of all the speakers in the data base and finds out the fittest refer-

ential sample. This sample then belongs to the speaker. Finally, using the statistcs or induction, the system can acquire the final result. In comparison of speaker identification system with speaker verification system, the main difference is that speaker identification system has N possible choices but not just either one or the other, and with the increase of speakers, the system becomes more complex. Therefore, the misjudgement rate will increase a little. The operation style of both "speaker verification" and "speaker identification" can be divided into text-dependent and text-independent. "Text-dependent" means the text used in the training system is the same as that the test system uses. This is simpler to the system. On the contrary, "text-independent" means that there is no limitation for the text used in the test system. This style, of course, is more complex to the system. And in comparison with text-dependent, the misjudgement rate of text-independent is higher. Although many scholars have investigated in speaker identification and have a good result on recognition [1]-[4], they lack an integrated structure. Thus, this paper proposes a two-stage recognition structure, using the average value of first formant (Vi) [7] and zero crossing rate as parameters and then preliminarily clustering the speech data by the distributed fuzzy rules to eliminate unnecessary silence and consonants. In order to fasten the operation speed to achieve the real time system, we also use the genetic algorithm to screen out unnecessary fuzzy rules. The results of the experiment show that the distributed fuzzy rules do effectively cluster the speech data. The genetic algorithm can almost screen the fuzzy rules to 1/4 of its original number, and the difference of recognition rate is under 1%. As to the recognition structure, we use the back-propagation neural network, which has the highest accuracy, to do the last identification of the speakers. Since the system has eliminated the silence and less stable consonants of contineous speech, we find that, from the results of the experiment, the whole recognition rate of the system can also get well-improved. This two-stage recognition structure proposed in the paper makes speaker identification automatic.

The contents of the following sections are: Section II introduces the application of the distributed fuzzy rules and the genetic algorithm in the preliminary classification of the speech data; Section III introduces the back-propagation neural network and the speaker recognition structure mentioned in this paper; Section IV is the result and evaluation of the experiments and Section V is the conclusion.

2. The preliminary classification of the speech

The preliminary classification of the speech data is very important to a good speaker identification system. We will use Vi and zero crossing rate as the parameters and then use the distributed fuzzy rules to cluster the characters of the speech data. In order to make the system optimal, we also use the genetic algorithm to screen the fuzzy rules to make the system reach the goal of speedy operation and achieve the real time system.

2.1 Distributed fuzzy rules

Hisao, Ken, and Hideo[5] proposed the "Distributed Fuzzy Rules" to cluster the numerical data using the triangular membership function. For only 3 classes and 9 training samples, the correct ratio for clustering unknown samples is up to 90%. There are two conclusions: (1) Under the same fuzzy partition, the correct ratio by the distributed fuzzy rules is higher than that by the ordinary fuzzy rules; (2) Even for fewer training samples, the correct ratio by the distributed fuzzy rules to cluster unknown samples is still higher. These properties are beneficial for clustering the features of large speech data without many training data. The common by used types of the membership function of the fuzzy rules are triangular membership function, exponential membership function, Mexico hat membership function, and so on. Exponential membership function is shown below:

$$\mu_i^k(x) = \exp(-\beta^2(x - a_i^k)^2) \qquad i = 1,2,.....,k \ (k \geq 2) \qquad (1)$$

where μ_i^k is the membership function of subspace A_i^k and

$$a_i^k = (i-1)/(k-1) \qquad i = 1,2,.....,k \qquad (2)$$

The ordinary fuzzy rules can be described as

If x is X_i^L and y is Y_j^L
then [x,y] belongs to $[X_i^L, Y_j^L]$
i , j = 1,2,.....,L (3)

