AT&T Bell Laboratories Technical Journal
Vol. 63, No. 7, September 1984
Printed in U.S.A.

On the Performance of Isolated Word Speech Recognizers Using Vector Quantization and Temporal Energy Contours

By L. R. RABINER,* K. C. PAN,* and F. K. SOONG*

(Manuscript received February 3, 1984)

In this paper we present results of a series of experiments in which combinations of vector quantization and temporal energy contours are incorporated into the standard framework for the word recognizer. We consider two distinct word vocabularies, namely, a set of 10 digits, and a 129-word airlines vocabulary. We show that the incorporation of energy leads to small but consistent improvements in performance for the digits vocabulary; the incorporation of vector quantization (in a judicious manner) leads to small degradation in performance for both vocabularies, but at the same time reduces overall computation of the recognizer by a significant amount. We conclude that a high-performance, moderate-computation, isolated word recognizer can be achieved using vector quantization and the temporal energy contour.

I. INTRODUCTION

The most popular form for an isolated word recognition system is the classic statistical pattern recognition implementation shown in Fig. 1. In this model the speech signal is first analyzed by the feature measurement block, which produces a test pattern consisting of a temporal sequence of (spectral) feature vectors characteristic of the speech sounds in the word. Most typically, the feature measurement system is either a bank of highly overlapping (in frequency) bandpass filters, or a Linear Predictive Coding (LPC) analysis. In either case

* AT&T Bell Laboratories.

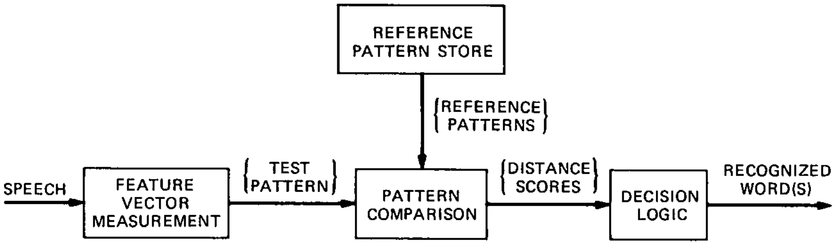

Fig. 1—Pattern recognition model of isolated word recognition system.

the feature vector, for a given time interval, is an estimate of the short-time spectrum of the speech signal. The job of the pattern comparison block is to register, in time, the test pattern with each of a set of stored reference patterns, and to determine a similarity (distance) score for each such pair of patterns. It has been shown that one must use some type of dynamic programming algorithm to achieve the required degree of time alignment for arbitrary word vocabularies.[1] Associated with the time alignment is a spectral distance measure for comparing frames of the test and reference patterns. Such distance measures are often as simple as summing spectral magnitude differences, or as complex as the likelihood ratio measure.[2] The final stage of the pattern recognition model of Fig. 1 is the decision rule that makes a recognition decision, or possibly a set of decisions, based on the distance or similarity scores provided by the pattern comparison block. The most widely used decision rule is the nearest-neighbor rule, which chooses the recognized word as the one whose pattern has the smallest distance score. Alternative decision rules are variants of the K-Nearest-Neighbor (KNN) rule.[3]

A wide variety of isolated word recognizers have been designed, based on the structure given in Fig. 1, and have been shown to yield good performance for several types of word vocabularies and talker sets.[4,5] The major obstacle to the widespread use of such recognizers for simple applications (home computers, terminals, etc.) is the inherent cost of the implementation. This cost, either in terms of computation or actual dollars, is primarily due to the cost of implementing the pattern comparison block with a dynamic programming algorithm. Several alternative recognition structures have been proposed for reducing the cost of the recognizer. These include replacing the nonparametric model of Fig. 1 with a parametric model [e.g., a Hidden Markov Model (HMM)],[6] using recognition structures without time alignment procedures,[7] and using some coding technique on the feature vectors to significantly reduce computation in the dynamic programming algorithm.[8] The first two alternative recognition strategies are still under investigation, but at the present time they yield degraded performance for several standard vocabularies in the speaker-independent mode. The third alternative is the subject of this paper. Shikano has presented some results on trade-offs between computation and performance achievable using coding techniques. We extend his results and apply them to two useful and interesting word vocabularies, and consider their applicability in a speaker-independent mode.

The actual recognition structure used in this paper is an LPC-based system, which uses the likelihood distance metric in a Dynamic Time Warping (DTW) implementation of the pattern similarity block of Fig. 1. The coding technique used to reduce computation in the DTW algorithm is LPC Vector Quantization (VQ).[9,10] The way to reduce computation is to replace each feature vector in the reference pattern by one of a set of fixed LPC vectors from a code book (designed from an appropriate training set). If we similarly replace each feature vector of the test pattern by the closest vector in the code book, then, by precomputing the matrix of distances of each code-book vector to every other code-book vector, the distance computation of the DTW algorithm becomes a simple table-lookup operation. Since the distance computation dominates the overall computation of the recognizer, significant reductions in computation are achieved with this technique. It remains to be shown that performance degradation (due to the distortion introduced by vector quantization) can be kept small.

