INRS · TÉLÉCOMMUNICATIONS

versité du Québec · Institut national de la recherche scientifique

Speaker Identification, Verification and Recognition Using Cepstral Matched Filters

FINAL REPORT

April 30, 1976

Kenneth J. Bures,
Université du Québec-INRS,
3 Place du Commerce,
Verdun, Quebec,
H3E 1H6

This research was sponsored under CRC Contract 02SU.36100-5-0314 Serial OSU5-0093.

List of Figures

1. Purpose and Scope of Research

The purpose of this research project was to determine the feasibility of using a matched filter speech processor to identify/verify speakers, and to recognize words from a limited vocabulary spoken by a single speaker.

Speaker verification is defined as the determination of whether or not a speech sample was spoken by a pre-specified person. Speaker identification is defined as the selection of which of a limited number of speakers has spoken a word or phrase. Speech recognition is defined as the identification of words or sounds of a known speaker.

This research was sponsored under CRC Contract 02SU.36100-5-0314. The contract stipulates that the following tasks be performed:

(1) Produce cepstra for representative samples of speech from several speakers.

(2) Produce cepstral matched filters for the phonemes occuring in the speech samples of one speaker.

(3) Perform the matched filtering of the filters of (2) with the cepstra of (1).

(4) Determine the effectiveness of the cepstral matched filtering technique as a speaker identifier/verifier.

(5) Determine the effectiveness of the cepstral matched filtering technique as a speech recognizer for a single speaker and for a limited vocabulary.

(6) Determine the sensitivity of the matched filtering to utterance duration.

2. Theoretical Principles

The speech mechanism has been accurately modeled (ref.1) as a vocal excitation, $e(t)$, driving a slowly time-varying vocal tract with impulse response, $h(t)$. The actual speech signal, $s(t)$, is the convolution $e(t) * h(t)$; or, in the frequency domain, $S(\omega) = E(\omega)H(\omega)$.

We attempt the separation of e(t) and h(t) by calculating the cepstrum (ref.2), $\hat{s}(t)$, defined as the inverse Fourier transform of the logarithm of the magnitude of the Fourier transform of s(t).

$$\hat{s}(t) = \mathcal{F}^{-1}\{\log|\mathcal{F}\{s(t)\}|\} = \mathcal{F}^{-1}\{\log|E(\omega)|\} + \mathcal{F}^{-1}\{\log|H(\omega)|\} \qquad (1)$$

It is seen that the cepstrum "deconvolves" e(t) and h(t), and displays the effects of each additively. For speech the two cepstral terms are essentially non-overlapping in time. There is a low-time term, $\hat{s}_L(t)$, due to the vocal tract term, and (for voiced sounds) a series of high-time term, $\hat{s}_H(t)$, due to the excitation. These are illustrated in Figure 1.

It was proposed that the two terms, $\hat{s}_L(t)$ and $\hat{s}_H(t)$, could be used in a speech recognition/identification/verification (RIV) system. The idea was that a series of low-time cepstra, $\hat{s}_L(t)$, for a series of phonemes spoken by a specific person could be calculated and stored in a computer. These "templates", or matched filters, could be compared with the low-time cepstrum of an input speech sample. The closest match (i.e. the smallest rms difference) between the speech cepstrum and the template would indicate that the phoneme of the speech sample was the same as that associated with the selected template.

Since the low-time cepstrum is related to the vocal tract impulse response, h(t), the matching process just described can be viewed as a technique of determining the configuration of the vocal tract used by the speaker when he uttered the speech sample. There is, however, still the unanswered question of whether or not the speech was voiced or unvoiced. For example, the phonemes /b/ and /p/ have identical vocal tract configurations; but /b/ is voiced and /p/ is unvoiced. The voiced/unvoiced decision can be made by looking at the high-time portion of the cepstrum, $\hat{s}_H(t)$. For

unvoiced speech this portion is essentially zero. For voiced speech there is a series of pulses which occur at the pitch rate. Thus, the voiced/unvoiced decision is made by determining the presence/absence of the pitch pulses in $\hat{s}_H(t)$.

In the experimental work of this project, only vowel and resonant sounds were used. These are, of course, always voiced; so that the voiced/unvoiced decision was unnecessary. This choice also has the advantage that these sounds are all high-energy sounds; whereas unvoiced sounds, and voiced stops and fricatives tend to be low-energy sounds.

The complete RIV system would operate by grabbing non-overlapping segments of speech. These segments must be long enough to include several pitch periods; but short enough to appear time-stationary. It has been found that 32ms of speech, about three pitch periods, is optimum. At an 8KHz sampling rate, this means that there are 256 samples of speech per analysis segment. These segments are then independently analyzed using the cepstrum approach.

A single-syllable spoken word would typically be made up of 8 to 20 segments. A given matched filter would match well only for a small number of segments within the word. For example, a matched filter may be made for the /v/ in the word "five". A good match would be made only on the last few segments of the word; namely, on those few segments corresponding to the sound /v/.

There are several reasons for believing that this cepstral matched filter technique might be an attractive method for RIV. First, the low-time portion of the cepstrum has a degree of time-normalization associated with it that the original speech waveform does not have. This is because the vocal

tract impulse reponse is separated from the highly variable excitation function. Secondly, the cepstral matched filters require substantially less storage than direct speech waveforms. In the experimental work of this project, 256 samples of speech were converted to 32-sample low-time cepstra.

The cepstrum technique is quite different in philosophy from other types of RIV systems (Ref.3) in two respects. First, the cepstrum system performs its segmenting into analysis segments without regard to speech boundaries. That is, the system does not isolate word boundaries before the actual cepstral analysis begins. It segments and analyzes silent periods as well as active periods. This does not necessarily increase computing time. Secondly, the cepstrum technique analyzes all segments in the same way. Other techniques make preliminary decisions (e.g. voiced/unvoiced, or vowel/consonant, etc.) and then branch to specific analyzers for these subclasses. One would expect that these processors would be more complicated than the cepstral processor; but that they would also be more accurate because each subclass of speech can be processed in a manner optimized for that subclass.

Before the effectiveness of the cepstral matched filter RIV technique could be evaluated, an even more fundamental question had to be answered: namely, whether or not the cepstrum is a suitable signal on which to base RIV decisions. In particular, two specific questions had to be answered.

The first had to do with how the cepstrum varies with changes in the position of the sample interval. This is illustrated in Figure 2. The cepstral analysis is performed on a 256-sample block of digitized speech.

This data block is grabbed asynchronously with respect to the speech pitch pulses (or any other speech-related cue). This eliminates the complications of pitch-synchronous processing. However, the asynchronous analysis

is valid only if the cepstrum for each interval (see Fig. 2) of identical speech is approximately invariant. This invariance was confirmed by computing the cepstrum of all possible shifts of the data intervals. Synthetically-generated vowel sounds were used to insure uniformity of the speech over the interval; and because it was possible to independently vary the various speech parameters. For a given vowel sound, the RMS difference between cepstra of different intervals was about a factor of 10 less than the RMS differences between different vowel sounds. Thus, it was concluded that the cepstrum is indeed approximately invariant to shifts in the analysis interval.

The second specific question had to do with whether the low-time portion of the cepstrum varies with pitch frequency. The effects of pitch variations should be noticed only in the high-time portion of the cepstrum. Any observable variation of the low-time cepstrum with pitch would indicate "leakage" of high-time cepstral components into the low-time period (see Fig. 1).

