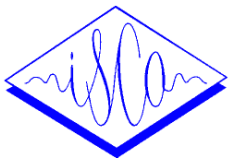

3rd International Conference on
Spoken Language Processing (ICSLP 94)
Yokohama, Japan
September 18-22, 1994

A TEXT-INDEPENDENT SPEAKER IDENTIFICATION SYSTEM BASED ON NEURAL NETWORKS

Jialong He, Li Liu and Günther Palm

Abteilung Neuroinformatik
University of Ulm, 89069 Ulm, Germany

ABSTRACT

A text-independent automatic speaker identification system was constructed and evaluated with the TIMIT database. All voiced parts of speech signals were automatically located by measuring the short-term energy of the signals. For each segment of the voiced signals LPC based cepstrum were calculated to compose a feature vector. Multilayer perceptron (MLP) and learning vector quantization (LVQ) networks were used as classifiers. The codebooks of the LVQ classifiers were initialized by the LBG algorithm and then were trained by the LVQ3 algorithm. The MLP classifiers were standard feed forward networks with one hidden layer and were trained in two steps by the conjugate gradient method. Speech data from 112 male speakers in the test subdivision of the TIMIT database were used to evaluate our system. For each speaker, we randomly selected eight sentences as training data and the remaining two as the testing ones. The results showed that the best correct identification rates were 88.4% by LVQ classifiers and 99.1% by MLP classifiers for a population of 112 speakers.

I. INTRODUCTION

Speaker recognition is a general term which refers to any task to discriminate people based upon their voice characteristics. Within this general task description there are two specific tasks that have been studied extensively. These are referred to as *speaker identification* and *speaker verification*. The speaker identification task is to classify an unlabeled voice token as belonging to one of a set of N reference speakers, whereas the speaker verification task is to decide whether or not an unlabeled voice token belongs to a claimed reference speaker. It is natural to expect that, all other factors being equal, the speaker identification performance will be lower than that of the speaker verification. Although the verification is more tractable and has more practical applications, the identification task is more suitable for comparing the performance of different systems. Speaker recognition systems may also be classified as *text-dependent* systems and *text-independent* systems. In a text-dependent system, the utterances used for training and testing are pre-selected, normally have the same texts. In contrast, a user may provide any utterance to train or test a text-independent system. Generally speaking, a text-independent speaker recognition system is more difficult, since the system must now cope with the additional variability due to the differences in the texts of the unknown and the reference utterances. The population size is another critical performance parameter for a speaker identification system since the probability of misclassification approaches 100% for infinitely large populations. To our knowledge, most of the researchers employ only a small number of speakers (10-30 speakers) to investigate some properties of their system, there is no report about systems which have been evaluated with significantly larger samples of speakers and utterances. Excellent reviews in this research area can be found from the papers written by Atal [1], Rosenberg [2] and Doddinton [3]. Like other pattern recognition systems, a speaker recognition system consists of two parts: feature extraction and classification. To extract speech parameters which are capable of efficiently representing the speaker dependent information is an important step towards achieving successful speaker recognition. Features such as pitch period and LPC based cepstrum have been found belonging to the most efficient parameters for speaker recognition [4][5]. Traditionally, classifications are mainly based on the distance criteria which quantify the degree of dissimilarity between the testing feature vector and the reference vectors. There are many distance metrics that have been investigated. In recent years, connectionist models have been widely accepted as an alternative tool for static pattern classification. Their main useful properties are their discriminative power and their ability to capture input-output relationships [6]. This paper presents an on-line text-independent speaker identification system based on LVQ3 and MLP neural networks. This system had been evaluated with the TIMIT database and it was shown that, with a population of 112 speakers, the correct identification rates were 88.4% for the LVQ classifiers and 99.1% for the MLP classifiers.

II. SYSTEM ARCHITECTURE

The schematic block diagram of our on-line text-independent speaker identification system is shown in figure 1. When operating in the on-line mode, two classifiers (LVQ and MLP) are simply combined to implement a sequential decision procedure [3]. That is, if the classified results from both networks are consistent the system will accept this result, otherwise, the system will prompt the user to speak another sentence. In the present study, however, the testing sentences extracted from the TIMIT database were classified by each of two classifiers independently in order to compare their performances.