This paper also discusses the application of temporal energy contours to the recognizer structure of Fig. 1. Previous work by Brown and Rabiner[11] shows that by treating the energy contour (normalized over the entire word duration) as a new feature, and by incorporating this energy feature into the distance metric as an independent, additive feature, performance of the conventional DTW recognizer was improved. Work by Rabiner et al.[12] shows how vector quantization design algorithms can incorporate energy directly into the standard code-book design procedure, yielding a joint quantization of the LPC vector and its energy value. In this paper we integrate both these results into a common framework. We also implement an isolated word recognizer, incorporating vector quantization to reduce computation and using temporal energy contours to achieve performance comparable to that of the DTW system without using either VQ or energy.

The organization of this paper is as follows. In Section II we describe the implementation of the isolated word recognizer using vector quantization and temporal energy contours. In Section III we describe and give results from a series of evaluation tests on two distinct sets of word vocabularies to show the performance of the overall word recognizer in a speaker-independent mode. In Section IV we discuss the results and compare them to those obtained in other studies of computational reduction techniques. Finally, in Section V we summarize our findings.

II. STRUCTURE OF THE OVERALL ISOLATED WORD RECOGNIZER

Figure 2 is a block diagram of the word recognizer incorporating vector quantization of the LPC-feature vectors and using temporal energy contours. The speech signal, recorded off a dialed-up telephone line and digitized at 6.67-kHz rate, is first blocked into 45-ms frames and analyzed to give LPC vectors every 15 ms (100 samples) using the autocorrelation method. A Hamming window is applied to the 45-ms (300 samples) section of speech, which has been preemphasized using a first-order digital network, and a set of $(p + 1)$ autocorrelations are computed. In our implementation we use $p = 8$ poles for the telephone bandwidth signal. The signal energy (unnormalized) for the lth frame of speech is the zeroth-order autocorrelation, $R_l(0)$. We denote the pth-order LPC vector for the lth frame as $\mathbf{a}_l$, and the log energy for the lth frame (after normalization, which will be described later) as $\hat{E}_l$.

The next stage in the processing of Fig. 2 is vector quantization of the LPC vector (with or without the energy parameter). To perform vector quantization, we need a predesigned code book of vectors and an appropriate distance metric for comparing the LPC vectors of the speech signal with the prestored code-book vectors. If we denote an arbitrary test vector as the pair $\hat{\mathbf{a}} = (\mathbf{a}, \hat{E}^T)$, and an arbitrary code-book vector as the pair $\hat{\mathbf{b}} = (\mathbf{b}, \hat{E}^R)$, then an appropriate distance for comparing $\hat{\mathbf{a}}$ and $\hat{\mathbf{b}}$ is[11]

$$d(\hat{\mathbf{a}}, \hat{\mathbf{b}}) = \left(\frac{\mathbf{b}^t V^T \mathbf{b}}{\mathbf{a}^t V^T \mathbf{a}} - 1\right) + \alpha f(|\hat{E}^T - \hat{E}^R|), \tag{1}$$

where V^T is the Toeplitz matrix of autocorrelations of the test frame, α is a suitable weighting factor on the energy distance, and $f(x)$ is a nonlinearity of the type

$$f(x) = \begin{cases} 0, & |x| < E_{\text{LO}} \\ |x| - E_{\text{LO}} + E_{\text{OF}}, & E_{\text{LO}} \leqslant x \leqslant E_{\text{HI}} + E_{\text{LO}} - E_{\text{OF}} \\ E_{\text{HI}}, & |x| > E_{\text{HI}} + E_{\text{LO}} - E_{\text{OF}}, \end{cases} \tag{2}$$

where E_{LO}, E_{HI}, and E_{OF} are appropriately chosen thresholds and energy offsets.

The first term in brackets in eq. (1) is the conventional Itakura LPC likelihood ratio (in its linear form),[2] and the second term is an energy distance that is added to the LPC distance. The weighting factor, α, accounts for the fact that energy distances bear significantly less information than LPC distances. The nonlinear function, $f(x)$,

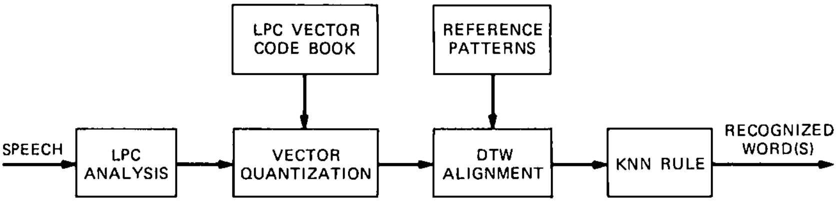

Fig. 2—LPC/DTW recognizer using vector quantization.

accounts for the fact that small energy differences (i.e., less than E_{LO} dB) are insignificant and hence should add no distance, and that large energy distances should be clipped at some appropriate value. Energy distances between these extremes are linearly weighted. In practice, because the Itakura likelihood ratio is essentially unbounded in value, a clipping function, g, is also applied in eq. (1) so that

$$g(x) = \begin{cases} x & x < D_{\mathrm{CLIP}} \\ D_{\mathrm{CLIP}} & x > D_{\mathrm{CLIP}}, \end{cases} \tag{3}$$

where x is the expression in brackets in eq. (1).