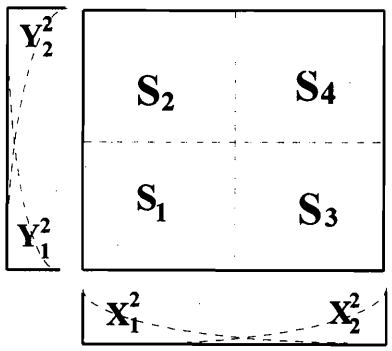
where [x , y] is one sample in unit square [0,1]x[0,1] ,and $[X_i^L, Y_j^L]$ is the subspace of unit square, and L is the fuzzy partitions. Making some modifications of eq. (3) as

If x is X_i^k and y is Y_j^k
then [x,y] belongs to $[X_i^k, Y_j^k]$
i, j =1,2,....,k; k = 2,.....,L (4)

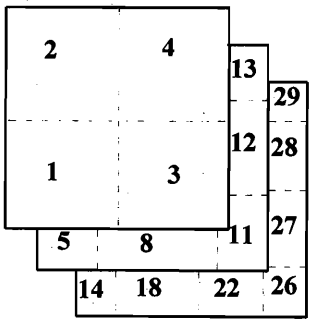
becomes the distributed fuzzy rules. Different number of fuzzy partitions is the difference between the ordinary fuzzy rules and the distributed fuzzy rules. For L partitions, the distributed fuzzy rules are operated on L-1 unit spaces, i. e. from 2 partitions to L partitions. The ordinary fuzzy rules are operated only on one unit space, i. e. the unit space with L partitions. Fig. 1 shows the representations of different fuzzy rules. Clustering every subspaces from the training data to obtain eq. (5) is the training purpose.

If x is X_i^k and y is Y_j^k
then [x,y] belongs to class-x (5)

In eq(5), class-x is the class of own data; the class with maximum membership among all subspace classes is the class of unknown data. More details can be seen from reference[5].



(a) Labels and indices of fuzzy if-then rules (L=2)



(b) Distributed Fuzzy Rules (L=4)

Fig 1. The representations of different fuzzy rules

2.2 The application of genetic algorithm in fuzzy rules

We can get the set of fuzzy rules by training some speech data. Our problem in this section is to select fuzzy rules from all the fuzzy rules to construct a compact rule set with high classification power. We can briefly describe the genetic algorithm operations as follows[6]:

i) Initialization: Generate some initial populations randomly that contain some string ($S=S_1S_2.....S_N$) where

N is the total number of fuzzy rules.

S_n=1 denotes that the nth rule is selected.

S_n=-1 denotes that the nth rule is not selected.

ii) Fitness: Our purpose is to maximize the number of correctly classified speech data by the selected fuzzy rules set S and to minimize the number of fuzzy rules in S. So, the fitness value of each string can be formulated as

$$F(s) = W_{NCP} * NCP(s) - W_S * |S| \tag{6}$$

where $F(s)$ is the fitness value($F(s) \geq 0$), NCP(s) is the number of correctly classified speech data by S and |S| is the number of fuzzy rules in S. $W_{NCP}$and W_s are positive weights.

iii) Reproduction: The selection probability of the individual S in new generation is proportional to its fitness value.

$$P(s) = \frac{f(s)-f_{\min}(\psi)}{\sum_{s' \in \psi} \{f(s')-f_{\min}(\psi)\}} \tag{7}$$

where $f_{\min}(\psi) = \min\{f(s) : S \in \psi\}$

iv) Crossover: We apply one point for crossover to the pair of selected individuals such that we can get new strings.(The step repeats p/2 times, where p is the number of populations).

v) Mutation: This operation can prevent strings to locate into local optimization.

$$S_n \rightarrow S_n * (-1) \tag{8}$$

Repeat steps iii, iv, and v until satisfy the stopping condition.

3. Speaker identification structure of the back-propagation neural network

The back-propagation neural network used in this paper is the most representative and common neural network nowadays. The basic rule of the back-propagation neural network is to use the concept of the gradient steepest decent method to minimize the error function. Because of the practicality and high recognition rate of the back-propagation neural network, this paper applies it to be the main recognition structure of speaker identification.

