Isolated Word Recognition Using A Two-Pass Pattern Recognition Approach

L. R. Rabiner
J. G. Wilpon

Bell Laboratories
Murray Hill, New Jersey 07974

Abstract One of the major drawbacks of the standard pattern recognition approach to isolated word recognition is that poor performance is generally achieved for word vocabularies with acoustically similar words. This poor performance is related to the pattern similarity (distance) algorithms that are generally used in which a global distance between the test pattern and each reference pattern is computed. Since acoustically similar words are, by definition, globally similar, it is difficult to reliably discriminate such words, and a high error rate is obtained. By modifying the pattern similarity algorithm so that the recognition decision is made in two passes, improvements in discriminability among similar words can be achieved. In particular, on the first pass the recognizer provides a set of global distance scores which are used to decide a class (or a set of possible classes) in which the spoken word is estimated to belong. On the second pass a locally weighted distance is used to provide optimal separation among words in the chosen class (or classes) and the recognition decision is made on the basis of these local distance scores. For a highly complex vocabulary (letters of the alphabet, digits, and 3 command words) recognition improvements of from 3 to 7 percent were obtained using the two-pass recognition strategy.

I. Introduction

The "standard" pattern recognition approach to isolated word recognition is a 3-step method consisting of feature measurement, pattern similarity determination, and a decision rule for choosing recognition candidates. This pattern recognition model has been applied to a wide variety of word recognition systems with great success [1-3]. However the simple, straightforward approach to word recognition runs into difficulties for complex vocabularies, i.e. vocabularies with phonetically similar words. For example, recognition of the vocabulary consisting of the letters of the alphabet would have problems with letters in the sets $\phi_1 = \{A,J,K\}$, $\phi_2 = \{B,C,D,E,G,P,V,T,Z\}$ etc. In the above case the problems are due to the inherent acoustic similarity (overlap) between sets of words in the vocabulary. It should be clear that this type of problem is essentially unrelated to vocabulary size (except when we approach very large vocabularies), since a large vocabulary may contain no similar words (e.g. the Japanese cities list of Itakura [2]), and a small vocabulary may contain many similar words (e.g., the letters of the alphabet).

It is the purpose of this paper, to propose, discuss, and evaluate a modified approach to isolated word recognition in which a 2-pass method is used. The output of the first recognition pass is an ordered set of word classes in which the unknown spoken word is estimated to have occurred, and the output of the second pass is an ordered list of word candidates within each class obtained from the first pass. The computation for the first pass is similar in nature but often reduced in magnitude from that required for the standard one-pass word recognizer. The computation of the second pass consists of using an "optimally" determined word discriminator to separate words within the equivalence class.

II. The Two-Pass Recognizer

Assume the word vocabulary consists of V words. The i^{th} word, v_i, is represented by word template $\mathbf{R}_i$, $i=1,2,...,V$, where each $\mathbf{R}_i$ is a multidimensional feature vector. Similarly we denote the test pattern as $\mathbf{T}$ (corresponding to spoken word q in the vocabulary) where $\mathbf{T}$ is again a multidimensional feature vector. For simplicity we assume that the pattern similarity and distance computation is carried out using the "normalize and warp" procedure described by Myers et al. [4]. A "standard" word duration of N frames is adopted, and each reference pattern is linearly warped to this duration. We call the warped reference patterns $\tilde{\mathbf{R}}_i$. Similarly the test pattern is linearly warped to a duration of N frames, yielding the new pattern $\tilde{\mathbf{T}}$. A dynamic time warping alignment algorithm then computes the "standard" distance

$$D(\tilde{\mathbf{T}},\tilde{\mathbf{R}}_i) = \frac{1}{N}\sum_{k=1}^{N} d(\tilde{\mathbf{T}}(k),\tilde{\mathbf{R}}_i(w(k))) \quad (1)$$

where $d(\tilde{\mathbf{T}}(k),\tilde{\mathbf{R}}_i(\ell))$ is the local distance between frame k of the test pattern, and frame ℓ of the i^{th} reference pattern, and $w(k)$ is the time alignment mapping between frame k of the test pattern, and frame $w(k)$ of the i^{th} reference pattern.