A. Speech database

The TIMIT database has been designed to provide speech data for the development and the evaluation of automatic speech recognition systems. It contains speech data from 630 speakers from 8 major dialects of American English, each speaking 10 phonetically rich sentences. The average length of the sentences is 6.15 seconds. The speech data were recorded in a quiet room and digitized at 16 kHz in 16 bit precision. Since the voice characteristics between male and female speakers are fairly different, it is more reasonable to evaluate a speaker recognition system using the data from the same gender. Hence, we used the speech data only from male speakers (112 speakers) in the test subdivision of the TIMIT. Eight out of 10 sentences spoken by each speaker were randomly selected as training data and the rest two as testing ones.

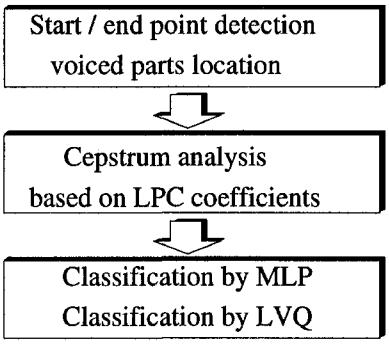

Figure 1. Functional block diagram of the on-line speaker identification system

B. Feature extraction

In order to implement a high performance speaker identification system, one should extract a set of acoustic features from speech signals which (1) occur naturally and frequently in normal speech, (2) vary as much as possible among speakers but as consistent as possible for each speaker. Some researches have been done to determine the relative merits of different phonemes. It is generally agreed that the voiced parts of a speech signal, especially vowels and nasals, are more effective than the unvoiced parts for speaker identifications [7][8]. To implement a text-independent system, we located only the voiced parts of a sentence regardless of their contents by using a simple energy measuring method. The length of the analysis window was 64 ms without overlapping. It should be noted that this analysis window size is a little larger than the commonly used sizes (normally 16-32 ms) since from our pilot experiments we had found that a shorter analysis window would degrade the identification performance. A possible explanation might be that the cost by rejecting a useful segment of speech signals is much lower than that by mistaking a segment of the speech signals which carries little speaker specific information. Whenever the short-time energy of a frame of the sentence was higher than a threshold, spectral features would be calculated. In our on-line system we used the sum of absolute instead of squared values of all samples as a measurement of the energy to speed up the calculation. The threshold was selected as the half of the average energy of the whole sentence. Among the many features related to speaker identifications, LPC based cepstral coefficients are thought to be one of the most efficient feature sets [5]. We calculate the cepstrum from each frame of signals using a fairly standard method [9]. It can be briefly described as following. (1) The samples are pre-emphasized by a first order digital filter with the factor 0.97. (2) 19th-order LPC coefficients are calculated from autocorrelation coefficients by the Durbin's recursive algorithm. (3) 19 cepstral coefficients are derived from the LPC parameters using the recursive relation defined in [5]. We decided to extract more cepstral coefficients (here 19) than that used by other researchers (normally 12-16). We had found that, from our pilot experiments, the identification performances could be slightly improved when higher order cepstral coefficients were included.

C. LVQ classifier

Learning Vector Quantization (LVQ) is a nearest-neighbor classifier proposed by Kohonen [10]. Unlike K-means algorithm, the LVQ is not used to approximate density functions of the class samples, but to directly define the class borders according to the nearest-neighbor rule. The class borders are represented piecewise linearly by segments of midplanes between code vectors of neighboring classes. There are three types of LVQ algorithms, namely LVQ1, LVQ2.1 and LVQ3, all of them assume a good initial state. This can be obtained by the traditional K-means clustering procedure or by the Self-Organizing Map algorithm. In our system, we mainly use the LVQ3 algorithm since it provides a better performance and is self-stabilizing. The initial placements of the code vectors are decided by the LBG vector quantization algorithm [11]. In the identification phase, each vector of a test sentence is labeled with a corresponding class symbol by nearest-neighbor comparison of this vector with all code vectors. The classification decision for the test sentence is determined by the majority voting of all vectors from this sentence.

