TIME-FREQUENCY VECTOR QUANTIZATION WITH APPLICATION TO ISOLATED WORD RECOGNITION

F. Rogers, P. Van Aken and V. Cuperman*

School of Engineering Science, Simon Fraser University, Burnaby, BC, Canada V5A 1S6
Tel: (604) 291-4371, Fax: (604) 291-4951, email: fmrogers@cs.sfu.ca, vladimir@cs.sfu.ca
* OMNI Microelectronics, Inc., 2620 Augustine Drive, Suite 130, Santa Clara, CA, USA 95054

ABSTRACT

In low-complexity speaker-independent isolated word recognition systems based on Vector Quantization (VQ) with multiple codebooks, the performance of the VQ has a big impact on the overall performance of the system. This paper introduces two techniques for combining spectral and temporal information in the VQ process, with the objective of improving the recognition performance, while maintaining or decreasing the storage requirement. The proposed techniques are compared to an existent method based on the probability distribution of the time of occurrence of spectral vectors in the quantization process.

The experimental results show that the proposed methods improve significantly the recognition performance and have similar or lower memory requirements to the reference method.

1. INTRODUCTION

Vector Quantization (VQ) has been previously used in low complexity speaker-independent isolated-word recognition for representing the spectral content of each dictionary word in a small size VQ codebook. In such a system, the recognizer has a number of VQ codebooks equal to the number of dictionary utterances.

Initially, the recognition was based on selecting the word corresponding to the codebook whose average spectral distortion with respect to the input token was minimum [1, 2]. Such an approach does not use any temporal information, i.e., is based on the occurrence of given spectral shapes in the input without using the information regarding the time of occurrence.

A procedure for incorporating the temporal information by subdividing each input word into a small number of non-overlapping regions and using a separate codebook for each region was proposed in [3]. This approach will be called below the segmented-codebook approach. Finally, a technique based on combining the spectral distortion with a temporal distortion using an estimated probability density function (pdf) of the time of occurrence on a normalized time scale was introduced in [4]. In this latter approach each codevector has an associated probability table which gives the estimated probability of occurrence of the codevector at a given normalized time.

In this paper we propose two alternative ways of incorporating the time information in the VQ codebook. In both cases the time information is incorporated directly into the codebook with the objective of generating a time-frequency characterization of the word. The proposed approaches are compared to the approach based on probability tables.

In the first approach, each word is represented by a codebook having codevectors which include spectral components and time components. The codebook is searched using a weighted Euclidean distance applied to the log-spectral components and to the time components. Both the spectral and the time components are obtained through a joint training process. In our experiments, the approach based on time components obtained better recognition results and lower memory complexity than the probability tables approach.

In the second approach, the time information is built implicitly into the codebook by training each codevector with input vectors corresponding to a given normalized time range. Each time-normalized section of the input is represented by a set of neighboring codevectors called sub-codebook and sub-codebooks are overlapped to a variable degree. This technique is a generalization of the segmented-codebook approach and achieves better time resolution without increasing significantly the required memory.

2. TIME-FREQUENCY VECTOR QUANTIZATION

Assume the input utterance is linearly time normalized to a fixed length L. The time-normalized utterance can be represented as a sequence of vectors $\underline{x}_i = (\underline{x}, i)$

where $\underline{x}$ is a spectral vector with components at log scale and i is the normalized time index (time of occurrence), $i = 1, 2, ..., L$. Assuming for the beginning only one time component per spectral codevector, the codevectors are of the form $(\underline{y}_k, t_k)$ where t_k is the time component (real number in the range 1 to L). The time component t_k represents the expected time of occurrence of the spectral vector $\underline{y}_k$ in the particular word for which the codebook is trained.

The distortion measure used in searching the codebook is given by

$$d(\underline{x}_i, \underline{y}_k) = ||\underline{x} - \underline{y}_k||^2 + \frac{{\sigma_{sk}}^2}{{\sigma_{tk}}^2}(i - t_k)^2 \qquad (1)$$

where σ_{sk} and σ_{tk} are the spectral and temporal variances estimated in the training process for cluster k. A block diagram of the codebook training procedure is shown in Figure 1. The codebook training is done in two steps: first the spectral components are trained independently of the time components, then the time components are trained with the clustering based on the spectral components only. The spectral and temporal variances are computed for each spectral cluster during the training process.

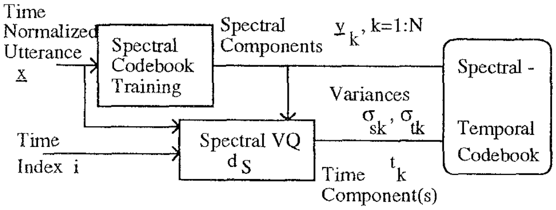

Figure 1: Training Procedure for a Codebook with Time Components

In a straightforward generalization of the above procedure, we considered the case where each spectral codevector has assigned a number M_k of time components representing different expected occurrence times. The index k indicates that the number of time components may be different for each spectral codevector. Intuitively, this corresponds to the case when a given spectral shape appears at different normalized times in a given utterance. For the case of multiple time components, the distortion measure used in search is based on the time component which is "closest" to the normalized time of occurrence of the input vector. Each time component has an associated variance estimated during the training process.

