"The use of dynamic segments in the automatic recognition of continuous speech," Rome Air Development Center, Griffiss Air Force Base, Rome, N. Y., Tech. Rep. RADC TR-70-22, 1970.

[16] B. Gold, "Word recognition computer program," Res. Lab. Electron., Mass. Inst. Tech., Cambridge, Mass., Tech. Rep. 452, June 1966.

[17] G. W. Hughes, "The recognition of speech by machine," Res. Lab. Electron., Mass. Inst. Tech., Cambridge, Mass., Tech. Rep. 395, 1961.

[18] A. N. Stowe, "Syllable segmentation in voiced environments," *J. Acoust. Soc. Amer.*, vol. 36, p. 1048, 1964.

[19] I. Lehiste and G. E. Peterson, "Transitions, glides, and diphthongs," *J. Acoust. Soc. Amer.*, vol. 33, pp. 268–277, 1961; also, I. Lehiste, Ed., *Readings in Acoustic Phonetics*. Cambridge, Mass.: Mass. Inst. Tech. Press, 1967.

[20] A. Holbrook and G. Fairbanks, "Diphthong formants and their movements," *J. Speech Hear. Res.*, vol. 5, pp. 38–58, 1962; also, I. Lehiste, Ed., *Readings in Acoustic Phonetics*. Cambridge, Mass.: Mass. Inst. Tech. Press, 1967.

[21] J. W. Forgie and C. D. Forgie, "Results obtained from a vowel recognition computer program," *J. Acoust. Soc. Amer.*, vol. 31, pp. 1480–1489, 1959.

[22] L. J. Gerstman, "Classification of self-normalized vowels," *IEEE Trans. Audio Electroacoust.*, vol. AU-16, pp. 78–80, Mar. 1968.

[23] G. E Peterson and H. L. Barney, "Control methods used in a study of the vowels," *J Acoust. Soc. Amer.*, vol. 24, pp. 175–184, 1952; also, I. Lehiste, Ed., *Readings in Acoustic Phonetics*. Cambridge, Mass.: Mass. Inst. Tech. Press, 1967.

[24] C. C. Tappert, W. D. Chapman, N. R. Dixon, and A. S. Rabinowitz, "Automatic recognition of continuous speech utilizing dynamic segmentation, dual classification, sequential decoding, and error recovery," Rome Air Development Center, Griffiss Air Force Base, Rome, N. Y., Tech. Rep. RADC-TR-71-146, 1971.

[25] O. Fujimura, "Analysis of nasal consonants," *J. Acoust. Soc. Amer.*, vol. 33, pp. 1865–1875, 1961; also, I. Lehiste, Ed., *Readings in Acoustic Phonetics*. Cambridge, Mass.: Mass. Inst. Tech. Press, 1967.

[26] J. M. Heinz and K. N. Stevens, "On the properties of voiceless fricative consonants," *J. Acoust. Soc. Amer.*, vol. 33, pp. 589–596, 1961; also, I. Lehiste, Ed., *Readings in Acoustic Phonetics*. Cambridge, Mass.: Mass. Inst. Tech. Press, 1967.

[27] M. Halle, G. W. Hughes, and J.-P. A. Radley, "Acoustic properties of stop consonants," *J. Acoust. Soc. Amer.*, vol. 29, pp. 107–116, 1957; also, I. Lehiste, Ed., *Readings in Acoustic Phonetics*. Cambridge, Mass.: Mass. Inst. Tech. Press, 1967.

Minimum Prediction Residual Principle Applied to Speech Recognition

FUMITADA ITAKURA, MEMBER, IEEE

***Abstract*—A computer system is described in which isolated words, spoken by a designated talker, are recognized through calculation of a minimum prediction residual. A reference pattern for each word to be recognized is stored as a time pattern of linear prediction coefficients (LPC). The total log prediction residual of an input signal is minimized by optimally registering the reference LPC onto the input autocorrelation coefficients using the dynamic programming algorithm (DP). The input signal is recognized as the reference word which produces the minimum prediction residual. A sequential decision procedure is used to reduce the amount of computation in DP. A frequency normalization with respect to the long-time spectral distribution is used to reduce effects of variations in the frequency response of telephone connections.**

The system has been implemented on a DDP-516 computer for the 200-word recognition experiment. The recognition rate for a designated male talker is 97.3 percent for telephone input, and the recognition time is about 22 times real time.

