



OPTIMUM REFERENCE CONSTRUCTION AND UPDATING FOR SPEAKER RECOGNITION SYSTEMS

N.Fakotakis*, E.Dermatas*, G.Kokkinakis*.

ABSTRACT

An algorithm for establishing more representative initial reference data in a speaker recognition system is presented, together with five different methods for updating the reference data. These methods have been tested on a text-dependent speaker verification system and the results are presented.

INTRODUCTION

The establishing and updating of reference data in speaker recognition systems must ensure a high degree of recognition accuracy, a minimum of memory needed for the reference data and a minimum complexity of the algorithms used. Several techniques have been proposed to this end (ref 1,4).

This paper presents an algorithm for establishing initial reference data which gives more representative and therefore more effective results but also requires more complex calculations. Also, five different methods for updating the reference data are described, each having different requirements in memory and in computations.

The performance of the updating methods has been tested on a text dependent speaker verification system and the results are presented.

REFERENCE DATA ESTABLISHING

The reference data characterizing a speaker, result from several training utterances each one represented by a parameter vector. It is assumed that corresponding parameter values are approximately normally distributed and that the intra- and inter-speaker distance distributions are also normal. These assumptions have been proved in several experiments (ref 6).

Therefore, the reference data of the i^{th} speaker are given by the mean \underline{m}_i and the variance \underline{w}_i of the parameter vectors, and by the means μ_{1i} , μ_{2i} and the standard deviations σ_{1i} , σ_{2i} of the intra- and inter-speaker distance distributions respectively, which are used to construct the minimum- or the equal-error decision threshold (ref 2,6). These are calculated by the following relations with \underline{x}_{ij} the n -dimensional vector representing the j^{th} utterance of the i^{th} speaker:

$$\underline{m}_i = \langle \underline{x}_{ij} \rangle_j, \quad \underline{w}_i = \langle (\underline{x}_{ij} - \underline{m}_i)^T (\underline{x}_{ij} - \underline{m}_i) \rangle_j \quad (1)$$

*Wire Communications Lab., University of Patras Greece.

$$\mu_{1i} = \langle d[\underline{x}_{ij}, \underline{m}_i] \rangle_j, \quad \sigma_{1i}^2 = \langle (d[\underline{x}_{ij}, \underline{m}_i] - \mu_{1i})^2 \rangle_j \quad (2)$$

$$\mu_{2i} = \langle d[\underline{x}_{kj}, \underline{m}_i] \rangle_{jk}, \quad \sigma_{2i}^2 = \langle (d[\underline{x}_{kj}, \underline{m}_i] - \mu_{2i})^2 \rangle_{jk} \quad (3)$$

where $k=1, \dots, i-1, i+1, \dots, M$ the number of speakers, $j=1, 2, \dots, N$ the number of utterances necessary to create the reference data, $\langle \rangle_j$ indicates averaging over the subscript j , (T) indicates the transpose of the vector, and $d[\underline{a}, \underline{b}]$ denotes the weighted Euclidean distance between vectors \underline{a} and \underline{b} .

From (1), the pooled intra-speaker covariance matrix or weighting function \underline{w} , included in the reference data of the recognition system, is taken

$$\underline{w} = \langle \underline{w}_i \rangle_i \quad (4)$$

The proposed training procedure for establishing a speaker's reference data is as follows:

Initial reference data are set up using a relatively small number of training utterances (voice samples), e.g. 5 samples, and with the aid of the above relations. Then, each of these initial samples is verified using the initial reference data. If any of these samples is rejected, a new sample replaces it and the initial reference data are recalculated. The reference data are not accepted unless all training samples are positively verified. Next, another sample of the same class is verified and in case of positive verification, this sample is included in the training set. New reference data are then estimated and a new sample is verified as before. This goes on till the number N of the training set is reached.

The above procedure ensures that no samples are used for the reference data establishment that are not accepted by the system.

UPDATING OF THE REFERENCE DATA

Updating of the reference data can be performed in several ways depending on the time interval during which it is assumed that a speaker's voice remains sufficiently stable. If this interval is related to the frequency of accessing the system by the users, an updating according to how many times the system is successfully accessed could be considered.

Below, five updating methods for a speaker verification system are described, in which updating takes place after a different number of successful verifications. In all cases it is assumed that there is a minimum necessary number of utterances, e.g. 12, to construct representative reference data for a speaker.

Method 1: The reference pattern of the i^{th} -speaker and the weighting function are updated each time the speaker is successfully verified. The decision threshold which is calculated during the training procedure is not updated.

