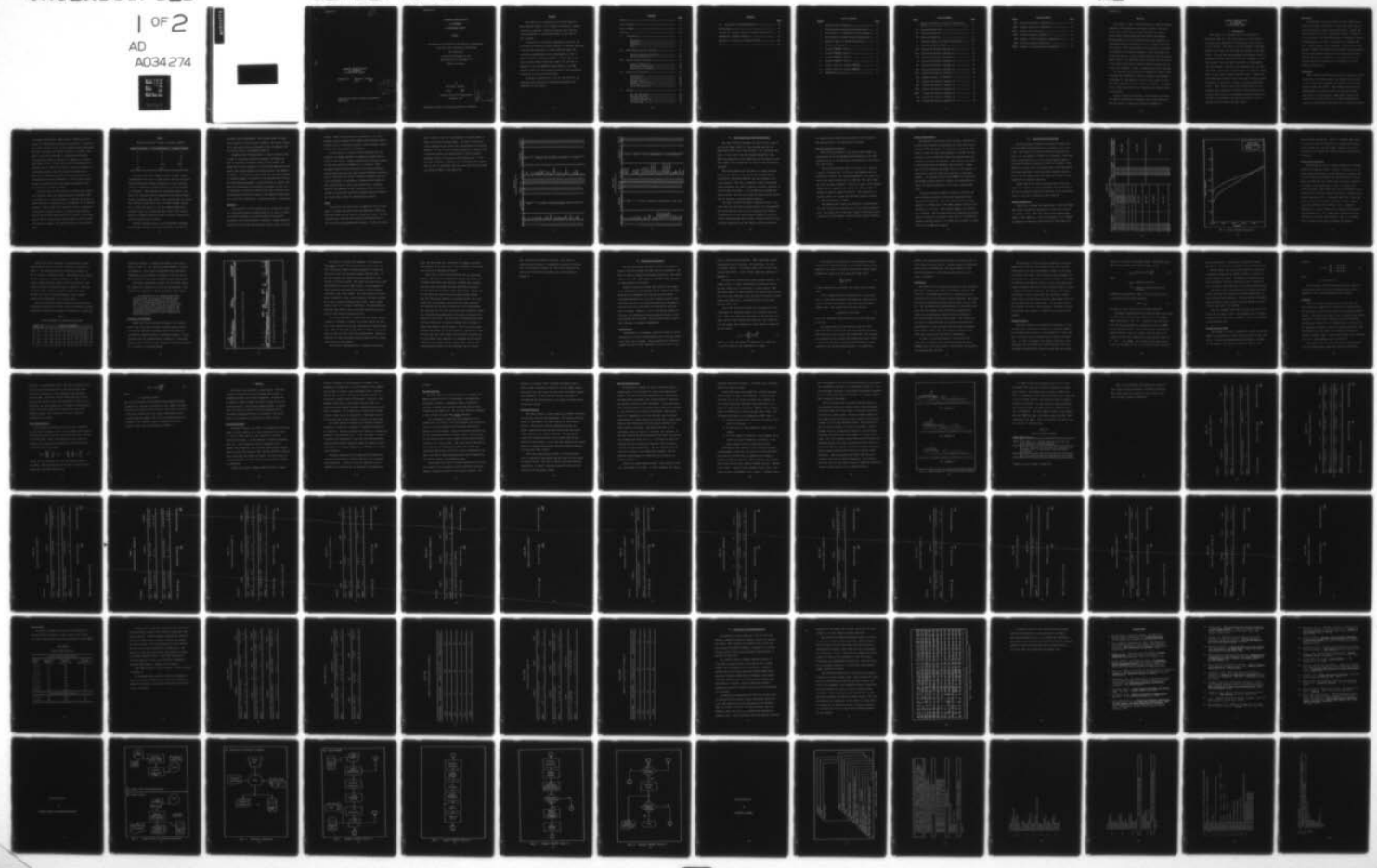
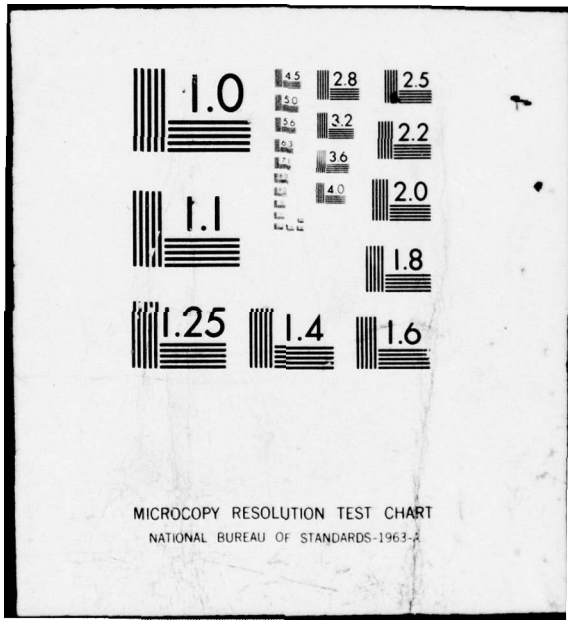
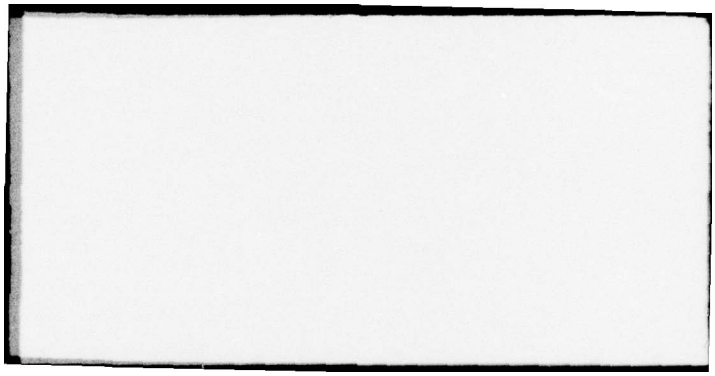
AD-A034 274 AIR FORCE INST OF TECH WRIGHT-PATTERSON AFB OHIO SCH--ETC F/G 17/2
COMPUTER IDENTIFICATION OF PHONEMES IN CONTINUOUS SPEECH.(U)
DEC 76 W R HENSLEY

UNCLASSIFIED GE/EE/76-24 NL

1 OF 2
AD
A034274




1.0
1.1
1.25
1.4
1.6
1.8
2.0
2.2
2.5
2.8
3.2
3.6
4.0
4.5
5.0
5.6
6.3
7.1
8.0
9.0
MICROCOPY RESOLUTION TEST CHART
NATIONAL BUREAU OF STANDARDS-1963-A



GE/EE/76-24

COMPUTER IDENTIFICATION OF PHONEMES IN CONTINUOUS SPEECH

GE/EE/76-24 William R. Hensley
Capt. USAF

Approved for public release; distribution unlimited

(See form 1473)

GE/EE/76-24

COMPUTER IDENTIFICATION OF PHONEMES IN CONTINUOUS SPEECH

THESIS

Presented to the Faculty of the School of Engineering

of the Air Force Institute of Technology

Air University

in Partial Fulfillment of the

Requirements for the Degree of

Master of Science

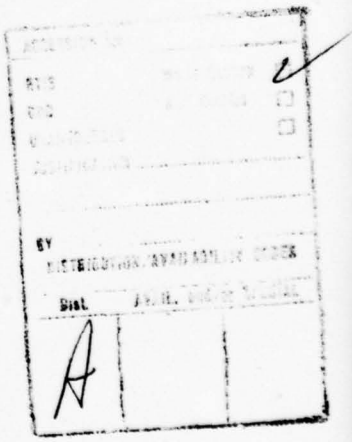
by

William R. Hensley

Capt. USAF

Graduate Electrical Engineering

December 1976



Approved for public release; distribution unlimited

Preface

This thesis is a continuation of the work begun by Major Ralph W. Neyman in the attempt to establish a phoneme recognition algorithm. Spatial filtering which has been used successfully in classifying words, is also used in this research.

I would like to gratefully acknowledge the advice and assistance provided by my thesis advisor, Dr. Matthew Kabrisky. I owe special appreciation to Major Ralph W. Neyman who insured my understanding of his investigation so that a smooth transition could be effected. I would like to express special thanks to William B. Hall, Jr. and Jack D. Capehart of the Analog/Hybrid System Branch of the ASD Computer Center for their patient support in the preliminary processing of the analog speech data.

My greatest appreciation is for my wife, Phyllis, who patiently and unselfishly encouraged me throughout the completion of this thesis.

Contents

Contents

List of Figures

List of Tables

List of Tables

Abstract

The purpose of this investigation was to identify phoneme segments as they appeared in continuous speech. The input device was an audio tape recorder from which the analog speech signal was digitized and fast Fourier transformed. The amplitudes of this transformed signal were combined in a logarithmic manner and printed out in a 16 channel digitized spectrogram. Sixty-one prototypes were selected to represent the phonemes of the English language. These prototypes were stored and used in a running crosscorrelation with the unknown speech signal. The amplitude values resulting from the correlation process were used to predict phoneme locations and the values were compared in order to identify the correct phoneme.

The phonemes were selected from Speaker A's speech signal and tests were conducted to analyze utterances from Speaker A and Speaker B. For Speaker A, location was rated at 81 percent while identification was rated at 45 percent. For Speaker B, location was found to be 70 percent with identification at 40 percent.

Spatial filtering techniques, uniform length prototypes, and various normalization procedures were investigated next with the result of improving location for Speaker B.

COMPUTER IDENTIFICATION OF PHONEMES IN CONTINUOUS SPEECH

I. Introduction

This thesis is a continuation of a study begun by R.W. Neyman (Ref 21). The long term goal of this study is to advance the possibility of unrestricted speech recognition by machine. For centuries man has dreamed of building machines that could hear and speak the language of men (Ref 18:45). For more than three decades, concentrated efforts of combined scientific disciplines have been expended to solve this problem. None have been successful in understanding continuous speech. Some men, after spending years of fruitless effort, have grown so discouraged as to label all energy spent in this area as wasted time (Ref 23:41). Others have said, "Engineers working in this area with continuous speech recognition in mind, have a right to be discouraged" (Ref 18:58). Nevertheless, just as men watching birds fly for centuries were inspired to countless trials before success, the mere fact that man can understand continuous speech in a variety of environments, motivates the attempt to build a machine that can achieve the same result.

Motivation

In experiments involving speech and other communication modes like typing, information is transferred almost twice as fast with speech as without speech (Ref 28:41). Neyman (Ref 21:2) computes the rate of information transfer in speech to be on the order of 50 bits/second based on Flanagan's estimate of approximately five bits of information per phoneme (Ref 7:4). Besides speed, other advantages of being able to communicate verbally with machine are constantly being expounded. Man will have both his hands free to do required work while actively passing on information to the system, and a substantial amount of training can be eliminated in the man-machine interface area.

Background

While several isolated word recognition systems for small vocabularies with known speakers are commercially available, it may be years before machines can recognize normal conversational speech (Ref 28:40). The problems associated with understanding of continuous speech are much more complex than those of isolated speech. Experiments indicate that one-fourth to one-half the words in normal conversational speech are unintelligible when taken out of context and heard

in isolation (Ref 28:41). This seems to indicate that the system for understanding continuous speech must, of necessity, use context related rules. In fact, psychoacoustic experiments show that listeners use semantic, syntactic, prosodic, pragmatic, and acoustic knowledge to understand acoustically corrupted speech (Ref 19 and Ref 27). Whether one accepts this theory or not, it seems clear that some system must be employed that can "hear" and perform a one-to-one mapping to a perception space so that the system can "know" what it heard even if additional analysis must be performed before the meaning and use are determined. A look at some current methods of analysis reveals that memory requirements limit the efficiency of today's systems.

Since the most accurate system of isolated word recognition available today uses template matching techniques (Ref 11 and Ref 29), it seems reasonable to consider the amount of memory required to represent various breakdowns of the English language. Table I (Ref 8:91) shows the relative frequency of occurrence of sounds and words in ordinary spoken English. One can see that 732 words constitute 75 percent of the words used in normal conversational speech, whereas only 19 sounds are required to make up the same percentage of total sounds used.

TABLE I

Relative Frequency of Usage of Sounds and Words

Number of Sounds	% of Time Used	Number of Words
4	25	9
9	50	69
19	75	732
--	78.6	1027
40+	100	--

As long as the total number of words is small, memory considerations will not be a prime factor, but for continuous speech systems with sizable vocabularies, a more efficient coding or decomposition system would be to use the phonemes as prototypes. This approach has considerable appeal, and much of the automatic speech research has concerned automatic phoneme recognizers (Ref 28:48). Even systems that use stored word templates could profit from a reliable phoneme recognizer to reduce the amount of time for template matching by selective recall of stored words (Ref 28:45). The ultimate hope for a phoneme recognizer would, of course, eliminate the need for word prototype storage.

Another motivating force to use the phonemic breakdown and prototype storage is the ease with which a correlation

procedure can be implemented. This process holds the additional promise of being closely related to the process carried out in the human cortex as proposed by Fano and Huggins (Ref 15), Cherry (Ref 3), and Kabrisky (Ref 12).

McLachlan (Ref 20) demonstrates a visual correlator that is able to locate and identify prototypes, and Neyman was able to construct an auditory analog of this system. His method was to first construct a digital spectrogram that would display the energy spectrum of successive short time-segments of speech. It is generally agreed that the information needed to recognize speech is contained in the spectrum (Ref 17:115). The spectrogram development is explained fully in Chapter III. After the spectrogram was developed, prototypes for the various phonemes of speech were selected and then correlation was accomplished with decisions based on the maximum crosscorrelation value that occurred over a specified length of utterances.

Objective

The objective of this research was to continue the original investigation begun by Neyman (Ref 21) in order to locate and identify phonemes in continuous speech using pattern recognition and crosscorrelation techniques. Neyman achieved excellent location and identification results using a 10 class

problem. When the prototypes were extended to a 47 class problem, location dropped only slightly while phoneme identification fell to 34 percent; however, correct category identification was only reduced to 62 percent.

In the analysis of results, certain phonemes were not looked for as Neyman believed an adequate prototype did not exist for that sound. Neyman suggested that follow-on studies in this area extend the phoneme set to include at least nasalized vowels and some prototypes from ending and beginning phonemes that were the same sound but different structure. He also suggested that spatial filtering techniques that had proven successful in recognition of hand-written letters by Carl and Hall (Ref 2) and in the recognition of isolated words for two speakers by Daily and Sutton (Ref 4) be incorporated to extract the important information while minimizing the "noise" that clouds the identification process.

Scope

The scope of the project was to expand the set of prototypes to include nasalized vowels and additional ending and beginning sounds and at least one combination sound. Low-pass filtering was tried next and required the modification of the previously used normalization process. It was also neces-

sary to select a new set of prototypes of uniform length in order to use the filtering scheme. Two sets of seven sentences composed by the author were analyzed with no filtering applied. One set of seven sentences spoken by a speaker with a different dialect was analyzed with no filter present. Low-pass filters of varying size were tested next. In all the cases analyzed, a complete set of prototypes was assumed. The two sets of prototypes that were used and their key words are listed in Table II and Table III.