We define the local distance of the k^{th} frame of the test pattern to the $w(k)^{th}$ frame of the i^{th} reference pattern as $d_i(k)$ where

$$d_i(k) = d(\tilde{\mathbf{T}}(k),\tilde{\mathbf{R}}_i(w(k))) \quad (2)$$

so $D(\tilde{\mathbf{T}},\tilde{\mathbf{R}}_i)$ of Eq. (1) can be written as

$$D(\tilde{\mathbf{T}},\tilde{\mathbf{R}}_i) = \frac{1}{N}\sum_{k=1}^{N} d_i(k) \quad (3)$$

If $\tilde{\mathbf{R}}_i$ corresponds to the correct reference for the spoken word $\tilde{\mathbf{T}}$ (i.e. $i=q$), then we would theoretically expect the local distance $d_q(k)$ to be *independent* of k, with d assuming values from a χ^2 distribution with p (8 for the system we are using) degrees of freedom [2] for the case where the speech features are those of an LPC model and the log likelihood distance measure is used for the local distance. Thus if we plotted $d_q(k)$ versus k, we would expect it to vary around some expected value $\hat{d}$, where

$$\hat{d} = E[d_q(k)] = E[\chi_p^2] \quad (4)$$

An example of a typical curve of $d_q(k)$ versus k is given in Figure 1a.

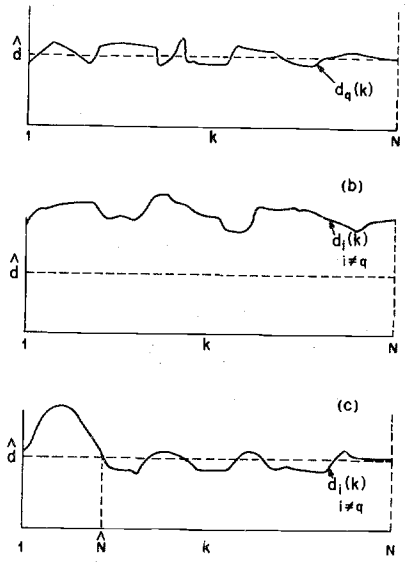

Fig. 1 Curves of $d_i(k)$ versus k for 3 cases.

CH1610-5/81/0000-0724 $00.75

If we now examine typical behavior of the curve of $d_i(k)$ versus k when $i \neq q$, we see that one of two types of behavior generally occurs. When word q is acoustically very different from word i, then $d_i(k)$ is generally large (compared to $\hat{d}$ of Eq. (4)) for *all* values of k, and the overall distance score D of Eq. (3) is large. This case is illustrated in Figure 1b. However when we have acoustically similar words, then generally $d_i(k)$ will be approximately equal to $d_q(k)$ for all values of k in acoustically *identical* regions, and will be larger than $d_q(k)$ only in acoustically dissimilar regions. An example in which the dissimilar region occurs at the beginning of the word (the first N frames) is shown in Figure 1c.

The key point to be noted from the above discussion is that when the vocabulary contains words that are acoustically similar, and one of these similar words is spoken (i.e. it is the test utterance), then the total distance scores for these similar words consist of a random component (due to the variations of $d(k)$ in the similar regions) and a deterministic difference (due to the differences in the dissimilar regions). In cases when the size of the dissimilar region is small (i.e. $N \ll N$ in Fig. 1c), then the random component of the distance score can (and often does) outweight the true difference component, causing a potential recognition error. For highly complex vocabularies (e.g. the letters of the alphabet) this situation occurs frequently.

One solution to the above problem would be to modify the overall distance computation so that more weight is given to some regions of the pattern than others. For example we could consider a weighted overall distance of the form

$$D(\tilde{\mathbf{T}},\tilde{\mathbf{R}}_i) = \frac{\sum_{k=1}^{N} W(k) d(\tilde{\mathbf{T}}(k),\tilde{\mathbf{R}}_i(w(k)))}{\sum_{k=1}^{N} W(k)} \qquad (5)$$

where $W(k)$ is an arbitrary frame weighting function, and the denominator of Eq. (5) is used for distance normalization. The problem with Eq. (5) is that a "good" weighting function is difficult to define since the "optimal" set of weights is clearly a function of the "actually" spoken word (q) and the reference pattern being used (i). Furthermore, any weighting that would help discriminate between acoustically similar words, would tend to hurt the discrimination between acoustically different words.

The above discussion suggests that a reasonable approach would be a 2-pass recognition strategy in which the first pass would decide on an ordering of word "equivalence" classes (in which sets of acoustically similar words occurred), and the second pass would order the individual words within each equivalence class. For the first pass recognition an unweighted (normal) distance would be used, and for the second pass a weighted distance would be used. We now discuss the issues involved in implementing such a two-pass recognizer.