A complete specification of the distance metric of eq. (1) requires specification of values for the LPC clipping threshold, D_{CLIP}, and the energy thresholds, E_{LO}, E_{HI}, and E_{OF}. These values are obtained by experimentation in a small pilot test, and specific values will be given in Section III.

Once the distance metric of eq. (1) is given, the implementation of the vector quantization stage of Fig. 2 using a code book with M^* vectors, for the lth feature vector, is a computation of the form

$$d^* = \min_{1 \le m \le M^*} d(\hat{\mathbf{a}}_l, \hat{\mathbf{b}}_m) \tag{4a}$$

$$m^* = \arg\left\{ \min_{1 \le m \le M^*} d(\hat{\mathbf{a}}_l, \hat{\mathbf{b}}_m) \right\} \tag{4b}$$

$$\hat{\mathbf{a}}_l \; => \hat{\mathbf{b}}_{m^*}, \tag{4c}$$

i.e., we find the code-book vector $\hat{\mathbf{b}}_{m^*}$ such that its distance to the analysis vector $\hat{\mathbf{a}}_l$ is minimum over all code-book entries, and we replace $\hat{\mathbf{a}}_l$ by $\hat{\mathbf{b}}_{m^*}$. (Equivalently, all we need to save is the index m^*, which gives the code-book vector with the minimum distance, since the code-book vectors are fixed.)

Once a test feature vector has been vector quantized, and similarly, each reference feature vector has been vector quantized, then the calculation of distance between test and reference feature vectors (as required in the DTW alignment procedure) becomes simply a table-lookup procedure from a table of all possible distances between code-book vectors. Thus, if we define the M^* by M^* matrix of code-book vector distances as

$$D_{CB}(i, j) = d(\hat{\mathbf{b}}_i, \hat{\mathbf{b}}_j), \qquad 1 \leqslant i \leqslant M^*, \qquad 1 \leqslant j \leqslant M^*, \tag{5}$$

and precompute $D_{CB}(i, j)$ and store it, then the distance between a test vector coded by code-book vector i^*, and a reference vector coded by code-book vector j^*, is simply $D_{CB}(i^*, j^*)$, and is computed in the time for a single table-lookup. Hence, the number of distance computations [nominally needing about $(p + 1)$ multiplications and additions] is essentially reduced to zero. In this manner the computation of distance from the DTW alignment is substantially reduced. The technique of vector quantizing both the test and reference patterns, and then using the matrix of distances for a table-lookup computation is called a double-SPLIT VQ by Shikano.[8] For the double-SPLIT method, a full LPC analysis (i.e., the Levinson recursion) does not need to be carried out since the prediction residual is common to all distances in the minimization of eq. (4) and hence, need not be computed.

It should be noted that the process of vector quantization of $\hat{\mathbf{a}}_l$ leads to a distortion error, ϵ_l, given by

$$\epsilon_l = \hat{\mathbf{a}}_l - \hat{\mathbf{b}}_m \cdot \tag{6}$$

since the actual feature vector does differ from the code-book vector. One way of avoiding this distortion in the test feature vector, due to Sakoe,[13] is to save the vector of distances, $\hat{d}(l, m)$, of the form

$$\hat{d}(l, m) = d(\hat{\mathbf{a}}_l, \hat{\mathbf{b}}_m) \tag{7}$$

for all frames, l, of the test pattern, and all code-book indices, m. In this manner whenever a distance is required between the true test feature vector, $\hat{\mathbf{a}}_l$, and a reference vector quantized to code-book vector, $\hat{\mathbf{b}}_q$, then the distance can be looked up in the distance vector for frame l as the qth entry. Thus, we eliminate storage for the M^* by M^* distances of the code-book vectors, but we instead need storage for the M^* by L distances, for a word of L frames, of the vector quantizer. Since $L < M^*$, in most cases, this simplification of the vector quantization generally both increases performance of the recognizer (since no distortion of the test vectors is incurred) and decreases storage of the system. This technique is called a single-SPLIT VQ by Shikano.[8]

The remaining steps in the recognizer of Fig. 2 are essentially those of a conventional DTW-based word recognizer. The DTW alignment compares the test pattern (in some type of VQ format) to each reference pattern (coded as a series of code-book vectors) and generates a distance score. The KNN rule examines the best K scores for each vocabulary word and gives an ordered list of word distances based on the average of the K scores. The "recognized" word is selected as the word whose best-K patterns have the smallest average score.