To test for this leakage, rms differences were measured between the low-time cepstra of the same synthetic phonemes, but with different pitch frequencies. The pitch frequency was varied between 80 and 120 Hertz. It was found that the rms differences were, in the worst case, about half the value of the rms differences between different phonemes with the same pitch frequency. This indicated that there was leakage, but that it was probably manageably small.

3. Preparation of the Matched Filters

The first step in evaluating the cepstral matched filter RIV technique is to generate suitable cepstral matched filters. It was originally proposed that the speech samples from which these cepstral matched filters were to be made be averages over several similar utterances, rather than single "raw" speech samples. The rationale for this was that the averaging process would retain

the similar aspects of these signals, but would reduce the "noise" portions. As an example of producing averaged utterances, consider the four different examples of the /o/-phonemes illustrated in Figure 3. They were all obtained from the digitized speech of a male speaker, who uttered the word "zero" four times. Several pitch periods of the /o/ phonemes were isolated from each "zero". One of these was arbitrarily picked as a reference, and correlated with each of the others. This gave the amount of "shift" necessary in each case to get the best alignment of the four waveforms. Once aligned, the four waveforms were normalized to contain equal energy, and then averaged on a point-by-point basis. The resulting average /o/-phoneme waveform is illustrated in Figure 4. Not surprisingly, there is a distinct similarity between the raw and averaged /o/ waveforms. Unfortunately, the averaging also tends to "wash out" much of the detail of the signal. In addition, the point-by-point averaging requires us to get speech samples which have the same pitch periods. Thus, the averaging technique does not appear to be the ideal signal to use for generating cepstral matched filters.

A second method has been developed for generating suitable "average" speech waveforms for use in constructing cepstral matched filters. This approach, based on linear prediction, produces "average" waveforms with markedly superior fidelity to the original utterances than do the straight averages.

The technique assumes that speech can be modelled as the convolution of a periodic excitation (whose fundamental frequency is the pitch frequency) with the time varying impulse response of the vocal tract (represented as an all-pole filter). Significantly, these are the same assumptions made for the cepstral analysis; and they have been shown to be reasonable assumptions by many researchers. A block diagram of the speech production model is

shown in Figure 5. The n^{th} speech sample, s_n, is given by:

$$s_n = \sum_{k=1}^{P} a_k s_{n-k} + G e_n \qquad (2)$$

where P is the number of poles of the vocal tract transfer function, the a_k are the coefficients of the vocal tract impulse response (transversal filter coefficients), G is a gain factor, and e_n is the n^{th} sample of the excitation.

We wish to determine the a_k for a given utterance. We do not know the excitation function, e_n, since we have available for analysis only the speech samples, s_n. Therefore, we attempt to determine the a_k in such a way as to minimize the mean squared error between the actual speech samples, s_n, and estimated (predicted) values, $\hat{s}_n$, neglecting the excitation function, e_n. From equation (2), it is seen that the estimated samples can be expressed as:

$$\hat{s}_n = \sum_{k=1}^{P} a_k s_{n-k} \qquad (3)$$

so that the mean squared error over an interval of N samples is:

$$E = \sum_{n=1}^{N} (s_n - \hat{s}_n)^2 = \sum_{n=1}^{N} \left(s_n + \sum_{k=1}^{P} a_k s_{n-k}\right)^2 \qquad (4)$$

The total error, E, is minimized with respect to the a_k. The result is a matrix equation of the form:

$$\begin{bmatrix} R_0 & R_1 & \cdots & R_{P-1} \\ R_1 & R_0 & \cdots & R_{P-2} \\ R_2 & R_1 & \cdots & \\ R_{P-1} & R_{P-2} & \cdots & R_0 \end{bmatrix} \begin{bmatrix} a_1 \\ a_2 \\ \\ a_P \end{bmatrix} = \begin{bmatrix} R_1 \\ R_2 \\ \\ R_P \end{bmatrix} \qquad (5)$$

where

$$R_i = \sum_{n=0}^{N-1-i} s_n s_{n+i} \qquad i>0 \qquad (6)$$

is the autocorrelation function of the speech, s_n. This matrix equation can be solved for the a_k (using brute force matrix inversion or faster techniques which depend on the symmetries of the autocorrelation matrix). The result is a set of P coefficients, a_k, which characterize a particular speech utterance.

It was originally thought that a suitable "average" waveform could be obtained by first determining the coefficients, a_k, for each version of an utterance; and then by averaging the coefficients to give a set of average coefficients for the utterance, and finally by re-synthesizing a speech waveform using equation (2) with an impulse excitation function. In practice, however, it was not so simple. The coefficients did not cluster well, indicating that the coefficients change dramatically with only small changes in pole positions. Therefore, the technique adopted was to factor the characteristic polynomials to determine the pole positions of each speech utterance. As expected, the poles clustered well (see Figure 6). These pole positions were averaged, and average pole positions were used to calculate average polynomial coefficients, a_k. These coefficients were used to re-synthesize an "average" speech waveform for each utterance. Figure 7 shows the /o/ phoneme that was formed in the manner just outlined. Note that it is a more faithful representation of the raw speech waveforms than is the true average.

The final step in constructing the cepstral matched filter is, of course, to calculate the cepstrum of the re-synthesized speech waveform.

The cepstrum is calculated in the following manner:

(1) 256 samples of re-synthesized speech are normalized with respect to energy, and Hamming windowed.

(2) The windowed signal is Fourier transformed using an FFT algorithm.

(3) The logarithm of the magnitude of the transformed signal is computed.

(4) The result of (3) is inverse Fourier transformed, yielding the cepstrum.

(5) The first 32 samples of the cepstrum are stored as the cepstral matched filter.

Filters were made for male speaker A for 10 phonemes. The phonemes selected were:

1. /r/ in zero
2. /o/ in zero
3. /ə/ in one
4. /n/ in one
5. /o/ in four
6. /r/ in four
7. /aI/ in five
8. /v/ in five
9. /aI/ in nine
10. /n/ in nine

Figure 8-A thru 8-I show the average pole positions which were calculated using the linear prediction pole averaging technique on five distinct utterances of each phoneme. The speech was modeled by a 12-pole model. Ten poles are complex, and are related to vocal tract parameters. The two real roots are related to the frequency-dependent characteristics of the glottal pulses, and to the acoustic radiation from the mouth. The phonemes were extracted from the word (zero, one, four, five or nine) by means of a computer program

which allowed the operator to extract a single pitch period from a speech file, and then listen to the sound. If the correct sound was selected, the operator could save that pitch period as a separate file, which was used in the linear prediction pole averaging programs.

The step-by-step linear prediction pole averaging technique was as follows:

(1) Select a pitch period of speech (program LISTEN)

(2) Calculate the linear prediction coefficients (LPCOF)

(3) Calculate the pole positions (LPROOT)

(4) Arrange poles in a standard order (ORDER)

(5) Average the pole positions of 5 speech samples, and compute the average coefficients (AVCOF)

(6) Re-synthesize 256 samples of speech (LPSYN)

The output of step (6) is used to compute the cepstral matched filter as outlined earlier.

4. Speech Recognition

A large number of recordings were made for possible use in the analysis. A smaller subset was actually used for testing the feasibility of the cepstral matched filter scheme. In particular, the recordings by male speaker A of the digits 0,1,4,5 and 9 (repeated 5 times) were used to compute the cepstral matched filters. The "unknown" input speech was the telephone number 451-9150 spoken by three male speakers A, B and C. It was decided that the unknown should be a sample of connected speech, because this seemed more realistic in an operational environment.

As detailed in the previous section, two phonemes per digit for the digits 0, 1, 4, 5 and 9 were used in making the matched filters. It was intended that a digit would be recognized if its two distinctive phonemes were detected in the proper order and proximity.

