Speaker Recognition Based on Source Coding Approaches

Biing-Hwang Juang and F. K. Soong

AT&T Bell Labs, Murray Hill, NJ 07974

ABSTRACT

Recent developments in source coding algorithms allow efficient encoding/quantization of speech signals. When used as classifiers, these source coders have the potential of offering high performance in speaker recognition tasks as previously demonstrated in a vector quantizer based system [1]. In this paper, we study the use of non-memoryless source coders in speaker recognition problems and discuss the effects of source variations, including speaking inconsistency and channel mismatch, in source coder designs for the intended application. It was found that incorporation of memory in source coders in general enhances the speaker recognition accuracy but more remarkable improvements can be accomplished by properly including potential source variations in the coder design/training. Our experiment with a 100-speaker database showed a 99.5% recognition accuracy, the best reported performance on the particular set.

1. Introduction

The task of automatic speaker recognition is to identify an unknown talker in a (registered) population or to verify the identity of a talker as claimed, through processing of the speech signal uttered by the talker. Applications of automatic speaker recognition range from access control to credit card verification/authorization. In some applications such as premises access control, the speech signal may be processed right at the control point while in applications like telebanking, it needs to be transmitted through some telephone network before the verification processes take place.

The design principle of a speaker recognizer is essentially that of a pattern recognizer and many known techniques are applicable. Nevertheless, a number of factors make recognition of a speaker a unique problem. It is well known that the physical condition of the talker, such as having a cold, can greatly affect the performance of the system. When recorded over long periods of time (e.g. months/years), human speech often change considerably [2]. Also, people often do not speak consistently due to their psychological state of mind. Finally, the ambient characteristics of the system, including the acoustic background or the telephone channel when remote processing is required, will also affect the speech characteristics. These difficulties make the speaker recognition problem an interesting, albeit difficult, one.

In this paper, instead of addressing the general problem of talker recognition, we investigate one particular methodology, namely source coding, with the hope that it will provide a solid framework for devising an effective talker recognition system.

This work is motivated by the recent success in applying vector quantization techniques (VQ) [3] to talker recognition [1]. In a basic VQ-based talker recognition technique [1], each speaker is characterized by a VQ codebook constructed from a large set of short time spectral vectors obtained from a series of training utterances provided by the talker. During recognition, the unknown talker's utterance is vector-quantized with the same codebooks and the resultant average distortions are used as the dissimilarity measure upon which the recognition decision is based.

The vector quantization technique used in [1] was of a memoryless type, meaning the processing of the current short time spectral vector is independent of other spectral vectors in the utterance. Source coding techniques, such as VQ with memory, provide designs that take into account the inherent memory (inter-frame dependence) in the signal to enhance the coding efficiency. Since speech is not purely random, incorporating the memory characteristics in the codebook representation of the talker may be advantageous in speaker recognition problems. Therefore, it is one purpose of this paper to report the results of using certain source coders with memory for speaker recognition.

As mentioned earlier, the channel through which the speech passes before being processed also affects the signal characteristics of the talker. The telephone network is one such source of distortion. In addition, there is the issue of lack of consistency of speech from a single speaker over time. This leaves us with the serious question of how to design robust source coders, *to be used as classifiers*, that can work reliably in spite of the known inconsistency in the source characteristics. In particular, we shall address the problem of vector codebook training in terms of maintaining reliable classification when source inconsistency is the issue. Furthermore, the aforementioned memory structure inherent in speech signals needs to be reexamined in the presence of varying channel characteristics.

In the next section, we discuss three basic source coder design algorithms that were used in the study. These are the block (n-tuple) vector or matrix quantizer [4], the pruned trellis vector quantizer [5] and the original memoryless vector quantizer [3] for the sake of comparison.

2. Source Coder Design

Consider a signal source whose output is observed at regularly paced time t. We denote this observation by x_t. In speech processing, x_t often represents a spectrum in terms of some parameter vector. Let $C = \{y_i\}_{i=1}^{N}$ be a set of reproduction codewords (e.g. spectral vectors) and $d(x, y)$ be a prescribed distortion measure between the source observation, x, and the reproduction codeword, y. Source coders attempt to encode x_t by the index i of a certain codeword $y_i \in C$, chosen so as to minimize $E[d(x_t, y_i)]$ where E denotes expectation. The design objective of source coders is then to find the best reproduction codeword set C to achieve the minimum expected distortion.

