MODELING TIME SERIES SIGNAL PATTERNS BY STATISTICAL DISTRIBUTION OF PREDICTION ERRORS AND ITS APPLICATION TO SPEAKER IDENTIFICATION

*Qian-Rong Gu and Tadashi Shibata**

Department of Electronic Engineering, The University of Tokyo
tt07142@mail.ecc.u-tokyo.ac.jp
*Department of Frontier Infomatics, School of Frontier Science, The University of Tokyo
shibata@ee.t.u-tokyo.ac.jp

ABSTRACT

A new method that uses statistical distribution of prediction error vectors to build models of time series patterns has been developed. A universal predictor is firstly established from universal training data. Then, properties common to all the patterns are removed from the training data by the predictor. The residuals, i.e., the prediction errors, hold the characteristics of individual patterns. After clustering the prediction errors to a universal codebook, the predictor and the codebook are applied to individual training data sets to obtain the usage histograms of code vectors in the universal codebook, namely, the statistical distribution of prediction error vectors. These histograms represent the properties of individual patterns, and can be used as models in pattern recognition applications. This method is not restricted to any specific signals. As a demonstration, we utilized it to speaker identification application. It performed as well as other modeling methods under the text-dependent condition.

Keywords: Pattern Model, Prediction Error, Vector Quantization, Statistical Distribution, Speaker Identification.

1. INTRODUCTION

Previous attempts at pattern recognition applications have used a variety of methods to create pattern models, including neural networks, statistical approach and syntactic approach [1]. Neural networks can be viewed as massively parallel computing systems consisting of an extremely large number of simple processors with many interconnections. By adjusting connecting weights between the neurons, the neural networks have the ability to learn complex nonlinear input-output relationships from sequential training procedures, and adapt themselves to the training data to form the model of patterns [2]. In the statistical approach, each pattern is represented in terms of d parameters and is viewed as a point in a d-dimensional space. The boundaries of patterns', i.e., the models of patterns, are determined by the probability distributions of the training data during the training phase [3]. In the syntactic approach, complex patterns are considered as being composed of simple sub-patterns, which are themselves built from much simpler sub-patterns. The simplest sub-patterns are called primitives and the model of given complex pattern is represented in terms of the interrelationships between these primitives [4].

This paper proposes a modeling method that is not restricted to any specific type of time series data. It uses statistical distribution of prediction error vectors to represent patterns of time series signals. The underlying concepts are very straightforward. Current signal can be predicted by the preceding signals since there are common properties in the signals. While, these common properties are not very helpful for the pattern recognition purposes because the recognition is based on individual characteristics. Our method is landed on the fact that the individual characteristics are contained in the prediction errors. For each pattern, its model can be established by analyzing statistical distribution of the prediction error vectors extracted from its training data set. We tested this modeling method by utilizing it to speaker identification application.

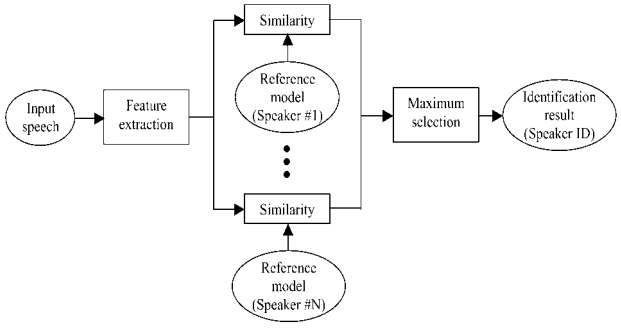

Fig. 1 Speaker identification system

Fig. 1 illustrates a speaker identification system. The goal of speaker identification is to recognize the unknown speaker from a set of N known speakers by matching his or her speech with stored reference models. At present, most common methods used to create speaker models are Hidden Markov Models, Neural Networks, Auto-Regressive Vector Model [5]. Though, the proposed method did not perform over the common ones, it reached a recognition rate as good as the others under the condition of text-dependent.

The rest of this paper is organized as follows. A detailed explanation of our method is given in section 2. Experiments are described in section 3. Experimental results and discussions are arranged to section 4. Finally, conclusions are drawn in section 6.