2.1 Generation of LPC-vector code book

The generation of vectors in the code book of the vector quantizer is straightforward and follows the procedures outlined in Refs. 9, 10, and 12. We used a training set of 39,000 LPC vectors with energy values. The vectors were extracted from isolated digit sequences spoken by 100 talkers (50 male, 50 female). Code books of size $M^* = 2, 4, 8, 16, 32, 64$, and 128 were generated. All results presented in this paper are for code-book size 128. Results, on digit evaluation tests, for smaller-size code books are given in Ref. 12.

2.2 Normalization of the energy contours for words

The raw energy value for the lth frame of a word, E_l, is computed as

$$E_l = 10 \cdot \log_{10}(R_l(0)), \qquad l = 1, 2, \cdots, L, \tag{8}$$

where L is the number of frames in the word. The normalization of energy is performed by finding the maximum energy value, E_{MAX}, over the word as

$$E_{\text{MAX}} = \max_{1 \leq l \leq L} (E_l) \tag{9}$$

and by subtracting E_{MAX} from E_l to give

$$\hat{E}_l = E_l - E_{\text{MAX}}. \tag{10}$$

In this manner the peak energy value of each word is 0 dB, and the recognition system is relatively insensitive to differences in gain between recordings. Of course the computation of eq. (9) means that word energy contour normalization cannot take place until the end of the word is located. This constraint poses no real difficulty since there are ways of implementing an approximate gain normalization in "real time" based on some realistic assumptions about the rate of change of system gain.

III. EVALUATION TESTS OF THE RECOGNIZER

To evaluate the effects of the vector quantizer and the use of energy contours on the performance of the isolated word recognizer, we performed a series of three sets of recognition tests. For the first two sets of tests, the word vocabulary was the ten digits (zero through nine), and for the third set of tests, a 129-word airlines vocabulary was used.[14,15] All tests were conducted in a speaker-independent mode in which all test recordings were made off a standard, dialed-up, local telephone line. A set of 12 speaker-independent reference patterns, obtained from a conventional clustering analysis[16] of 100 tokens per word, were used for each vocabulary word.

We obtained three test sets, denoted TS1, TS2, and TS3, in the following way. TS1 consisted of 100 talkers (50 male, 50 female) who each spoke each digit once. These talkers were the same ones who generated the training tokens used to create both the isolated-digits reference patterns, and the code-book vectors. However, different recordings were used for the training sets than for the test set. The second test set, TS2, consisted of ten talkers (five male, five female) who each spoke each digit 20 times. These talkers were not members of the 100-talker training set and were chosen from a set of 100 talkers used in a large-scale evaluation of a combined digit recognition, talker identification system.[17] The set of ten talkers was chosen on the basis of preliminary experimentation, since they had an error rate somewhat above the average of the 100 talkers used in the experiment. In this manner it was hoped that the TS2 data would amplify differences in test performance results.

The third test set, TS3, consisted of 20 talkers (10 male, 10 female) who each spoke the entire airlines vocabulary a single time. These 20 test talkers were different from those who provided the training tokens used to give the word reference templates.

3.1 Results on digits (TS1)

A series of recognition tests were performed in which temporal energy and vector quantization were tried in all combinations with the basic LPC-based DTW recognizer. A total of six test results are given in Table Ia for the following cases:

Run 1—Standard LPC-based DTW recognizer without energy and without VQ.

Table I—Average digit error rates for the top β word candidates for six runs

Run Number	Energy Used	VQ Ref.	VQ Test	Error Rate (%) for Top β Candidates					
				1	2	3	4	5	6
			(a) TS1 data						
1	No	No	No	2.7	0.8	0.2	0.2	0.2	0
2	Yes	No	No	2.1	0.4	0.2	0.2	0.1	0
3	No	Yes	No	3.2	0.9	0.2	0.1	0.1	0
4	Yes	Yes	No	2.4	0.5	0.1	0.1	0	0
5	No	Yes	Yes	4.0	1.1	0.3	0.2	0	0
6	Yes	Yes	Yes	3.5	0.6	0.1	0.1	0.1	0
			(b) TS2 data						
1	No	No	No	3.6	1.1	0.3	0.1	0	0
2	Yes	No	No	2.8	1.1	0.5	0.2	0.1	0
3	No	Yes	No	3.8	1.2	0.3	0.2	0.1	0
4	Yes	Yes	No	4.2	1.5	0.7	0.3	0.1	0
5	No	Yes	Yes	4.1	1.1	0.4	0.2	0.1	0
6	Yes	Yes	Yes	4.0	2.0	0.9	0.3	0.1	0.1

Run 2—Energy contour used, but no VQ.

Run 3—Energy contour not used and VQ used only on the reference pattern—i.e., this is a single-SPLIT VQ.

Run 4—Energy contour used combined with VQ only on the reference pattern.