3.1 The application of genetic algorithm in back-propagation neural network

Although the back-propagation neural network has satisfying results, including high learning accuracy, fast recollection speed, etc.., it still has some unavoidable defects. Many scholars are making researches in this aspect.

Local minimum is the most troublesome problem in the defects of the back-propagation neural network. Because the back-propagation neural network is based on the gradient steepest descent method, it will unavoidably be puzzled by the local minimum. Even in the process of minimizing the error function, the weighting of the network falls into a local minimum of the error function and can't jump out, so that the convergence is incomplete and the error function does not reach the global minimum. This phenomenon is caused by two major reasons: (1) the order of the training samples: Because the adjustment of the weighting of the networks adopts single pattern learning, that is, the weighting adjusts one time in each training sample, the order of the samples will influence the learning result. Fortunately, from the experiment, we know the order of the samples doesn't influence much on the last training result of the system. This paper will not consider this factor. (2) the initial value of the weighting of the networks: the initial value of the weighting will influence the efficiency of the whole system, and if the value is good, the system will approach the optimum quickly; on the contrary, if the value is bad, what the system will get is the local minimum.

Because the genetic algorithm uses the multiple points search, it has a good effect on the optimum of the system. Therefore, this paper uses the advantages of the genetic algorithm to improve the problem of the local minimum that the back-propagation neural network faces. Of course, the algorithm can't fully prevent the network from the trouble of the local minimum. But, it can make the network reach the multiple points search to avoid the result of local minimum as much as possible. Also, to increase the search ability of the system, this paper randomly adds a perturbation to the process of searching. According to the experiment, we find

that adding the random perturbation really helps the search of the system. The way of perturbation is listed below:

$$W^{n+1} = W^n + \Delta W + \xi \tag{9}$$

The whole process of applying the optimum of the genetic algorithm to improve the back-propagation neural network is described below:

(i) Initial population: Randomly giving the weighting on some different groups and regarding them as the initial population in the whole operation.

(ii) Fitness function: Because the training of the back-propagation neural network is to acquire the minimum of error function, the fitness function of every group can be identified as the total error of the neural network.

(iii) Acquiring the weightings of every group by Error Back-propagation. Then figuring out the total error of every group.

(iv) Mutation: Randomly choose one group from the population. Then, randomly assign the position of the selected group and change the weighting value to produce a new group. The mutation times are based on the fitness of the group. That is, the group which has large fitness value has more mutation times; on the contrary, the group which has small fitness value has an obvious declination of its mutation times. This process needs to be done once to all groups.

(v) Cross-over: Randomly choose two groups from the population. Then, use the cross-over method to exchange part of the weighting value to produce new groups. This process is usually executed P/2 times (P is the group number of the population).

Repeat step (iii), (iv), and (v) until the fitness value fits the requirement of the system.

3.2 Speaker identification structure

In the way of speaker identification, because the change of the speech of text independent is too complex, it is necessary to preliminarily cluster and screen the speech data. On the other hand, vowels hold the most part in Chinese text and are more stable, so this paper uses zero-crossing rate and Vi as the parameters of preliminary classification to eliminate unnecessary silence and consonants. In the preliminary classification, this paper uses the distributed fuzzy rules to screen the vowels, because it can get quite good result with very few training samples.

In the last recognition, we use the highly accurate back-propagation neural network to be the last speaker recognition structure. The whole recognition process is represented in Fig. 2.