Table II
Expanded Phoneme Set

Key Word	Computer Representation	Length (Sec)	Key Word	Computer Representation	Length (Sec)
1. eve	.I	.154	31. no	N	.154
2. it	I	.102	32. sing	NG	.192
3. met	<E	.102	33. we	W	.102
4. at	$E	.115	34. you	Y	.102
5. not	A	.154	35. read	R	.102
6. all	¢	.154	36. late	L	.102
7. obey	O	.154	37. will	.L	.102
8. put	U	.154	38. when	HW	.102
9. boot	OO	.102	39. church	CH	.102
10. up	-A	.102	40. judge	DZ	.154
11. ate	AE	.192	41. he	H	.102
12. came	EI	.154	42. ahead	XH	.102
13. ate	E	.102	43. vote	V	.102
14. church	UR	.154	44. then	TH	.115
15. I	$I	.154	45. zoo	Z	.102
16. boy	¢I	.192	46. pleasure	ZH	.102
17. out	AU	.192	47. but	.B	.077
18. new	IU	.192	48. feel	F	.102
19. but	UH	.102	49. thin	TS	.102
20. urn	ER	.154	50. see	S	.102
21. late	TT	.051	51. she	SH	.102
22. judge	J	.102	52. instead	ST	.154
23. or	OR	.154	53. pay	P	.102
24. embers	.E	.102	54. to	T	.102
25. and	.A	.192	55. key	K	.102
26. on	.O	.154	56. came	.C	.051
27. boon	.W	.154	57. Christ	CR	.102
28. under	.U	.102	58. tight	.T	.102
29. in	.N	.154	59. be	B	.051
30. me	M	.077	60. day	D	.051
			61. go	G	.051

Table III
Revised Phoneme Set

Key Word	Computer Representation	Length (Sec)	Key Word	Computer Representation	Length (Sec)
1. eve	.I	.154	31. journal	J	.077
2. it	I	.102	32. journal	JE	.077
3. met	>E	.102	33. journal	JL	.077
4. at	$E	.115	34. of	OF	.077
5. not	A	.154	35. of	F	.077
6. all	¢	.154	36. speech	P	.077
7. obey	O	.154	37. speech	SE	.077
8. put	U	.154	38. speech	CH	.077
9. boot	OO	.102	39. and	.A	.077
10. up	-A	.102	40. hearing	H	.077
11. ate	AE	.192	41. hearing	HE	.077
12. came	EI	.154	42. research	R	.077
13. ate	E	.102	43. research	RE	.077
14. church	UR	.154	44. breeds	B	.077
15. I	$I	.154	45. recognition	RA	.077
16. boy	¢I	.192	46. recognition	RK	.077
17. out	AU	.192	47. recognition	RG	.077
18. vitamins	V	.077	48. basic	BA	.077
19. vitamins	VI	.077	49. basic	BE	.077
20. vitamins	VT	.077	50. basic	BC	.077
21. vitamins	VM	.077	51. computation	C	.077
22. vitamins	VU	.077	52. computation	CP	.077
23. vitamins	VN	.077	53. computation	CY	.077
24. vitamins	VS	.077	54. computation	SN	.077
25. taste	T	.077	55. obey	OB	.077
26. taste	TA	.077	56. obey	AA	.077
27. taste	TE	.077	57. judge	DZ	.077
28. good	G	.077	58. not	NT	.077
29. good	GU	.077	59. closure	ZH	.077
30. good	GD	.077	60. will	W	.077
			61. cause	Z	.077

II. Data Acquisition and Pre-Processing

The same recording equipment was used for this study as was used by Neyman (Ref 21). The recorder used was the Ampex Model F4450 stereo tape recorder. The recordings were made in a very quiet room with minimum background noise. These recordings were easily understood by the human ear and were judged to be satisfactory for input to the digitization equipment.

The speech samples were recorded at a normal speaking level on one channel of the stereo tape recorder while a periodically interrupted 2000 Hz tone was recorded on the second channel. The tone, provided by a Model III Wavetek signal generator, was used to indicate recording intervals for the digitization problem. A one-second tone preceded each speech record. The tone was turned off during speech recordings to eliminate crosstalk between channels.

The tape recordings provided a permanent record in the event that the digitization process had to be reaccomplished. The recordings were also an aid in analyzing the computer representations of the various speech samples to ascertain exactly which phonemes were uttered. Another benefit of this recording system was that the signals could be recorded at

one speed and then played back at another, thus increasing the sampling rate in the digitization procedure.

Analog-to-Digital Conversion

The initial processing of the analog speech signal was accomplished by the Analog/Hybrid Systems Branch of the ASD Computer Center in the same manner as processing of the Neyman data (Ref 21:16).

The recording had been made at a 7½ ips rate. By using a speed of one-half that (3 3/4 ips), the sampling rate was effectively doubled. The accepted bandwidth of the amplifiers used in the analog system was 0 to 2500 Hz. The audio signal was first low-pass filtered to 2500 Hz to insure a band limited signal, and the sampling rate was set at 5 KHz in order to satisfy the Nyquist sampling criteria. This resulted in an over-all effect of a signal that had been low-pass filtered to 5 KHz and sampled at 10 KHz.

The sampled input signal was amplified to approximately 100 volts to make more effective use of the analog representation. The signal was fed through a Comcor Ci-5100 high speed interface to a Xerox Sigma 7 general purpose digital computer.

Signal Transformation

The digitized analog speech data was then converted into an equivalent frequency representation by using fast Fourier transform (FFT) techniques. By selecting a relatively wide window to input the time domain samples to the FFT, the time resolution of the transformed signal was enhanced while the frequency resolution was degraded. This selection was based on the previous work of Oppenheim (Ref 22:57-62). Neyman (Ref 21:17-18) selected the window size to be 128 samples in length and to step the window thru the data in 128 sample segments. An in-house program called AMPSPC was used by the Analog/Hybrid System Branch to compute the forward FFT and return the absolute magnitude of the values computed (Ref 9:42).

Using the conjugate symmetry property of the FFT, the above procedure resulted in 64 discrete amplitude values separated by 78.125 Hz. Since the original data was being sampled at a 10 KHz rate, a 128 sample segment occurred every $128/10^4$ sec or 12.8 ms. These sample segments are referred to as "frames". The resulting data was converted into decimal form by dividing by the largest array value in a transformed sentence and then written on a library tape (L-tape) in proper format for the CDC-6600 computer.

III. Digital Signal Processing

The pre-processed information received on L-tape from the Analog/Hybrid System Branch was contained in an m x 64 array. The length of the speech utterance determined the value of m, the number of frames in an utterance. Since each frame represented 12.8 ms of the original speech sample, a one-second utterance would have 1/.0128 or 78 frames. Each element in a frame was a four decimal digit that represented the signal amplitude in that particular frequency channel. Each of the 64 channels had 78 Hz separation between center frequencies of adjacent channels.

Neyman (Ref 21:19) used a restructuring of this data format in a manner that would approximate the sensitivity of the ear to frequency changes by simulating to the logarithmic nature of the ear at frequencies above 1000 Hz.

Channel Compression

Table IV is included for completeness to show how Neyman (Ref 21:20) grouped the frequencies to reduce the original 64 channels to 16. Since the energy of the channels were added in each subgroup, it was not necessary to use standard preemphasis of 6db/octave (Ref 26:311) for the higher fre-

Table IV

Speech Frequencies

Center Frequency Original Data	Center Frequency Original Data	Center Frequency Original Data	Center Frequency Original Data
78.125	78.125	2578.125	
156.250	156.250	2656.250	
234.375	234.375	2734.375	
312.500	312.500	2812.500	2812.500
390.625	390.625	2890.625	
468.750	468.750	2968.750	
546.875	585.940	3046.875	
625.000		3125.000	
703.125	742.188	3203.125	
781.250		3281.250	
859.375	898.440	3359.375	
937.500		3437.500	
1015.625	1132.810	3515.625	3554.690
1093.750		3593.750	
1171.875		3671.875	
1250.000		3750.000	
1328.125	1445.310	3828.375	
1406.250		3906.250	
1448.375		3984.375	
1565.500		4062.500	
1640.625	1793.380	4140.625	
1718.750		4218.750	
1796.875		4296.875	
1875.000		4375.000	
1953.125		4453.125	
2031.250	2226.560	4531.250	4453.125
2109.375		4609.375	
2187.500		4687.500	
2265.625		4765.625	
2343.750		4843.750	
2421.875		4921.875	
2500.00		5000.000	

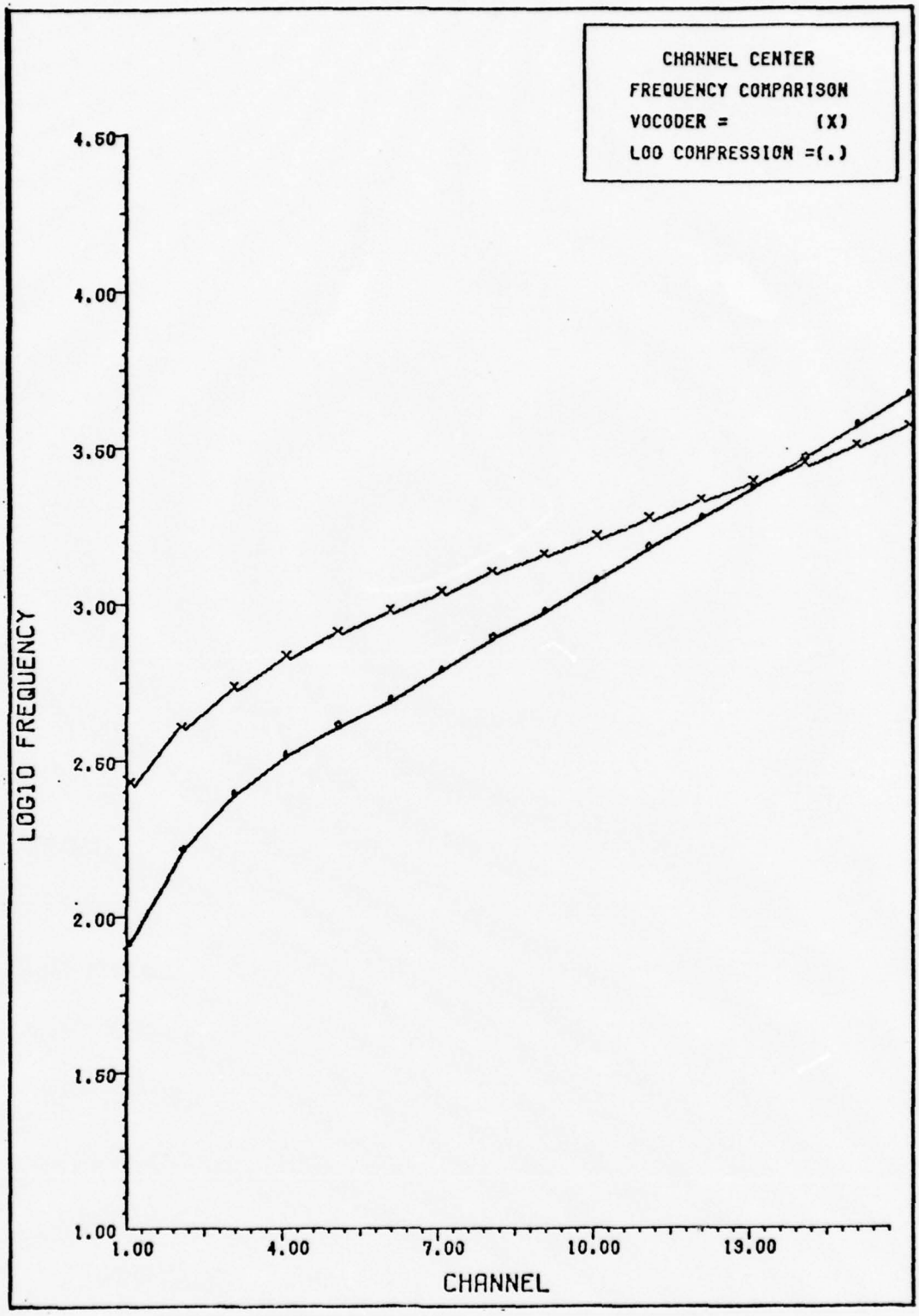


Fig. 1 Channel Center Frequencies

quencies before processing. Figure 1, by Neyman (Ref 21:21), is included to show the comparison of frequency distribution of this system and a vocoder system. Reduction from 64 to 16 channels also reduced the computer storage requirement by 75 percent.

Spectrogram Development

Spectrograms are used in pattern recognition to visually display the frequency context of speech. Although it is not known exactly what accuracy can be achieved in visually reading high quality spectrograms, the extensive work of Potter, Kopp, and Green (Ref 24) indicate that sufficient information is encoded in the spectrogram in order to reproduce the original message. More recent tests on the usefulness of the spectrogram in continuous speech recognition, indicate visual reading successes of 85 - 100 percent (Ref 13:6). Such high success rates were attainable only when the test subjects were given additional cues; however, this is viewed as comparable to a person listening to a message with "context" and associated cues. If a spectrogram contains sufficient information for visual interpretation, then it is feasible that a computer may be able to decipher the message.

Neyman (Ref 21:23) developed a limited-detail digital spectrogram by using an overprint technique as specified in Table V. His program printed the spectrogram adjacent to the 16 channels of numerical data. Each channel had a threshold for overprint; a round-up procedure was used to form integer values and these integer values corresponded to the overprint "level of darkness" figures of Table V. Although the array values could be studied to observe the energy changes and locate low energy phonemes, a more complete depiction was considered to be of great value.

Since this research program ultimately performed energy normalization before the decision space was reached, an energy normalized spectrogram was judged as invaluable in finding the

Table V

Overprint Symbols for Speech Spectrograms

Number of Overprints	LEVEL OF DARKNESS									
	0	1	2	3	4	5	6	7	8	9
1			+	x	x	x	x	x	x	x
2					-	+	0	0	0	0
3								-	-	#
4									+	+
5										*

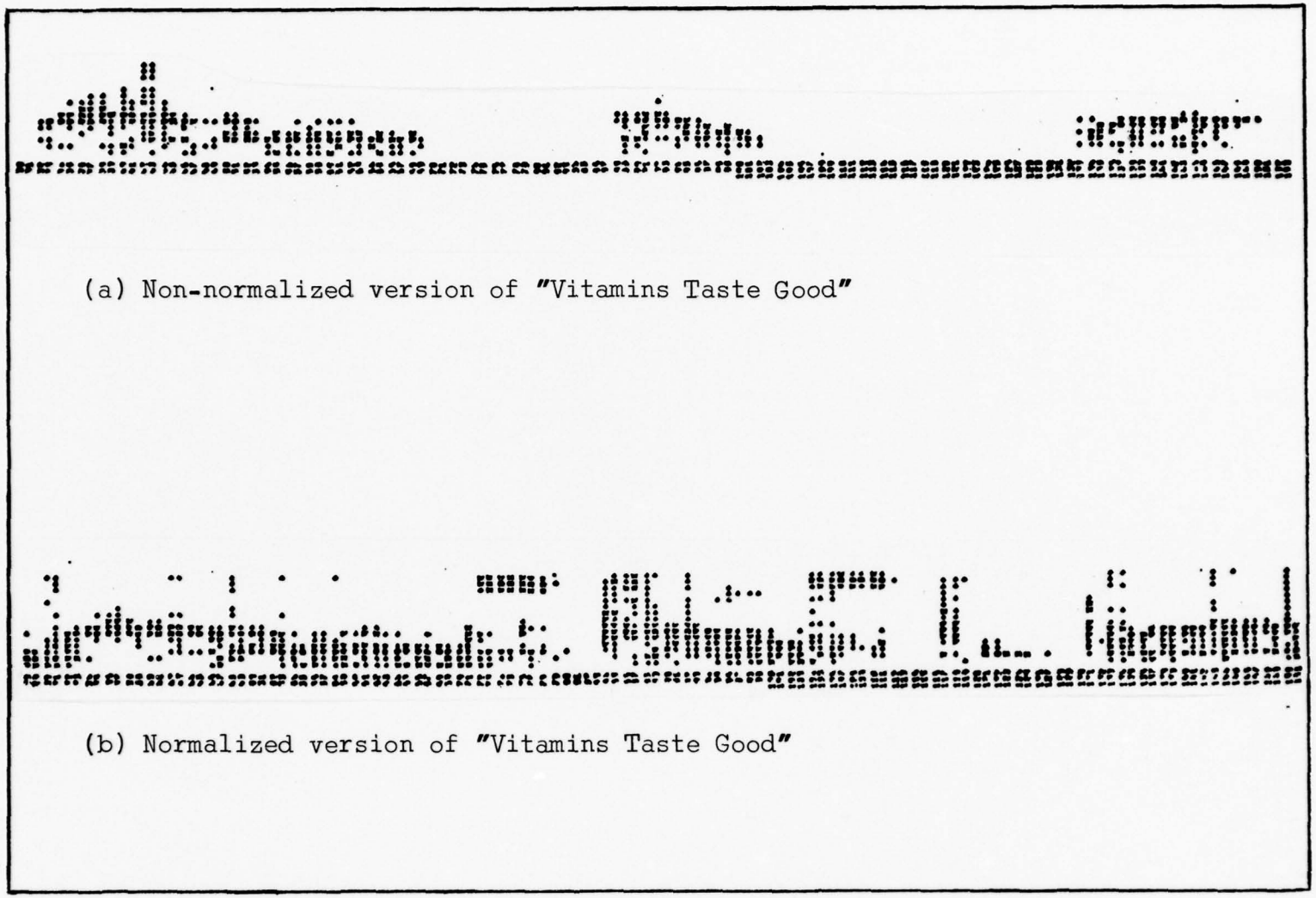
individual phonemes. A sample spectrogram of each type is shown in Figure 2. The modified program OCTAVE is included in Appendix B. This program accomplishes the logarithmic compression of the original 64 channels of data as well as the generation of the energy normalized speech spectrograms.