Run 5—Energy contour not used and VQ used on both test and reference patterns—i.e., this is a double-SPLIT VQ.

Run 6—Energy contour used combined with VQ on both test and reference patterns.

For each of the energy-based runs, the parameters of the energy distance were set to

$$E_{\mathrm{LO}} = 6(\mathrm{dB}), \qquad E_{\mathrm{HI}} = 20(\mathrm{dB}), \qquad E_{\mathrm{OF}} = 0(\mathrm{dB}),$$

and the LPC distance-clipping threshold, D_{CLIP}, was set to 2.5. (Some experimentation was done with values of $E_{\mathrm{OF}} = 6$ dB, as used in Ref. 12, but results were almost always worse using this parameter setting because of the sensitivity of the DTW path to matching energy contours with the 6-dB energy offset).

An examination of the results given in Table Ia, which gives digit error rates in percent for the top β word candidates ($\beta = 1$ to 6) shows the following:

1. For each of the three consecutive pairs of runs (where each pair differs only in regard to the inclusion of the temporal energy contour), the inclusion of energy reduces the error rate in the top candidate by about 0.6 percent (±0.1 percent).

2. Applying VQ to the reference alone (Runs 3 and 4) increases the average digit error rate in the top candidate by about 0.4 percent. However, using energy, the error rate in the top candidate (2.4 percent) is still below the error rate for Run 1, the standard DTW recognizer without energy or VQ.

3. Applying VQ to both test and reference patterns (Runs 5 and 6) increases the average digit error rate in the top candidate by about 1.3 percent over that for Runs 1 and 2. In these cases the performance is degraded from that of the recognizer without either temporal energy or VQ.

3.2 Results on digits (TS2)

The results for the same six runs using the 2000 digits of TS2 are given in Table Ib. For this set of data the average digit error rate for Run 1 is about 1 percent higher than for TS1 data. This is due to the inclusion of talkers in the database with higher than average error rates. When energy is included in the recognizer (without VQ), the average top candidate error rate falls by 0.8 percent.

The results of Runs 3 and 4 show that using VQ on the reference patterns alone leads to a small increase in error rate (0.2 percent) for the case without energy and a larger increase in error rate (1.4 percent) for the case with energy. In these two cases, the runs using the energy contour provided essentially the same performance as the run without the energy contour.

Runs 5 and 6 show a slight increase in error rate for the case without energy (Run 5 compared to Run 3) and a slight decrease in error rate for the case with energy (Run 6 compared to Run 4). For these two cases, the performance is essentially identical.

A summary set of curves showing the error rate versus candidate position, β, for the best sets of results of Table I is given in Fig. 3, where, for each consecutive pair of runs, we have plotted the best results. These sets of curves show the slight degradations introduced by applying a VQ to the reference alone and to both the reference and test patterns.

3.3 Results on airlines words (TS3)

For the airlines vocabulary a set of four runs were made. These runs correspond to the first four runs on the digits vocabulary. No tests were made with VQ of both test and reference patterns for this vocabulary. The results of the four runs are given in Table II and plotted in Fig. 4.

The results show that using energy contours for this medium-size, complex vocabulary led to essentially no significant improvement in performance for either of the pairs of runs. For the case of no VQ of the references, the performance with energy was 0.2 percent worse than without energy; for the case of using VQ on the references, the performance with energy was 0.5 percent better than without energy.

It can also be seen that using VQ of the references led to a 4- to 4.5-percent increase in error rate for the top candidate and somewhat smaller increases for higher-position candidates. These results indicate that a VQ with 128 code-book entries is just too small for a vocabulary of this size and complexity.

IV. DISCUSSION OF RESULTS

The results presented in Section III showed the following:

1. The addition of the temporal energy contour as an additive feature to the LPC vector generally improved the performance of the recognition system by a small amount. This result was more the case for the digits vocabulary than for the airlines vocabulary.

2. For the digits vocabulary, using VQ on just the reference pattern (the single-SPLIT case) slightly increased the error rate; for the airlines vocabulary a significant increase in error rate occurred, indi-