A block diagram of an isolated-word speaker-independent recognizer using spectral-temporal codebooks is shown in Figure 2. Each dictionary utterance is represented by a spectral-temporal codebook and the input utterance is recognized by selecting the minimum spectral-temporal distortion D_j, $j = 1, 2, ..., V$, where

$$D_j = \sum_{i=1}^{L} \min_{\underline{y}_k \in C^j} d(\underline{x}_i, \underline{y}_k) \qquad (2)$$

The previous equation shows that D_j is a distortion computed between the input utterance and the codebook with index j. The distortion with respect to the codebook C^j is obtained here by accumulating (over the entire duration of the utterance) the distortions (1) computed between each spectral-temporal vector in the input utterance and the "closest" codevector in the codebook C^j.

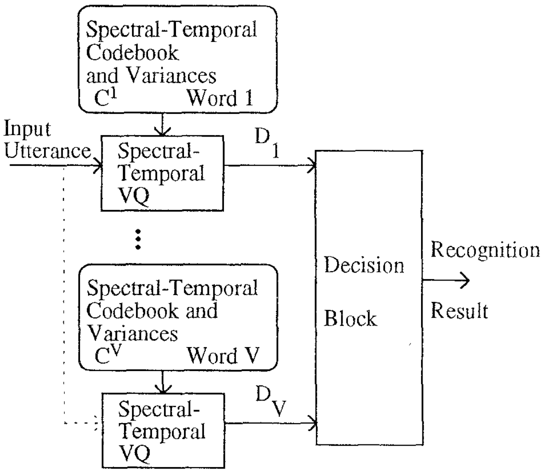

Figure 2: Recognition System with Time Components

The time component approach presented above is equivalent to a compact representation of (partial) temporal pdf information integrated within the codebook. A time component requires at most 3 parameters (component value and two variance values) and in our experimental system even in the multiple time-component system the average number of time parameters per spectral vector is only about 4. On the other hand, the probability table representation requires L parameters per codevector, where typically $L = 40$. Despite the significant reduction in the memory requirements provided by the compact time-component approach, as it will be seen in the last section, the performance improves with respect to the probability tables technique. This indicates that the distortion measure defined by (1, 2) is more efficient than the distortion measure used in the probability table approach (the temporal distortion based on the scaled log-probability as described in [4] was used for the probability tables approach).

3. OVERLAPPED CODEBOOKS

In this approach the temporal information is built implicitly into codebooks by defining a search space for each input vector $\underline{x}_i$ consisting of codevectors $\underline{y}_k$ with indices in the interval $k_{i,min} \leq k \leq k_{i,max}$. These codevectors form a sub-codebook and the sub-codebooks for different neighboring indices are overlapped.

A simple example of overlapped codebooks is given in Figure 3. The numbers enclosed in parentheses in Figure 3 represent the sub-codebooks used for a given utterance segment - the segment 2 is processed using the sub-codebooks 2, 3, and 4. The codebook training is based on the fact that a codevector $\underline{y}_k$ is accessed during the search by input vectors with the normalized time indices in the range $i_{k,min} \leq i \leq i_{k,max}$, and hence should be trained only by these input vectors. The interval limits for training can be determined easily based on the interval limits used for search.

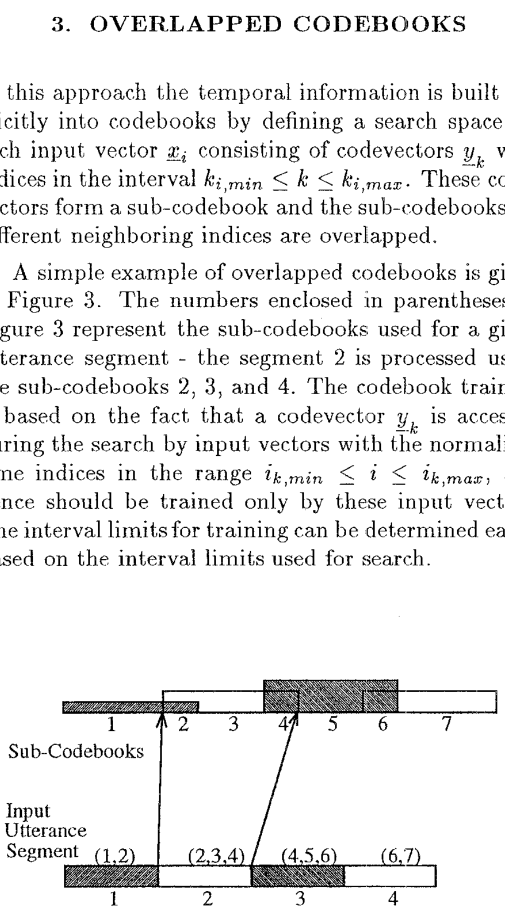

Figure 3: VQ with Overlapped Codebooks

We believe that this structure of the VQ codebook is a natural way to represent an input utterance for two reasons. First, linear time normalization results in an imperfect temporal match and as a consequence, a given spectral shape may appear at a range of normalized times in different repetitions of the same utterance. Second, during a local-stationary segment, similar spectral shapes may appear for a range of time-normalized indices.