The author gained great insight into the speech process by studying the patterns of the various spectrograms. This confirms the conclusion of Klatt and Stevens (Ref 13:27):

> In conclusion, it is suggested that every serious worker in the area of automatic speech recognition should undertake to read spectrograms in an organized way similar to the projects that we have described. It is an excellent way of learning a great deal about speech, and it is the only way to convince yourself of the complexities involved and of the necessity for approaching the problem with more sophisticated forms of analysis.

Selection of Prototypes

Neyman (Ref 21:26) selected prototypes from isolated words since this method offered a straight forward method of selecting individual prototypes with little chance of accidentally combining frames of different phonemes. He also suggested that the phoneme-word be included in a structured sentence (Ref 21:74-75) to more adequately produce the phoneme as it occurred in continued speech.



(a) Non-normalized version of "Vitamins Taste Good"

(b) Normalized version of "Vitamins Taste Good"

Fig. 2. Non-Normalized and Normalized Spectrograms

The words of interest were embedded in the sentence "Say (word) instead." The enunciation was very precise in this setting and tended to produce phonemes of length that agreed very well with predicted lengths (Ref 8:59-67). It was also evident that there is great variability in the duration of certain sounds. The vowels and vowel-like sounds showed the most variation (Ref 26:315). A good example of vowel variation was noted by the effect of the following consonant on the vowel length. The vowel tends to be longer when followed by a final voiced consonant than when followed by a final voiceless consonant (Ref 14:18). Vowel length tends to fall in the range of 80-360 ms. The consonants are generally much shorter than vowels with many being as short as 70 ms (Ref 40:19-68 and Ref 8:59-67).

It is also important to realize that the energy concentrations or formants in vowels are constant during the duration of the sound but must have a beginning and ending transition. This is true even if the vowel is uttered in isolation since it does not start and stop instantaneously. Therefore, selection of vowel prototypes should be made from the steady-state section of the vowels.

The pictorial representations of phonemes from Potter,

Kopp, and Green (Ref 24), along with the computer generated spectrograms and an audio tape of the utterances, facilitated the selection of phoneme prototypes.

There were at least three different sets of prototypes chosen. The first set followed very closely the pattern set by Neyman (Ref 21:28) and included, as Neyman had suggested, nasalized vowel prototypes and some additional prototypes for the beginning and ending sounds (Ref 21:74). Another set of prototypes was chosen using the same procedure except that the vowels were uttered in isolated context. The sound lasted over a two-second interval and the vowel prototype was selected from the most uniform area of the spectrogram. The consonants for this set were chosen from deliberate speech with carefully enunciated words in the hope of capturing the essence of each sound. The third set of prototypes was selected from normal rate of speech sentences with no attempt to modify the speaker's speech pattern. This set of prototypes limited each phoneme to the same duration. The basis for this selection was that vowel sounds can be located consecutively for long vowels, that each part of a dipthong can be located separately and restructured by context rules, and that the uniform duration chosen for the prototypes was no shorter

than the shortest sound that can occur. This last set selection was necessary to accommodate the spatial filtering that is discussed in Chapter IV. The results obtained from the use of the different prototype sets are discussed in Chapter V.

IV. Recognition Processing

The recognition phase operates on the m x 16 arrays of digital data and includes all tasks that are performed on the data in order to complete the phoneme recognition. The upper limit on m is 500. This allows an utterance with a duration of approximately 6.25 seconds.

Neyman's recognition scheme (Ref 21:30-46) was judged to be exceptionally well designed and was changed only where necessary to accommodate the filtering routine and the increased prototype set. As in the original program, after prototype selection, the complete program from microphone to decision print out could easily be converted to near real time if desired. However, to aid in the manual analysis of the data, the normalized and non-normalized versions of the spectrograms were generated. The revised program as used in this research is included in Appendix B.

Normalization

Normalization, an extremely important concept in speech recognition, is used to help minimize some of the many variations that occur in speech. Using normalization techniques enables the use of fewer templates or special rules to rep-

resent a speech sound faithfully. These techniques include normalization by (1) velocity, (2) amplitude, (3) time, (4) speaker spectra, (5) dynamic range, and (6) noise subtraction (Ref 28:51). Each of these terms are explained in Appendix C.

In some cases normalization might actually mislead. One example occurs in faster speech where articulatory targets are less likely to be reached than in slower speech. When the faster speech is time-stretched, the target values reached will still have different values from those obtained by slower speech and might lead to the identification of the wrong phonemes (Ref 5:761).

One of the most obvious needs for normalization is the requirement for something similar to an automatic gain control. Under this amplitude normalization, the phoneme prototypes and the input word/sentence data were unit normalized for each frame. Each component of every frame is normalized by the formula

$$x_{nj} = x_j / \left[\sum_{i=1}^{16} (x_i)^2 \right]^{\frac{1}{2}} \qquad (1)$$

where x_{nj} is the normalized j^{th} component of a frame and i is used to index all the components of a frame.

To minimize the possibility of non-information bearing intervals and intentional stops in the speech utterance being changed to the point of entering the decision scheme, Neyman checked each frame by the following rule (Ref 21:31)

$$\sum_{i=1}^{16} (x_i)^2 < 0.5 \qquad (2)$$

If the inequality was satisfied, the vector was not normalized.

A unit normalization was performed next on each prototype to insure that prototypes with excessive energy did not falsely correlate with higher values than the true weaker energy terms. The normalization that was used was

$$x_{nj} / \text{length of prototype} \qquad (3)$$

since the individual frames had been previously normalized by Eq (1).

One problem that was discovered by using the rules implied by Eqs (2) and (3) was that some unvoiced fricatives and stops did not have every frame normalized. This resulted in an apparent loss of energy that caused these quiet sounds to have weak correlation values and little chance of being selected in the phoneme selection phase. To remedy this

problem, two spectrograms were printed. One used the rule of Eq (2) while the other did not. Figure 2 shows the difference between the two spectrograms. The actual method of using these spectrograms to aid in the decision process is discussed in Chapter V.

Correlation

The "heart" of this recognition process is the correlator. Basically no changes were made to the Neyman correlator (Ref 21:33-38). The method Neyman chose to accomplish the correlation was to use the discrete Fourier transform. The actual fast Fourier transform algorithm used was known as Fourt (Ref 10). The two-dimensional crosscorrelation of the model prototypes with the unknown sentence data was accomplished by taking the two-dimensional discrete Fourier transform of both the prototypes and the sentence data. The conjugate of one array of transformed data was found and point-by-point multiplication of this new array with the other transformed array yields a third array. The inverse transform of the third array produced the correlation coefficients.

In order to avoid the problem of "end effect" that occurs with correlation using discrete transforms, Neyman imbedded each of the data arrays in zeros before the transform was performed (Ref 21:35-36).

The mechanics of the correlation sequence are given by Neyman (Ref 21:36-38). The largest section that could be transformed at one time using Neyman's scheme was 48 frames of original input data. This limitation can be changed consistent with the constraints of the Fourt routine. An overlap of eight was used between sections to solve the problem with larger prototypes that did not have sufficient space to effect a complete correlation sequence. The values of the arrays are defined in such a manner that the correlation coefficients that are printed agree with the frame numbers that are printed along side the coefficients. A correlation vector was computed for each of the prototypes. Following the decision process, the sequence is repeated for the next speech segment.

Phoneme Location

The first process in the decision strategy was to find possible areas of phoneme occurrence in a sentence segment. To facilitate this decision, it was necessary to insure that the correlation value was high enough to warrant consideration. In order to determine the maximum correlation value obtainable, the prototypes were autocorrelated. Since the prototypes and speech data had been normalized, the maximum

value was a function of prototype length. The maximum value that could be obtained was found by Neyman to be

$$z_{max} = \left[(4.19 \times 10^{6})(Q) \right]^{\frac{1}{2}} \qquad (4)$$

where

z_{max} = maximum correlation

Q = number of frames defining the prototype

A phoneme was considered to exist if the correlation value z_i satisfied the following inequality

$$z_i \geq C \; z_{max} \qquad (5)$$

The value of C was chosen to be 0.86 (Ref 21:38-39).

In Neyman's program there were differing data range values for each level of the decision process, i.e., the maximum number specified by Eq (4) existed at the correlation level and was transformed by a normalizing factor (X NORM) for the prototype vector. None of the arrays contained the actual correlation value in a manner that was easy to use. A change was made that combined the normalization factors of Eq (4) and X NORM. This caused all the array values to fall in the range of 0.1 - 1.0, with the latter represent-

ing autocorrelation. On the basis of empirical results, the value of .86 was still considered a good threshold value.

Another factor that had to be considered in accepting a candidate phoneme was the number of times it occurred in a short segment of speech. If additional occurrences were to be considered, they were required to have a correlation value of greater than 96 percent of the prime location value.

The third area of consideration was to insure that additional locations fell outside the duration established for the prototype being correlated. This was done by considering high correlation values near the original maximum to be part of that occurrence of the phoneme.

Once the candidate areas were selected, the rest of the vector was set equal to zero. The maximum value of correlation was put into the vector a number of times corresponding to the prototype size.

Phoneme Classification

The program, as listed in Appendix B, selects the phoneme based on the magnitudes of the prototype vectors in the final array. The overlap allowed between prototypes is variable in the program. The following scheme was used for this

analysis

$$\text{overlap} = \begin{cases} 1 & 1 \leq Q \leq 8 \\ 2 & 9 \leq Q \leq 11 \\ 3 & 12 \leq Q \leq 15 \end{cases} \qquad (6)$$

where

Q = prototype size

The correlation coefficient arrays were also useful for studying areas where incorrect decisions had been made to determine if the correct phoneme had been located.

Filtering

Spatial filtering techniques were used by Daily and Sutton (Ref 4) to improve the recognition of isolated words. The same type of filtering was used in the prototype matching process. The decision was made to use a variable length filter inserted in the FFT where correlation was being performed. The FFT array contains 64 x 32 complex terms. An easy filter to implement consisted of replacing unwanted terms with zeros. The dimensions of the filter were varied by changing two integer variables. When no filter was desired, these variables were set to these maximum values of 64 and 32.

Experiments with the filter revealed an incompatibility with the normalization scheme that had been used without

filtering. A normalization factor had been included to bring all the correlation values back to the same general magnitude after correlation so that comparison type decisions could be made. The filter removes energy from the correlation process and this causes the normalization factors to be incorrect. No easy method exists to change the normalization factors since they would have to change with each filter dimension change. The solution was to use a different normalization procedure.

Filter Normalization

The prototype and sentence data were still normalized by time frame as before to serve as an automatic gain control. The unit normalization process of the prototype was relocated to the FFT array. Since each component of this array was complex, the normalization consisted of dividing each term of the FFT array by Eng where

$$\text{Eng} = \left[\sum_{i=1}^{64} \sum_{j=1}^{32} R_{ij}^2 + \sum_{i=1}^{64} \sum_{j=1}^{32} I_{ij}^2 \right]^{\frac{1}{2}} \qquad (7)$$

and R_{ij} and I_{ij} represent the real and imaginary terms of the array. The normalization factor that is used with this method was found empirically to be

$$\text{Good} = (175)\left[\frac{15}{Q}\right]^{\frac{1}{2}} \tag{8}$$

where

Q = prototype length

The 15/Q relationship existed because the maximum prototype length was 15, and with a prototype of this length, the maximum value for autocorrelation was 175. This value was stored in the array "Good" and is the single normalization factor in this modified program. The filter and normalization are included in the computer program of Appendix B, and the results of their use are discussed in Chapter V.

V. Results

The results are presented in three phases. The first attempts to duplicate the work of Neyman and includes an extended prototype set as Neyman suggested. In phase two an attempt is made to improve the work of phase one by correcting an error in the original Neyman program. In phase three the results of spatial filtering combined with the necessary program modification are presented. The result phases are preceded by a discussion of rating results.

Scoring Philosophy

Existing ratings of the results of recognition of various types of speech signals, as a rule, are based on the value $p = (m/n) \times 100\%$, where m is the quantity of correctly identified patterns; n is the quantity of patterns presented (Ref 6:9). Even though this rule is generally used to measure the recognition rate of speech understanding systems, there are many other measures that could be defined if desired that would cause the ratings to differ. For this reason, it is very important to insure that the exact method of scoring is understood.

Unlike the results of Neyman (Ref 21:47-72), in these

results a complete set of prototypes is assumed. This assumption is rather poor in the last phase of this chapter but was made to reflect a more meaningful score. Another measure that is used to reflect the secondary quality of a recognition system is that of "location." Location in the broadest sense means to accurately state the time in a particular speech segment in which a phoneme occurred, given that it occurred. It is important to realize that without location, there can be no recognition. This broad view of location was used in the analysis of results in this study.

The actual method of analysis also warrants attention. Generally, a fixed set of phonemes is expected and this set is looked for. Scoring is based on the success in finding the members of the predicted list. In the last phase of results it becomes more meaningful to see what was predicted before deciding what phonemes should be looked for because in many cases there is no unique combination of phonemes for a particular utterance.

Although recognition can be substantially improved by training the prototypes, training requires a lot of manual processing time. In order to keep the selection process adaptable for real-time use, no training of prototypes was

allowed.

Expanded Test Set

The first phase of results consisted of expanding the Neyman program from 47 to 61 prototypes. The additional prototypes consisted of additional ending or beginning versions of sounds and nasalized vowels that had not been included in the Neyman set. All the test words were embedded in the sentence structure "Say (word) Instead."

Just as Neyman's recognition rate dropped as he expanded from a 10 class to a 47 class problem, the recognition rate for a 61 class problem fell below that achieved by Neyman's 47 class problem. Location percentage remained high but the increased prototype set had a larger overlap region in the decision space as was evident from multiple prototype locations for a single frame. Two useful facts found during this phase were (1) combination sounds such as "st" were identified 100 percent of the time, and (2) dipthongs can be split into "short vowel-transition-short vowel" phonemes for the decision segment and recombined later.

During this first phase it became apparent that an error had existed in the preliminary signal processing throughout Neyman's analysis and much of this current research. The

problem, an incorrect shift and sample procedure within a buffer stage, essentially resulted in one 128 sample segment being used four times while the next three 128 sample segments were discarded. At this point the decision was made to reaccomplish the entire process in the hope of getting better recognition results.

Corrected Test Set

The entire process, through sampling, prototype selection, and decision stage was reaccomplished. The results reflected almost no improvement over those obtained in the previous section. This was not entirely unexpected because the Nyquist sampling rate had not been violated and the speech signal had been only slightly modified. The possibility exists that psuedo-filtering of this nature might be more beneficial than harmful. It has also been observed that speech signals can undergo considerable distortion without becoming unintelligible (Ref 16:536).

After the system had been tested, it became apparent that no substantial improvement had been made over the original Neyman system. The idea of spatial filtering grew more appealing as it seemed a maximum recognition rate had been achieved with the present system.

Pre-Filtered Test Set

As discussed in Chapter IV, before filtering could be accomplished, it was necessary to change the normalization scheme. Once the filter was designed and the normalization reaccomplished, prototypes were autocorrelated to ascertain the maximum correlation value attainable. It was discovered that unvoiced sounds of low energy content would not correlate to the same level as a similar voiced sound. This was because of a segmentation rule that had been used to prevent normalization of frames having less energy than a fixed amount. Removing this restriction from the program resulted in an almost perfect correlator. Everything that went in the correlator, came out just as it occurred. For instance, if the word "church" had been pronounded "ch-ur-ch", the correlation scheme would print "ch h ur ch h" with the extra h's representing the unintentional aspiration that occurred as a result of strong enunciation. The only problem with this correlation scheme is the segmentation problem. The two different spectrograms that represent this situation are shown in Figure 2.

Figure 2(a) shows "energy groups." These groups in this case just happen to be words. In other examples, the groups

represent individual syllables. In either case, the groups contain at least one vowel.

Figure 2(b) shows every sound that occurred including throat noise, lip noise, and breathing. A rule was used such that (1) all the groups of Figure 2(a) existed, and (2) at most each group could have associated with it six frames on either side of the group. Markoul used a similar device for detection of silence and gaps (Ref 1:249). The following rules help to solve ambiguities (Ref 25:85- 4)

1) Fricatives often have a short dip in energy at the start of frication.