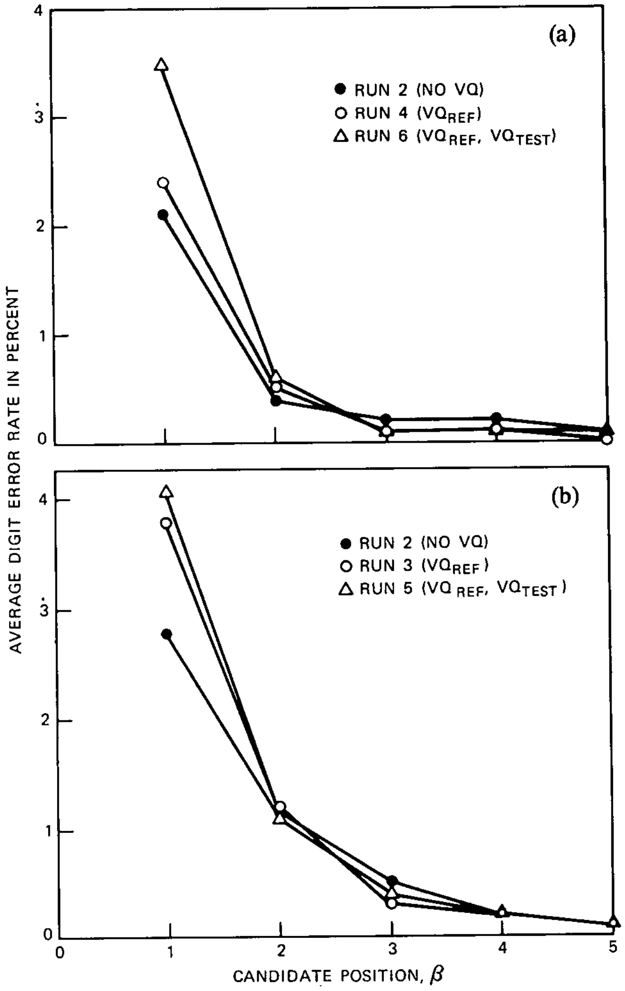

Fig. 3—Average word error rate versus candidate position for the digits vocabulary for (a) TS1 and (b) TS2.

Table II—Average word error rates (percent) for tests on airline vocabulary

Run Number	Energy Used	VQ Ref.	Error Rate (%) for Top β Candidates				
			1	2	3	4	5
1	No	No	10.0	4.5	2.9	2.2	1.6
2	Yes	No	10.2	4.2	2.4	1.9	1.6
3	No	Yes	14.5	6.0	3.5	2.6	2.1
4	Yes	Yes	14.0	6.3	4.0	3.0	2.2

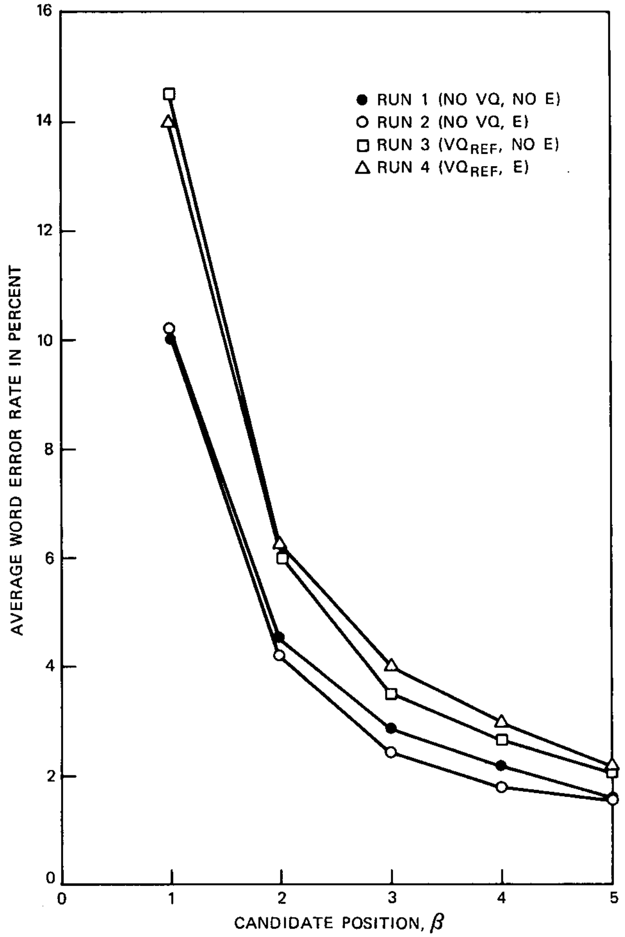

Fig. 4—Average word error rate versus candidate position for the airlines vocabulary.

cating that the size of the VQ was too small for the size and complexity of the vocabulary.

3. For the digits vocabulary, using VQ on both the test and reference patterns (the double-SPLIT case) increased the error rate to a larger degree.

These results indicate that using VQ on just the reference pattern, combined with using the energy contour as an additional feature, can lead to a recognition system with only marginally poorer performance than the system without VQ (at least for the digits vocabulary), and we assert that if a large enough VQ were used for the airlines vocabulary (e.g., M^* on the order of 512), similar results would have been attained.

The basic results are consistent with the findings of Shikano,[8] who studied a speaker-trained system with a large vocabulary (641 words) of Japanese city names, and with earlier work on the digits vocabulary.[12]

If we compare the results reported here to alternative computationally efficient approaches—such as the method proposed by Shore and Burton,[7] or the Hidden Markov Model (HMM) approach[12]—we see that the performance of the DTW approach using a VQ and temporal energy is significantly better than the alternatives. Shore and Burton presented results on a 20-word vocabulary (consisting of the digits and ten control words) for an eight-male talker population and had a 12-percent word error rate; for the digits vocabulary the error rate was still 4.1 percent. Rabiner et al.[12] reported error rates of about 3.5 percent for TS1 data using the HMM approach on a digits vocabulary, and about 15 percent on the airlines vocabulary. Thus, at the current time, since the computation of the LPC recognizer with DTW processing using energy and VQ is comparable to that of alternative approaches, and since its performance is better, it is the most attractive proposal for significant reductions in computation in an isolated word recognizer.