Practical source coder designs often rely on a (large) training set $\{x_t\}_{t=1}^{N}$, for the source distribution is usually unknown. The expected distortion, in this case, is replaced by an average distortion over the entire training set. In the current study, we consider three types of source coders

CH2847-2/90/0000-0613 $1.00 © 1990 IEEE

commonly used in speech processing. These are the memoryless vector quantizer, the matrix quantizer and the trellis vector quantizer.

2.1 Memoryless Vector Quantizer

The original vector quantizer design based on the generalized Lloyd algorithm [3] is essentially memoryless; i.e. the encoding/quantization of the current vector x_t is independent of other observations. The codebook $C = \{y_i\}_{i=1}^{N}$ is designed to minimize

$$\bar{D} = \frac{1}{T} \sum_{t=1}^{T} d(x_t, \hat{x}_t) \tag{1}$$

where

$$\hat{x}_t = \arg \min_{y_i \in C} d(x_t, y_i) . \tag{2}$$

The combination of the minimum distortion rule of (2) and the generalized Lloyd algorithm always leads to a good quantizer (at least a fixed point solution) for the training sequence. The memorylessness comes from the fact that $\hat{x}_t$ is chosen independent of other x_τ, or $\hat{x}_\tau$ at any different time τ.

2.2 Matrix Quantizer

Matrix quantizers [4] in their very simplest form can be designed using the same VQ design algorithm. If we encode k vectors at the same time, the codebook $C = \{Y_i\}_{i=1}^{N}$ is then designed to minimize

$$\bar{D} = \frac{1}{T-k+1} \sum_{t=1}^{T-k+1} d'(X_t, \hat{X}_t) \tag{3}$$

where

$$X_t = [x_\tau]_{\tau=t}^{t+k-1} \tag{4}$$

and

$$\hat{X}_t = \arg \min_{Y_i \in C} d'(X_t, Y_i) . \tag{5}$$

Simultaneous (block) quantization of a sequence of vectors, as defined by (4), implies that the codewords Y_i incorporate certain block memory constraints. The distortion d' of (3) and (5) could be defined more generally for the matrix pair X and Y but we often use a sum of individual vector distortions for simplicity.

2.3 Trellis Vector Quantizer

A trellis vector quantizer is a finite state vector quantizer [5], specified by a finite state space $\mathcal{S}$, an initial state s_0, and three functions: 1) an encoder α: $\mathcal{A} \times \mathcal{S} \rightarrow \mathcal{N}$ where $\mathcal{A}$ denotes the observation space and $\mathcal{N}$ is a finite alphabet, e.g. the index set; 2) a next state function or transition function f: $\mathcal{S} \times \mathcal{N} \rightarrow \mathcal{S}$; and 3) a decoder β: $\mathcal{S} \times \mathcal{N} \rightarrow \hat{\mathcal{A}}$ where $\hat{\mathcal{A}}$ is the space of reproduction vectors. During encoding, the encoder α encodes the input x_t to a $u_t \in \mathcal{N}$ based on the status of the current state s_t; i.e. $u_t = \alpha(x_t, s_t)$. The state advances according to $s_{t+1} = f(s_t, u_t)$, which is reproduced at the decoder. Upon receiving u_t, the decoder reconstructs x_t by $\hat{x}_t$ based on the function $\hat{x}_t = \beta(s_t, u_t)$. The encoder always has a copy of the decoder so as to accomplish the minimum distortion requirement:

$$u_t = \alpha(x_t, s_t) = \arg \min_{u \in \mathcal{N}} d(x_t, \beta(s_t, u)) . \tag{6}$$

The codebook C and the next state function f are designed to minimize

$$\bar{D} = \frac{1}{T} \sum_{t=1}^{T} d(x_t, \beta(s_t, \alpha(x_t, s_t))) . \tag{7}$$

For a given next state function f, (7) is quite similar to (1) and (3) and the codewords can be designed using the well known Lloyd algorithm. For the design of the next state function, f, we follow the pruning method of [5]. The pruning method starts with a regular memoryless VQ codebook design. It then constructs the trellis, which defines the next state function in terms of allowable search range of the codewords, by pruning non-essential transitions. Several advantages of the method were discussed in [5]. The memory structure of the coder is realized in the way the trellis is constructed and is of a sequential nature as opposed to the block constraints in the matrix quantizer case.