I. INTRODUCTION

RECENTLY time-domain speech analysis based on linear predictability of signal waveform has been successfully adopted for efficient coding of a redundant speech signal [1], [2]. Motivated by these successes, several efforts have been made toward application of the linear predictor coefficients (LPC) for speech recognition [3], [4]. But the immediate use of LPC as feature parameters was not so successful as might be expected [5]. The lack of success may be partly due to the fact that the feature space spanned by LPC is too complicated to introduce a simple and effective measure of distance between elements in the space.

It may be natural to raise the question: what kind of distance measure should be used in the framework of the linear prediction technique? In order to discuss this question, let us consider a more simplified problem; given a short segment of signal, what is the optimal distance measure to test a hypothesis that the segment can be regarded as one generated by a model having specified LPC? The answer to this question might be extended to test a more complicated hypothesis; namely, that some input utterance can be regarded as a word having a specified pattern of LPC.

In this paper, an approach to this problem will be described from a statistical point of view, and it will be shown that the log likelihood ratio, which is the best criterion to test the hypothesis, is reduced to the logarithm of the ratio of prediction residuals, and can be used as a

Manuscript received March 29, 1974; revised October 2, 1974. This paper was presented at the IEEE Symposium on Speech Recognition, Carnegie-Mellon University, Pittsburgh, Pa., April 15–19, 1974.

The author is with the Acoustics Research Department, Bell Laboratories, Murray Hill, N. J. 07974 and the Electrical Communications Laboratories, Nippon Telephone and Telegraph Public Corporation, Musashino, Tokyo, Japan.

powerful distance measure. This result is applied to automatic recognition of isolated words, wherein a sequential likelihood ratio test is adopted to reduce the amount of computation.

II. DISTANCE MEASURE FOR AN ALL-POLE MODEL

An all-pole model of speech signal is as follows. The discrete-time signal $x(n)$ ($t = nT$, when T is the interval between samples) in a stationary segment of signal satisfies the system of difference equations

$$x(n) + a(1)x(n-1) + \cdots + a(p)x(n-p) = e(n) \tag{1}$$

where $\{a(i)\}$ are constants and $\{e(n)\}$ is a white noise or a quasi-periodic signal, with mean-squared value s. Because p is usually much less than the period of $\{e(n)\}$, $\{e(n)\}$ can be regarded as an uncorrelated signal as far as correlations between $q(<p)$ adjacent samples are concerned. In this paper, $\{-a(i)\}$ or simply $\{a(i)\}$ is called the LPC, and the mean-squared value s of $\{e(n)\}$, or power, is called the prediction residual.

The problem in this section is to derive a measure of distance, or dissimilarity, between a segment of signal $\mathbf{X} = (x(1),\cdots,x(N))$ and the model defined by (1). Here, for mathematical tractability, we assume that $\{e(n)\}$ is a Gaussian white noise. Then a set of parameters $\mathbf{P} = (s,a(1),\cdots,a(p))$ specifies the conditional joint probability density $p(\mathbf{X}/\mathbf{P})$, and if $N \gg p$, the logarithm of $p(\mathbf{X}/\mathbf{P})$ is approximately given by [6], [7]

$$L(\mathbf{X}/\mathbf{P}) = -(N/2)[\log 2\pi s + (1/s)\mathbf{aVa}'] \tag{2}$$

where $\mathbf{a}$ is a row vector $(1,a(1),\cdots, a(p))$, and

$$\mathbf{V} = [v(|i-j|)], \qquad (i, j = 0,1,\cdots p)$$

is a correlation matrix whose elements are defined by

$$v(i) = (1/N) \sum_{n=1}^{N-i} x(n)x(n+i). \tag{3}$$

Supposing that we have no knowledge about the absolute value of s and s is a free parameter (Case I), it is replaced by its estimate which maximizes $L(\mathbf{X}/\mathbf{P})$

$$\partial L(\mathbf{X}/\mathbf{P})/\partial s = 0, \qquad s = \mathbf{aVa}' \tag{4}$$

thus, from (2), we obtain

$$\begin{aligned} L'(\mathbf{X}/\mathbf{a}) &= \max_{s} L(\mathbf{X}/\mathbf{P}) \\ &= -(N/2) \log \mathbf{aVa}' + C. \end{aligned} \tag{5}$$