The input speech (451-9150) for each speaker was processed in 256-samples to give a 32-point cepstrum for each segment. This was compared on a point-by-point basis with each of the ten stored cepstra of speaker A, and the rms difference was computed. This difference was plotted for each 256-sample segment of the input speech. The input speech contained 120 such segments. Figure 9 shows the results: the rms difference between speakers A, B and C saying "451-9150" and the 10 matched filters of speaker A. Figure 9-A also shows the locations of the word boundaries. The computations which produced these figures were done using program MATCH2.

A small rms difference indicates a close match; whereas a large rms difference indicates a bad match. It is immediately noted in Figure 9-A that the silent portions (the beginning, end and middle pauses) give bad matches. It is also apparent that each matched filter gives a low output (i.e. a good match) for almost all input phonemes. Thus, the technique is not at all reliable in selecting the correct phoneme.

It is pertinent to determine why the technique is not reliable for speech recognition. If there is an inherent weakness, then it can never be modified to perform speech recognition successfully. A recent paper by Rabiner and Sambur (ref.4) indicates that the schemes for the recognition of connected digits require "greatly different" implementations than for recognition of isolated digits. This is because of coarticulation effects which are present in connected digits. This indicates that one cannot use matched filters made from isolated digits, as was done in this project, to detect phonemes in connected digits. To test this idea, it was determined

to perform the matched filtering on the string of isolated digits 0, 1, 4, 5 and 9, using the matched filters calculated using isolated digits. This arrangement would eliminate all coarticulation effects of connected digits. A series of tests showed that the cepstral matched filter speech recognition system is not noticeably better on isolated digits. Thus, we conclude that coarticulation effects of connected digits is not a cause of the difficulties. There must be some deeper problem with the cepstral approach. Two possible explanations of this deeper problem come to mind.

First, in view of the fact that the cepstral matched filter speech recognition uses so little information about the speech (viz. 32 points of the cepstrum) as compared to other systems (the system of ref. 4 uses zero crossings, log energy, linear predictor coefficients, linear predictor error, and autocorrelation coefficients), it is quite likely that we are simply asking too much of the one measurement.

A second possible explanation is that there is too much leakage of the high-time cepstrum region into the low-time cepstrum region. This hypothesis is not easily tested, beyond what was done in the experiment described in an earlier section. One can speculate as to the source of such leakage to see if it might be feasible to circumvent the problem. Along this line, we recognize that the cepstrum was broken up into high- and low-time portions because initially we assumed the traditional linearly separable vocal tract, $h(t)$, and excitation function, $e(t)$. Some recent work by Flanagan et al (refs. 5, 6) shows that this model puts limits on the physiological realism of the model; and hence, of the speech output. Flanagan avoids these limitations by modeling the speech as sound source which interacts with the resonant system. Such a model does not have a separable excitation function and vocal tract response. Thus the cepstral matched filtering theory, if applied to this model, would not lead to complete separation of

e(t) and h(t). Hence, we conclude that with "natural" (i.e. real) speech, the cepstral leakage is unavoidable. An interesting point is that one effect which the Flanagan model is capable of handling is coarticulation between phonemes. The coarticulation is present with connected digits as explained earlier; but it is also present with isolated digits, because isolated digits are made up of connected phonemes. In fact, on physiological grounds, it is reasonable to expect that phoneme-to-phoneme coarticulation is greater than digit-to-digit coarticulation. The phoneme-to-phoneme coarticulation effects were not noticed in the test for cepstral leakage described in an earlier section because synthetic speech consisting of isolated phonemes was used.

The conclusion, therefore, is that cepstral leakage is a sufficient reason for the failure of the cepstral matched filter speech recognition technique.

5. Speaker Identification and Verification

In this section, we consider the problem of selecting which one of N speech samples was spoken by a specified speaker, given that we have a set of matched filters for that speaker. This is the verification problem. If we have a set of matched filters for each possible speaker, and only one test phrase, then we have the recognition problem. If we solve one, we have solved the other; therefore, we concentrate here on the verification problem.

Strange as it seems, even though the cepstral matched filter technique fails as a speech recognition system, it is quite successful at speaker identification/verification, at least for the small sample size used in this feasibility study. Figure 10 shows plots of the rms differences between the matched filter for /ə/ of "one" of speaker A, matched with the telephone number "451-9150" spoken by speakers A, B, and C. Recall that each point represents the rms difference for one 256-sample segment of speech. We wish to determine

from the data that speaker A spoke the utterance illustrated in Figure 10-A.

The key to the analysis of these figures is to note that one could fit a smooth curve to these points. If this were done, and if one then computed the rms error between the actual data and this smooth curve, one would find that almost always the rms error would be considerably greater for the false matches than for the true match. This rms difference will be called the spread.

Referring to Figure 11-A (which is identical to Figure 10-A), it is seen that the largest rms differences occur during periods of silence at the beginning, end, and in the middle pause of the telephone number utterances. Since we are interested in the spread only during periods of speaking, we can further improve the decision process by calculating the spread only for those data points which can be identified as active speech periods. Periods of silence are ignored.

The active/silent decision is made by setting a suitable threshold on Figure 11-A; and by assuming that all data samples above that threshold are silence,and all data points below the threshold are speech. A suitable threshold was found experimentally to be 75 per cent of full scale. If all points above the threshold are ignored, Figure 11-B results. For the remaining data points in Figure 11-B, we calculate the rms difference with the smoothed data of Figure 11-C. The resulting spread does not include the influence that silent periods had on the previously defined spread calculation. Figure 12 lists the values of spread using the 10 phonemes of speaker A matched with the three speech samples "451-9150" spoken by speakers A, B, and C. Note that in only one case is there an error: the spread for the /o/ of "4" matched with speaker B is larger than for speaker A. However, the results for the other 9 phonemes indicate (correctly) that A is the correct match. The utterance of speaker C is never incorectly identified.

Figure 11-C shows the smoothed version of the raw data of Figure 11-A. The smoothed data was computed from the raw data on a point-by-point basis by extrapolating the best (least squares) fit of a cubic polynomial to the five preceding data points.

This smoothed data is subtracted on a point-by-point basis from those points of the raw data which are below the threshold. These differences are squared and summed to give the total squared error. This error is square-rooted, and that result is divided by the number of data points below threshold. The resulting number, the spread, is the rms error of the least squares fit to the raw data.

There is another test which can be made on the data of Figure 10 to help determine the correct identification. This test is especially helpful in cases where the spread test is not conclusive (i.e. where the spreads in any row of Figure 12 do not differ by a large amount). For example, the differences are not large between the spreads of the /o/ in "4" or the /aI/ in "5" for speakers A and B.

The basis of the test can be seen by referring to the raw data shown in Figure 10. Recall that the silent and active regions can be separated by thresholding at 75 per cent of full scale. It is seen that the silent portions have large values of rms differences (the vertical axis variable of Figure 10), and the active portions have lower values of rms differences. The test is to calculate the average value of the rms difference for the silent periods and for the active periods, and then form the ratio active/silent. Figure 12 lists the ratios for the 10 phonemes of speaker A matched with the test phrases of speakers A, B, and C. Note that in most cases the ratio is considerably less than 0.50 (usually about 0.30 to 0.40). However, in a few cases the ratio is greater than 0.50.