2.1 Generation of Word Equivalence Classes

Given the V vocabulary words $v_1, v_2, \ldots, v_V$, we would like to find a procedure for mapping words into acoustic equivalence classes ϕ_j, $j=1,2,\ldots,J$, where $J \leq V$. One approach is to use real tokens of the vocabulary words and do dynamic time warping of the feature sets and obtain word distances. From the word-by-word distance matrices, word equivalence classes may be obtained using the clustering procedures of Levinson et al. [5] in which the vocabulary words are grouped into clusters (equivalence sets) based entirely on pairwise distance scores.

As an example of the use of the above techniques, consider the 39 word vocabulary consisting of the 26 letters of the alphabet, the 10 digits, and the 3 command words STOP, ERROR, and REPEAT. These 39 words become clustered into the sets ϕ_1 = B,C,D,E,G,P,T,V,Z,3,REPEAT, ϕ_2 = A,J,K,8,H, ϕ_3 = F,S,X,6, ϕ_4 = I,Y,5,4, ϕ_5 = Q,U,2, ϕ_6 = L,M,N, ϕ_7 = O, ϕ_8 = R, ϕ_9 = W, ϕ_{10} = STOP, ϕ_{11} = ERROR, ϕ_{12} = 0, ϕ_{13} = 1, ϕ_{14} = 7, and ϕ_{15} = 9, We will be discussing this vocabulary and the resulting equivalence sets in Section III.

2.2 Determination of Class Distance Scores

Once all the vocabulary words have been assigned to one of the J classes, the first recognition pass estimates an ordering of the word classes in terms of class distance scores. The class distance scores are computed as the minimum of the word distance scores, for all words in the class, i.e.

$$\tilde{d}(\phi_j) = \min_{v_i \in \phi_j} D(\tilde{\mathbf{T}},\tilde{\mathbf{R}}_i), \quad j=1,2,\ldots,J \qquad (6)$$

This computation is similar to the one used by Aldefeld et. al. [6] for directory listing retrieval.

2.3 Choice of Weighting Functions for the Second Pass of Recognition

The output of the first recognition pass is an ordered set of word class distance scores. For the second recognition pass, all words *within* the top class (or classes) are compared to the unknown test word pattern ($\tilde{\mathbf{T}}$) using a weighted distance of the type discussed in Eq. (5), and an ordering of words *within* the class is made. If several classes have similar class distance scores, the words within each of these classes are ordered in the same manner. Based on a simple theoretical model, a reasonable choice for frame weighting is

$$W^{j,i}(k) = \frac{|<\hat{d}_{ii}(k)> - <\hat{d}_{ji}(k)>|}{[\sigma^2_{\hat{d}_{ii}(k)} + \sigma^2_{\hat{d}_{ji}(k)}]^{1/2}} \qquad (7)$$

where $\hat{d}_{ii}(k)$ is the local distance between repetitions of word i for frame k, and $\hat{d}_{ji}(k)$ is the local distance between spoken words j and i for frame k, and where the expectations are performed statistically over a large number of occurrences of the words v_i and v_j.

By way of example, Figure 2 shows examples of plots of $<d_{ji}(k)>$ versus k and $W(k)$ versus k for some typical cases. Figure 2 shows a series of plots for the following cases:

1. (Fig. 2a) Curves of $<\hat{d}_{ji}(k)>$ and $\sigma_{\hat{d}_{ji}(k)}$ for the case where word i was the letter I, and word j was the letter Y. It can be seen that $<\hat{d}_{ii}(k)>$ (the solid curves) is approximately constant whereas $<\hat{d}_{ji}(k)>$ differs from $<\hat{d}_{ii}(k)>$ only at the beginning of the word (i.e. the first 8 frames). It can also be seen that the curves of $\sigma_{\hat{d}_{ji}(k)}$ (the dashed curves) are comparable for the cases $j=i$ and for $j \neq i$, with only small differences occurring in the first 8 frames.
2. (Fig. 2b) Curves of $<\hat{d}_{ji}(k)>$ and $\sigma^2_{\hat{d}_{ji}(k)}$ for the case where word i was the letter A, and where j corresponded to the letters J, K and 8. Similar behavior to that of Fig. 2a is seen in that $<\hat{d}_{ii}(k)>$ is approximately constant, and $<\hat{d}_{ji}(k)>$ is larger than $<\hat{d}_{ii}(k)>$ at the beginning of the word, for words J and K, and at the end of the word, for word 8. For the word 8, the curve of $\sigma_{\hat{d}_{ji}(k)}$ is also fairly large at the end of the word, indicating the high degree of variability in the plosive release of the word 8.
3. (Fig. 2c) This part shows the results of averaging the data of Fig. 2b over all $j \neq i$ with j in the class of word i - i.e. class weighting templates. In this case the curve of $<\hat{d}_{ji}(k)>$ shows flat behavior except at the beginning (due to J,K) and end (due to 8). If storage of word weighting curves is burdensome, the use of class weighting curves could be considered as a viable alternative.