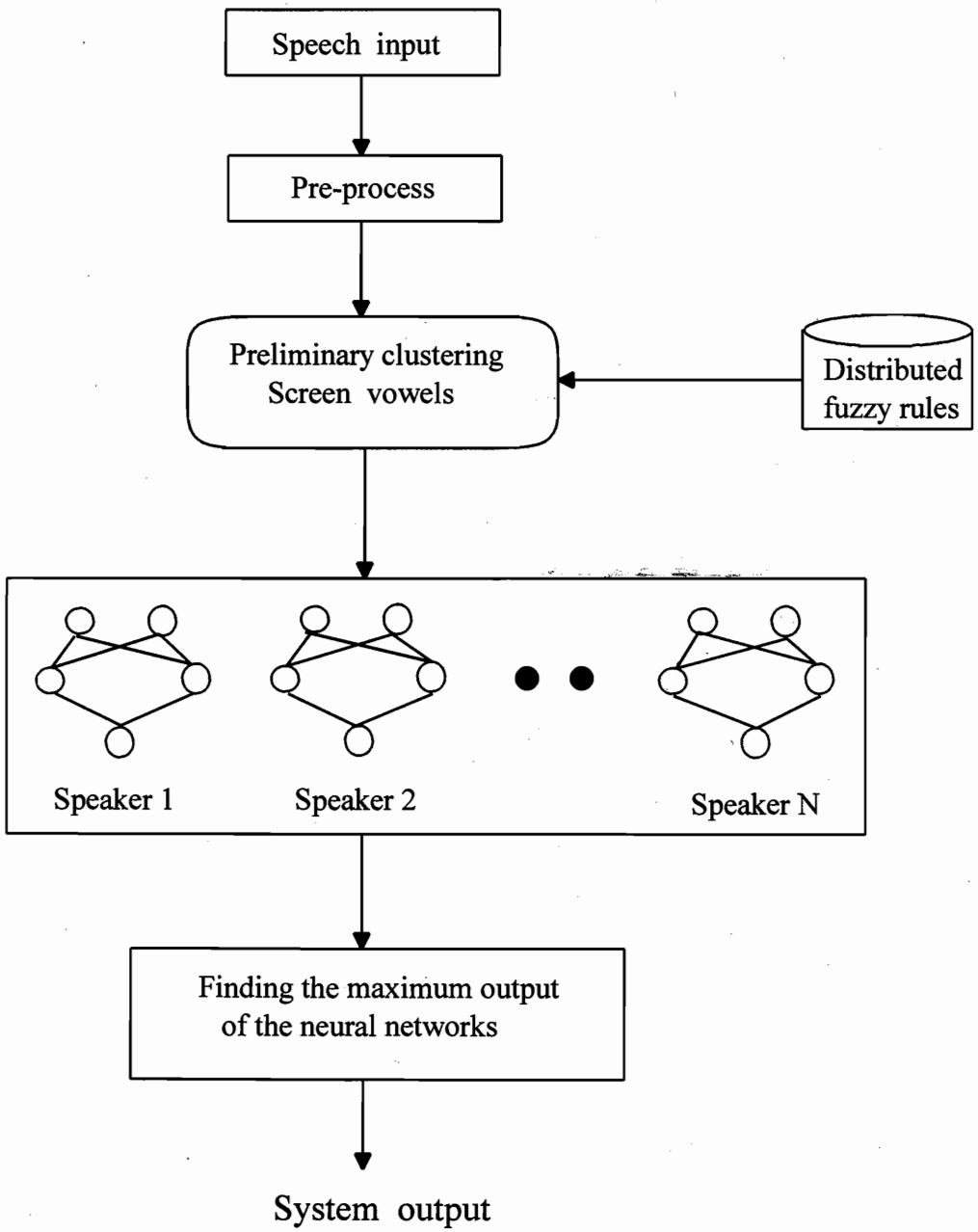


Fig. 2 The recognition structure of two-stage speaker identification

4. Experimental results and evaluation

This section will describe an experiment in the two-stage recognition structure, including the preliminary classification and the screening of the speech data in the first part and the speaker identification structure of neural network in the other part.