Referring to the corresponding spread values in Figure 12, it is seen that these high ratios correspond to matches which gave ambiguous values of spreads. Thus, this test can be used as an indicator for the validity of the spread test. If the ratio test is high, then the result of the spread test is ambiguous; and it may be best to make no decision on the basis of that spread test. Incidentally, the value of 0.50 used above is not optimized; but represents an "eye ball" best choice of ratio test threshold.

REFERENCES

1. R.W. Schafer, L.R. Rabiner: "Digital Representations of Speech Signals". IEEE Proc. Vol. 63, no. 4, April 1975, pp. 662-677.

2. A.V. Oppenheim, R.W. Schafer: "Homomorphic Analysis of Speech". IEEE Trans. AV-16 (1968), pp. 221-226.

3. M.R. Sambur, L.R. Rabiner: "A Speaker-Independent Digit-Recognition System", BSTJ, vol. 54, pp. 81-102, Jan. 1975.

4. L.R. Rabiner, M.R. Sambur: "Some Preliminary Experiments in the Recognition of Connected Digits", IEEE Trans. ASSP-24, no. 2, April 1976, pp. 170-182.

5. J.L. Flanagan, K. Ishizaka: "Automatic Generation of Voiceless Excitation in a Vocal Cord-Vocal Tract Speech Synthesizer", IEEE Trans. ASSP-2 , no. 2, April 1976, pp. 163-170.

6. J.L. Flanagan, K. Ishizaka, K.L. Shipler: "Synthesis of Speech from a Dynamic Model of the Vocal Tract and Vocal Cords", BSTJ, vol. 54, March 1975, pp. 485-505.