2.4 Generation of Distance Scores for the Second Recognition Pass

We have now shown how to assign words to classes, how to get class distance scores for the first recognition pass, and how to assign weights for pairs of words within a word class. The next step in the procedure is the determination of the distance for the second recognition pass based on the pairwise weighted distance scores.

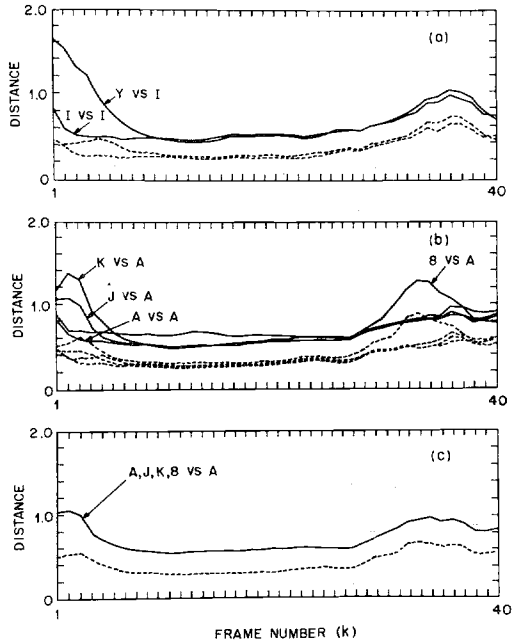

Fig. 2 Examples of frame-by-frame distances for words within word equivalence classes.

To see how this is accomplished, we define a pairwise weighted distance $D_{j,i}$ as

$$D_{j,i} = \frac{\sum_{k=1}^{N} W^{j,i}(k) d_i(k)}{\sum_{k=1}^{N} W^{j,i}(k)} \tag{8}$$

where i is the index of the reference pattern (i.e. one of the words in the equivalence class) and j is the (assumed) index of the test pattern (again one of the words in the equivalence class).

The quantity $D_{j,i}$ of Eq. (8) is computed for all i,j pairs (with $i \neq j$) in the word class with minimum class distance, and a matrix of pairwise distances **D** is obtained. The word distance, D_i, is obtained as

$$D_i = \sum_{\substack{j \\ j \neq i}} D_{j,i} \tag{9}$$

2.5 Overall Distance Computation

If we can make the assumption that the probability of a class error on the first recognition pass is significantly smaller than the probability of a word error on the first pass, then the final distance for each word of the minimum class is the distance as obtained on the second recognition pass. However there are applications in which it is desirable to have a distance score for *every* word in the vocabulary. Hence, in these cases, it is necessary to combine the ordering from the second pass, with the distances from the first pass. The basis for such a strategy is that distances on the first pass are statistically more reliable than distances on the second pass, whereas order statistics (within the class) are more reliable on the second pass than on the first pass. One very simple way of combining distances and word orders is to obtain second pass ordering for every word in the vocabulary (i.e. apply the method of Section 2.4 to all word classes), and then reorder the word list using distances from the first pass, and ordering within the class from the second pass.

2.6 Summary of the Two-Pass Recognizer

Figure 3 shows a block diagram of the full two-pass isolated word recognition system. In the next section we demonstrate how this procedure works in some practical recognition examples.

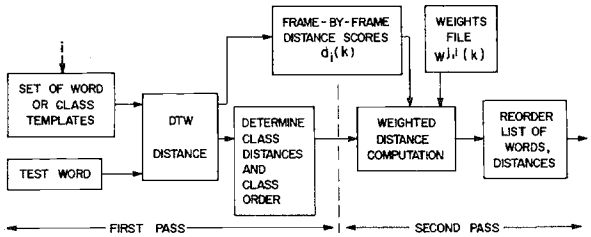

Fig. 3 Block diagram of the overall two-pass recognizer.