2. USING HISTOGRAM OF PREDICTION ERROR VECTORS TO REPRESENT PATTERNS

Almost all the time series signals in the real world can be predicted by observing its past samples. The linear estimation of signal X at time *n* can be written as:

$$\hat{x}_n = \sum_{k=1}^{m} a_k x_{n-k} \qquad (1)$$

Where $\hat{x}_n$ is the prediction of the signal X at time *n*, x_{n-m}, …, x_{n-1} are *m* observations of the signal X before time *n*. $a_1,\ldots,a_m$ are corresponding coefficients of x_{n-1}, …, x_{n-m}. *m* is order of the predictor. The prediction error at time *n* can be written as:

$$e_n = x_n - \hat{x}_n \qquad (2)$$

If the predictor is created from a universal training data set, it represents general properties of the training data that contains the patterns need to be recognized. Thus, it can be used to remove the properties common to all the patterns from the input signals; and leave the individual pattern characteristics in prediction errors, which can be utilized for pattern recognition applications.

Vector Quantization (VQ) is an efficient method to map a given set of vectors X = { x_i | i = 1,...,L} into K (K<< L) clusters such that similar vectors are grouped together and vectors with different features belong to different groups [6]. The Generalized Lloyd Algorithm (GLA) [7] is a well-known algorithm to extract codebook being necessary in the mapping process from training data set. In training phase, by applying GLA to prediction error vectors resulted from the predictor, a codebook universal to the entire individual patterns can be generated. While, each pattern has its own characteristics that distinguish it from others. The individual characteristics are expressed in the form of different statistical distribution of prediction errors for different pattern. If we vector quantize the prediction errors calculated from individual training data set with the universal codebook, a corresponding usage histogram of each code vector is obtained. The histogram is different for different pattern because the individual properties of the patterns are different. In other words, these histograms can be used as models to represent individual patterns. Fig. 2 illustrates the above concepts.

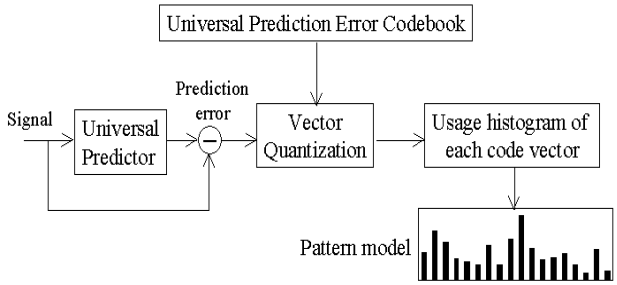

Fig. 2 The procedure of creating pattern models

Since the predictor and the codebook are learned from training data set, the special knowledge of particular signal is not necessary to create pattern models. Thus, this modeling method is not restricted to any specific signals. With proper pre-process of signals, it can be applied to many different pattern recognition applications.

In the training phase, the detailed procedures of creating models from training data sets are:

Stage I:

1. Prepare a universal training date set.
2. Design a predictor from the universal training date set.
3. Calculate the prediction error data set.
4. Generate a universal prediction error codebook from the prediction error data set. (By GLA)
5. Optimize the predictor and the codebook for each other.

Stage II:

1. Prepare training data set for individual patterns.
2. Apply the predictor and the codebook to the individual training date set.
3. The usage histogram of code vectors in the universal codebook is the model of this individual pattern.

In the recognition phase, the calculating procedure is similar to the Stage II of the training phase except that the predictor and the codebook are applied to the real-time input signals instead of the individual training data set. Then the real-time calculated histogram is matched against all the models saved in the database. The model with minimum distance from the real-time histogram is output as the recognition result.

3. EXPERIMENTS

In this research, as a demonstration, we utilized the proposed modeling method to speaker identification application.

A universal training data set was collected from 14 speakers (6 males and 8 females). Each speaker has a 150-second-length speech in the universal training data set. Then, 10 different sentences around 15 seconds length were recorded three times for each speaker. One record was used as individual training data set to create the histogram of the speaker, i.e., the speaker model. The other two were used for identification purpose. These 15-second-length speeches were further divided into five different subsequences of the length 15s, 12s, 9s, 6s and 3s to investigate the influence of speech length on the identification rate, namely the efficiency of the modeling method. In addition, several other sentences with various length and different contents were also recorded for the text-independent speaker recognition test. All of the speeches were sampled at a rate of 8.0 kHz with a resolution of 16 bits per sample.