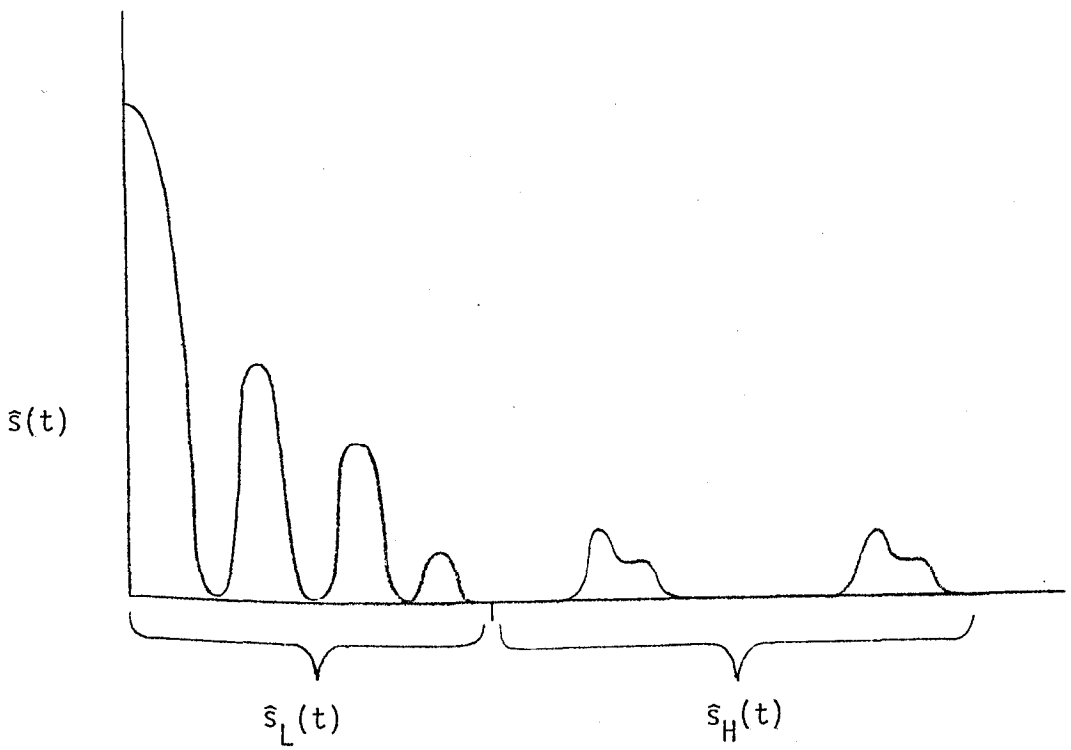


Figure 1: Diagrammatic Representation of Speech Cepstrum

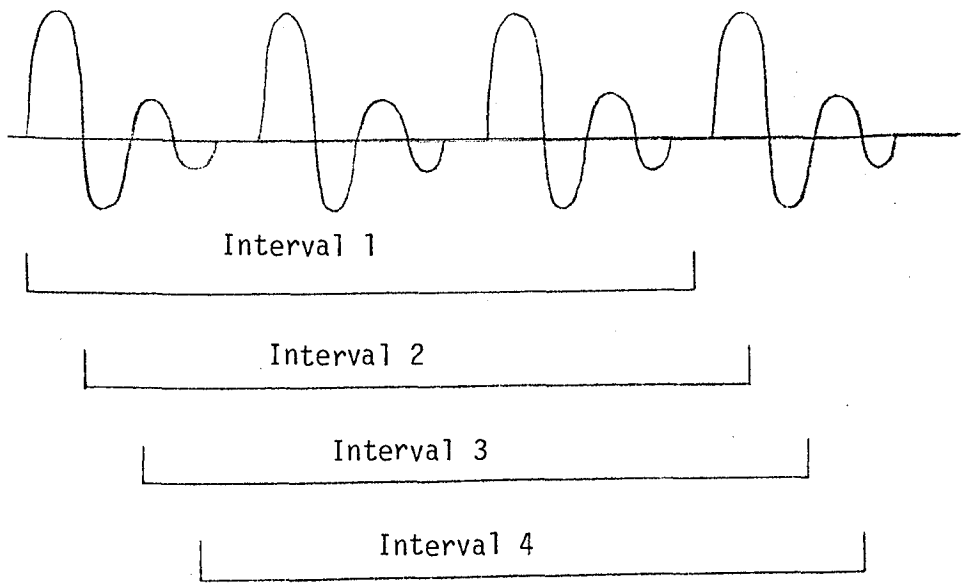


Figure 2: Data Analysis Intervals

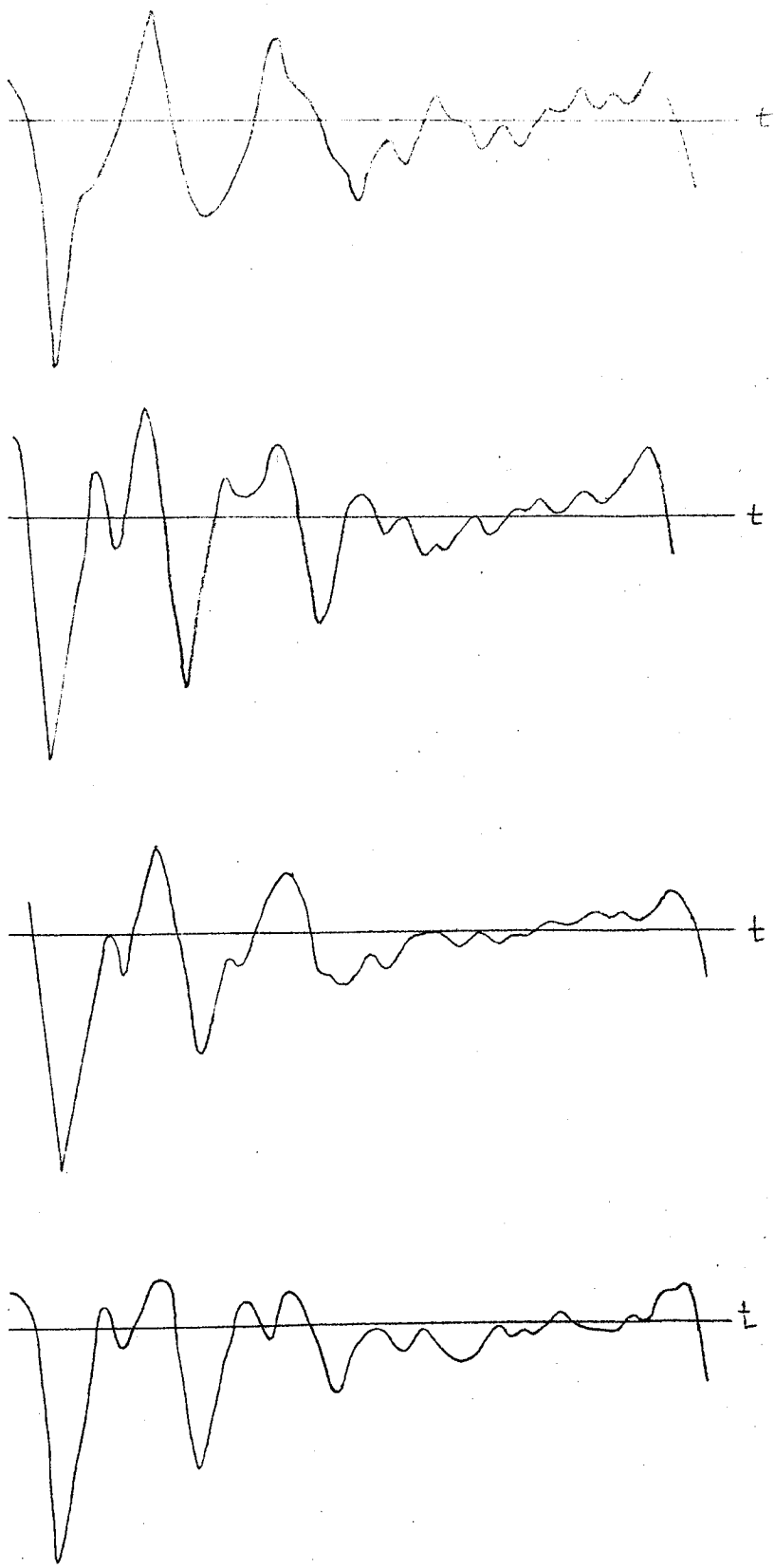


Figure 3: Waveforms of /o/-Phonemes of "zero".