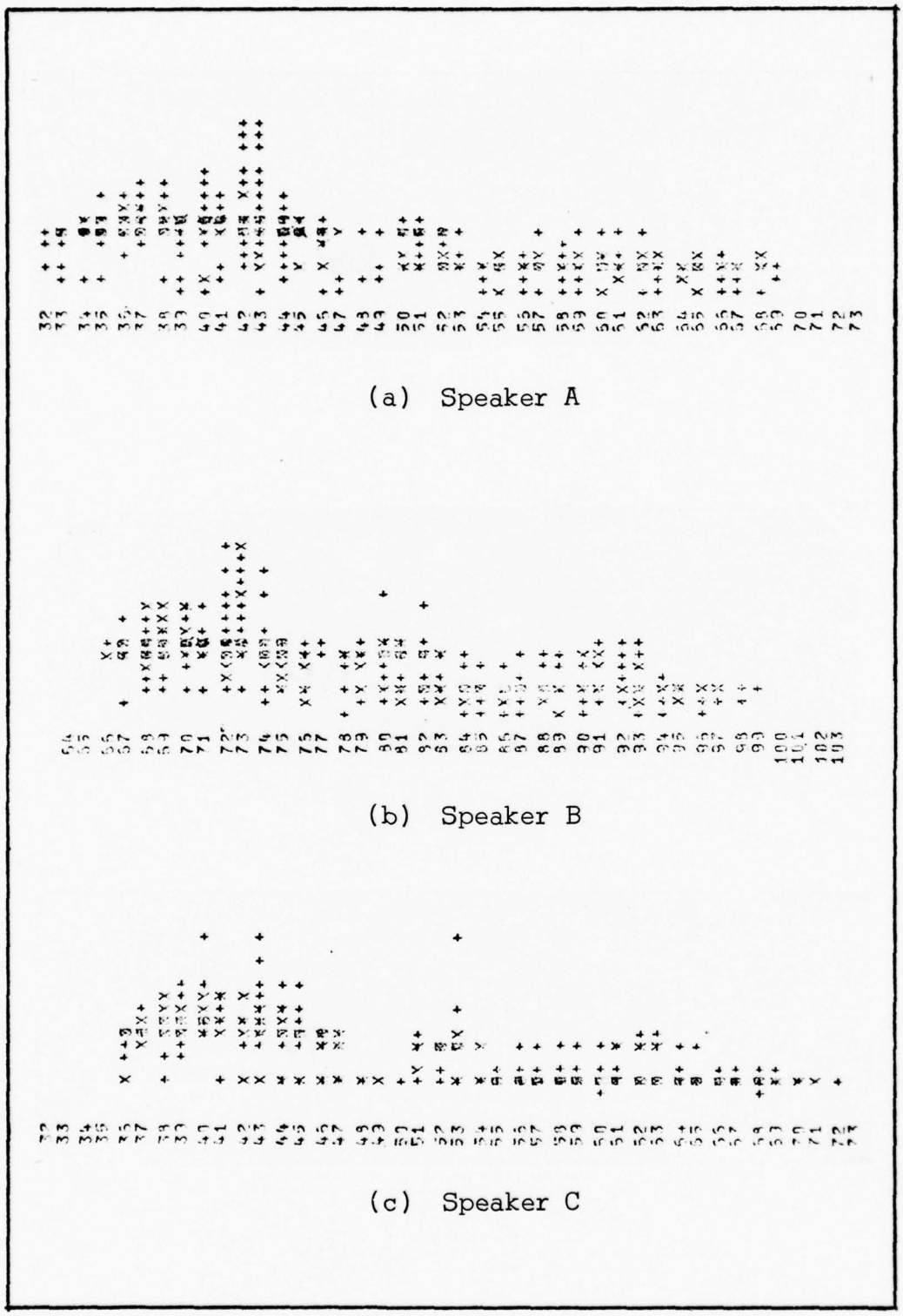
2) A short nasal is often marked by a short drop in energy.

3) A silent segment followed by a noisy segment can be either a plosive followed by a fricative, or the whole sequence can be an aspirated plosive.

Although the main impetus of this research concerned single-speaker recognition, the success of Daily and Sutton with spatial filtering (Ref 4) suggested the attempt at multiple speaker recognition. Seven sentences were recorded three times each by three separate speaker subjects. Speaker A was a male - southern accent, Speaker B was a male - mid-eastern accent, and Speaker C was a female - southern accent.

The spectrograms of the word "vitamins" spoken by each speaker show remarkable similarity as is displayed in Figure 3. This across-speaker invariance in the speech spectrogram representation of speech which might be the basis of a speaker independent recognition algorithm.

The first lesson that was gained from this portion of the experiment was that prototypes chosen from deliberately pronounced words had little chance of correctly correlating with those of normal speech. The main problem seemed to be the length of the prototypes as they occurred in slow speech compared to the normal shortened speech. The normalization that was being used did not rectify the problem. Another problem, that has already been alluded to, was the shorter prototype correlation with noisy segments of longer prototypes. This last problem worsened when spatial filtering was attempted because the longer prototypes had more energy removed from them by filtering than did the shorter prototypes. These problems motivated the selection of uniform length prototypes. These prototypes were taken from the seven sentences that were recorded by Speaker A. The sentences that were used for this were not used in compiling recognition results as this would not be an unbiased scoring.



(a) Speaker A

(b) Speaker B

(c) Speaker C

Fig. 3. Spectrograms for "Vitamins" by Three Speakers

In order to keep the prototype set at 61, 17 of the prototypes that came from continuous utterances of vowels were retained. The complete set is listed in Table III. There is redundancy in the selection and there are sounds that are not included; however, the scoring is made as though a complete set existed. In some cases the longer prototypes did correlate highly, but when they did it generally was a case of agreement. One good example of this is the dipthong prototype "AE" from hate correlating higher with "TA - TE"* combination from taste. Table VI describes the symbols used for analysis of sentence data.

Table VI

Sentence Analysis Symbols

Symbol	Definition
Blank	No symbol (or a blank) indicates that this was the accepted, recognized phoneme
L	An "L" indicates that although not recognized, the maximum value of correlation indicated proper location
X	An "X" indicates that this phoneme was not located

* Symbols used are listed in Table III.

Table VII through Table XIV display the analysis of each sentence spoken by Speaker A and Table XV through Table XXII contain the analysis of one of each of the seven sentences spoken by Speaker B.

Table VII

Sentence #1 Analysis (Speaker A)

Sentence

	Vitamins	Taste	Good
1(a)	/ V VI VT $E VM HE VN VS /	/ T TA TE Z VS T /	/ G GU GD /
Symbol	L X	L L X	L L
1(b)	/ V VI VT $E VM VU SE VN VS /	/ T BA HE VS T /	/ RG GU W D /
Symbol	L X L	L L	L X

Total Phonemes 35 Phonemes located 31 89% Phonemes Recognized 21 60%

TABLE VIII

Sentence #2 Analysis (Speaker A)

Sentence

Journal of speech and hearing

2(a)	/	J	JE	VN	VU	JL	/		/	OF	F	/		/	VS	P	SE	CH	/		/	.A	N	D	/		/	H	TE	R	/	*
Symbol		L		L	L						L					L						L	L	X					L	X			
2(b)	/	J	JE	VN	VU	JL	/		/	OF	F	/		/	VS	P	SE	CH	/		/	.A	N	D	/		/	H	HE	R	/	*
Symbol		L		X	L											L							L	X						X			

Total Phonemes 34 Phonemes Located 29 85% Phonemes Recognized 17 50%

*Rest of sentence not digitized.

Table IX

Sentence #3 Analysis (Speaker A)

Sentence	Research	breeds	recognition
3(a)	/ R RE VS JE Z CH /	/ B R HE W D VS /	/ R RA JE RK CY RG / *
Symbol	L	L X	X L
3(b)	/ R RE VS ◄E JE W CH /	/ B R HE D VS /	/ R RA RK VU RG / *
Symbol	L	L L X L	X L L

Total Phonemes 37

Phonemes Located 33

89%

Phonemes Recognized 24

65%

*Rest of Sentence not digitized.

Table X

Sentence #4 Analysis (Speaker A)

Sentence	Central	nervous	system
4(a)	/ VS RA VN T R -A L /	/ VN JE V VU VS /	/ VS VU VS T -A VM /
Symbol	L X X X L L	L L L	L L L X
4(b)	/ VS EI VN T R -A L /	/ VN -E JE V VU VS /	/ VS VU VS T VU VN /
Symbol	L X L X L X	L L L	X L X L

Total Phonemes 37 Phonemes Located 28 76% Phonemes Recognized 11 30%

Table XI

Sentence #5 Analysis (Speaker A)

Sentence	Basic								computation										element						
5(a)	/	B	BA	TE	VS	Z	I	C	/	/	C	VU	VM	CP	VU	T	CH	VU	VN	/	/	E	JL	VM	HE	VN	/	*
Symbol		L			L		X	X			X		X	L	L	X			L			L	L	X	L	L		
5(b)	/	B	BA	TE	VS	I	RK	/	/	C	VU	.A	VM	P	VU	T	E	SN	HE	VN	/	/	<E	L	VM			/	*
Symbol		X		L	L	L							L	X		X				L			L	L	X				

Total Phonemes 41

Phonemes Located 31

76%

Phonemes Recognized 15

37%

*Rest of Sentence not digitized.

Table XII

Sentence #6 Analysis (Speaker A)

Sentence	Obey	and	judge	not
6(a)	/ OB B AE /	/ .A VN GD /	/ J -A DZ /	/ VN OF T /
Symbol	L L	L L X	L	L X
6(b)	/ OB B AE /	/ .A VN GD /	/ J -A DZ RG /	/ VN A T /
Symbol	L L	X X L	L L	X L

Total Phonemes 25 Phonemes Located 20 80% Phonemes Recognized 8 32%

Table XIII

Sentence #7 Analysis (Speaker A)

Sentence	Tight	closure	will	cause
7(a)	/ T VI $I T /	/ C JL O Z JE /	/ W HE JL /	/ C A OB Z /
Symbol	L L X	X X L L	L X	X L

(Only one replicate existed for the sentence.)

Total Phonemes 16

Phonemes Located 11

69%

Phonemes Recognized 5

31%

Table XIV

Summary Analysis for Speaker A

Total Phonemes 225	Phonemes Located 183	Phonemes Recognized 101
	81%	45%

Table XV

Sentence #1 Analysis (Speaker B)

Sentence	Vitamins										taste						good				
	/	V	VI	$E	VT	VU	VM	VU	VN	VS	/	T	TA	TE	Z	VS	T	/	RG	G	GU	GD	/
Symbol		X			L	L		X	L			X	L				X						

Total Phonemes 19 Phonemes Located 15

79%

Phonemes Recognized 11

58%

Table XVI

Sentence #2 Analysis (Speaker B)

Sentence

Journal of speech and hearing

	/ J JE VN -A JL /	/ -A F /	/ S P SE CH /	/ E VN D /	/ H BE R TE VN G /
Symbol	X X		L L L	L L X	L X X L

Total Phonemes 20 Phonemes Located 15 75% Phonemes Recognized 8 40%

Table XVII

Sentence #3 Analysis (Speaker B)

Sentence	Research	breeds	recognition
	/ R HE VS JE W CH /	/ B R HE D VS /	/ R RA RK OF RG VN I VS SN /
Symbol	L L	X L L X L	L X L L X L L

Total Phonemes 20

Phonemes Located 16

80%

Phonemes Recognized 6

30%

Table XVIII

Sentence #4 Analysis (Speaker B)

Sentence	Central							nervous						system					
	/	VS	I	VN	T	R	OF	JL	/	VN	JE	V	VU	VS	/	VS	I	VS	T	VU	VM	/
Symbol		X	L	L	X	L		X		L		X	L	X		X	X		X		X	

Total Phonemes 18

Phonemes Located 9

50%

Phonemes Recognized 4

22%

Table XIX

Sentence #5 Analysis (Speaker B)

Sentence	Basic	computation	element
	/ B E VS TE RK /	/ C A VM P CY T TE SN /	/ * /
Symbol	X X	L X L X X L L	

Total Phonemes 13 Phonemes Located 8 62% Phonemes Recognized 4 31%

*Rest of Sentence not digitized.

Table XX

Sentence #6 Analysis (Speaker B)

Sentence	Obey	an	judge	not
	/ OB B >E TE /	/ .A VN /	/ J GU DZ /	/ VN OF * /
Symbol	X	L L	X L	X L

Total Phonemes 11 Phonemes Located 8 73% Phonemes Recognized 4 36%

*Rest of Sentence not digitized.

Table XXI

Sentence #7 Analysis (Speaker B)

Sentence

Tight					closure							will			cause				
/	T	VI	VT	TA	/	/	RK	JL	OB	ZH	JE	R	/	/	W	I	L	/	/	C	A	-A	Z	/
Symbol	L		L				X	X	L	L	X	L			X	L				X	L	L	L	

Total Phonemes 17 Phonemes Located 12 71% Phonemes Recognized 3 18%

Table XXII

Summary Analysis for Speaker B

Total Phonemes 118	Phonemes Located 83	Phonemes Recognized 4
	70%	40%

Filtered Test

In order to establish the filter size and whether to normalize before filtering or after, several test filters were used. These filter results are presented in Table XXIII.

Table XXIII

Results With Filtering

Filter Size	Number of Phonemes	Phoneme Location (%)	Phoneme Recognition (%)
Filter before Normalization			
5 x 13	7	29	0
7 x 7	15	80	40
9 x 13	15	93	33
7 x 15	16	81	25
15 x 7	16	88	50
15 x 15	16	100	63
17 x 33	16	94	50
17 x 64	15	93	47
32 x 33	16	100	63
Filter after Normalization			
15 x 15	16	100	19
25 x 45	16	100	56

Filtering after normalization gave the same location as filtering before; however, the correlation magnitudes were greatly reduced. Filtering before normalization caused the correlation coefficients to greatly increase and crowded the decision space. The filter chosen for filter analysis was the 25 x 45 filter placed after normalization. The energy lost with this large filter was minimal but seemed to maximize phoneme location. Table XXIV presents the filtered analysis of five of the sentences of Speaker A and Table XXV presents a summary of the analysis.

Table XXVI contains a filter analysis of three sentences of Speaker B.

No improvement was gained by filtering for Speaker C. Only one sentence, "Vitamins taste good," was analyzed for Speaker C and location was rated at 89 percent and recognition at 19 percent.

Table XXIV

Filtered Analysis (Speaker A)

	Vitamins								taste					good				
Sentence 1(b)	/	V	VI	VT	$E	VM	VU	VN	VS	/	/	T	TA	HE	VS	T	/	/	RG	G	GU	GD	/
Symbol		L		L		L		L				L			L				L	L		X	

	Journal							of			speech					and				hearing				
Sentence 2(b)	/	RG	J	JE	VN	VU	JL	/	/	OF	F	/	/	VS	CP	SE	CH	/	/	.A	VN	GD	/	/	H	HE	R	/
Symbol			L		X	L	L								L					L	X				L		X	

	Central							nervous						system					
Sentence 4(a)	/	VS	RA	VN	T	R	RA	JL	/	/	VN	JE	V	VU	VS	/	/	VS	TE	VS	T	.A	/
Symbol			X		X	L	X	L			L		L		L				L		L		

Table XXIV (Cont'd.)

	Basic						computation									element					
Sentence 5(a)	/ B	BA	BE	VS	Z	RK /	/ C	OF	VM	P	CY	VT	VU	CH	VU	VN /	/ -A	JL	VM	HE	VN /
Symbol	L		L	L		L	X	L	X	X	X	X			L				X	L	L

	Obey			and			judge			not		
Sentence 6(c)	/ OB	B	E /	/.A	VN	GD /	/ J	-A	DZ /	/VN	A	T /
Symbol	L	X		L	L	X	X		L	L		L

Table XXV

Summary of Filtered Results for Speaker A

Sentence Number	Total Phonemes	Phoneme Location (%)	Phoneme Recognition (%)
1(b)	17	94	47
2(b)	18	83	50
4(a)	17	82	29
5(a)	21	71	33
6(c)	12	75	25
Total	85	81	38

Table XXVI

Filtered Analysis (Speaker B)

	Vitamins									taste							good					
Sentence 1(a)	/	A	V	VI	VT	$E	VM	VN	VS	/	/	VT	TA	TE	Z	VS	T	/	/	H	RG	G	GU	GD	/
Symbol		B	L		L		L	L				L	L				L			B					

	Central								nervous							system						
Sentence 4(a)	/	VS	.I	VN	T	R	RA	JL	/	/VN	JE	V	VU	VS	/	/	VS	TE	VS	T	.A	VM	/
Symbol			L	L	L	X		X		L	L	X	L	L				L	L	L	L	X	

	Basil							computation												
Sentence 5(a)	/	B	E	VS	TE	RK	/	/	C	-A	VM	P	CY	T	E	VS	SN	/		
Symbol		X				X			L	L	L	L	X	X		L	L			

Table XXVII

Summary of Filtered Results (Speaker B)

Sentence Number	Total Phonemes	Phoneme Location (%)	Phoneme Recognition (%)
1(a)	17	100	59
4(a)	18	78	17
5(a)	14	71	29
Total	49	84	35

VI. Conclusions and Recommendations

The objective of this study was to try to solve the phoneme recognition problem by computer analysis of continuous speech. More directly, the objective was to take the basic program developed by Neyman, to subject it to further testing, and by incorporating meaningful modifications, to improve its operation.