D. MLP classifier

It has been demonstrated that arbitrary complex decision regions can be approximated if more than one hidden layers are used [6]. The MLP networks we examined are standard feedforward ones which have only one hidden layer. From our pilot experiments we noticed that the networks with a single hidden layer converged more easily to an acceptable local minimum. In this evaluation, all MLP networks have 19 input nodes corresponding to the order of cepstral parameters, 112 output nodes which equals to the total classes, or the number of speakers in the system. Various numbers of hidden nodes (50 - 300) have been tried. The networks were trained by the conjugate gradient method [12]. The conjugate gradient algorithm is usually able to locate the minimum of a multivariate function much faster than the gradient-decent procedure that is customarily employed with BP. Furthermore, there is no need to choose the learning rate and the momentum parameter which are critical to the success of the optimization in conventional BP algorithms. However, we had found that a MLP classifier trained by the conjugate gradient method worked well with a small number of speakers but almost always converged rapidly to a local minimum which was too high to be acceptable when the number of speakers in the system was relatively larger (more than 50 speakers). We found a method to overcome this problem by training the network in two steps. At the first step, we trained the

network with partial training vectors that could be correctly classified by a LVQ network. That is, the MLP network was first trained with "good" or easily identified speech data, these speech data could be correctly classified at frame level by the LVQ network. Based on our own experience, it is better to select these vectors with a small codebook, e.g. 4-8 code vectors per speaker. If a larger codebook is used, too many training vectors will be selected for the first training phase. Since in this case the problem will get more complex than that with less training vectors the network may converge very slowly or even converge to an unacceptable local minimum. After the network reaches a local minimum in the first training phase we fine tune the network over the entire training sets. The network trained with the above procedure usually can find a relatively small local minimum. During the training, the target pattern is a vector with all elements set to 0 except for the element corresponding to the speaker of the input pattern. This element of the vector is set to 1. In the identification phase, the outputs of each output nodes are accumulated over all vectors from the current testing sentence. The classification decision is made corresponding to the output node with the highest accumulated value.

III. RESULT AND DISCUSSION

Figure 2a shows the percent of correct identification, at the sentence level, as a function of the number of code vectors per speaker. In this figure, the solid lines represent the identification results for the training data, correspondingly, the identification accuracy for the testing sentences are shown in the dashed lines. The intermediary identification results obtained from the initial codebooks are shown in open circles, the final results from the fine turned codebooks are displayed in filled circles. Since we had randomly selected 8 out of 10 sentences spoken by each speaker as training data and the rest two as testing ones, there were 896 judgments for each data point in the solid curves and 224 tests for each data point in the dashed curves. Unlike the curves for the training data which rise steadily with the size of codebook enlarged, the identification performances for the testing data are not monotonical functions of the codebook size and can not be further improved by simply increasing the number of code vectors when it is more than 12 code vectors per speaker. The explanation might be that, in natural speech, there are mainly about 10 different voiced phonemes, e.g. vowels, nasals (diphthongs viewed as composed of two vowels), each voiced phoneme of one speaker forms a cluster which will be approximated by a code vector. If more code vectors are used, some special characteristics of training data may be captured but not the general properties of this phoneme so that the identification performance for testing data will not be improved. Furthermore, it is noticed from Fig. 2a that all identification performances are improved after applying the LVQ3 algorithm to the initial codebooks. This is easy to understand if we think of the fact that the goal of a classical VQ algorithm, such as LBG algorithm, is to approximate the probability density function of input vectors, consequently, to minimize the errors which are caused by representing input vectors with their code vectors. However, the objective of the LVQ algorithm is to approximate the Bayesian decision surfaces by adjusting the borders between classes, therefore, to minimize the number of misclassified vectors. This result shows that the LVQ algorithm is superior to the commonly used VQ classifiers for speaker recognition [13][14]. In sum, our evaluation shows that, with a population of 112 male speakers, the best identification result by LVQ classifiers is 100% for the training sentences and 88.4% (198/224) for the independent testing data.