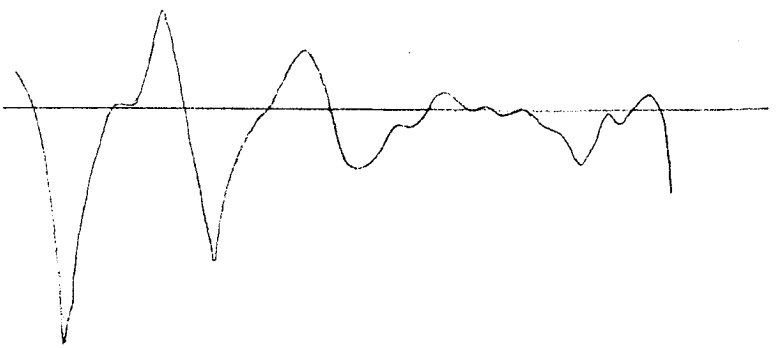


Figure 4: Waveform of Average /o/-Phoneme

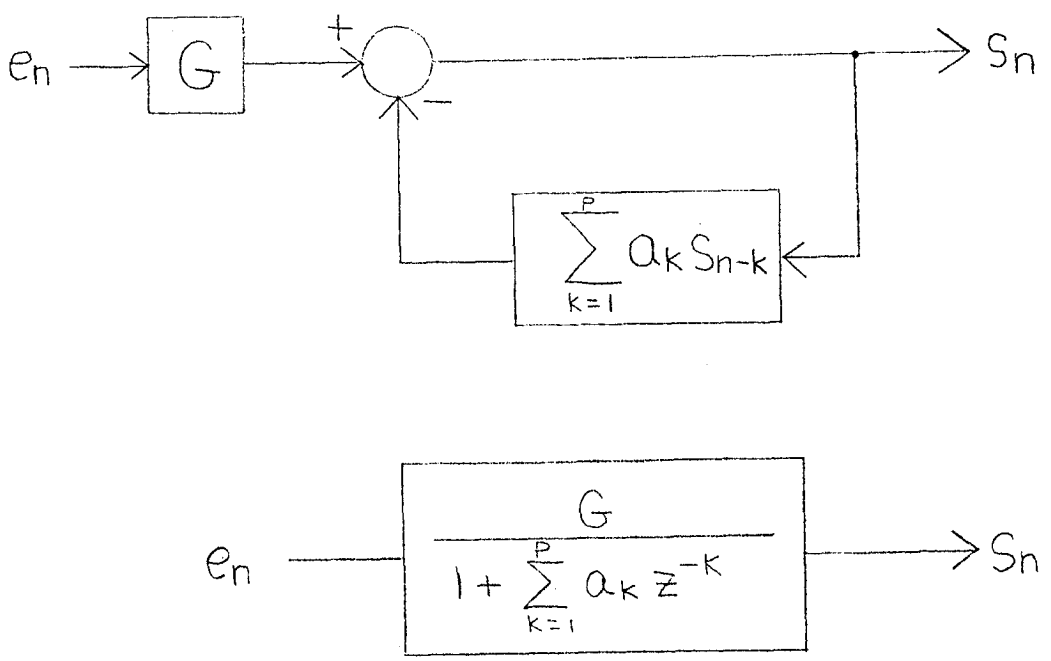


Figure 5: Speech Production Model

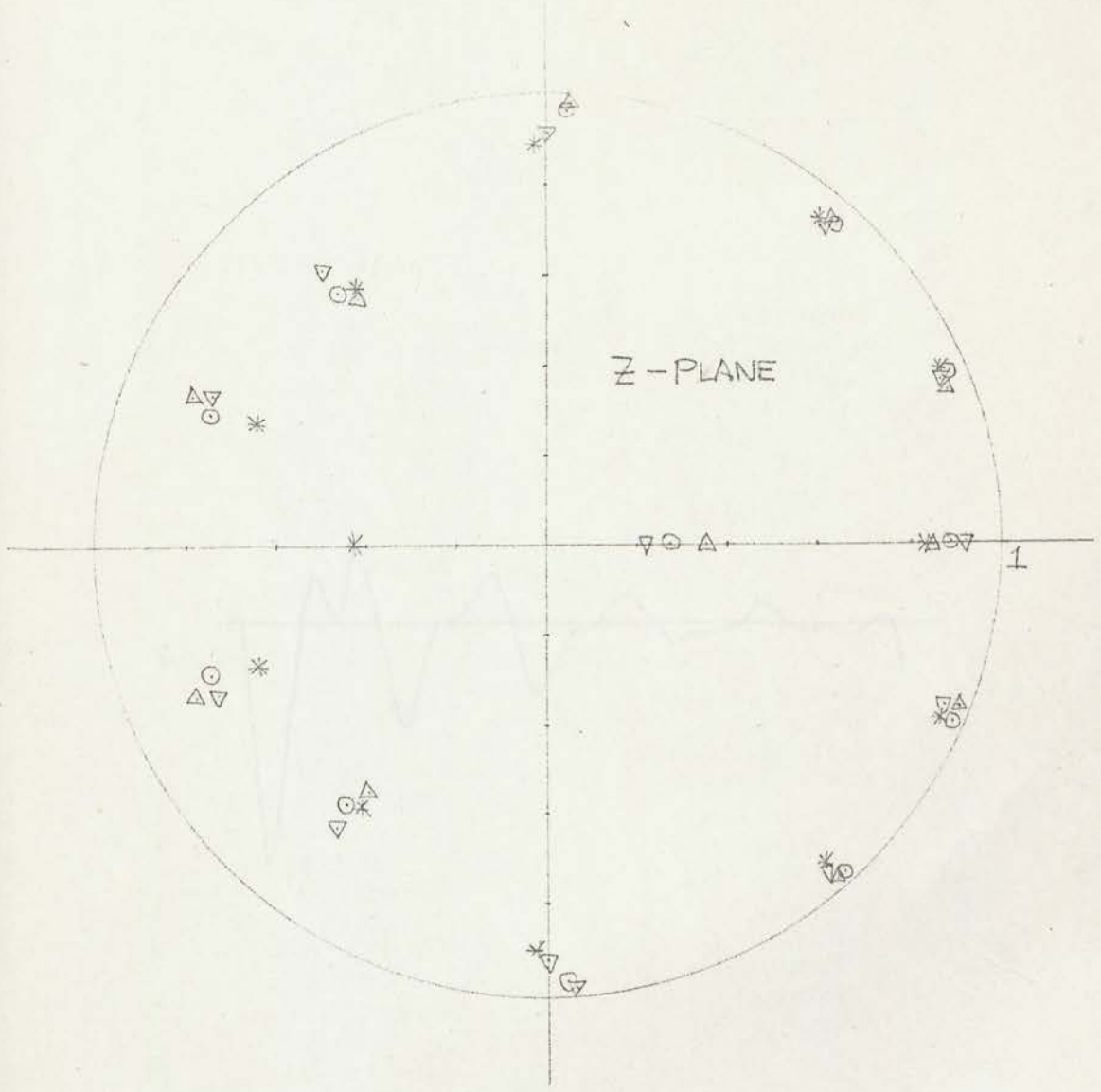


Figure 6: /o/-Phoneme Pole Positions

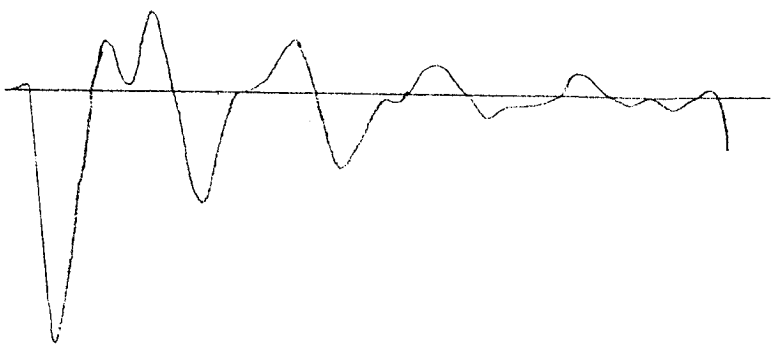


Figure 7: /o/-Phoneme Re-synthesized Using Linear Prediciton Technique.

AVERAGE POLES: /r/	/r/ in zero
0.1095182	0.0000000
0.8319680	0.0000000
0.9215992	0.2957688
0.9215992	-0.2957688
0.3618614	0.8726538
0.3618614	-0.8726538
0.0374973	0.8923460
0.0374973	-0.8923460
-0.4101246	0.4934236
-0.4101246	-0.4934236
-0.7399815	0.3791264
-0.7399815	-0.3791264

AVERAGE POLES:	/o/ in zero
0.4424442	0.0000000
0.8924758	0.0000000
0.8818176	0.3759260
0.8818176	-0.3759260
0.6180211	0.7346640
0.6180211	-0.7346640
0.0712229	0.9504004
0.0712229	-0.9504004
-0.3927494	0.5365630
-0.3927494	-0.5365630
-0.7707663	0.3579074
-0.7707663	-0.3579074

AVERAGE POLES:	/ə/ in one
-0.3955063	0.0000000
0.8973941	0.0000000
0.8862568	0.3524979
0.8862568	-0.3524979
0.7082098	0.6759152
0.7082098	-0.6759152
0.1254074	0.7433629
0.1254074	-0.7433629
-0.3739439	0.8015275
-0.3739439	-0.8015275
-0.7738258	0.4608557
-0.7738258	-0.4608557

AVERAGE POLES:	/n/ in one
0.1288262	0.0000000
0.9398875	0.0000000
0.9091274	0.2080333
0.9091274	-0.2080333
0.3553351	0.7633860
0.3553351	-0.7633860
0.0310725	0.8432968
0.0310725	-0.8432968
-0.4997548	0.7668499
-0.4997548	-0.7668499
-0.6923749	0.2463440
-0.6923749	-0.2463440

Figure 8: Phoneme Average Pole Positions
(Real and Imaginary Parts)

AVERAGE POLES: /o/ in four

-0.0800432	0.0000000
0.8499187	0.0000000
0.9095355	0.2954679
0.9095355	-0.2954679
0.7488239	0.5771269
0.7488239	-0.5771269
-0.1432497	0.9171206
-0.1432497	-0.9171206
-0.1150963	0.4009684
-0.1150963	-0.4009684
-0.7770389	0.4377142
-0.7770389	-0.4377142

AVERAGE POLES: /r/ in four

0.0367483	0.0000000
0.9239879	0.0000000
0.9072701	0.3203424
0.9072701	-0.3203424
0.7450810	0.6106638
0.7450810	-0.6106638
-0.0354407	0.9172264
-0.0354407	-0.9172264
-0.3561942	0.5765684
-0.3561942	-0.5765684
-0.7481564	0.3633996
-0.7481564	-0.3633996

AVERAGE POLES: /aI/ in five

0.2358079	0.0000000
0.8309318	0.0000000
0.7928910	0.3986267
0.7928910	-0.3986267
0.6061636	0.7159086
0.6061636	-0.7159086
-0.1732347	0.9311450
-0.1732347	-0.9311450
-0.0045646	0.4669212
-0.0045646	-0.4669212
-0.7565032	0.4058339
-0.7565032	-0.4058339

AVERAGE POLES: /v/ in five

0.0944021	0.0000000
0.9381915	0.0000000
0.8509296	0.3747892
0.8509296	-0.3747892
0.3573231	0.5585222
0.3573231	-0.5585222
0.3100193	0.8578342
0.3100193	-0.8578342
-0.2244939	0.9066396
-0.2244939	-0.9066396
-0.8163342	0.3135582
-0.8163342	-0.3135582

Figure 8 (cont'd)

AVERAGE POLES: /aI/ in nine

-0. 0746314	0. 0000000
0. 8944305	0. 0000000
0. 8466437	0. 3432857
0. 8466437	-0. 3432857
0. 4886777	0. 7208534
0. 4886777	-0. 7208534
0. 2490693	0. 7802069
0. 2490693	-0. 7802069
-0. 3547992	0. 8297969
-0. 3547992	-0. 8297969
-0. 7664526	0. 4549700
-0. 7664526	-0. 4549700

AVERAGE POLES: /n/ in nine

0. 2641472	0. 0000000
0. 9628884	0. 0000000
0. 9044339	0. 2834057
0. 9044339	-0. 2834057
0. 3987909	0. 7071466
0. 3987909	-0. 7071466
-0. 0313535	0. 8521464
-0. 0313535	-0. 8521464
-0. 4068614	0. 8035302
-0. 4068614	-0. 8035302
-0. 8369936	0. 3866441
-0. 8369936	-0. 3866441