3. Speaker Recognition Based on Source Coding

As explained above, there exist basically three issues in source coding: design of the coder (codewords), minimum distortion search, and assessment of performance as measured by a prescribed distortion function. Given a source coder q which includes the search mechanism and the codeword set and structure, encoding of an input sequence, say $\mathbf{z} = \{z_t\}_{t=1}^{L}$, results in an average distortion

$$D_q = \frac{1}{L} \sum_{t=1}^{L} d(z_t, \hat{z}_t) \tag{8}$$

which is reminiscent of (1), (3) and (7). Alternately, we can view the performance figure D_q as the dissimilarity between an input sequence $\mathbf{z}$ and the source, represented by q, of which the codebook C_q and the encoding mechanism are two vital components. For speaker recognition, we exploit this dissimilarity to differentiate an unknown speaker from a known source.

In a speaker identification mode of operation, an utterance from an unknown talker is to be identified as one from M pre-registered talkers. Initially, for each of the M speakers in the pool, an individual source coder q_i, $i = 1, 2, ..., M$ is designed. During recognition, the input sequence $\mathbf{z}$ is encoded/quantized by each of the M source coders, resulting in

$$D_{q_i} = \frac{1}{L} \sum_{t=1}^{L} d(z_t, \hat{z}_t^{(i)}) \tag{9}$$

where $\hat{z}_t^{(i)} \in C_{q_i}$, the codebook pertaining to encoder q_i, $i = 1, 2, ..., M$. The recognizer then determines that $\mathbf{z}$ is spoken by talker j if

$$D_{q_j} = \min_i D_{q_i} . \tag{10}$$

In the current study, we use the likelihood ratio measure [5] as the prescribed distortion function. Other measures are known to produce better recognition performance [6] but since our goal is to compare the three source coding methods, it is not essential that the performance for each system be the best known.

4. Database and Analysis Conditions

The database used to test the speaker recognition capabilities of the various source coders consists of a total of 20,000 isolated digit utterances spoken by 100 speakers, 50 male and 50 female. The utterances were recorded over dialed-up local telephone lines using an ordinary telephone handset with a carbon microphone. The speakers were seated in a sound booth so that there was virtually no spurious acoustic background noise in the recording. The recording of the 200 utterances from each talker was completed in five sessions held over a period of up to three months. In each recording session, each talker uttered the 10 digits in 4 randomized sequences.

The characteristics of the recording channel, which included the microphone, the telephone line as well as the A/D conversion circuitry, were not measured nor compensated for. This means that the recorded speech signal contains variations inherent in both the talker and the channel.

The speech signal was bandlimited to 200-3200 Hz and digitized at a 6.67 kHz rate with a 16-bit linear A/D converter. (The signal was not specifically adjusted to maximize the precision of the digitized results.) The sampled speech was then preemphasized by a first order differentiator $1-0.95z^{-1}$, followed by an 8th order LPC analysis. The autocorrelations were computed every 15 msec using overlapping 45 msec Hamming windows. These autocorrelation vectors (of dimension 9) were then normalized by the corresponding residual energy, as appropriate for the calculation of the likelihood ratio distortion.

5. Experiments and Results

As discussed in Section 1, our goal is to study the effects of different source coder designs and of speaker/source consistency as reflected in the training data to design various coders. These two issues can be studied experimentally by constructing two different training sets representing the source for each speaker. The first set consisted of the 80 utterances recorded during the first two sessions. Since there were 100 speakers, the total number of such training sets is 100. Note that even though these different speaker utterances were designated by the same session numbers, they were not recorded at the same time for each speaker. We call this training set A for convenience. The second set was formed by grouping together the first half of each recording set, i.e. 20 utterances, from the first 4 sessions, again resulting in 80 training utterances for each speaker. This second training set has the same number of tokens as the first set but is richer in its sampling of talker and channel variability over times since it includes data from 4 sessions. We shall call the second training set B again for convenience.

The rest of the data, 120 utterances from each speaker, was then used as the test set. The following recognition (identification) results all pertain to this open test condition.