The vector $\hat{\mathbf{a}}$ which maximizes $L'(\mathbf{X}/\mathbf{a})$ is determined as a solution of the following system of equations:

$$\sum_{j=0}^{p} v(i-j)\hat{\mathbf{a}}(j) = 0, \qquad (i = 1,\cdots,p) \tag{6}$$

and the maximum value is as follows:

$$\begin{aligned} L''(X) &= \max_{\mathbf{a}} L'(\mathbf{X}/\mathbf{a}) \\ &= -(N/2) \log \hat{\mathbf{a}}\mathbf{V}\hat{\mathbf{a}}' + C. \end{aligned} \tag{7}$$

$L''(\mathbf{X})$ is the maximum value of the likelihood function when both s and $\mathbf{a}$ are assumed to be free parameters (Case II). From (5) and (7), the likelihood ratio for the Case I and II of free parameters is proportional to

$$d(\mathbf{X}/\mathbf{a}) = \log (\mathbf{aVa}'/\hat{\mathbf{a}}\mathbf{V}\hat{\mathbf{a}}'). \tag{8}$$

The quadratic forms $\mathbf{aVa}'$ and $\hat{\mathbf{a}}\mathbf{V}\hat{\mathbf{a}}'$ in (8) are prediction residuals, when the input signal $\mathbf{X}$ is operated by the model LPC $\mathbf{a}$ and the estimated LPC $\hat{\mathbf{a}}$, respectively.

If the model defined by $\mathbf{a}$ is close to the actual process which generates $\mathbf{X}$, then the $\hat{\mathbf{a}}$, which is the maximum likelihood estimate of $\mathbf{a}$, are close to the $\mathbf{a}$ and $d(\mathbf{X}/\mathbf{a})$ is close to zero; if not, $d(\mathbf{X}/\mathbf{a})$ is significantly large. More precisely, $d(\mathbf{X}/\mathbf{a})N$ will be asymptotically a χ^2 variate with p degrees of freedom, if $\mathbf{X}$ is a realization of a model having $\mathbf{a}$ as the parameter(null hypothesis) [12]. If the $d(\mathbf{X}/\mathbf{a})N$ is larger than the $\chi^2_{1-\alpha}(p)$, then the null hypothesis should be rejected at a given probability α of false rejection. In this sense, $d(\mathbf{X}/\mathbf{a})$ can be regarded as a distance measure between $\mathbf{X}$ and the hypothesized model (1) specified by LPC $\mathbf{a}$.

For small deviation of $\hat{\mathbf{a}}$ from $\mathbf{a}$, the distance of (8) can be approximated by

$$d'(\mathbf{X},\mathbf{a}) = [(\mathbf{a} - \hat{\mathbf{a}})\mathbf{V}(\mathbf{a} - \hat{\mathbf{a}})']/[\hat{\mathbf{a}}\mathbf{V}\hat{\mathbf{a}}']. \tag{8$'$}$$

This measure is apparently different from those which might be intuitively suggested in the space of $\{a(i)\}$ such as

$$d = (\mathbf{a} - \hat{\mathbf{a}})\mathbf{W}(\mathbf{a} - \hat{\mathbf{a}})'. \tag{9}$$

The main difference is in that, the weighting of the quadratic form in (9) is a constant matrix $\mathbf{W}$, while that of (8′) depends on the autocorrelation $\mathbf{V}$ of $\mathbf{X}$. This dependence of the weighting matrix on $\mathbf{V}$, or equivalently on $\hat{\mathbf{a}}$, is a natural consequence from the fact that the covariance characteristics of estimation error of $\mathbf{a}$ depends on the location of $\hat{\mathbf{a}}$ in the space of LPC. It is well known that the estimated covariance matrix of $(\mathbf{a} - \hat{\mathbf{a}})$ is proportional to the inverse of $\mathbf{V}/[\hat{\mathbf{a}}\mathbf{V}\hat{\mathbf{a}}']$ [12]. In the distance measure of (8′), the weighting for the difference $(\mathbf{a} - \hat{\mathbf{a}})$ is automatically adjusted in accordance with the probable error of the estimated parameter $\hat{\mathbf{a}}$.