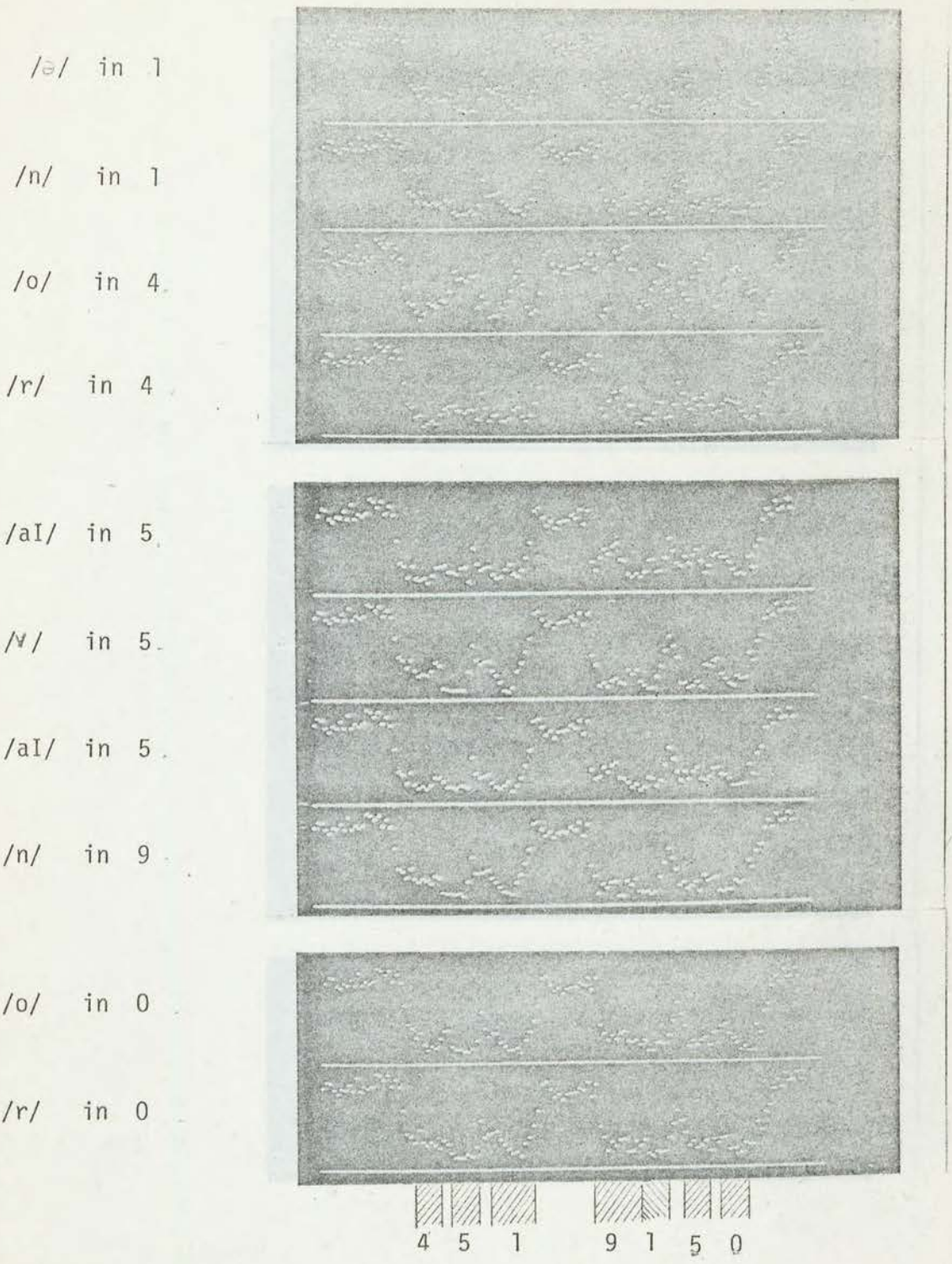


Figure 9(a): Cepstral Matched Filter for 10 Phonemes of Speaker A Matched to Test Phrase of (a) Speaker A, (b) Speaker B, (c) Speaker C.

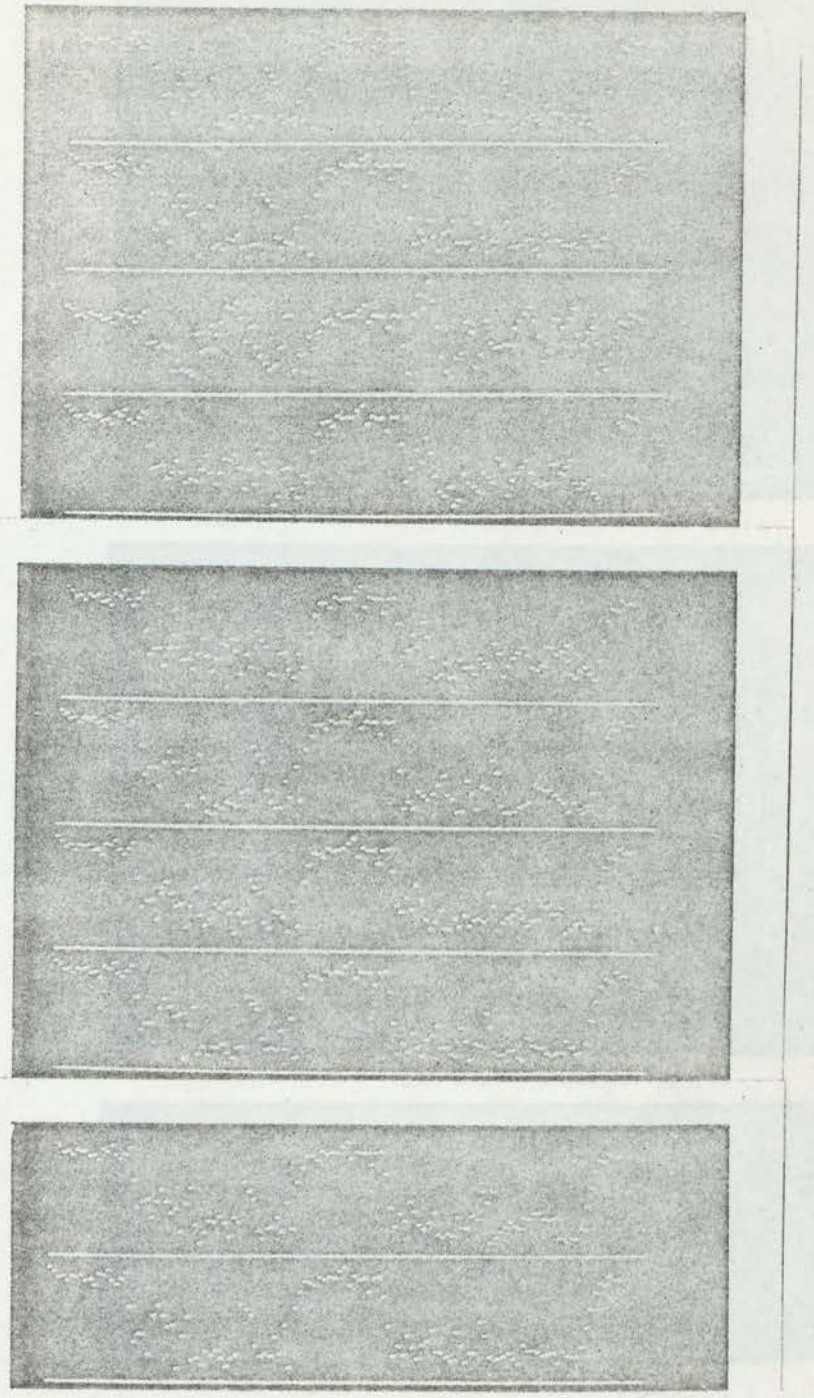


Figure 9 (b)