4.1 The experiment of the preliminary classification of the speech data

The speech data in this experiment include seven blocks for each of two males and two females. In each block, they read any article for four seconds. The sampling rate is 10kHz. Each analysis frame is 25.6 ms and the overlap between analysis frames is 15.6 ms. The training data come from the first five blocks and the test data come from the last two. Table I shows the classification of speech phonemes. Table II shows the correct recognition rate of classification from partition number L=6 to L=15. From Table II, we can find that the recognition rate in different fuzzy partition has different results for different classes. But, on the whole, the average correct recognition rate increases with the partition number, and the correct recognition rate of each class gradually converges. However, though the larger fuzzy partition number helps to enhance the average correct recognition rate, the fuzzy rules increases tremendously with the increase of partition number (the distributed fuzzy rules are sum of square of the fuzzy partition; if fuzzy partition number is L, then the distributed fuzzy rules $S_{ALL} = 2^2 + 3^2 + ... + L^2$). So, in considering how to increase the recognition rate, we should also think about the execution efficiency of the system. In order to test the distributed fuzzy rules to see if it can still have good result with fewer training data, we use different blocks to be the training data. Table III shows the results. Fig 3 and Fig 4 are the classified diagrams from one training and five training blocks (0 ,1 and 2 denote silence, consonants and vowels, respectively). From table III, we find that the whole classification rate does not increase with the growth of the training data. Thus the distributed fuzzy rules really can have good classification result with very few data.

In order to understand the relation between the adjustable coefficient of membership function and the result of classification, we use 3 blocks as the training data and the fuzzy partition number L=6. Table IV shows the results of the experiment. We find that the rates for silence and vowels decline and the rates for consonants increases with the increase of adjustable coefficient; and the whole classification rate almost doesn't change. Thus, we may know that the setting of adjustable coefficient affects the recognition rate of each class.

Table I. Relationship between Classes and Phonemes

Class	Phoneme
Silence	silence
Consonants	f,d,t,g,k,h,j,b,p,y,m,ch,sh,tz,ts,s,r,l
Vowels	i,u,a,o,e,io,ai,ei,au,ou,el,ia,ie,iau,iou,ua, uo,uai,uei,ue,iue,iua,iu,n,ng

Table II. Classification results with different values of L by distributed fuzzy rules using triangular membership function.

	L=6	L=12	L=13	L=14	L=15
Silence	94.35%	84.85%	84.85%	84.77%	85.21%
Consonants	70.11%	83.54%	83.77%	83.84%	84.14%
Vowels	98.76%	96.02%	95.95%	95.91%	95.87%
average	87.74%	88.14%	88.19%	88.17%	88.41%
fuzzy rules	90	649	818	1014	1239

Table.III The classification result with different training blocks (L=6, β=6)

Training Blocks	1	2	3	4	5
Silence	86.9%	89%	90%	90%	92.1%
Consonants	72.2%	73%	72.8%	71.3%	69.6%
Vowels	99.3%	99%	99%	98.9%	99%
Average	89.9%	90.2%	90.3%	89.8%	89.7%

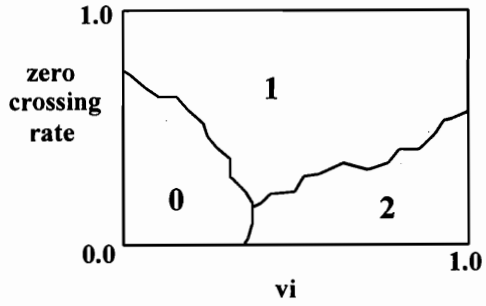


Fig 3. The distribution of classes for 1 training block

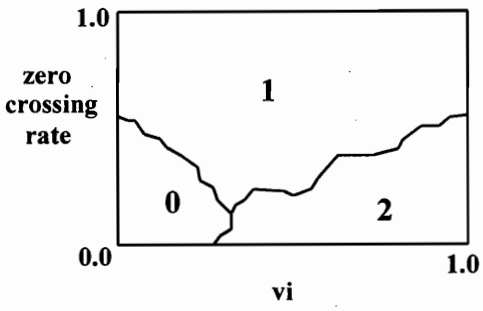


Fig 4. The distribution of classes for 5 training block

Table IV. Classification results with different values of β by distributed fuzzy rules using exponential membership function(L = 6).