Let \underline{x} be the n-dimensional vector representing the test utterance, which has been successfully classified in the i^{th} -speaker. The following recursive formulas for the updated mean vector \underline{m}_i^* variance \underline{w}_i^* and weighting function \underline{w}^* have been introduced. The test utterance \underline{x} , which is more representative of the current speaker's voice, is weighted by $1/N$.

$$\underline{m}_i^* = ((N-1)\underline{m}_i + \underline{x})/N, \quad \underline{w}_i^* = ((N-1)\underline{w}_i + (\underline{x} - \underline{m}_i^*)^2)/N - (\underline{m}_i - \underline{m}_i^*)^2 \quad (5)$$

$$\underline{w}^* = \underline{w} + (\underline{w}_i^* - \underline{w}_i)/M \quad (6)$$

Method 2: The reference data of the i^{th} -speaker are updated each time the speaker is successfully verified using the last N test utterances. To update the reference data the relations (1), (2), (3) and (6) are used.

Method 3: The reference data of the i^{th} -speaker are updated after every $N/2$ successful verifications of the speaker using the last N test utterances. To update the reference data the same relations as in method 2 are used.

Method 4: The reference data of the i^{th} -speaker are updated after every N successful verifications of the speaker using the last N test utterances. To update the reference data the same relations as in method 2 are used.

Method 5: This method combines the methods 1 and 2. The reference data of the i^{th} speaker are updated each time the speaker is successfully verified and the vectors \underline{m}_i^* , \underline{w}_i^* , \underline{w}^* are calculated using the relations (5) and (6), as in method 1. Nevertheless, the statistical values for the intra- and inter-speaker distance distributions are updated using the last N test utterances and the relations (2) and (3), as in method 2.

EXPERIMENTAL RESULTS

The training procedure and the updating methods, have been tested using a speaker recognition system and a large voice data base. The recognition system was developed earlier in our laboratory (ref 3). It is a text-dependent system based on formant frequency parameters. An utterance is represented by a n-dimensional vector, the parameters of which are the first three formant frequencies on $n/3$ peaks corresponding to vowels.

The voice data base, consisting of the reference and test parameter vectors, was created by means of a simulation procedure. On the basis of former experimental results, the following assumptions were made for the parameters: (a) normal distribution, (b) intra-speaker variance of 10% of the mean values, (c) inter-speaker variance of 30% of the mean values, (d) random shifting of the mean values for the successive repetitions in a range of 0.1%, (e) no correlation between the parameters.

Using the above assumptions the data base was created with the following procedure: An 18-dimensional pattern vector, extracted from a real speech sample was taken as the basic vector (as the mean vector of the inter-speaker distribution). From the basic vector and in a range of $\pm 30\%$, initial mean vectors for each of the $M(=15)$ classes were randomly estimated, using a uniform distribution generator. With the mean vectors and a variance of 10%, the test vectors for each class were established, using a Gaussian distribution generator (ref 5). For every new test vector the initial mean vector, given to the normal generator, was shifted randomly within 0.1%. Each of the 18-parameters comprising the vectors was estimated separately.

The data base created by the above simulations includes: 12 patterns made for each of the 15 classes, for the reference data construction, and 700 patterns made for each of the 15 classes, for the test procedure.

The five updating methods and verification without updating were tested experimentally. Each of the test patterns was used as belonging both to a true speaker and to an impostor to estimate the false rejection (FR) and the false acceptance (FA) error rates, respectively. Table 1 shows the error rates of the speaker verification experiments using the five updating methods and one without updating.

Table 1 verification error rates of six experiments.

	No Updat.	Method1	Method2	Method3	Method4	Method5
FA (%)	40.32	30.12	2.32	2.26	2.42	1.68
FR (%)	33.11	23.46	0.14	0.20	0.21	0.13
(FA+FR)/2	36.71	26.79	1.23	1.23	1.31	0.90

CONCLUSION

A new algorithm for establishing initial reference data and five methods for updating the data have been tested on a speaker verification system. The results have shown that the method combining a weighted updating of the mean and the variance and a normal updating of the intra- and inter-speaker distances, gives the best performance.

REFERENCES

1. G R Doddington, proc. ELECTRO-76, P 22, (1976).
2. N Fakotakis, E Dermatas, G Kokkinakis, EUSIPCO-86, V2, p585.
3. N Fakotakis, G Kokkinakis, MELECON-85, (1985).
4. S Furui, Trans. IEEE ASSP, vol. 29, pp. 254-272, (1981).
5. G A Mihram, Simulation (Academic Press NY, 1972).
6. L R Rabiner, et. al., Tr IEEE, ASSP-26, p575, (1978).