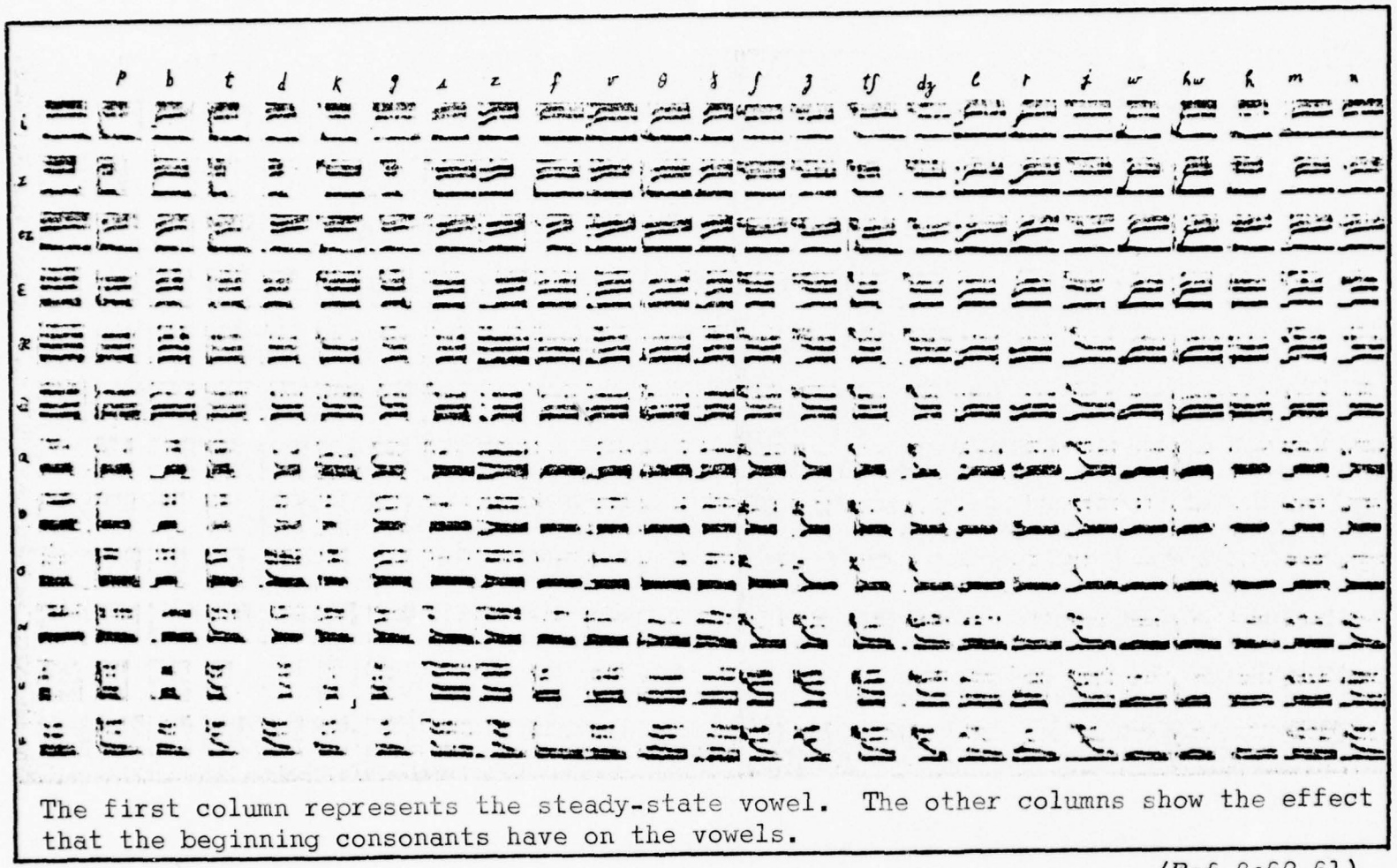
For sentence analysis, Neyman achieved location of 92 percent and identification of 34 percent for a single speaker and an incomplete set of prototypes. Assuming a complete set of prototypes and using uniform length prototypes and different normalization techniques, this program increased identification by 11 percent while location was decreased by 11 percent. Using an additional speaker, recognition was still 6 percent better, but location dropped by 21 percent.

Filtering was investigated next with the overall effect of improving location while slightly decreasing identification. The identification was degraded more for Speaker A. This is related to the fact that the prototypes came from Speaker A while the filter size analysis was performed on Speaker B data. Daily and Sutton found that spatial filtering

designed for one speaker was not best suited for the other speaker or for both speakers together (Ref 4:36).

The feature that shows the greatest promise of success is the use of uniform length prototypes. Speech segments that are longer than the prototypes can have consecutive identification periods. This allows the use of prototypes for transitions. Figure 4 (Ref 8:60-61) suggests that there might be as many as 300 prototypes to cover the vowels, consonants, and interphonemic transitions. Even this would be an acceptable solution if it would offer substantially higher recognition results.

The increased identification attained by this program continues to warrant further study. The prototype set should have the missing phonemes added by deleting phonemes that prove to be redundant. The redundancy should be identified by actual correlation tests so as not to destroy unique prototypes. The correlation process obtains best results when the prototypes are taken from actual speech. The warning that must be issued here is that there is a high degree of probability of selecting portions of adjacent phonemes as is surely the case in a few of the existing prototypes of this program.



The first column represents the steady-state vowel. The other columns show the effect that the beginning consonants have on the vowels.

(Ref 8:60-61)

Fig. 4. Spectrograms of Vowel-Consonant Combinations

If female voices are to be used with male prototypes, frequency normalization of some type will be necessary. From observing Figure 3(c), it is evident that substantial improvement could be gained by setting the first two frequency channels of both the prototypes and the sentence data to zero since they are missing from the female voice.

Bibliography

1. Broad, David J. and June E. Shoup. "Concepts for Acoustic Phonetic Recognition." Speech Recognition (D. Raj Reddy, ed), Academic Press, New York, 1975.

2. Carl, Joseph W. and Charles F. Hall. "The Application of Filtered Transforms to the General Classification Problem." IEEE Transactions on Computers, C-21:785-790 (July 1972).

3. Cherry, Colin. "Two Ears - But One World" in Sensory Communication, edited by Walter A. Rosenblith. Cambridge, Massachusetts: The M. I. T. Press, 1961.

4. Dailey, Keith G. and Frankie S. Sutton. An Automatic Speech Recognition System Using a Vocoder Input. M.S. Thesis GE/GGC/EE/72-18. Wright-Patterson Air Force Base, Ohio: Air Force Institute of Technology (1972).

5. Denes, P. "Effect of Duration on the Perception of Voicing." Journal of the Acoustical Society of America, 27:761-764 (July 1955).

6. Epifantsev, B.N. "An Investigation of the Cross Correlation of Speech Signals." A foreign Technology Division translation from Polytechnic Institute Transactions, Leningrad. 291:134-140 (1968).

7. Flanagan, James L. Speech Analysis Synthesis and Perception. New York: Academic Press, Inc., 1965.

8. Fletcher, Harvey. Speech and Hearing in Communication. New York: D. Van Nostrand Company, Inc., 1953.

9. Hall, William B. Jr. A Digest and Reference Organization of Fast Fourier Transform Literature and Software. VNA, Internal Memo 72-2. Wright-Patterson Air Force Base, Ohio: Analog/Hybrid Systems Branch Computer Center, February, 1972.

10. Haller, Mark. The Cooley-Tukey Fast Fourier Transform in USASI Basic Fortran. A computer Program for the CDC 6600. Wright-Patterson Air Force Base, Ohio: ASD Computer Center, 1972.

11. Itakura, F. "Minimum Prediction Residual Principle Applied to Speech Recognition." IEEE Transactions on Acoustics, Speech and Signal Processing, Vol. ASSP-23, No. 4:373-379 (August 1975).

12. Kabrisky, Matthew. A Proposed Model for Visual Information Processing in the Human Brain. Urban, Illinois: University of Illinois Press, 1966.

13. Klatt, Dennis H. and Kenneth N. Stevens. Strategies for Recognition of Spoken Sentences from Visual Examination of Spectrograms. Bolt, Bernack, and Newman Report No. 2154 (June 1971).

14. Klatt, Mary M. and Kenneth N. Stevens. Study of Acoustical Properties of Speech Sounds. Bolt, Bernack, and Newman Report No. 8 (August 1968).

15. Licklider, J.C.R. "Basic Correlates of the Auditory Stimulus" in Handbook of Experimental Psychology, edited by Herbert S. Langfield. New York: John Wiley and Sons, Inc., 1951.

16. Lindblom, Björn E.F. and Stig-Goran Svensson. "Interaction Between Segmental Factors in Speech Recognition." IEEE Transactions on Audio and Electroacoustics, AU-21: 536-545 (December 1973).

17. Lindgren, Nilo. "Machine Recognition of Human Language - Part I." IEEE Spectrum, 2:114-136 (March 1965).

18. -----. "Machine Recognition of Human Language - Part II." IEEE Spectrum, 2:45-59 (April 1965).

19. Marslen-Wilson, W.D. "Sentence Perception as an Interactive Parallel Process." Science, Vol. 189 (July 1975).

20. McLachlin, Dan Jr. "The Role of Optics in Applying Correlation Functions to Pattern Recognition." Journal of the Optical Society of America, Vol. 52, No. 4:454-459 (April 1962).

21. Neyman, Ralph W. Computer Identification of Phonemes in Continuous Speech. M.S. Thesis GE/EE/76-10. Wright-Patterson Air Force Base, Ohio: Air Force Institute of Technology (1976).

22. Oppenheim, Alan V. "Speech Spectrograms Using the Fast Fourier Transform." IEEE Spectrum, 7:57-62 (August 1970).

23. Pierce, J.R. "Whither Speech Recognition?" Journal Acoustical Society of America, 46:1049-1051 (July 1975).

24. Potter, Ralph K., et al. Visible Speech. D. Van Nostrand Co., Inc., 1947.

25. Schwartz, Richard and John Makhoul. "Where the Phonemes Are: Dealing with Ambiguity in Acoustic-Phonetic Recognition." Proceedings IEEE Symposium on Speech Recognition, 148-153 (April 1974).

26. Ullmann, J.R. Pattern Recognition Techniques. New York: Crane, Russak and Company, Inc. 1973.

27. Warren, A.M. and R.P. Waren. "Auditory Illusions and Confusions." Scientific American. Pp 30-36 (December 1970).

28. White, George M. "Speech Recognition: A Tutorial Overview." Computer. Vol. 9, No. 5:40-53 (May 1976).

29. White, G.M. and R.B. Neely. "Speech Recognition Experiments with Linear Prediction, Bandpass Filtering and Dynamic Programming." IEEE Transactions on Acoustics, Speech, and Signal Processing, Vol. ASSP-24, No. 2:183-188 (April 1976).