Equation (8) can be rewritten in the form

$$d(\mathbf{X}/\mathbf{a}) = c + \log [(\mathbf{br})/(\hat{\mathbf{a}}\mathbf{r})] \tag{10}$$

where $(\mathbf{xy})$ means the inner product of two vectors, $\mathbf{r} = (v(i)/v(0))$, $(i = 0,\cdots,p)$, $c = \log (\mathbf{aa})$, and $\mathbf{b}$ is a vector $(1,b(1),\cdots,b(p))$ whose elements are defined by

$$b(i) = 2 \sum_{j=0}^{p-i} a(j)a(j+i)/(\mathbf{aa}). \tag{11}$$

The $\{b(i)/2\}$'s are the autocorrelation coefficients associated with the inverse filter of the all-pole model. The c is the log power of its impulse response. These modified parameters **b** and c are more convenient to compute $d(\mathbf{X}/\mathbf{a})$ than the **a** themselves.

III. ISOLATED WORD RECOGNITION

Each isolated word to be recognized can be expressed as a time pattern of LPC, which is called the reference pattern. The process in recognition is to find a reference pattern which produces the minimum distance to an input utterance.

Reference Pattern: The reference pattern $\mathbf{R}(k)$ for each word is stored as a matrix of the form

$$\mathbf{R}(k) = [c(m;k),\mathbf{b}(m;k)] \qquad (m = 1,\cdots,M(k),k = 1,\cdots,K) \qquad (12)$$

where $c(m;k)$ and $\mathbf{b}(m;k)$ are the modified parameters of LPC at the mth segment of the kth reference pattern, $M(k)$ is the number of segments in the reference pattern $\mathbf{R}(k)$, and K is the number of words to be recognized. Elements of the matrix $\mathbf{R}(k)$ are computed from a training utterance using (3), (6), and (11).

Recognition: An input utterance is expressed as a time pattern of autocorrelation coefficients at the first p delays

$$\mathbf{r}(n), \qquad n = 1,\cdots,N \qquad (13)$$

where N is the number of segments in the input utterance. The distance between the nth segment of the input and the mth segment of a reference pattern $\mathbf{R}(k)$ is

$$d(n,m;k) = c(m;k) + \log [(\mathbf{b}(m;k)\mathbf{r}(n))/(\hat{\mathbf{a}}(n)\mathbf{r}(n))]. \qquad (14)$$

The value of $(\hat{\mathbf{a}}(n)\mathbf{r}(n))$ is obtained in the process of solving the linear equation (6).

If we assume statistical independence of $d(n,m;k)$ for $n = 1,\cdots,N$, it is reasonable to sum up $d(n,m;k)$ over the entire input utterance to give the total distance between the input and the reference pattern. Of course, m must be determined as a function of n

$$m = w(n). \qquad (15)$$

This function $w(n)$, which maps the input time axis onto the reference time axis, is called the time-warping function. This function should satisfy some boundary conditions as well as some continuity conditions. For brevity in the following discussion, it is assumed that $w(n)$ is subject to the following conditions.

Boundary Conditions:

$$w(1) = 1, \qquad w(N) = M(k). \qquad (16)$$

Continuity Conditions:

$$\begin{aligned} w(n+1) - w(n) &= 0, 1, 2 \qquad (w(n) \neq w(n-1)) \\ &= 1, 2 \qquad (w(n) = w(n-1)). \end{aligned} \qquad (17)$$

Fig. 1 shows the domain of possible (n,m) coordinates and an example of $w(n)$. The continuity conditions imply that the ratio of instantaneous speed of the input utterance to that of the reference is bounded between 1/2 and 2 at every point. Let us denote the minimum value of the sum of $d(n,m;k)$ for all possible choices of the time-warping function by

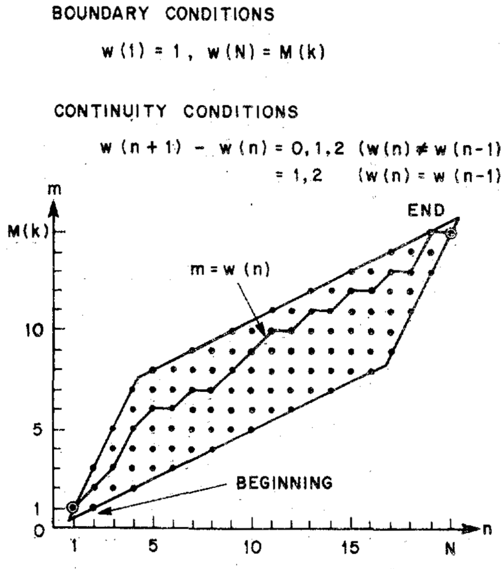

Fig. 1. An example of time-warping function. The parallelogram shows the possible domain of (n,m) coordinates.

$$D(k) = \min_{\{w(n)\}} \sum_{n=1}^{N} d(n,w(n);k). \qquad (18)$$

$D(k)$ is a distance between the input utterance and a hypothesized word k. A decision can be made on the basis of the minimum distance among $D(k),k = 1,2,\cdots,K$.