	β = 6	β = 7	β = 8	β = 9	β = 10
Silence	88.70%	88.24%	87.77%	87.38%	87.23%
Consonants	79.75%	82.04%	82.96%	83.40%	83.70%
Vowels	98.05%	95.66%	95.30%	94.90%	94.68%
average	88.83%	88.65%	88.68%	88.56%	88.54%

Next, we observe the effect of the genetic algorithm used in this paper. When the fuzzy partition number L changes, we use the genetic algorithm to screen the fuzzy rules. Table V and Table VI are the results of the experiment. From the Tables, we find that although fuzzy rules have been greatly eliminated, the whole classification rate doesn't have obvious declination. We also find that using the exponential membership function to do the classification is better than using the triangular membership function. The number of average fuzzy rules can decline to 1/4 of its original amount. Thus, using the genetic algorithm to screen the fuzzy rules really works. Fig 5 shows the diagram of the distribution of classes after being operated by the genetic algorithm. There is no obvious difference between the distribution of each class before and after the operation of the genetic algorithm.

Table V. Classification results with different values of L by genetic algorithm (Using triangular membership function).

	L=6	L=12	L=13	L=14	L=15
Silence	89.93%	81.13%	83.44%	87.87%	83.52%
Consonants	78.63%	85.17%	85.84%	81.25%	85.47%
Vowels	96.63%	96.79%	95.20%	96.89%	95.14%
average	88.40%	87.70%	88.16%	88.67%	88.04%
fuzzy rules	38	239	314	384	485

Table VI. Classification results by genetic algorithm.
(Using exponential membership function)

	Silence	Consonant	Vowel	fuzzy rules
1	83.13	85.62	94.19	20
2	90	80.43	96.01	21
3	87.15	82.28	95.48	15
average	86.76	82.78	95.23	18.7

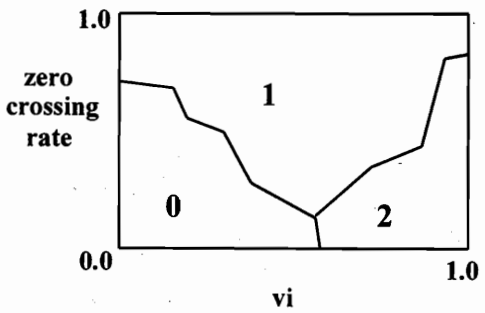


Fig 5. The distribution of classes with GA for 3 training blocks (exponential membership)

4.2 Performance evaluation

This evaluation of speaker identification experiment was conducted in the following manner. First, the test speech was produced a sequence of feature vectors $\{x_1, x_2, ..., x_t\}$. To evaluate different test utterance lengths, the sequence of the feature vectors was divided into overlapping segments of T feature vectors. The first two segments from a sequence would be

$$\underbrace{x_1, x_2, ..., x_T}_{\text{Segment 1}}, x_{T+1}, x_{T+2}, ...$$

$$x_1, \underbrace{x_2, ..., x_T, x_{T+1}}_{\text{Segment 2}}, x_{T+2}, ...$$

A test segment length of 5 seconds corresponds to T=250 feature vectors at a 20 ms frame rate. Each segment of T vectors was treated as a separate test utterance.

The identified speaker of each segment was compared with the actual speaker of the test utterance and the number of segments correctly identified was tabulated. The above steps were repeated for test utterances from each speaker in the population. The final performance evaluation was then computed as the percent of correctly identified T-length segments over all test utterances

$$\%\ \text{correct identification} = \frac{\#correctly\text{–}identified\text{–}segments}{total\#ofsegments} \times 100 \qquad (10)$$

The evaluation was repeated for different values of T to evaluate performance with respect to test utterance length [1].