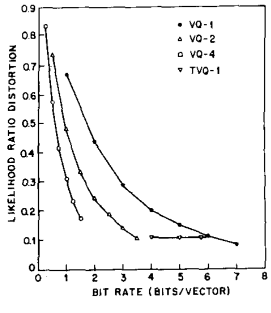

Fig. 1 Source coder performances for a typical speaker in set A.

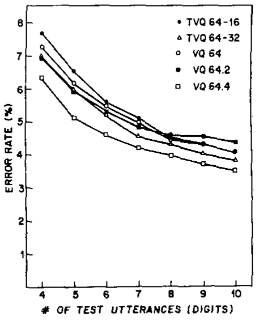

Fig. 2 Speaker identification error rate for different test token lengths in set A.

5.1 Training Set A

The three sets of source coders, as described in Section 2, were designed for each speaker. We shall use VQ-1, VQ-2, and VQ-4 to designate the traditional memoryless vector quantizer, the 2-tuple matrix quantizer, and the 4-tuple matrix quantizer respectively. We use TVQ-1 to denote the trellis vector quantizer as it also encodes 1 vector at a time. Fig. 1 shows the distortion-rate curve for a typical speaker for the three types of source coders. The performance of these source coders were measured in terms of the likelihood ratio distortion [5]. The matrix quantizer outperforms the memoryless vector quantizer in that for a given bit rate it always achieves a lower distortion. The trellis vector quantizer was designed based on a 64-entry memoryless VQ codebook. Similar to what was reported in [5], the performance of TVQ-1 did not degrade as the trellis was pruned from 64 (6-bit) to 16 (4-bit), i.e. from 64 codebook entries at each state to 16 codebook entries at each state.

For speaker recognition experiments, the following coders were used: 1) 64-entry VQ-1 (VQ 64), 2) 64-entry VQ-2 (VQ 64.2, rate=3 bits/vector), 3) 64-entry VQ-4 (VQ 64.4, rate=1.5 bits/vector), 4) 32-branch TVQ-1 with 64 codewords (TVQ 64-32, rate=5 bits/vector), and 5) 16-branch TVQ-1 with 64 codewords (TVQ 64-16, rate=4 bits/vector). The recognition (identification) was based on the decision rule of (10). Fig. 2 shows plots of the performance of these coders in terms of the recognition error rate versus the number of digit utterances used in each recognition trial, ranging from 4 to 10. As seen from the figure, the speaker recognition performance always improved when some memory constraints were incorporated in the source coder design. These memory constraints may be representative of a speaker's particular speaking characteristics. The block type of memory constraints, as implied in VQ 64.4, appears to be the most effective among all coders tested.

5.2 Training Set B

The same experiment was run using training set B and the corresponding test data. The distortion performances of the various coders are shown in Fig. 3. Even though training set B is thought to have richer session variations, the distortion performances of the source coders so designed were not significantly different from those obtained with training set A.

The speaker recognition performance in this case is, however, very different from the previous result as is seen from Fig. 4. For the trellis vector quantizers, the recognition

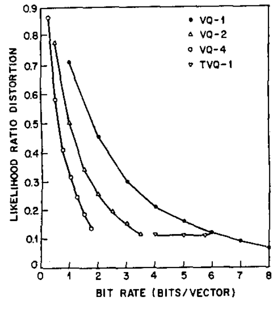

Fig. 3 Source coder performances for a typical speaker in set B.

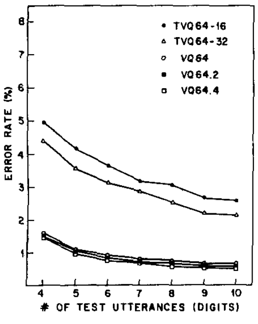

Fig. 4 Speaker identification error rate for different test token lengths in set B.

error rate was reduced from 4% to around 2.5% with 10 test utterances. Most dramatically, the error rate of VQ-1 was reduced from 4% to 0.68% again using 10 test utterances, an almost 6 fold reduction in error rate. There is about a half percent difference in error rate between TVQ 64-32 and TVQ 64-16, but the recognition performances of VQ 64, VQ 64.2 and VQ 64.4 designed on training set B are virtually identical with minor improvements due to incorporation of memory.