IV. DYNAMIC PROGRAMMING AND SEQUENTIAL DECISION

The distance $D(k)$ in (18) can be efficiently computed using the algorithm of dynamic programming (DP) [8]–[10]. Let us introduce the partial distance measure, in which the boundary conditions are $w(1) = 1$ and $w(n) = m$, and the continuity conditions are the same as the above, denoted by

$$D(n,m;k) = \min_{\{w(j)\}} \sum_{j=1}^{n} d(j,w(j);k). \qquad (19)$$

Then there follows the recurrence relation;

$$\begin{aligned} D(n+1,m;k) = d(n+1,m;k) + \min\, (&D(n,m;k) \\ &\cdot g(n,m), \\ &D(n,m-1;k),D(n,m-2;k)) \end{aligned} \qquad (20)$$

where

$$\begin{aligned} g(n,m) &= 1(w(n) \neq w(n-1)), \\ &= \infty\,(w(n) = (n-1)). \end{aligned} \qquad (21)$$

In the recurrence relation (20), it is assumed that $d(n,m;k)$ outside the allowable domain in the (n,m) coordinates, shown in Fig. 1, is infinitely large. $D(k)$ is found at the last stage in this recurrence formula, that is, $D(k) = D(N,M(k);k)$.

As shown in the recurrence relation, at every lattice point, $d(n,m;k)$ must be calculated; therefore the amount of computation to obtain $D(k)$ is approximately proportional to the number of lattice points, which is nearly

$$L = (2N - M(k) + 1)(2M(k) - N + 1)/3. \quad (22)$$

For example, if $N = 40$, $M(k) = 40$, $K = 200$, the total number of lattice points to be examined in recognizing one word is nearly 112 000, and the processing time is 49 s, if one lattice point is processed in 420 μs, as is the case described later.

One method to reduce the computation time might be the procedure of sequential decisions. Now, let us define $D(n;k)$ by

$$D(n;k) = \min_{m} D(n,m;k). \quad (23)$$

$D(n;k)$ is the minimum distance between the first n segments of the input and a reference pattern $\mathbf{R}(k)$. If a reference pattern $\mathbf{R}(k^*)$ coincides with the input, it is expected that $D(n;k^*)$ takes lower values for all stages n. Therefore, if $D(n;k)$ is sufficiently large compared with probable excursion of $D(n;k^*)$, the reference pattern can be immediately rejected at an early stage without examining further segments. If not, the next stage is examined.

The threshold for rejection should be as low as possible under the constraint that the probability of false rejection is sufficiently small. Although it is desirable to get a threshold $T(n)$ which meets this requirement theoretically, it is difficult to find it in practice.[1] But, if we notice that the final decision is made on the basis of the relative values of $D(k)$, $T(n)$ might be determined depending on the actual realization of $D(n;k)$. The method used in the experiment is as follows.

The rejection threshold is assumed to be of the form

$$T(n) = S(n) + M \quad (24)$$

where M is a constant decision margin and $S(n)$ is the variable part of $T(n)$ and is updated at every stage in the DP. Initially $S(n)$ is so selected that it is not less than $D(n;k^*)$ for all cases when k^* coincides with the input word. Beginning with $n = 1$, $D(n;k)$ is sequentially computed. If $D(n;k)$ is less than $T(n)$ at the stage n, $S(n)$ is replaced by $D(n;k)$ to give a new threshold which will be used to compute $D(n;k + 1)$. If a reference which is similar to the input is found, the threshold $T(n)$ is set to a lower value. Thereafter only references similar to the input are examined in detail and other references will be rejected at earlier stages. In the final decision only reference patterns which arrived at the last stage N are candidates for the recognized word. If no reference arrives at the last stage, the input is rejected as an inadequate input.