III. Evaluation of the Two-Pass Recognizer

In order to test the ideas behind the two-pass recognizer, a data base of existing recordings was used. The word vocabulary consisted of the V=39 word vocabulary of the letters of the alphabet, the digits (0-9), and the three command words STOP, ERROR, and REPEAT. The training data for obtaining word and class reference templates, and pairwise word weighting curves, consisted of 1 replication of each word by each of 100 talkers (50 men, 50 women). The word reference templates (12 per word) were obtained from a clustering analysis of the training data [3,5]. The pairwise word weighting curves were obtained by cross-comparing all word tokens within a word class, averaging the time aligned distance curves, and computing both the averages and standard deviations for each frame.

To test the performance of the overall system, two test sets of data were used. These included:

1. TS1 - 10 talkers (not used in the training) spoke the vocabulary one time over a dialed up telephone line.
2. TS2 - 10 talkers (included in the training) spoke the vocabulary one time over a dialed-up telephone line.

Two sets of performance statistics were measured. For the first recognition pass the ability of the recognizer to determine the correct word class was measured. For the second recognition pass the improvement in word recognition accuracy (over the standard one-pass approach) was measured. The results obtained are presented in the next two sections.

3.1 Class Recognition Accuracy for the First Pass

The ability of the recognizer to determine the "correct" word class of the spoken word was measured using word templates and obtaining class distance scores from the word distance scores. The number of templates per word varied from 1 to 12 in the tests to see the effects of the number of reference templates on the class accuracy. The K-nearest neighbor (KNN) rule was used to measure class scores with values of KNN=1 (minimum distance), and KNN=2 (average of two best scores).

The results of the class recognition accuracy tests are given in Figure 4. Figure 4 shows plots of class error rate (based on the top C classes) as a function of the number of templates per word for values of KNN=1 and 2, and for C=1 (top candidate), C=2 (2 best classes), and C=3 (3 best classes).

Several interesting observations can be made from Figure 4. These include:

1. The KNN=1 rule performs consistently better than the KNN=2 rule for class discrimination, for *all values* of C and Q. This result is in contradiction with the results of Rabiner et al. [3] who found significantly better performance for KNN=2 than for KNN=1. The explanation of this behavior is that the KNN=2 rule provides significantly improved, within-class discrimination, (at the expense of slightly worse between class discrimination) and that when the only function is to determine the class, the KNN=1 rule is superior.

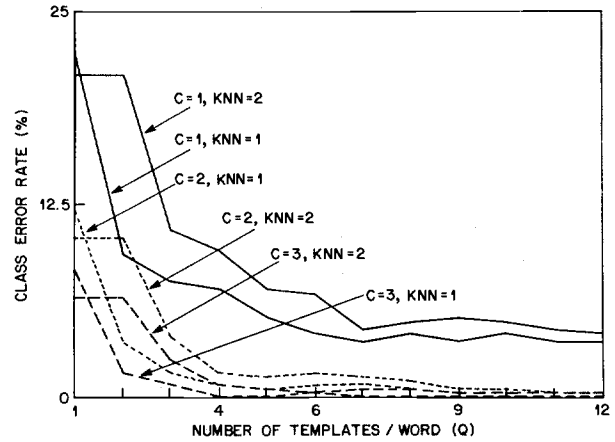

Fig. 4 Plots of class accuracy as a function of the number of templates per word (Q), class position (C), and KNN rule (K). For a 15 class vocabulary.

2. With 6 templates per word, error rates of about 4% (C=1), 1% (C=2), and 0% (C=3) are obtainable, indicating that the full contingent of 12 templates per word is unnecessary for proper class determination. Using 6, rather than 12 templates per word reduces the computation in the first recognition pass by 50%. If we *always* use 2 or more word classes, the required number of templates per word for the first pass can be reduced to 4 with no serious loss in class accuracy.

The results shown in Figure 4 indicate that high accuracy can readily be achieved in determining the correct equivalence class for each word in a very complex vocabulary. Hence there would appear to be no problems in implementing the first pass of the recognition system.