6. Discussion

The above results clearly demonstrate the effects of both incorporating memory in source coder designs and the inherent source variability on the task of speaker recognition. Furthermore, they also show how different source coders can handle the types of source variations in the data. To illustrate this point, Fig. 5 shows the recognition error rate from using TVQ 64-16 and VQ 64.2 for the two training sets A and B respectively, again as a function of the number of test utterances used. The test set for these results came from only the last session ("out-of-session") of 40 utterances. Therefore, it is an entirely open (independent) test as the recording conditions in the last session were not included in either of the two training sets. There appears to be little difference between the two cases of TVQ 64-16 for training set A and B respectively. The source coder VQ 64.2, designed for source set A, also performed similarly with only small degradation at a high number of test utterances. The coder VQ 64.2 designed for source set B, however, achieved a much better performance than any of the other three source coders. With 7 test utterances used in each trial, the recognition error rate was only 2% and with only 3 test utterances, its performance was almost the same as the other coders at 10 test utterances per trial. The trellis in the trellis vector quantizer is thus seen to contribute little towards discrimination of speaker characteristics, particularly when the source contains inevitable variability.

When the source is more homogeneous, as in the case of training set A, the memory factor incorporated in the TVQ or matrix quantizers does help improve the recognition rate. This can be seen from the results shown in Fig. 2 and 4. However, when we examine the various performance scores, we also have to consider the complexity of each coder. The trellis vector quantizer, when compared with the traditional memoryless VQ, has a reduced search complexity (also lower encoding rate) but incurs a small increase in storage for the accompanying trellis. The matrix quantizer, or *m*-tuple VQ, nevertheless, requires *m* times the storage and computation of a corresponding memoryless VQ. Since we are not interested in coding rate in speaker recognition tasks, it seems difficult to justify such an increase in complexity, particularly when the performance improvement in recognition error rate is small compared to the memoryless VQ design.

Inclusion of as much speaker source variation as possible in training is seen to be the most important factor in speaker recognition tasks based upon source coding approaches.

7. Summary

We addressed in this paper two main issues associated with speaker recognition problems based on source coding approaches. These two issues are the effects of incorporating source memory in the coder design for speaker recognition and the source inconsistency due to speaker or channel variations. We investigated three different kinds of vector quantizers, namely, the traditional memoryless VQ, the matrix quantizer, and the trellis vector quantizer respectively. It was shown that, in general, incorporation of source memory in the coder design enhances the discriminating power of the source coders used as classifiers. The improvement, however, was only moderate compared to what could be achieved when the coders were designed explicitly to include as much source variation as possible. The vector quantizer design without the separate trellis memory constraints is particularly attractive in its discrimination capacity. With proper training and inclusion of source variations, a vector quantizer (with or without memory) was shown to be able to achieve a recognition accuracy of 99.5% for a 100-speaker database. This is the best result reported so far on this particular database.

REFERENCES

1. F. K. Soong, A. E. Rosenberg, L. R. Rabiner, and B. H. Juang, "A vector quantization approach to speaker recognition," *AT&T Technical Journal*, vol. 66, pp. 14-26, 1987.
2. J. D. Markel, B. Oshika, and A. H. Gray, Jr., "Long-term feature averaging for speaker recognition," *IEEE Trans. Acoust. Speech Signal Processing*, vol. ASSP-25, pp. 330-337, Aug. 1977.
3. Y. Linde, A. Buzo and R. M. Gray, "An algorithm for vector quantization," *IEEE Trans. on Communications,* COM-28, pp. 84-95, 1980.
4. D. Y. Wong, B. H. Juang, and D. Y. Chang, "Very low data rate speech compression with LPC vector and matrix quantization," *Proc. ICASSP-83*, pp. 65-68, Apr. 1983.
5. B. H. Juang, "Design and performance of trellis vector quantizers for speech signals," *IEEE Trans. Acoust. Speech Signal Processing*, ASSP-36, pp. 1423-1431, Sept. 1988.
6. 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*, 22, 143-157, 1989.

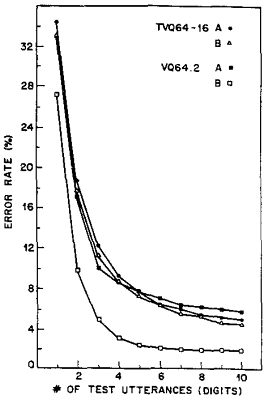

Fig. 5 Comparison of speaker recognition performance for set A and B.