V. EXPERIMENTAL PROCEDURE AND RESULTS

The recognition scheme described earlier has been implemented on a DDP-516 computer for a 200-word recognition experiment. The flow chart of the system is shown in Fig. 2. The 200 words are Japanese geographical names and were pronounced by a male speaker. The mean duration of the reference utterances is 600 ms, and the mean number of syllables is 3.5.

Speech Input: The experimental arrangement for the isolated word recognition is shown in Fig. 3. Each utterance is inputted to the computer using a conventional telephone set dialed through the Bell Laboratories PBX. The telephone set is placed beside the computer console and the noise level around it is about 68 dB(A). After passing through a lowpass filter whose 6 dB cutoff frequency is 3.0 kHz, the speech is sampled at 6.667 kHz and temporarily stored in disc memory. Each utterance is made within a fixed time interval of 1.2 s after listening to the start signal or manually pressing an initiating switch.

Autocorrelation Analysis: Hamming window of 200 samples (30 ms) is applied to the digitized signal. The window is advanced in steps of 100 samples (15 ms) to get the next segment. The instantaneous power within each window is computed, and if it exceeds the noise level, the first eight coefficients are computed. The speech signal duration is detected automatically by examining the power envelope from the forward and backward and neglecting low-level noise.

Normalization of Long-Time Spectrum: The gross spectral distribution of the input signal may be greatly affected by physical factors, such as transducer and line response, as well as by human factors, such as stress and physical condition. These factors may have serious effects on the stability of the system using LPC. Therefore, we have applied a normalization technique of input spectral distribution in the following way. The first two autocorrelation coefficients are averaged over the utterance interval after weighting by the instantaneous power level. For every utterance, a second-order inverse filter is designed by solving a two-variable linear equation. This filter is used to normalize the gross spectral distribution of the utterance. This is done by convolving the original autocorrelation coefficients and the autocorrelation coefficients of the impulse response of the second order inverse filter. The first six normalized autocorrelation coefficients are used to make both the reference patterns to recognize unknown inputs.

Making a Reference Pattern: The reference pattern for each word is generated by the method described in Section III. In this experiment p is 6, $\overline{M(k)}$, the average number of reference segments is about 40 (600 ms), and K, the number of words, is 200. The memory capacity for storing all the reference patterns is $(p + 1)\overline{M(k)}K = 56\,000$ words, and each computer word consists of 16 bits.

[1] If the reference pattern $R(k^*)$ is the true one and the assumptions of the model and the statistical independence of $d(n,m;k^*)$ for $n = 1,2,\cdots, N$ hold, the optimal $T(n)$ is given by $x_{1-\alpha}{}^2(pn)$, where α is the probability of false rejection.

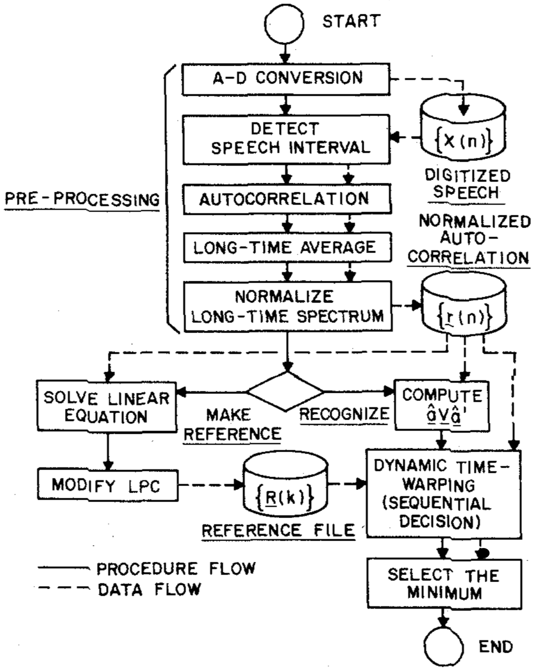

Fig. 2. Flow chart of isolated word recognition.