APPENDIX

A

SEQUENCE CHART FOR PHONEME RECOGNITION

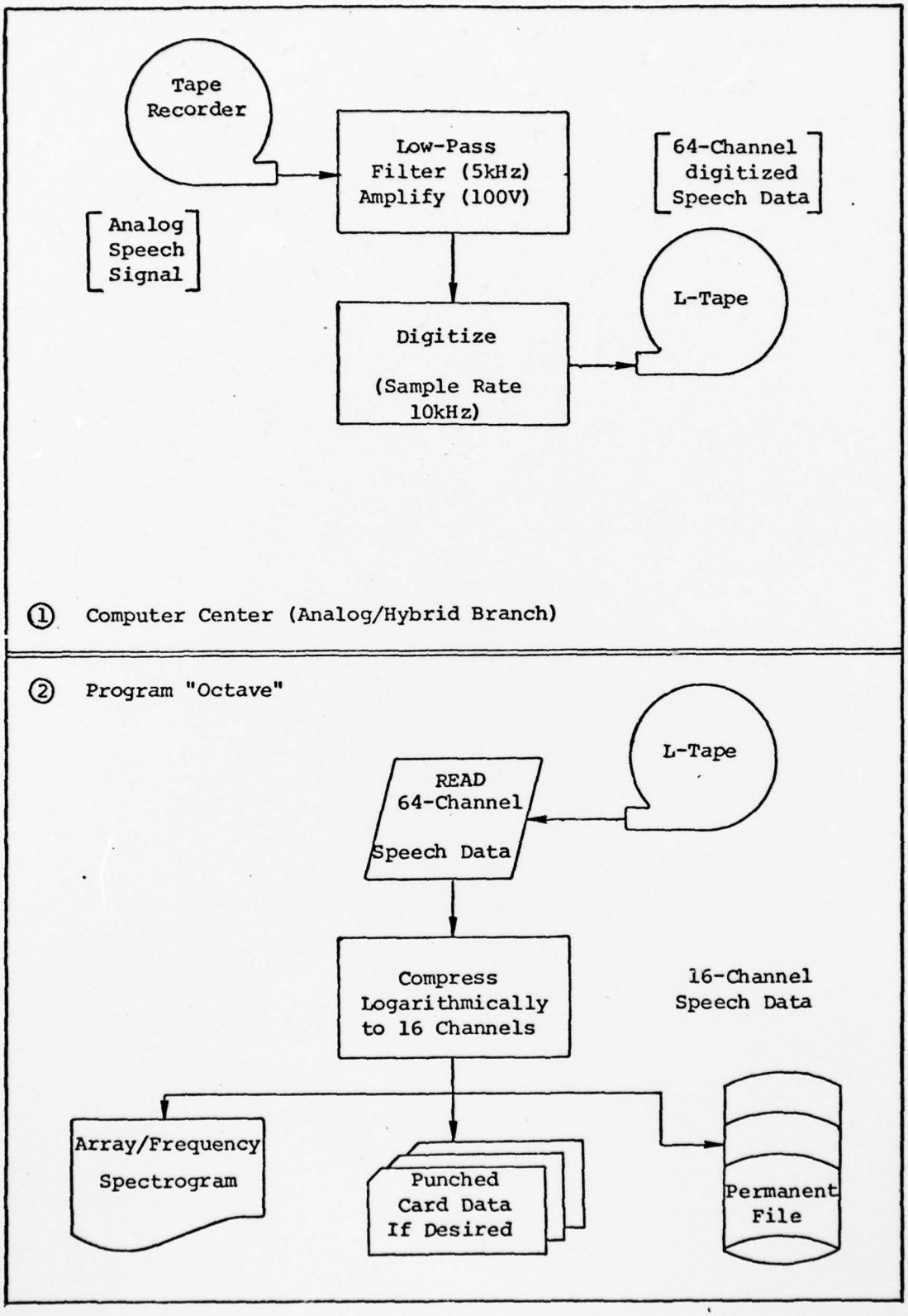


Fig. 5. Sequence Chart for Phoneme Recognition

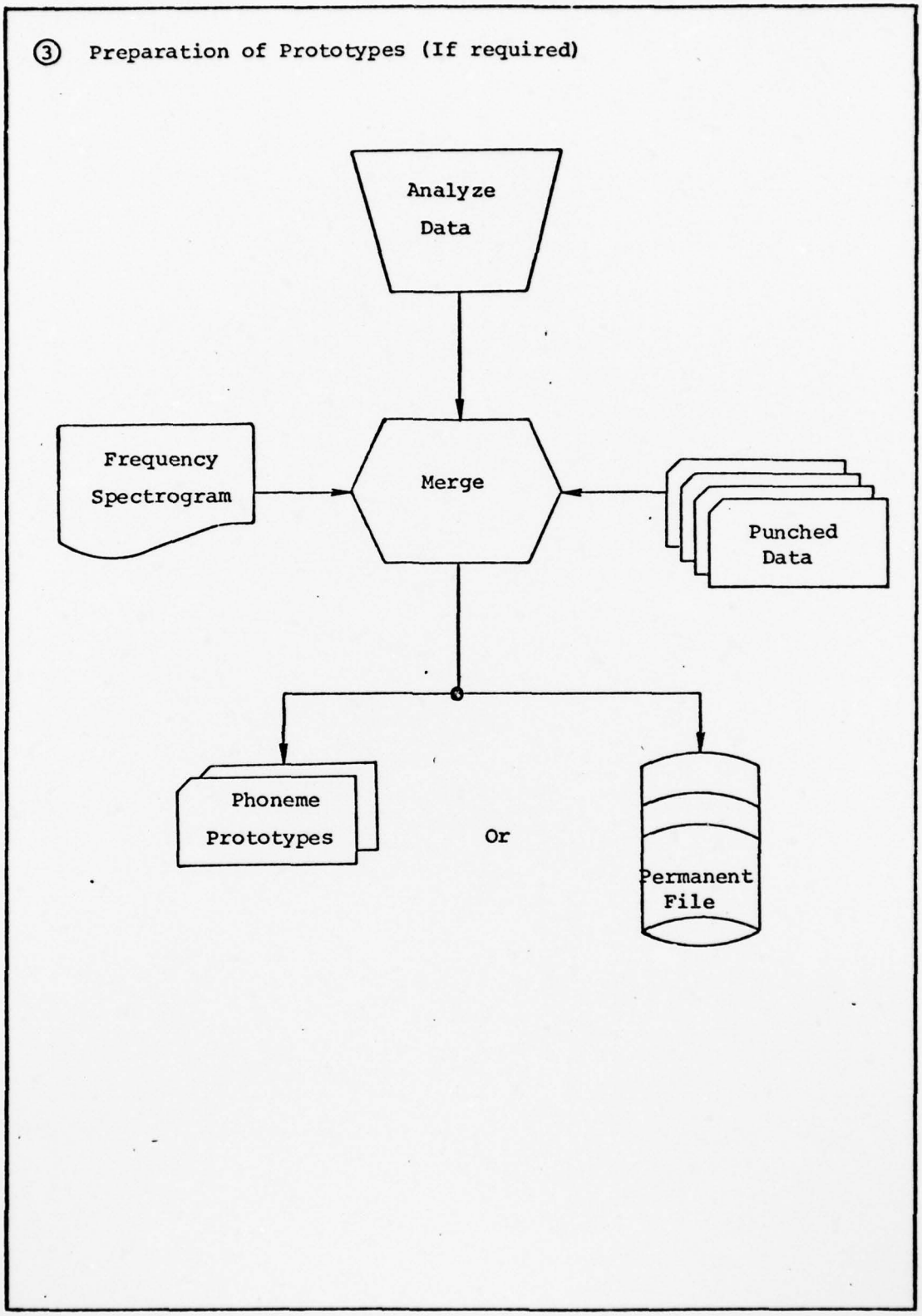


Fig. 6. Prototype Preparation

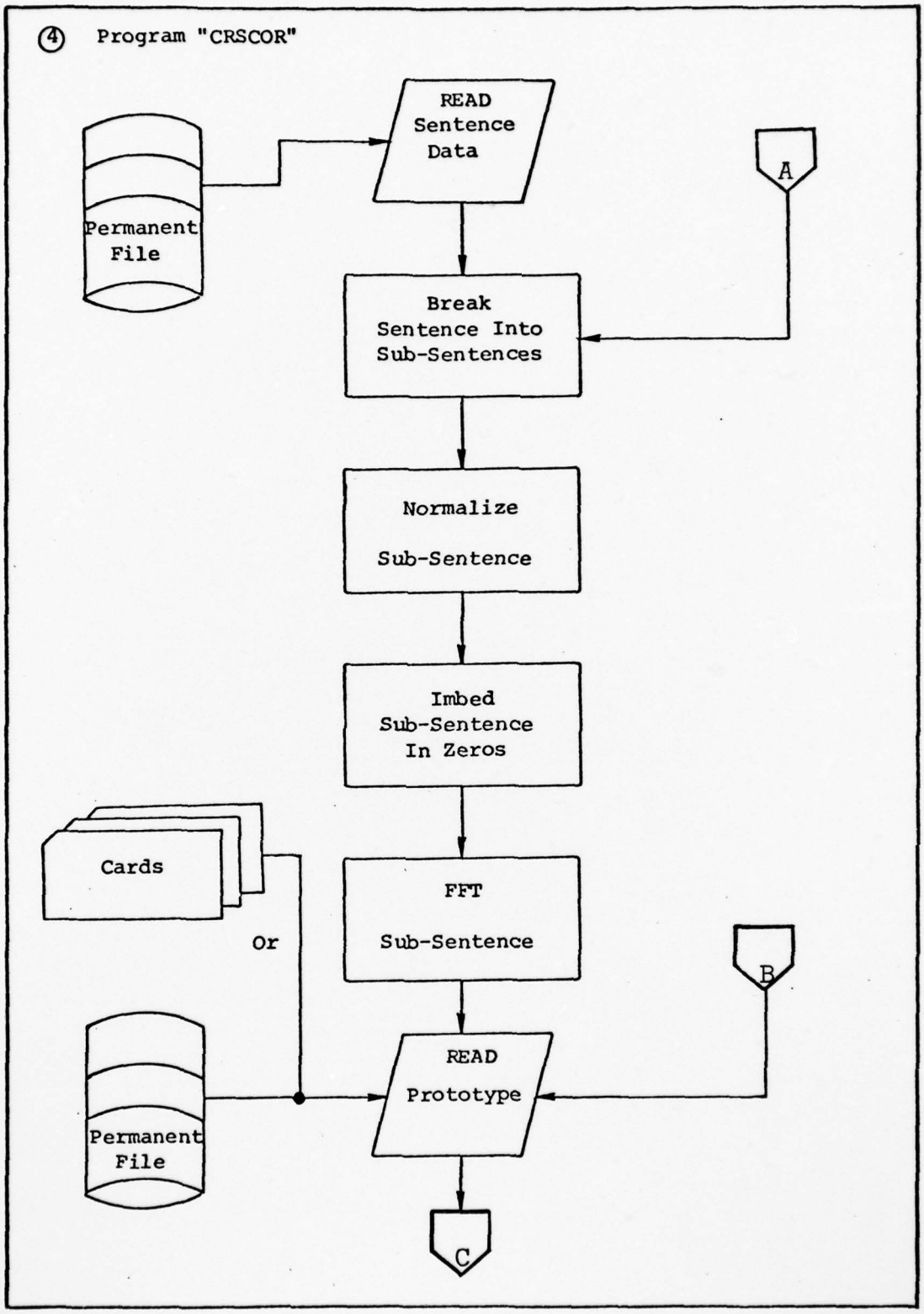


Fig. 7. Program "CRSCOR" (Plate 1)

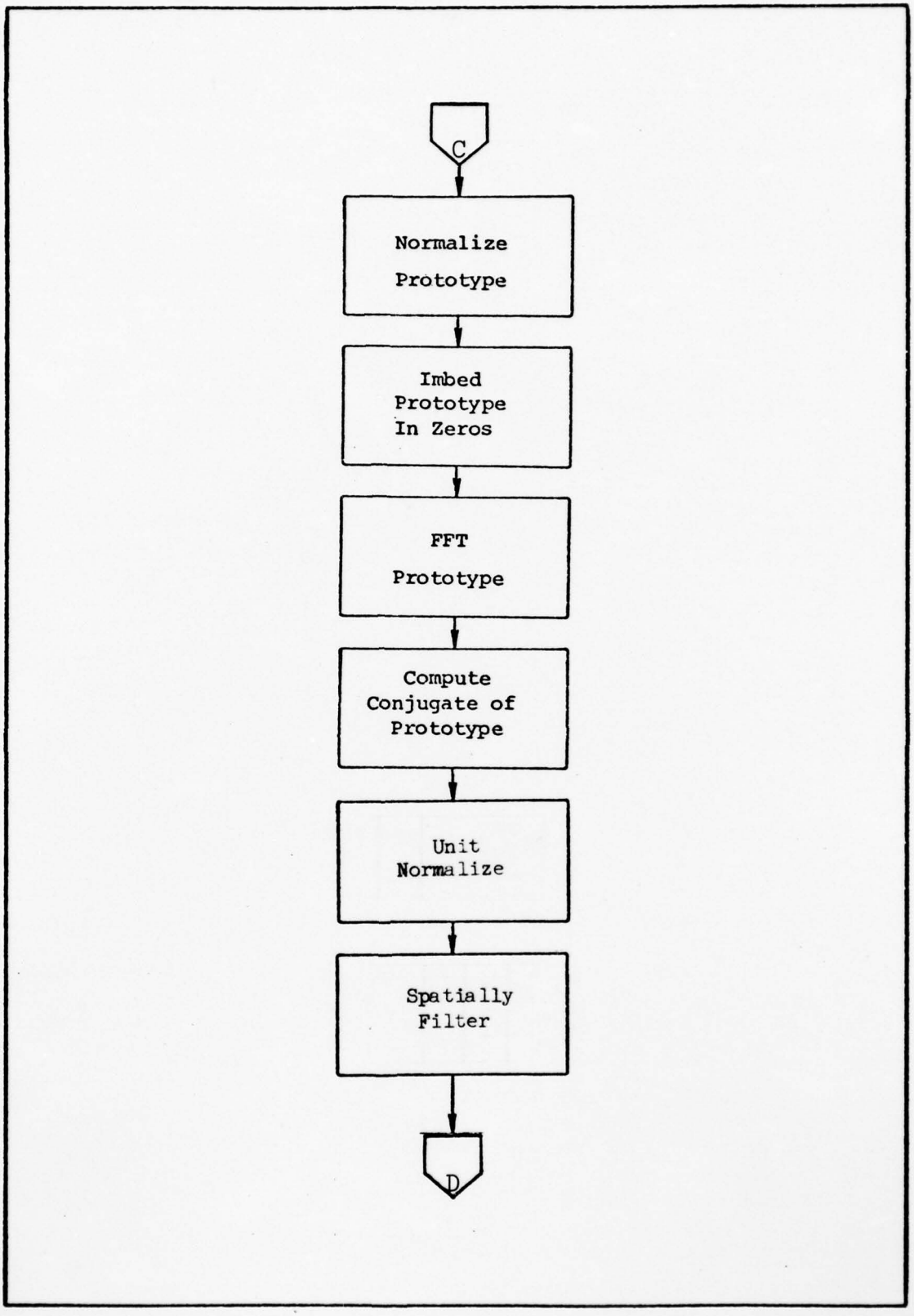


Fig. 8. Program "CRSCOR" (Plate 2)

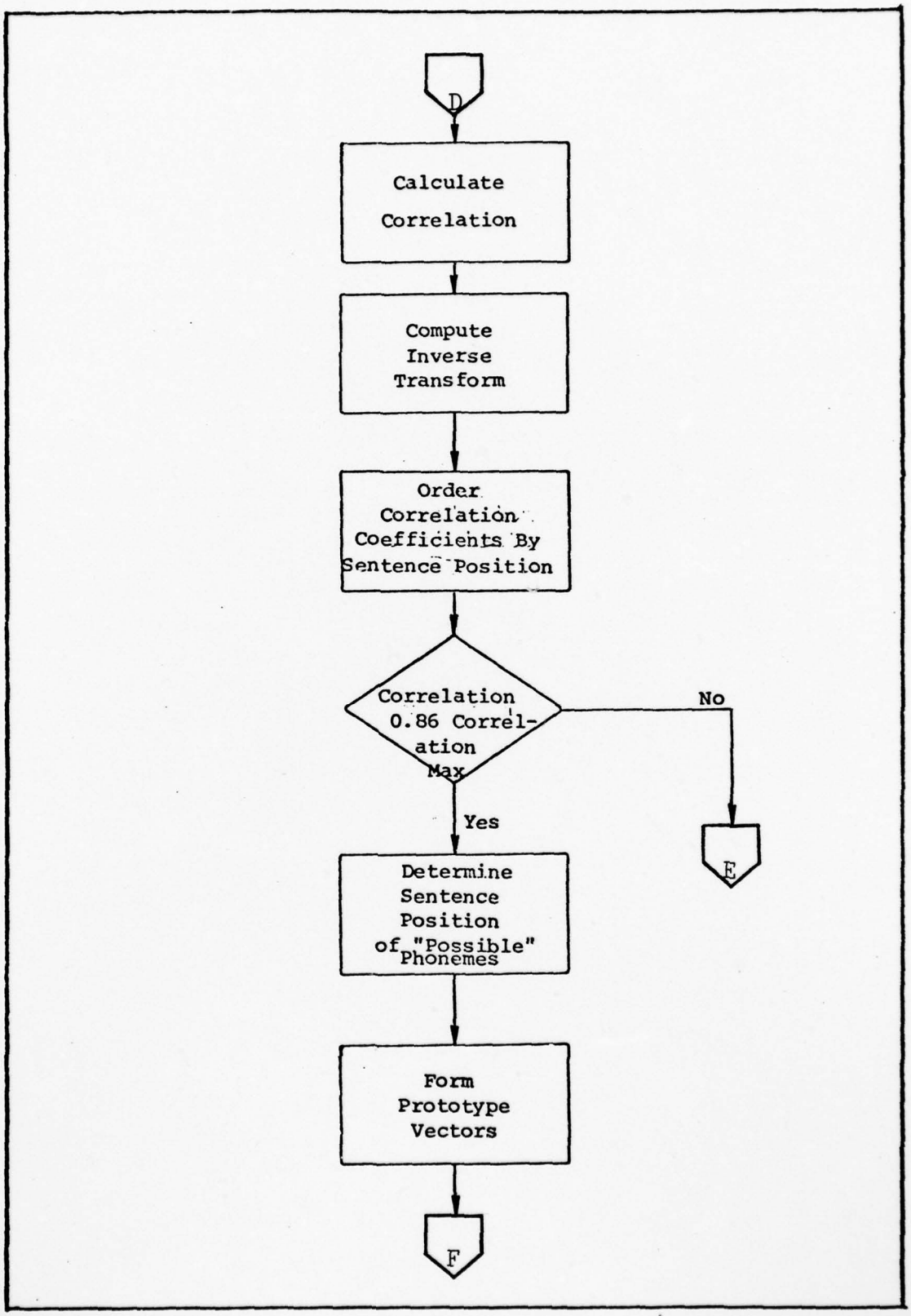


Fig. 9. Program "CRSCOR" (Plate 3)

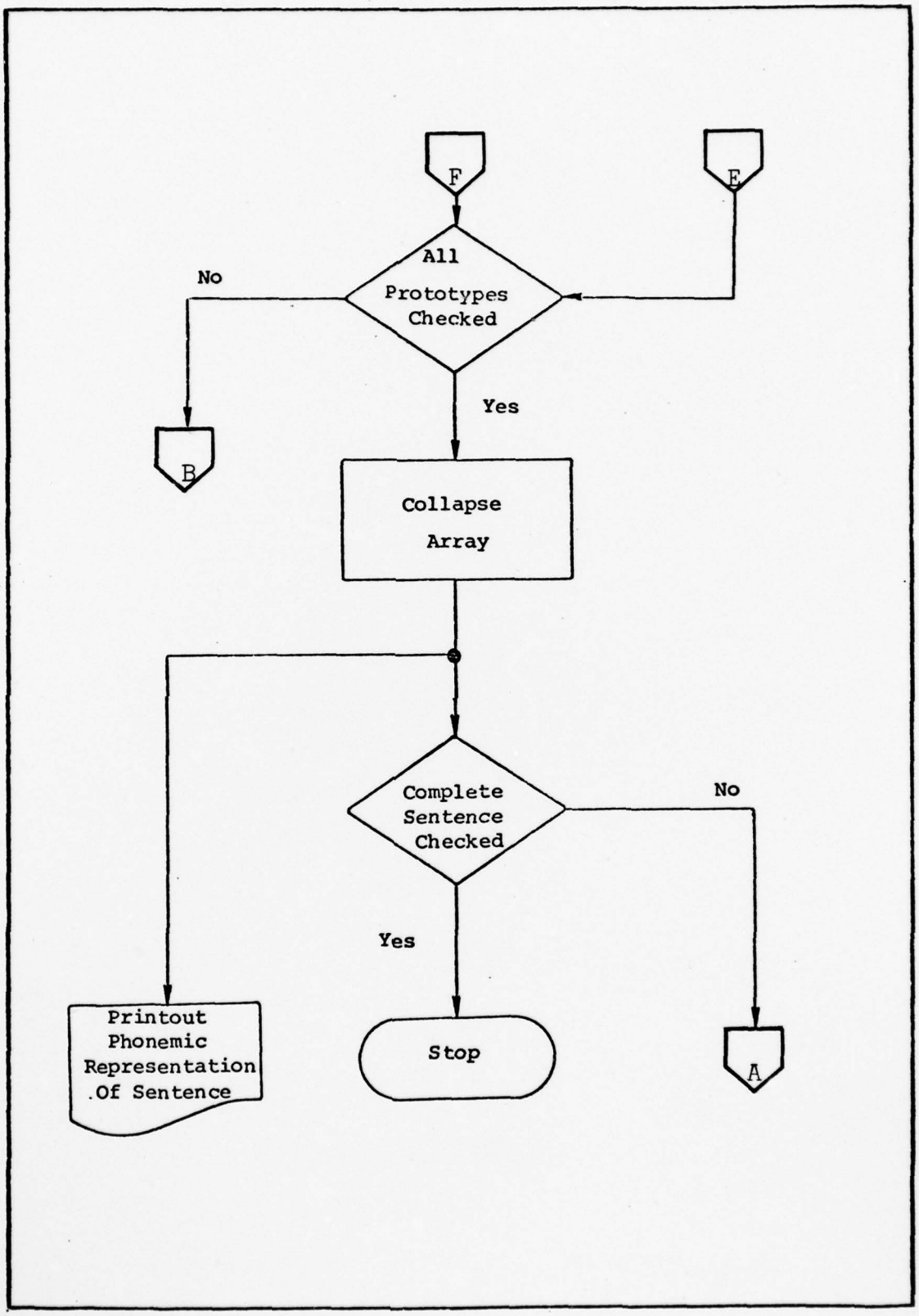


Fig. 10. Program "CRSCOR" (Plate 4)