Before the experiments begun, the following pre-processes were carried out to extract the acoustic features from the speech signals [8]:

1. Using short-term energy calculations to remove unvoiced parts in the speech samples.
2. Pre-emphasizing the speech signals by a high pass filter with a transfer function of $H(z) = 1 - 0.95z^{-1}$.
3. Using a 20 ms Hamming windows with an overlap of 10 ms to divide the speech signals into frames.
4. Using short-term spectral analysis to extract Mel-Frequency Cepstrum Coefficients (MFCC) [9] from the speech signals.
5. Discarding coefficient 0 of MFCC, because it corresponds to the total energy of the frame.
6. Using $x^*_k = (x_k - \mu_k) / \sigma_k$, $\forall$ k = 1,…, P to normalize the MFCC vector. Here x_k and x^*_k are the original and normalized vector components, respectively; μ_k and σ_k are the mean and the standard deviation of the kth component over all MFCC vectors [7].

After that, the normalized MFCC vectors extracted from the universal training data set were used to design the universal predictor and the universal codebook of prediction errors. The dimension of MFCC vector was selected as 16, and the order of predictor was set to 16. The code vectors in the universal codebook were sorted according to the distance between them before being applied to create models of individual speakers

By using these test data, we investigated:

1. If the histograms calculated from speeches with different contents for the same speaker were similar?
2. If the histograms calculated from speeches with same contents, while recorded at different times for the same speaker were similar?
3. If the histograms calculated from speeches with same contents for different speakers were different?
4. Which percent of the recognition rate could be achieved?
5. The influence of speech length on the recognition rate.
6. The influence of the codebook size on the recognition rate.

4. EXPERIMENTAL RESULTS AND DISCUSSIONS

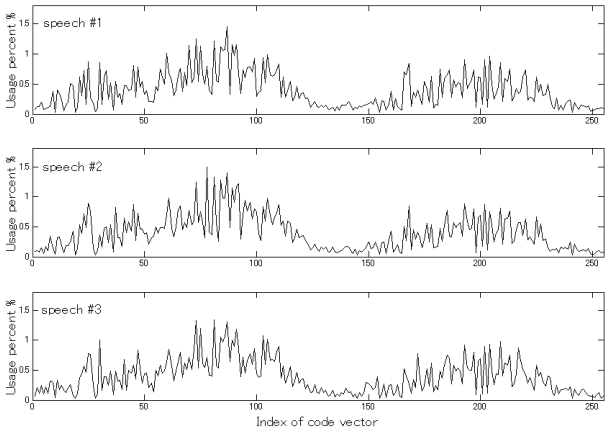

Fig. 3 Histograms of three speeches with different contents for the same speaker (codebook size: 256)

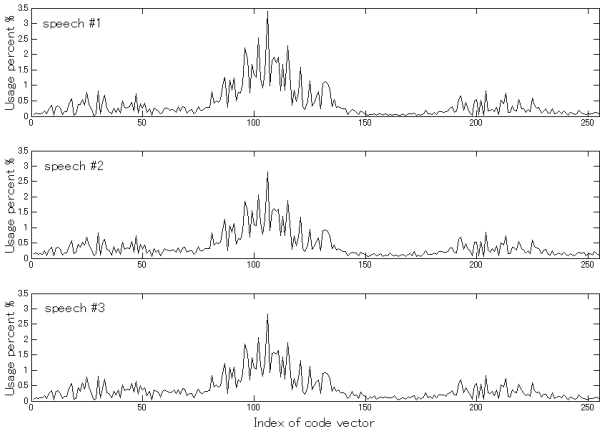

Fig. 4 Histograms of a same sentence recorded at three different times for the same speaker; the speaker is different from the one in Fig. 3 (codebook size: 256)

Figs. 3 illustrates the usage histograms calculated from three speeches with different contents for one speaker. Fig. 4 shows the usage histograms calculated from a same speech recorded at three different times for another speaker. From Fig. 3 and 4, we can see that the histograms are robust to both the change of speech contents and the change of recording conditions.

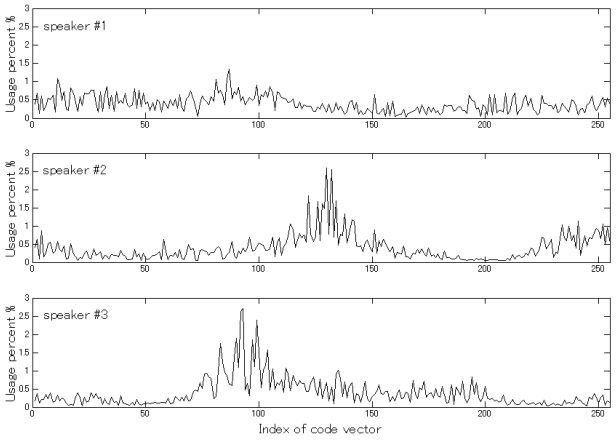

Fig. 5 Histograms of a speech with same contents for three different speakers (codebook size: 256)