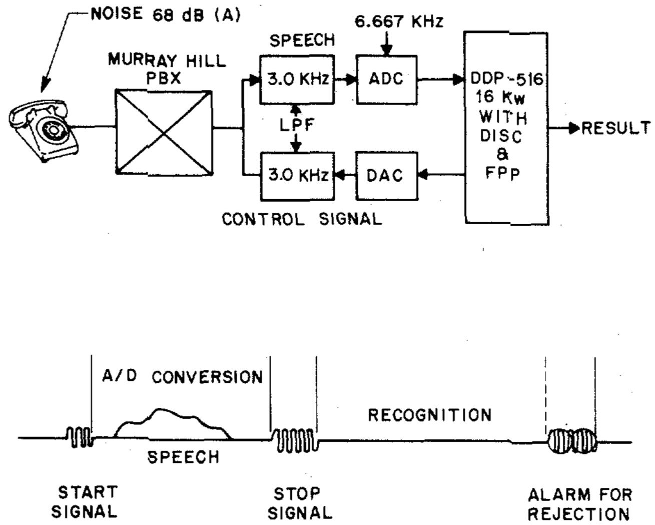

Fig. 3. Experimental arrangement of on-line isolated word recognition.

Recognition and Result: The recognition procedure has been described in Sections III and IV. The major processing in this phase is to compute the distance $d(n,m;k)$ defined by (14), and it is programmed using assembly language. The base of the logarithm is set to 2, and $\log x(1/2 \leq x < 1)$ is approximated by $2(x - 1)$. The computation time for the basic recurrence formula (20) at every lattice point is 420 μs including the computation of $d(n,m;k)$. The threshold margin is chosen as $M = 4$ on the basis of preliminary experiments. The average number of lattice points actually examined is 69 per reference pattern; that is, only 12 percent of L expressed by (22) in which the sequential decision scheme is not used. The total recognition time including the autocorrelation analysis and other preprocessing is about 12 s for one utterance, which is 22 times real time. The recognition rate is 97.3 percent and the rejection rate is 1.65 percent, when the designated talker inputs 2000 test utterances over an time span of three weeks.

VI. Discussion

The main objective of this study is not to develop a particular isolated word recognition system, but is focused to assess experimentally the effectiveness of the proposed distance measure and the sequential decision scheme based on the distance. For this reason, the word recognition system is a straightforward realization of the proposed method without any ad hoc modification. Some parameters, such as intensity, voicing and pitch patterns, are intentionally not used, although they must be crucial in discriminating some words which have very similar patterns of LPC. Despite existence of some room for improvement, the experimental results may give a promising impression, considering the quality of the input signal and vocabulary size. But it must be admitted that the recognition rate is strongly influenced by a particular choice of vocabulary set. For example, if the vocabulary set is the English alphabet and digit

$$\{A,B,\cdots,Y,Z,1,2,\cdots,9,0\},$$

the recognition rate was 88.6 percent for 720 test utterances by the same speaker under the same conditions as in V. All digit input are correctly recognized, although "H" is incorrectly recognized as "8" twice out of 20 trials. Major misclassifications are listed in Table I. This result shows that the majority of confusions occur between pairs in which the vowel part is identical and the difference of the consonant part is relatively small and it is masked by the vowel part which is predominant in duration.

As compared with other word recognition methods of comparable vocabulary size [8], [11], the recognition rate of this method is nearly same or slightly better, despite, in this experiment, a conventional telephone set in a noisy environment is used as a speech input terminal. The complexity of recognition algorithm seems to be of the same order, judging from the processing time per word using computers of similar speed.

VII. Conclusion

A new measure of distance for an all-pole model of speech has been derived on the basis of the likelihood ratio criteria and is applied to automatic recognition of isolated words. An algorithm to find to the best match between the input pattern and a reference pattern is derived, in which the dynamic programming technique is used in conjunction with a sequential decision scheme. The system is implemented on a DDP-516 computer to recognize 200 isolated words. The validity of the scheme has been confirmed experimentally. Further work is in progress to

TABLE I
MAJOR MISRECOGNITIONS IN ALPHABET-DIGIT RECOGNITION

INPUT	RECOGNIZED	NO. of ERRORS
V	B or D	15
B	D	10
P	T or D	9
E	T	9
M	N	5
I	Y	5
	SUB TOTAL	53
	OTHER ERRORS	29
	TOTAL ERRORS / TOTAL TRIALS	82 / 720
	ERROR RATE	=11.4%

test the system for a greater number of talkers and for telephone connections switched over greater distances.

ACKNOWLEDGMENT

The author wishes to thank J. L. Flanagan for his guidance and stimulating discussions, and to acknowledge the help received from C. H. Coker and A. E. Rosenberg during implementation of the system.