APPENDIX

B

COMPUTER PROGRAMS

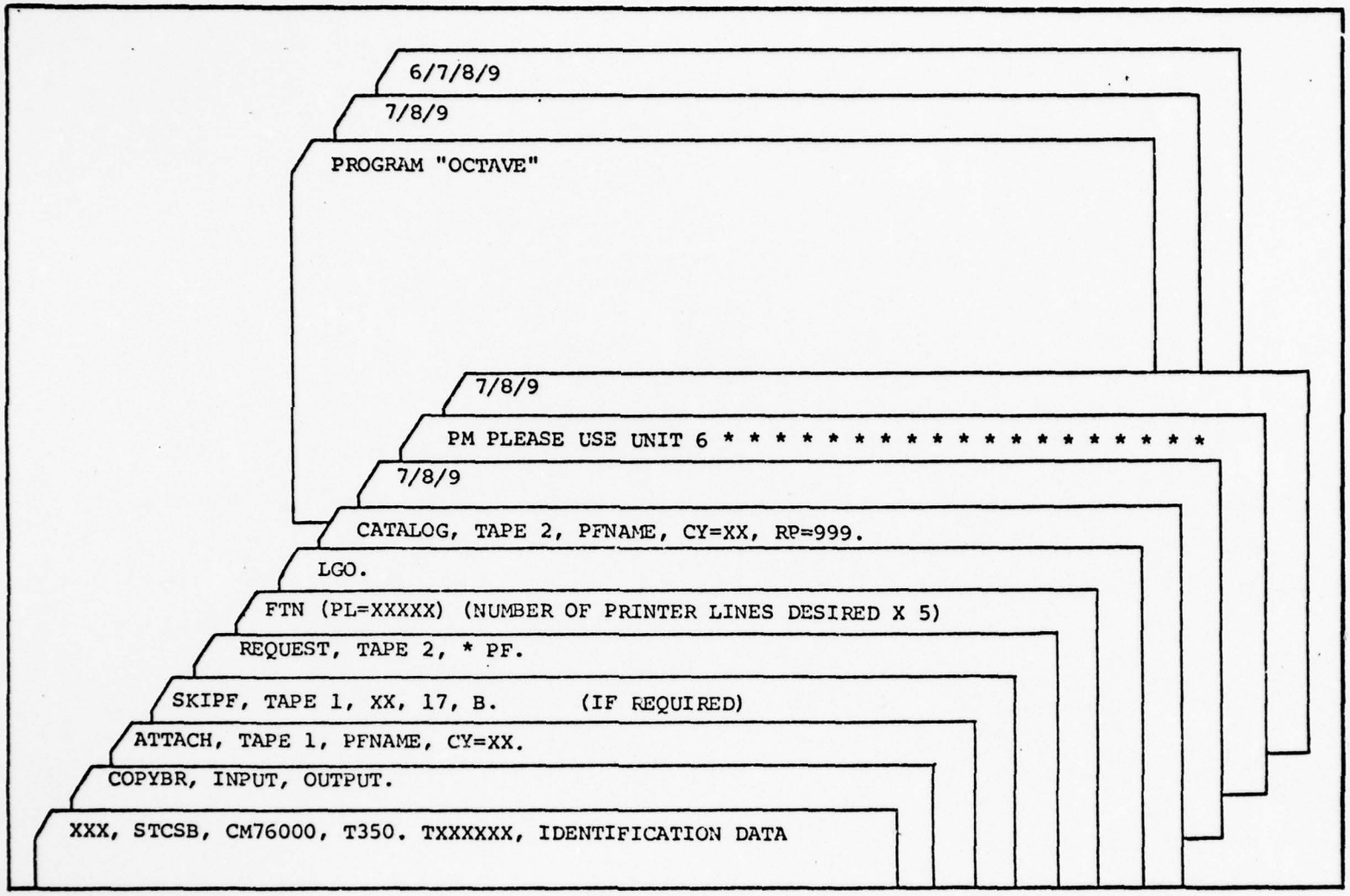


Fig. 11. Control Cards for Program "OCTAVE"

```
C*****************************************************************************
C*****************************************************************************
C**     THIS PROGRAM REDUCES 64 CHANNELS OF DIGITIZED SPEECH DATA TO 16      **
C**    CHANNELS. THE ORIGINAL 64 CHANNELS ARE COMBINED BY ADDING THE ENERGY   **
C**    CONTRIBUTIONS OF EACH ELEMENT WITHIN A 1/3 OCTAVE GROUP. THE OUTPUT    **
C**    IS THE COMPRESSED ARRAY AND AN ACCOMPANYING SPEECH SPECTROGRAM.        **
C*****************************************************************************
C*****************************************************************************
      PROGRAM OCTAVE(INPUT,OUTPUT,TAPE1,TAPE2,TAPE6=OUTPUT)
      DIMENSION SYMBOL2(10),SYMBOL3(10),SYMBOL4(10),SYMBOL5(10)
      DIMENSION A(64),B(19),SYMBOL1(10),BI(19),IBI(19)
C-----------------------------------------------------------------------------
C--                      SPECTROGRAM OVERPRINT SYMBOLS                      --
C-----------------------------------------------------------------------------
      DATA SYMBOL1/1H ,1H ,1H+,1HX,1HX,1HX,1HX,1HX,1HX,1HX/
      DATA SYMBOL2/1H ,1H ,1H ,1H ,1H-,1H+,1HO,1HO,1HO,1HO/
      DATA SYMBOL3/1H ,1H ,1H ,1H ,1H ,1H ,1H ,1H-,1H-,1H#/
      DATA SYMBOL4/1H ,1H ,1H ,1H ,1H ,1H ,1H ,1H ,1H+,1H+/
      DATA SYMBOL5/1H ,1H ,1H ,1H ,1H ,1H ,1H ,1H ,1H ,1H*/
C-----------------------------------------------------------------------------
C--                             PROGRAM VARIBLES                            --
C-----------------------------------------------------------------------------
C     NUMBER OF RECORDS TO BE READ
      NREC=21
C     MAXIMUM RECORD LENGTH
      NN2=800
C-----------------------------------------------------------------------------
C--                INPUT ARRAY LOGARITHMICALLY COMPRESSED                   --
C-----------------------------------------------------------------------------
      NN1=64
```

```
1     CONTINUE
      DO 305 I=1,NN2
5     READ(1,10)(A(J),J=1,NN1)
10    FORMAT(22F6.3)
      IF(EOF(1))310,30
30    CONTINUE
      JJ=1
      DO 40 J=1,6
      B(JJ)=A(J)
      JJ=JJ+1
40    CONTINUE
      DO 50 J=7,11,2
      B(JJ)=(A(J)+A(J+1))
      JJ=JJ+1
50    CONTINUE
      DO 60 J=13,17,4
      B(JJ)=(A(J)+A(J+1)+A(J+2)+A(J+3))
      JJ=JJ+1
60    CONTINUE
      SUM1=0
      DO 70 J=21,25
      SUM1=(SUM1+A(J))
70    CONTINUE
      B(JJ)=SUM1
      SUM2=0
      DO 80 J=26,31
      SUM2=(SUM2+A(J))
80    CONTINUE
      JJ=JJ+1
      B(JJ)=SUM2
```

```
      SUM3=0
      DO 90 J=32,40
      SUM3=(SUM3+A(J))
 90   CONTINUE
      JJ=JJ+1
      B(JJ)=SUM3
      SUM4=0
      DO 100 J=+1,50
      SUM4=(SUM4++A(J))
 100  CONTINUE
      JJ=JJ+1
      B(JJ)=SUM4
      SUM5=0
      DO 110 J=51,54
      SUM5=(SUM5+A(J))
 110  CONTINUE
      JJ=JJ+1
      B(JJ)=SUM5
C-----------------------------------------------------------------------
C--                ARRAY VALUES CONVERTED TO INTEGER FORM             --
C-----------------------------------------------------------------------
      DO 240 JJ=1,16
      BI(JJ)=(B(JJ)+.5)
      IBI(JJ)=IFIX(BI(JJ))
 240  CONTINUE
      IF(I.GT.1) GO TO 295
C-----------------------------------------------------------------------
C--          COMPRESSED ARRAY AND ASSOCIATED SPECTROGRAM OUTPUT       --
C-----------------------------------------------------------------------
      PRINT 250
```

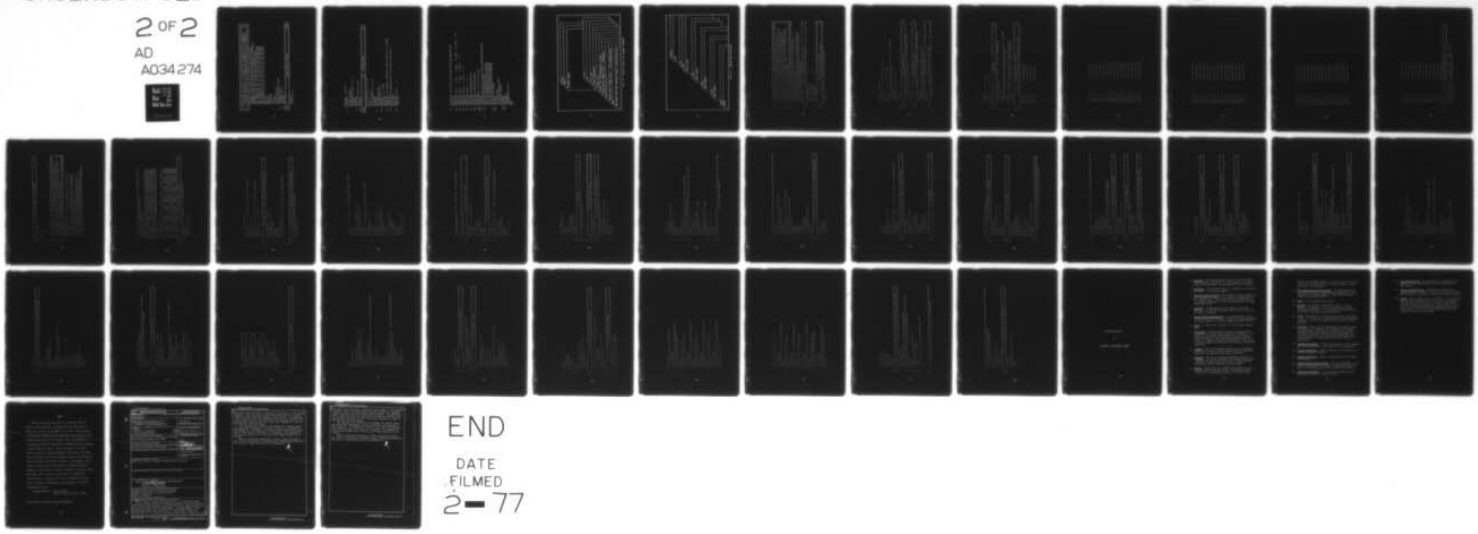
```
250   FORMAT(///,87X,*SYMBOLS REPRESENT INTEGER VALUES AS FOLLOWS:*)
      PRINT 260
260   FORMAT(83X,"0=BLANK",2X,"1=( )",2X,"2=(+)",2X,"3=(X)",
     12X,"4=(X)")
      PRINT 265
266   FORMAT("+",112X,"    - ")
      PRINT 261
261   FORMAT(83X,"5=(X)",2X,"6=(X)",2X,"7=(X)",2X,"8=(X)",2X,
     1"9=(X)")
      PRINT 262
262   FORMAT("+",82X,"    + "2X,"    0 ",2X,"    0 ",2X,"    0 ",2X,"    0 ")
      PRINT 263
263   FORMAT("+",95X,"    - ",2X,"    - ",2X,"    - ")
      PRINT 264
264   FORMAT("+",103X,"    + ",2X,"    + ")
      PRINT 265
265   FORMAT("+",110X,"    * ")
      PRINT 270
270   FORMAT(92X,*0000000001111111*)
      PRINT 280
280   FORMAT(92X,*1234567890123456*)
      PRINT 290
290   FORMAT(89X,*--------------------*)
295   CONTINUE
      PRINT 210,(B(JJ),JJ=1,16),I,(SYMBOL1(IBI(JJ)+1),JJ=1,16)
210   FORMAT(1X,16F5.2,8X,I3,16A1)
      PRINT 211,(SYMBOL2(IBI(JJ)+1),JJ=1,16)
      PRINT 211,(SYMBOL3(IBI(JJ)+1),JJ=1,16)
      PRINT 211,(SYMBOL4(IBI(JJ)+1),JJ=1,16)
      PRINT 211,(SYMBOL5(IBI(JJ)+1),JJ=1,16)
```

```
 211  FORMAT("+",91X,16A1)
C-------------------------------------------------------------------------------
C--     COMPRESSED ARRAY WRITTEN TO TAPE2 TO ALLOW DATA TO BE TRANSFERRED      --
C--    TO PERMANENT FILE UPON COMPLETION OF PROGRAM.                           --
C-------------------------------------------------------------------------------
      WRITE(2,315)(B(JJ),JJ=1,16)
 315  FORMAT(16F6.3)
 305  CONTINUE
 310  CONTINUE
      ENDFILE2
      PRINT*
      PRINT*
      NREC=NREC-1
      IF(NREC.GT.0) GO TO 1
      STOP
      END
```

AD-A034 274 AIR FORCE INST OF TECH WRIGHT-PATTERSON AFB OHIO SCH--ETC F/G 17/2
COMPUTER IDENTIFICATION OF PHONEMES IN CONTINUOUS SPEECH.(U)
DEC 76 W R HENSLEY