A key issue when using a VQ in the recognizer is the savings in computation over the conventional DTW approach without VQ. To quantify this concept, we define the following terms:

M^* = Number of vectors in code book

V = Number of vocabulary words

Q = Number of reference templates per vocabulary word

$\bar{L}$ = Average number of frames in a word

p = Order of LPC analysis.

For the conventional DTW approach, the computation in each DTW (for each reference pattern) is approximately:

$$\bar{C}_{\mathrm{DTW}} = \bar{C}_{\mathrm{DIST}} + \bar{C}_{\mathrm{COMB}} \tag{11a}$$

$$\bar{C}_{\mathrm{DIST}} = \frac{\bar{L}^2}{3} \cdot (p + 1)(*, +), \tag{11b}$$

where $\bar{C}_{\mathrm{DIST}}$ is the computation for distances in the DTW, and $\bar{C}_{\mathrm{COMB}}$ is the computation for combinatorics. In a serial processor the computation for combinatorics is on the order of one-fifth the computation for distances. Hence, a good approximation is

$$\bar{C}_{\mathrm{DTW}} = \frac{6}{5} \cdot \frac{\bar{L}^2}{3} (p + 1)(*, +) \tag{12}$$

per DTW, or a total of

$$\begin{aligned} C_{\mathrm{DTW}} &= \bar{C}_{\mathrm{DTW}} \cdot V \cdot Q \\ &= \frac{2}{5}\,\bar{L}^2(p+1)V \cdot Q(*,\,+) \end{aligned} \tag{13}$$

to recognize a test word.

Using the VQ in the recognizer leads to a front-end computation load of

$$C_{\mathrm{VQ}} = M^*\bar{L}(p+1)(*,\,+) \tag{14}$$

to code the $\bar{L}$ frames of the test, and a reduction of computation in the DTW to

$$C_{\mathrm{DTW/VQ}} \cong \frac{\bar{L}^2}{15}\,(p+1)V \cdot Q(*,\,+) \tag{15}$$

since all distance computation is eliminated.

The ratio of computation, R, of the DTW with VQ to the DTW without VQ is given as

$$R = \frac{C_{\mathrm{DTW/VQ}} + C_{\mathrm{VQ}}}{C_{\mathrm{DTW}}} \tag{16a}$$

$$= \frac{\dfrac{\bar{L}}{15} \cdot V \cdot Q + M^*}{\dfrac{2}{5}\,\bar{L} \cdot V \cdot Q}. \tag{16b}$$

For the digits vocabulary, with $\bar{L} = 40$, $M^* = 128$, $p = 8$, $V = 10$, and $Q = 12$, we get

$$R_{\mathrm{DIGITS}} = \frac{(128 + 320)}{1920} \approx 0.233,$$

i.e., a 4.3 to 1 reduction in computation. For the airlines vocabulary, we get about a 5.5 to 1 reduction in computation (even if we make $M^* = 512$). Hence, we conclude that the inclusion of VQ in the DTW-based word recognizer can indeed significantly reduce the computation without significantly lowering recognizer performance.

V. SUMMARY

In this paper we show that by adding a vector quantization stage to the standard DTW-based isolated word recognizer, and by incorporating temporal energy as an additional feature to the LPC vector, a high-performance, yet significantly reduced computation word recognizer can be implemented. By using the so-called single-SPLIT methods, in which the VQ is only directly applied to the reference patterns, we show that the resulting VQ distortion can be made sufficiently small such that only an insignificant increase in word error rate results.