REFERENCES

[1] F. Itakura and S. Saito, "Analysis synthesis telephony based on the maximum likelihood method," in *Proc. Int. Congr. Acoust.*, Tokyo, Japan, Rep. C-5-6, 1968.

[2] B. S. Atal and S. L. Hanauer, "Speech analysis and synthesis by linear prediction of speech waveform," *J. Acoust. Soc. Amer.*, vol. 50, pp. 637–655, 1971.

[3] M. Kohda, S. Hashimoto, and S. Saito, "Spoken digit mechanical recognition," *Trans. Inst. Electron. Commun. Eng. (Japan)*, vol. 55-D, no. 3, 1972

[4] Y. Nakano, A. Ichikawa, and K. Nakata, "Evaluation of various parameters in spoken digits recognition," presented at the IEEE Conf. Speech Communication and Processing, Cambridge, Mass., Apr. 1972, Paper C4.

[5] H. Fujisaki and Y. Sato, "Evaluation and comparison of features in speech recognition," Faculty Eng., Univ. of Tokyo, Tokyo, Japan, Annu. Rep. Eng. Res. Inst., 1973, vol. 32, pp. 213–218.

[6] F. Itakura and S. Saito, "A statistical method for estimation of speech spectral density and formant frequencies," *Trans. Inst. Electron. Commun. Eng. (Japan)*, vol. 53-A, pp. 36–43, 1970.

[7] S. Saito, T. Fukumura, and F. Itakura, "Statistically optimum discrimination of speech spectra," *J. Acoust. Soc. Japan*, vol. 23, no. 5, 1967.

[8] V. M. Velichiko and N. G. Zagoruiko, "Automatic recognition of 200-words," *Int. J. Man-Machine Studies*, vol. 2, pp. 223–234, 1970.

[9] H. Sakoe and S. Chiba, "A dynamic programming approach to continuous speech recognition," in *Proc. Int. Congr. Acoust.*, Budapest, Hungary, Rep. 20-C-13, 1971.

[10] ——, "Comparative study of DP-pattern matching techniques for speech recognition," Speech Res. Group, Acoust. Soc. Japan, Rep. S73-22, 1973.

[11] D. R. Reddy, "Segment synchronization problem in speech recognition," *J. Acoust. Soc. Amer.*, vol. 46, no. 1, p. 89, 1969.

[12] H. B. Mann and A. Wald, "On the statistical treatment of linear difference equations," *Econometrica*, vol. 11, pp. 173–217, 1943.

Pitch Detection by Data Reduction

NEIL J. MILLER, MEMBER, IEEE

***Abstract*—This paper presents an algorithm that determines the fundamental frequency of sampled speech by segmenting the signal into pitch periods. Segmentation is achieved by identifying those samples of the waveform corresponding to the beginning of each pitch period.**

The segmentation is accomplished in three phases. First, using zero crossing and energy measurements, a data structure is constructed from the speech samples. This structure contains candidates for pitch period markers. Next, the number of candidate markers within this structure is reduced utilizing syllabic segmentation, coarse pitch frequency estimations, and discrimination functions. Finally, the remaining pitch period markers are corrected, compensating for errors introduced by the data reduction process.

This algorithm processes both male and female speech, provides a voiced-unvoiced decision, and operates in real time on a medium speed, general purpose computer.

INTRODUCTION

THE pitch detection process described in this paper has the following characteristics. It computes a marker that identifies the beginning of each pitch period. It permits the analysis of both male and female speech by detecting pitch frequencies over a range of 50 to 500 Hz. Finally, the process normally requires less than 20*N computer operations, where N is the number of samples in the speech signal.

The algorithm was developed for utilization within a pitch synchronous speech recognition system. However, determination of pitch rate is also essential in pitch synchronous time compression [1], in the study of prosodics [2], and in several bandwidth compression systems [3]–[5]. A variety of algorithms have been reported that perform pitch detection. These include heuristic methods [6], [7], autocorrelation methods [8], single Fourier transform methods [9], single Fourier transforms combined with histograms [10], double

Manuscript received March 29, 1974. This work was supported by the Advanced Research Projects Agency of the Office of the Secretary of Defense under Contract DAHC 15-73-C-0435. This paper was presented at the IEEE Symposium on Speech Recognition, Carnegie-Mellon University, Pittsburgh, Pa., April 15–19, 1974.

The author is with the Artificial Intelligence Laboratory, Department of Computer Science, Stanford University, Stanford, Calif.