3.2 Within-Class Word Discrimination for the Second Pass and Overall Performance Scores

The two-pass word recognizer was tested on the words of TS1 and TS2. For each test set a total of 390 words were used (39 words × 10 talkers). For TS1, the word recognition accuracy (for the best candidate) on the first pass was 78% and for TS2 (with talkers from the training set) the word recognition accuracy on the first pass was 85%. At the output of the second pass, the word recognition accuracy for the best candidate was 84.6% for TS1 and 88.5% for TS2, representing potential improvements of 6.6% and 3.5% respectively. The reason that a larger improvement in accuracy was obtained for TS1 data than for TS2 data was that the accuracy on the first pass was lower for TS1 than for TS2 (where the talkers were in the training set) and hence there was more room for improvement within the word classes.

Figure 5 shows plots of the changes in accuracy that are obtained for TS1 data when a threshold is imposed on the distance scores at the output of the first recognition pass. The threshold specifies that the second recognition pass is skipped if the distance of the second word candidate is more than the threshold greater than the distance of the first word candidate. Clearly this procedure is a strictly computational one since low distance scores for a single word on the first pass are highly reliable indicators that no second pass is necessary. The data plotted in Fig. 5 shows the percentage of cases where the actually spoken word comes in a lower position on the second pass than in the first pass within the word class; it also shows the percentage of cases when the spoken word comes in a higher position on the second pass than the first pass, and the difference (the improvement) between the two curves. All the results are plotted as a function of the distance threshold for performing the second pass computation. It can be seen from these figures that the two-pass recognizer is not ideal - i.e. there is a significant fraction of words for which a worse position results at the output of the second pass. However, on balance, it is seen that a real improvement in recognition accuracy results, and it it this improvement that makes the procedure a viable one.

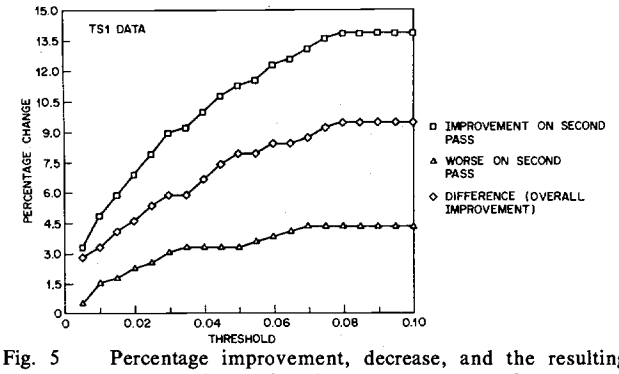

Fig. 5 Percentage improvement, decrease, and the resulting difference in word position at the output of the second recognition pass for TS1 data as a function of the distance threshold.

IV. Discussion

The results presented in the preceding section show that improved recognition accuracy can be obtained via a two-pass recognition algorithm. It was shown that the improvements were both global - i.e. in an absolute recognition sense, and local - i.e. within the classes of equivalent words. Although the proposed two-pass recognizer has a number of possible implementations, it was shown that the best choices were to use a reduced set of word templates on the first pass, and to use all word classes that had reasonably small distance scores on the second pass.

V. Summary

In this paper we have shown that a two-pass approach to isolated word recognition is a viable one when the word vocabulary consists of sets of acoustically similar words. The first recognition pass attempts to determine accurately the class within which the spoken word occurs, and the second recognition pass attempts to order the words within the class based on weighted distances of pairwise comparisons of all words within the class.

References

[1] T. B. Martin, "Practical Applications of Voice Input to Machine", *Proc. IEEE*, Vol. 64, pp. 487-501, Apr. 1976.

[2] F. Itakura, "Minimum Prediction Residual Principle Applied to Speech Recognition", *IEEE Trans. on Acoustics, Speech and Signal Proc.*, Vol. ASSP-23, No. 1, pp. 67-72, Feb. 1975.

[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. on Acoustics, Speech, and Signal Proc.*, Vol. ASSP-27, No. 4, pp. 336-349, Aug. 1979.

[4] C. S. Myers, L. R. Rabiner and A. E. Rosenberg, "Performance Tradeoffs in Dynamic Time Warping Algorithms for Isolated Word Recognition", *IEEE Trans. on Acoustics, Speech, and Signal Proc.*, (to appear).

[5] 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. on Acoustics, Speech, and Signal Proc.*, Vol. ASSP-27, No. 2, pp. 134-141, April 1979.

[6] B. Aldefeld, L. R. Rabiner, A. E. Rosenberg, and J. G. Wilpon, "Automated Directory Listing Retrieval System Based on Isolated Word Recognition", *Proc. IEEE*, Oct. 1980.