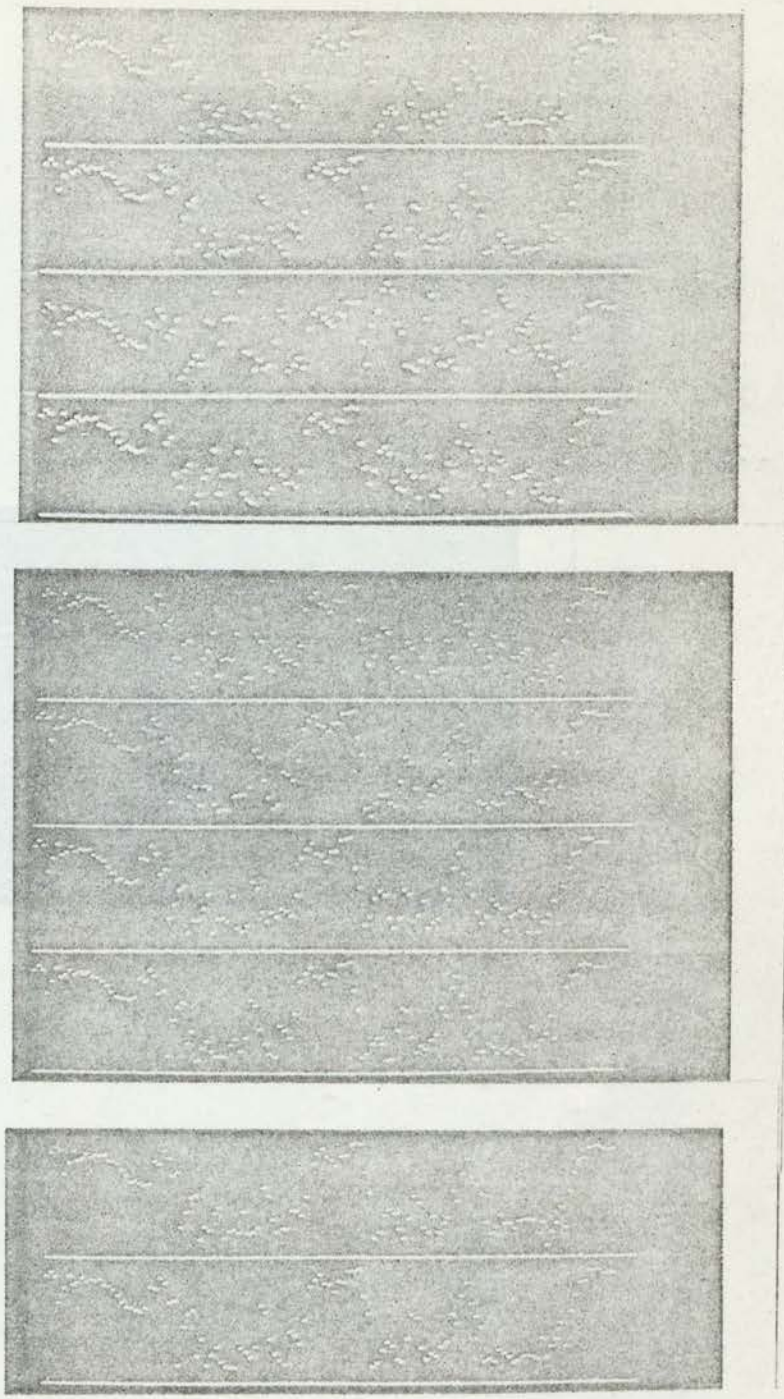


Figure 9 (c)

(a)

(b)

(c)

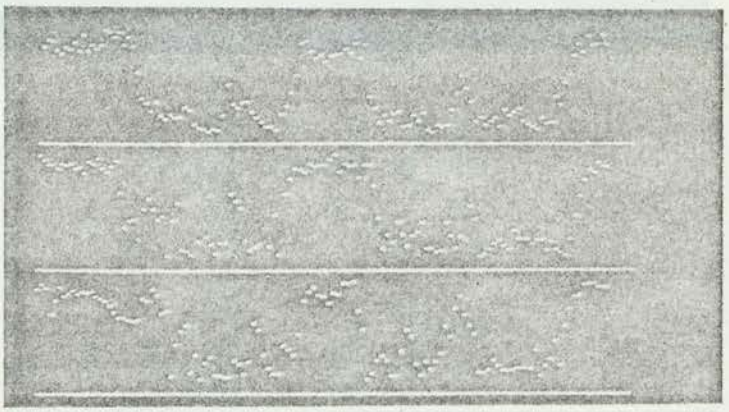
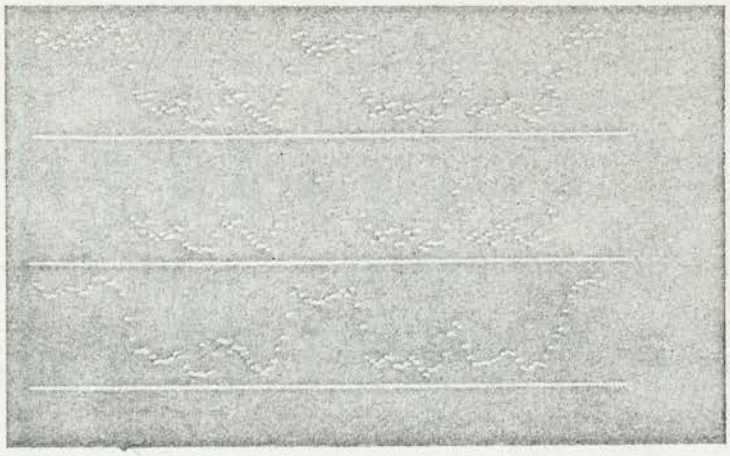


Figure 10: Cepstral Matched Filter Output for /ə/-Phoneme of "1" Matched to (a) Speaker A, (b) Speaker B, (c) Speaker C.

(a)

(b)

(c)



(ə) phoneme of "l"

(a)

(b)

(c)

(r) phoneme of "o"

Figure 11: Cepstral Matched Filter Output: (a) Raw Output
(b) Threshold Output (c) Smoothed Output

		SPEAKER A		SPEAKER B		SPEAKER C	
Item	Phoneme	Spread	Ratio .	Spread	Ratio .	Spread	Ratio .
1	/r/ in "0"	4571	.326	8234	.370	7594	.427
2	/o/ in "0"	4400	.353	6567	.382	6591	.436
3	/ə/ in "1"	4345	.325	7930	.357	7641	.415
4	/n/ in "1"	4456	.343	7949	.368	7315	.423
5	/o/ in "4"	5772	.549	5409	.649	7003	.605
6	/r/ in "4"	4413	.376	5501	.438	7269	.472
7	/aI/ in "5"	4917	.373	5535	.448	7636	.487
8	/v/ in "5"	4475	.333	7362	.371	6616	.427
9	/aI/ in "9"	4575	.346	6606	.403	7550	.313
10	/n/ in "9"	4452	.333	8374	.356	7668	.415

Figure 12: Table of Spread and Ratio Test Results

DATE DUE
DATE DE RETOUR

JUN 21 1988			

Université du Québec
Institut national de la recherche scientifique