4.3 The experiment of two-stage speaker identification

In this section, the training data includes twenty-seconds articles and six groups of telephone numbers; the test data are the four records of sentences and listed in Table VII. A 25.6 ms Hamming window is applied to the speech every 10 ms, and the feature consists of the 10 cepstral coefficients (LPC) and the 10 dynamic spetral coefficients. Table VIII shows the experimental results only using the back-propagation neural network as the recognition structure.Table IX shows the experimental results after the preliminary disposition by the

distributed fuzzy rules. From the tables, we can see that we can eliminate unnecessary silence and less stable consonants in the continuous speech when the system is preliminarily classified and screened by the distributed fuzzy rules. Then, we may screen the vowels to do the networks training. The proposed system of this paper not only can fasten the operation speed but also can enhance the total recognition rate.

Table.VII The contents of utterance.

	Contents
1	高壓（gao ya）
2	曲棍球（qu gun qiu）
3	雙管齊下（shuang guan qi xia）
4	淡水捷運站（dan shui jie yun zhan）
5	但願我們能夠永遠在一起（dan yuan wo men neng gou yong yuan zai yi qi）

Table.VIII The experimental results of text-independent speaker identification.
(Using the back-propagation neural network)

Segment length (sec)	5 speakers	10 speakers	20 speakers
1	79.5%	75.2%	68.4%
2	85.9%	80.3%	76.2%
3	89.2%	84.7%	81.2%
4	93.5%	90.1%	85.1%

Table.IX The experimental results of text-independent speaker identification.
(Using the distributed fuzzy rules and the back-propagation neural network)

Segment length (sec)	5 speakers	10 speakers	20 speakers
1	84.7%	78.8%	75.1%
2	92.8%	84.3%	81.5%
3	97%	90.5%	87.2%
4	100%	93.1%	90.3%

5. Conclusion

In the preliminary disposition, speaker identification system mentioned in this paper uses the zero-crossing rate and Vi as the parameters, and then preliminarily clusters and screens the speech data by the distributed fuzzy rules and the genetic algorithm, in order to eliminate the unnecessary silence and consonants. Besides, the system has the adaptability to independent speakers. As to the recognition, in order to reduce the rate of falling into local minimum, we use the characteristics of the genetic algorithm to do multiple points search to the neural networks. From the results of the experiment, we can find that the proposed two-stage recognition structure not only improves the recognition rate but also makes speaker identification autumatic.

Reference

[1] Dougla A. Reynolds and Richard C. Rose, "Robust Text Independent Speaker Identification Using Gaussian Mixture Speaker Models," IEEE Trans on speech and Audio Processing, Vol. 3, No. 1, pp. 72-83, Jan. 1995.

[2] Kevin R. Farrell, Richard J. Mammone and Khaled T. Assaleh, "Speaker Recognition Using Neural network and Conventional Classifiers," IEEE Trans on Speech and Audio Processing, Vol. 2, No. 1, Part II, pp. 194-204, Jan. 1994.

[3] T. Matsui and S. Furui, "A Text-independent speaker recognition method robust against utterance variations," in Proc. IEEE ICASSP, 1991, pp. 377-380.

[4] S. Furui, "Comparison of speaker recognition methods using statistical features and dynamic features," IEEE Trans. Acoust., Speech, Signal Processing, Vol. ASSP-29, pp. 342-350, June 1981.

[5] Hisao Ishibuchi, Ken Nozaki and Tanaka, "Distributed representation of fuzzy rules and its application to classification," Fuzzy Sets and Systems 52(1992), pp. 21-31.

[6] Hisao Ishibuchi, Ken Nozaki, Naohisa Yamamoto and Hiedo Tanaka, "Construction of fuzzy classification systems with retangular fuzzy rules using genetic algorithms," Fuzzy Sets and Systems 65(1994), pp. 237-253.

[7] C. T. Hsieh and S. C. Chien, "Speech segmentation and clustering problem based on fuzzy rules and transition states," Twelfth IASTED Int. Conf. Applied Informatics, 1994.