REFERENCES

1. G. M. White and R. B. Neely, "Speech Recognition Experiments With Linear Prediction, Bandpass Filtering, and Dynamic Programming," IEEE Trans. Acoustics, Speech, and Signal Processing, *ASSP-24*, No. 2 (April 1976), pp. 183–8.
2. F. I. Itakura, "Minimum Prediction Residual Principle Applied to Speech Recognition," IEEE Trans. Acoustics, Speech, and Signal Processing, *ASSP-23*, No. 1 (February 1975), pp. 67–72.
3. L. R. Rabiner, S. E. Levinson, A. E. Rosenberg, and J. G. Wilpon, "Speaker-Independent Recognition of Isolated Words Using Clustering Techniques," IEEE Trans. Acoustics, Speech, and Signal Processing, *ASSP-27*, No. 4 (August 1979), pp. 336–49.
4. G. R. Doddington and T. B. Schalk, "Speech Recognition: Turning Theory to Practice," IEEE Spectrum, *18* (September 1981), pp. 26–32.
5. L. R. Rabiner and S. E. Levinson, "Isolated Word Connected Word Recognition—Theory and Selected Applications," IEEE Trans. Commun., *COM-29*, No. 5 (May 1981), pp. 621–59.
6. L. R. Rabiner, S. E. Levinson, and M. M. Sondhi, "On the Application of Vector Quantization and Hidden Markov Models to Speaker-Independent, Isolated Word Recognition," B.S.T.J., *62*, No. 4 (April 1983), pp. 1075–105.
7. J. E. Shore and D. K. Burton, "Discrete Utterance Speech Recognition Without Time Alignment," IEEE Trans. Inform. Theory, *IT-29*, No. 4 (July 1983), pp. 473–91.
8. K. Shikano, "Spoken Word Recognition Based Upon Vector Quantization of Input Speech," Trans. Comm. Speech Res., (December 1982), pp. 473–80.
9. Y. Linde, A. Buzo, and R. M. Gray, "An Algorithm for Vector Quantization," IEEE Trans. Commun., *COM-28*, No. 1 (January 1980), pp. 84–95.
10. B. Juang, D. Wong, and A. H. Gray, Jr., "Distortion Performance of Vector Quantization for LPC Voice Coding," IEEE Trans. Acoustics, Speech, and Signal Processing, *ASSP-30*, No. 2 (April 1982), pp. 294–303.
11. M. K. Brown and L. R. Rabiner, "On the Use of Energy in LPC-Based Recognition of Isolated Words," B.S.T.J., *61*, No. 10 (December 1982), pp. 2971–87.
12. L. R. Rabiner, M. M. Sondhi, and S. E. Levinson, "A Vector Quantizer Combining Energy and LPC Parameters and Its Application to Isolated Word Recognition," AT&T Bell Lab. Tech. J., *63*, No. 5 (May–June 1984), pp. 721–35.
13. H. Sakoe, "Device for Recognizing an Input Pattern With Approximate Patterns Used for Reference Patterns on Mapping," U.S. Patent 4, 256, 924, issued March 17, 1981.
14. S. E. Levinson, A. E. Rosenberg, and J. L. Flanagan, "Evaluation of a Word Recognition System Using Syntax Analysis," B.S.T.J., *57*, No. 5 (May–June 1978), pp. 1619–26.
15. J. G. Wilpon, L. R. Rabiner, and A. Bergh, "Speaker-Independent Isolated Word Recognition Using a 129-Word Airline Vocabulary," J. Acoust. Soc. Amer., *72*, No. 2 (August 1982), pp. 390–6.
16. S. E. Levinson, L. R. Rabiner, A. E. Rosenberg, and J. G. Wilpon, "Interactive Clustering Techniques for Selecting Speaker-Independent Reference Templates for Isolated Word Recognition," IEEE Trans. Acoustics, Speech, and Signal Processing, *ASSP-27*, No. 2 (April 1979), pp. 134–41.
17. A. E. Rosenberg, K. L. Shipley, and D. E. Bock, "A Speech Data Base Facility Using a Computer Controlled Cassette Tape Deck," J. Acoust. Soc. Amer., Suppl. 1, *72*, (Fall 1982), p. 580.

AUTHORS

Kok-Chin Pan, S.B. and S.M., 1984 (Electrical Engineering and Computer Science), The Massachusetts Institute of Technology. From 1981 to 1984, Mr. Pan participated in a cooperative program in Electrical Engineering and Computer Science at AT&T Bell Laboratories. He worked on digital circuit design and vector quantization applied to speech recognition. He is currently a Teaching Assistant in the department of Electrical Engineering and Computer Science at MIT. His interests include speech processing and recognition and VLSI system design. Member, Tau Beta Pi.

Lawrence R. Rabiner, S.B. and S.M., 1964, Ph.D., 1967 (Electrical Engineering), The Massachusetts Institute of Technology; AT&T Bell Laboratories, 1962—. Presently, Mr. Rabiner is engaged in research on speech communications and digital signal processing techniques. He is coauthor of *Theory and Application of Digital Signal Processing* (Prentice-Hall, 1975), *Digital Processing of Speech Signals* (Prentice-Hall, 1978), and *Multirate Digital Signal Processing* (Prentice-Hall, 1983). Member, National Academy of Engineering, Eta Kappa Nu, Sigma Xi, Tau Beta Pi. Fellow, Acoustical Society of America, IEEE.

Frank K. Soong, B.S., 1973, National Taiwan University, M.S., 1977, University of Rhode Island, Ph.D., 1983, Stanford University, all in Electrical Engineering; AT&T Bell Laboratories, 1982—. From 1973 to 1975 Mr. Soong served as a teacher at the Chinese Naval Engineering School at Tsoying, Taiwan. In 1982 he joined the technical staff at AT&T Bell Laboratories, where he engaged in research in speech coding and recognition. Member, IEEE.