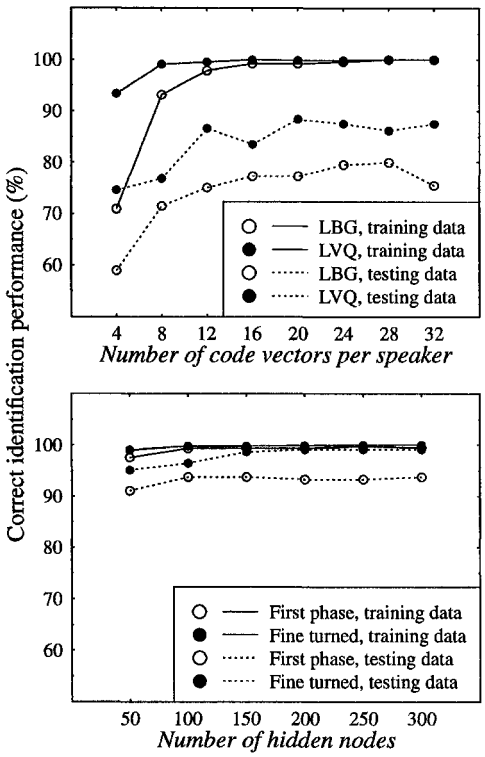

Figure 2a (top) Identification performance with LVQ classifiers as a function of the number of code vectors per speaker. Figure 2b (bottom) Identification performance with MLP classifiers as a function of the number of hidden nodes. In both graphs, Y-axes were scaled to start from 50%. Solid lines show the results for the training data based on 896 judgments and dashed lines show the results for the testing sentences based on 224 tests. The population size is 112 male speakers from the test subdivision of the TIMIT.

The results obtained from MLP classifiers are shown in Figure 2b. Similar to Figure 2a, the solid lines and dashed lines indicate the percent of correct identification rates for the training and testing data, respectively. The open circles show identification results after the first training phase. The final results are shown in filled circles. In this evaluation, the vectors used in the first training phase were selected based on a fine turned codebook with 8 code vectors per speaker. Surprisingly, 93.8% performance was achieved even if the networks had been only trained with partial training vectors. Comparing with Fig. 2a, the identification rate of the LVQ3 network is only 76.8% in the case of 8 code vectors per speaker, much lower than that of MLP networks. As mentioned above, in the first training phase, only vectors that could be correctly classified by

the LVQ3 network were used to train the MLP networks, at frame level, MLP networks could at most reach the performance of the LVQ3 network for the current training set. However, at sentence level, especially for the testing data, the performances of the MLP networks were much better than that of the LVQ3 network. Thus MLP networks with one hidden layer demonstrated more generalization capability than LVQ networks did. An explanation to this difference is that there is only one layer in the LVQ networks and the class borders are formed from the piecewises of midplanes between codebook vectors. On the other hand, the decision regions of two layer MLP networks can be any convex shape [6] or even more complex shape [15]. It can also be seen from Figure 2b that the performances of identification to the testing sentences can not be further improved by increasing the number of hidden nodes after they have reached a particular level (here 93.8%). The networks must be further trained with the whole training sets. Besides, it is worth while to mention that although the performances can be slightly improved by increasing the number of hidden nodes, the training time increases dramatically. A good compromise in our case is to select 150 hidden nodes, where the identification performances are 99.8% for the training data and 98.7% (221/224) for the 224 testing sentences.