Fig. 5 shows the histograms calculated from a same speech for three different speakers. Though the speech contents are the same, the histograms are different greatly from each other because the speakers are different. From Fig. 3, 4 and 5 obviously, the histogram represents the individual properties of speakers; and it can be used as speaker models for both text-dependent and text-independent identifications. Under the text-dependent condition, this modeling method worked as well as others. The recognition rate reached 100% for 15-second-length testing speeches. However, the performance degraded for text-independent condition. Only 80% recognition rate was achieved for 15-second-length speeches in the experiments.

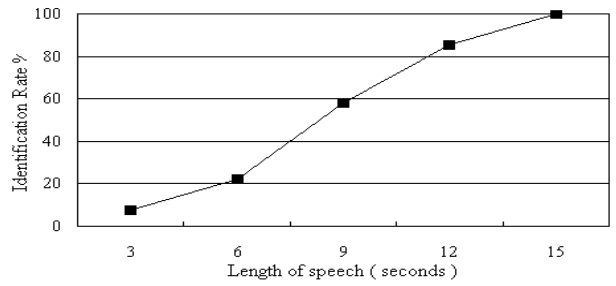

Fig. 6 Influence of speech length on identification rate codebook size = 256

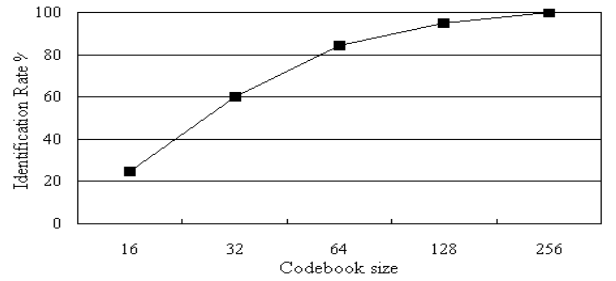

Fig. 7. Influence of codebook size on identification rate speech length = 15s

Figs. 6 and 7 illustrate the influence of codebook size and speech length on the identification rate under the text-dependent condition, respectively. As it was expected, the identification rate increased with respect to the codebook size and the speech length.

The experimental results verified our consideration that the statistical distribution of prediction error vectors could be used as models to represent patterns for recognition applications

6. CONCLUSIONS

A modeling method that uses statistical distribution of prediction error vectors to represent patterns for time series signals was proposed in this paper. The method is not restricted to any specific signals. We verified it by speaker identification application. Under the test-dependent condition, its performance is as good as other modeling methods such Hidden Markov Models, Neural Networks and so on. Currently, we are working on a facial expression recognition project by using this method to create expression models.

REFERENCES

[1] Anil K. Jain, Robert P.W. Duin, and Jian-chang Mao,Statistical Pattern Recognition: A Review. *IEEE Trans. on Pattern Analysis and Machin Intelligence*, pp.4-37, Jan 2000

[2] A.K. Jain, J. Mao, and K.M. Mohiuddin, ™Artificial Neural Networks: A Tutorial,º *Computer*, pp. 31-44, Mar. 1996.

[3] L. Devroye, L. Gyorfi, and G. Lugosi, A Probabilistic Theory of Pattern Recognition. Berlin: Springer-Verlag, 1996.

[4] K.S. Fu, Syntactic Pattern Recognition and Applications. Englewood Cliffs, N.J.: Prentice-Hall, 1982.

[5] T. Matsui and S. Furui, " Speaker recognition technology," *NTT Review*, 7(2), pp. 40-48, 1995.

[6] Gersho A., Gray R.M.: Vector Quantization and Signal Compression. (Dordrecht: Kluwer Academic Publishers, 1992).

[7] Linde Y., Buzo A., Gray R.M.: An algorithm for vector quantizer design. *IEEE Trans. on Communications*, 28(1): 84-95, January 1980.

[8] Deller Jr. J.R., Hansen J.H.L., Proakis J.G.: *Discrete-time Processing of Speech Signals*. Macmillan Publishing Company, New York, 2000.

[9] F. K. Soong et al. "On the Use of Instantaneous and Transitional Spectral Information in Speaker Recognition", *IEEE Trans. Speech and Audio Proc.*, Vol. SAP-36, No.6, pp.871-879, June 1988