4. EXPERIMENTAL RESULTS

In order to evaluate the performance of the different time-frequency representations, we built an isolated word speaker independent recognizer similar to the word-based pre-processor described in [4]. The feature extraction is based on a non-uniform filter bank with 16 pass-bands covering the frequency range 140Hz to 7200Hz with center frequencies at the Bark scale. The output of each bandpass filter is rectified, low-pass filtered, and resampled to obtain an estimate of the signal's spectral characteristics. Logarithmic compression followed by spectral and then temporal normalization are performed on the spectral vectors. During temporal normalization, the time component is added to the spectral characteristics vector as the 17th component. Each codebook is trained with feature vectors corresponding to a specific word in the recognizer's vocabulary. The test utterance is quantized with respect to each of the codebooks and the output is the index of the minimum distortion codebook.

The baseline recognizer using only the spectral components obtains a performance of about 97% for speakers who contributed to the training data or speakers with similar pronunciations recorded in conditions identical to those used for training. The performance degrades significantly for foreign pronunciations or different recording conditions. To enhance the performance differentiation we used a mixed test set with a total of 92 tokens per dictionary word (for 12 dictionary words) out of which 52 tokens were similar (pronunciation, recording conditions) to the training set and 40 tokens were recorded in different conditions using talkers with English as a second language. On this test set, the baseline recognizer's performance degrades to about 89%. Our objective was to improve this performance by combining spectral and temporal information. As it will be shown below, an improvement in the recognition rate from 89% to about 94% was achieved by using the proposed techniques.

The temporal probability tables method described in [4] was implemented to provide a reference for the comparison of the systems described above. For the optimal value of the parameter α, representing the mix of spectral and temporal distortions (see [4]), the introduction of probability tables reduced the recognition error rate on the test set by about 1.6% resulting in a recognition rate of about 90.7%.

The recognition error rates for the systems presented in this paper along with the required memory in words per dictionary entry is shown in Table 1. The results indicate that using only one time component results in better performance (about 92%) than the prob-

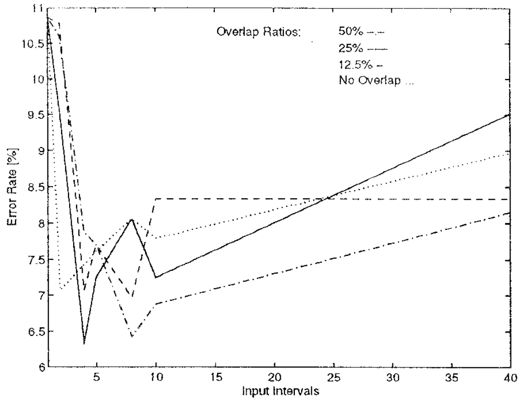

Figure 4: Error Rates for Different Overlap Ratios

VQ Method	Error Rate [%]	Memory / Codeword [KB]
Spectral Info. Only	10.87	1
Probability Tables	9.24	3.5
1 Time Comp. (TC)	8.06	1.06
2 TC	7.52	1.09
Overlapped Cbks	6.34	3.62

Table 1: Error Rates For Recognition Using Spectral Temporal VQ

ability table approach at a lower memory requirement. The multiple time components approach uses a variable number of time components per dictionary word with an average of about 1.5 components per dictionary entry. The result is a further performance improvement to 92.5% at the cost of a negligible increase in memory size. The overlapped codebooks approach shows the best recognition accuracy: 93.7% representing an improvement of about 2.9% with respect to probability tables at the expense of a larger memory requirement.

To obtain the result given in Table 1 for the overlapped codebooks method, a number of tests were performed to measure the recognition rate for a variable number of input intervals (partitions) and for different overlap ratios. The outcome of these tests is shown in Figure 4 and indicates the possible design trade-offs between the number of partitions in the input utterance and the amount of overlap between adjacent sub-codebooks.

5. REFERENCES

[1] K. Shikano, "Spoken Word Recognition Based Upon Vector Quantization of Input Speech," Trans. of Committee on Speech Research, pp. 473-480, December 1982.

[2] J. E. Shore and D. K. Burton, "Discrete Utterance Speech Recognition Without Time Alignment," IEEE Trans. on Information Theory, vol. IT-29 (4), pp. 473-491, July 1983.

[3] D. K. Burton, J. T. Buck, J. E. Shore, "Parameter selection for isolated word recognition using vector quantization", Proc. IEEE International Conference on Acoustics, Speech and Signal Processing, pp. 9.4.1 - 9.4.4, March 1984.

[4] K. C. Pan, F. K. Soong, L. R. Rabiner, A. F. Bergh, "An Efficient Vector-Quantization Preprocessor for Speaker Independent Isolated Word Recognition," Proc. IEEE International Conference on Acoustics, Speech and Signal Processing, April 1985, pp. 874-877.