The higher identification performance of this system indicated that neural networks are promising techniques for speaker recognition due to their discriminative power for static patterns. Unlike in speech recognition where the transition parts of speech signals play an important role, speaker recognition systems, especially text-independent systems, make mainly use of the steady parts of signals. In consequence, one avoids dealing with one of the major limitations of neural networks, namely, most of the neural network architectures are not suitable to capture the dynamic characteristics of speech signals. Furthermore, by examining the MLP networks more closely, we found that the corresponding output of a correctly identified sentence is generally much larger than the outputs of other neurons. However, no output is significantly larger than others if a sentence is misclassified by all networks, in another word, the classification results from different networks for the misclassified sentences are quite inconsistent. By a subjective examination of these easily misclassified sentences we have noticed that they usually contain many fricative consonants. This reminds us that it is useful, when operating in on-line mode, to combine several MLP networks with different number of hidden nodes, or even with the same architecture but trained with different initial values to form a sequence decision process. The method that a MLP network is trained in two steps can be extended to a multi-step training process. In a preliminary experiment, we trained a MLP network with 100 hidden nodes in three steps. In the first step, in order to have the network to form the rough decision borders quickly, we made directly use of code vectors from a small LVQ codebook as the training data to train the MLP network, and then we trained the network further with a larger codebook (32 code vectors per speaker), at last, we fine turned the network over all training data. For 326 speakers (all male speakers in the training subdivision of the TIMIT), more than 80% correct identification rate had been reached. It should be noticed that the high identification rates obtained with the TIMIT database do not imply that our system is a practical speaker identification system since the TIMIT database is a high quality database which is mainly used as a benchmark for speech recognition systems. Many researchers show that the recording environment, channel distortion and intersession variability have significant effects on a speaker identification system [2][3]. Further studies are being conducted to improve the performance of our speaker identification system.

IV. REFERENCES

[1] B. S. Atal, "Automatic recognition of speakers from their voices," Proc. IEEE, vol. 64, pp. 460-475, April. 1976.

[2] A. E. Rosenberg, "Automatic speaker verification: a review," Proc. IEEE, vol. 64, pp. 475-487, April. 1976.

[3] G. R. Doddington, "Speaker recognition - identifying people by their voices," Proc. IEEE, vol. 73, pp. 1651-1664, March. 1985.

[4] B. S. Atal, "Automatic speaker recognition based on pitch contours," J. Acoust. Soc. Am. 52, pp. 1687-1697, 1972.

[5] B. S. Atal, "Effectiveness of linear prediction characteristics of the speech wave for automatic speaker identification & verification," J. Acoust. Soc. Am. 55, pp. 1304-1312, 1974.

[6] R. P. Lippmann, "An introduction to computing with neural nets," IEEE ASSP Magazine, 3, pp. 4-22, April, 1987.

[7] J. J. Wolf, "Efficient acoustic parameters for speaker recognition," J. Acoust. Soc. Am. 51, pp. 2044-2056, 1972.

[8] M. R. Sambur, "Selection of acoustic features for speaker identification," IEEE Trans. ASSP, ASSP-23, pp. 176-182, 1975.

[9] J. Makhoul, "Linear Prediction: Tutorial Review," Proc. IEEE, vol. 63, pp. 561-580, April. 1975.

[10] T. Kohonen, "The self-organizing map," Proc. IEEE, vol. 78, pp. 1464-1480, Sept. 1990.

[11] Y. Linde, A. Buzo and R. M. Gray, "An algorithm for vector quantizer design," IEEE Trans. Comm., vol. 20, pp. 84-95, Jan. 1980.

[12] M. J. D. Powell, "Restart procedures for the conjugate gradient method," Mathematical Programming, vol. 12, pp. 241-254, 1977.

[13] A. E. Rosenberg and F. K. Soong, "Evaluation of a vector quantization talker recognition system in text independent and text dependent modes," Computer Speech and Language, vol. 22, pp. 143-157, 1987.

[14] N. Fakotakis, A. Tsopanoglou and G. Kokkinakis, "A text-independent speaker recognition system based on vowel spotting," Speech Communication, vol. 12, pp. 57-68, 1993.

[15] G. J. Gibson and C. F. N. Cowan, "On the decision regions of multilayer perceptron," Proc. IEEE, vol. 78, pp. 1590-1594, Oct. 1990.