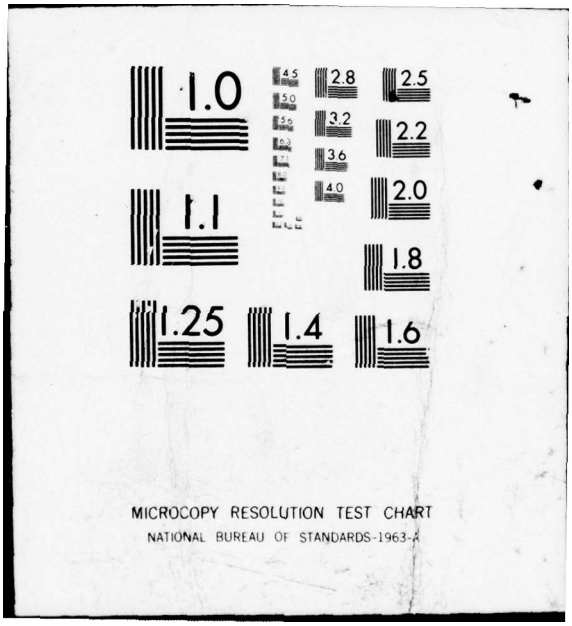
UNCLASSIFIED GE/EE/76-24 NL

2 OF 2
AD
A034274



END

DATE
FILMED
2-77


1.0
1.1
1.25
1.4
1.6
1.8
2.0
2.2
2.5
2.8
3.2
3.6
4.0
4.5
5.0
5.6
MICROCOPY RESOLUTION TEST CHART
NATIONAL BUREAU OF STANDARDS-1963-A

```
C*******************************************************************************
C*******************************************************************************
C**THIS PROGRAM ATTACHES THE PERMANENT FILE CONTANING THE 15 CHANNELS OF     ****
C**     DIGITIZED DATA AND GIVES A NORMALIZED VERSION OF THE SPECTROGRAM.    ****
C*******************************************************************************
C*******************************************************************************
      PROGRAM OCTAVE(INPUT,OUTPUT,TAPE1,TAPE2,TAPE6=OUTPUT)
      DIMENSION SYMBOL2(10),SYMBOL3(10),SYMBOL4(10),SYMBOL5(10)
      DIMENSION B(16),SYMBOL1(10),BI(16),IBI(16),A(16)
      DATA SYMBOL1/1H ,1H ,1H+,1HX,1HX,1HX,1HX,1HX,1HX,1HX/
      DATA SYMBOL2/1H ,1H ,1H ,1H ,1H-,1H+,1HO,1HO,1HO,1HO/
      DATA SYMBOL3/1H ,1H ,1H ,1H ,1H ,1H ,1H ,1H-,1H-,1H#/
      DATA SYMBOL4/1H ,1H ,1H ,1H ,1H ,1H ,1H ,1H ,1H+,1H+/
      DATA SYMBOL5/1H ,1H ,1H ,1H ,1H ,1H ,1H ,1H ,1H ,1H*/
      NREC=20
      NN1=16
      NSTART=45
      NSTOP=200
      L=0
 1    CONTINUE
      DO 305 I=1,NSTOP
 5    READ(1,10)(B(J),J=1,NN1)
 10   FORMAT(16F6.3)
      IF(EOF(1))310,30
 30   CONTINUE
CXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
CXXXX                         NORMALIZATION  ROUTINE                       XXXXX
CXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
      SUME=0.0
      DO 33  J=1,15
```

```
      SUME=SUME + (B(J))**2
   33 CONTINUE
      DO 34  J =1,16
      ENERGY=SQRT(SUME)
      IF(ENERGY.GT.0.50) GO TO 32
      ENERGY=1.0
   32 CONTINUE
      B(J) = (B(J)/ENERGY)*10.
   34 CONTINUE
CXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
CXXXX                                END    NORMALIZATION                XXXX
CXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
      IF(I.LT.NSTART) GO TO 305
      L=L+1
      DO 240 JJ=1,16
      IF(B(JJ).LE.9.0) GO TO 31
      B(JJ)=9.0
   31 CONTINUE
      BI(JJ)=(B(JJ)+.5)
      IBI(JJ)=IFIX(BI(JJ))
  240 CONTINUE
      IF(L.GT.1) GO TO 295
      PRINT 250
  250 FORMAT(///,87X,*SYMBOLS REPRESENT INTEGER VALUES AS FOLLOWS:*)
      PRINT 260
  260 FORMAT(83X,"0=BLANK",2X,"1=( )",2X,"2=(+)",2X,"3=(X)",
     12X,"4=(X)")
      PRINT 266
  266 FORMAT("+",112X,"   - ")
      PRINT 261
```

```
251   FORMAT(83X,"5=(X)",2X,"6=(X)",2X,"7=(X)",2X,"8=(X)",2X,
     1"9=(X)")
      PRINT 262
262   FORMAT("+",82X,"    + "2X,"    0 ",2X,"    0 ",2X,"    0 ",2X,"    0 ")
      PRINT 263
263   FORMAT("+",95X,"    - ",2X,"    - ",2X,"    - ")
      PRINT 264
264   FORMAT("+",103X,"    + ",2X,"    + ")
      PRINT 265
265   FORMAT("+",110X,"    * ")
      PRINT 270
270   FORMAT(92X,*0000000001111111*)
      PRINT 280
280   FORMAT(92X,*1234567890123456*)
      PRINT 290
290   FORMAT(83X,*--------------------*)
295   CONTINUE
      PRINT 210,(B(JJ),JJ=1,16),I,(SYMBOL1(IBI(JJ)+1),JJ=1,15)
210   FORMAT(1X,15E5.2,8X,I3,16A1)
      PRINT 211,(SYMBOL2(IBI(JJ)+1),JJ=1,16)
      PRINT 211,(SYMBOL3(IBI(JJ)+1),JJ=1,16)
      PRINT 211,(SYMBOL4(IBI(JJ)+1),JJ=1,16)
      PRINT 211,(SYMBOL5(IBI(JJ)+1),JJ=1,16)
211   FORMAT("+",91X,16A1)
305   CONTINUE
      DO 306 I=1,110
      READ(1,10)(B(J),J=1,NN1)
      IF(EOF(1)) 310,306
 306  CONTINUE
310   CONTINUE
      NREC=NREC-1
      IF(NREC.GT.0) GO TO 1
      STOP
      END
```

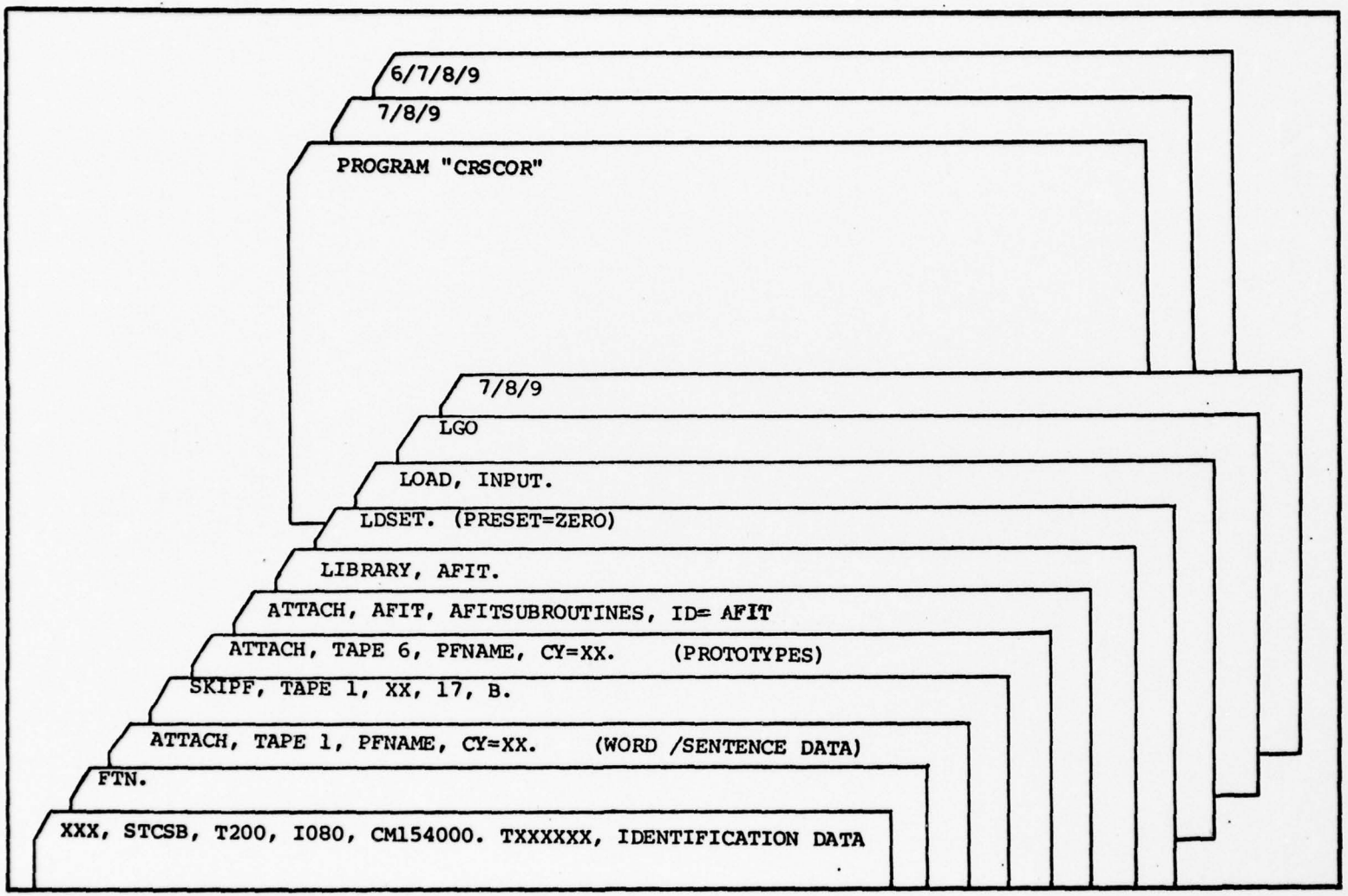


Fig. 12. Control Cards for Program "CRSCOR"

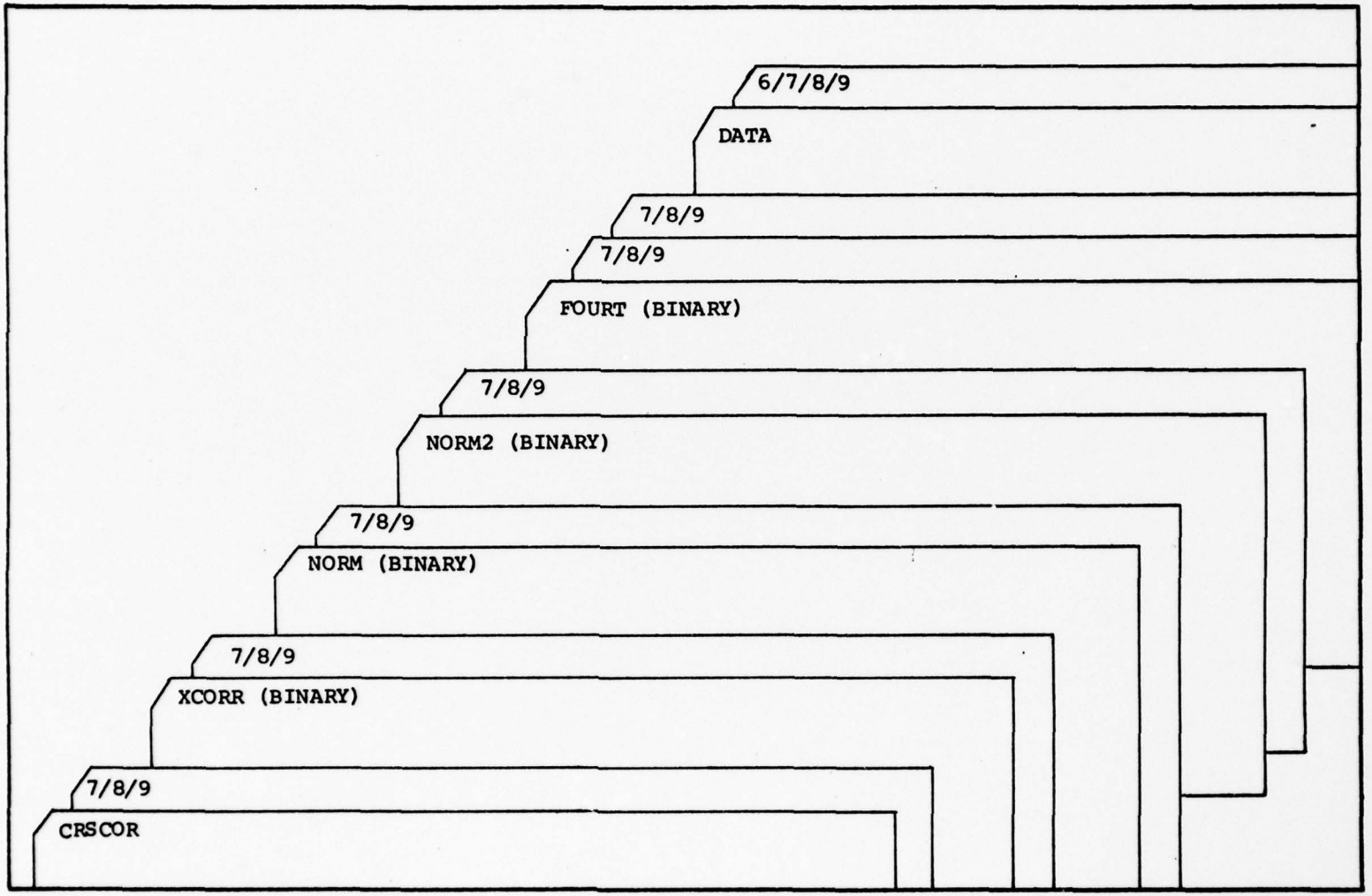


Fig. 13. CRSCOR Deck Structure

```
C**********************************************************************
C**********************************************************************
C**     THIS PROGRAM IS A SPEECH PHONEME RECOGNITION SCHEME BASED ON   **
C**    PROTOTYPE MATCHING. THE SPEECH DATA IS READ (ONE SENTENCE AT A TIME)  **
C**    FROM A FILE ATTACHED AS TAPE1. THE DATA MUST BE IN AN ARRAY 16XM,     **
C**    WHERE M<501. UP TO 61 PROTOTYPES OF SIZE 16XN,WHERE N<16,CAN BE       **
C**    ATTACHED AS TAPE6 OR READ FROM CARDS. THE PROGRAM VARIABLES ARE SET   **
C**    IN THE MAIN PROGRAM AND FED THROUGH COMMON TO THE SUBROUTINE XCORR     **
C**    WHERE ALL THE ANALYSIS TAKES PLACE.                                   **
C**********************************************************************
C**********************************************************************
      PROGRAM CRSCOR(INPUT,OUTPUT,TAPE1,TAPE2,TAPE6,TAPE9=OUTPUT)
      DIMENSION GOOD(64),ITYP(64)
      DIMENSION SYMBOL3(1),SYMBOL4(1),SYMBOL5(1),SYMBOL6(1),SYMBOL7(1)
      COMMON NSTART,NN2,NN3,NN5,ISJRLN,IOVLAP,NORMAL,NORMAR,ATOL,BTOL,
     1INHIB,LOOK,IDECID,GOOD,ITYP,ILIM,ILIN
C-----------------------------------------------------------------------
C--                    TITLE OF WORD/SENTENCE BEING READ                --
C-----------------------------------------------------------------------
      DATA SYMBOL3/10H          /
      DATA SYMBOL4/10H          /
      DATA SYMBOL5/10H          /
      DATA SYMBOL6/10H          /
      DATA SYMBOL7/10H          /
C######################################################################
C##                        VARIBLES USED BY PROGRAM                  ##
C##            (*) MUST BE SET FOR EACH SENTENCE/WORD CHANGE         ##
C######################################################################
C     *POSITION OF SENTENCE INFORMATION IN INPUT ARRAY
      NSTART=1
```

```
C     *MAXIMUM SENTENCE LENGTH
      NN2=200
C     *NUMBER OF THE SENTENCE/WORD BEING READ
      NN5=1
C     SIZE OF LARGEST PROTOTYPE + ONE
      NN3=16
C     LENGTH OF SUB-SENTENCE(THIS ESTABLISHES THE SIZE SECTIONS
C        THE SENTENCE IS BROKEN INTO)
      ISUBLN=48
C     DESIRED SUB-SENTENCE OVERLAP
      IOVLAP=8
C
C-----------------------------------------------------------------------
C--                          ENERGY NORMALIZATION                     --
C-----------------------------------------------------------------------
C     IF ENERGY NORMALIZATION OF DATA IS DESIRED,SET "NORMAL" TO "1"
C           OTHERWISE SET "NORMAL" TO "0"
      NORMAL=1
C-----------------------------------------------------------------------
C--                          SIGNIFICANCE VALUES                      --
C-----------------------------------------------------------------------
C     THE ALLOWED VARIABILITY OF A SINGLE PROTOTYPE WITHIN A SENTENCE
      ATOL=0.86
C      THE MINIMUM CORRELATION VALUE TOLERATED BASED ON:
C        (VALUE ACCEPTED>BTOL*MAX CORRELATION FOUND)
C-----------------------------------------------------------------------
      BTOL=0.86
C--                    THE ALLOWED OVERLAP OF PROTOTYPES              --
C-----------------------------------------------------------------------
C        PROTOTYPE SIZE "0<P<7"
```

```
      ILIL=1
C        PROTOTYPE SIZE "5<P<10"
      ILIM=2
C        PROTOTYPE SIZE "9<P<15"
      ILIN=3
C-------------------------------------------------------------------------------
C--                                PRINTOUTS                                  --
C-------------------------------------------------------------------------------
C     TO INHIBIT PRINTOUT OF PROTOTYPSET "INHIB" TO 1,OTHERWISE
C        SET "INHIB" TO 0.
      INHIB=0
C     IF PRINTOUT OF ALL CORRELATION COEFFICIENTS IS DESIRED SET "LOOK"
C        EQUAL TO "1" OTHERWISE SET "LOOK" EQUAL TO "0"
      LOOK=1
C     TO COMBINE DECISION STATEGY AND TOTAL CORRELATION PRINTOUT
C         SET IDECID EQUAL TO 1 OTHERWISE SET IDECID EQUAL TO 0
      IDECID=1
C-------------------------------------------------------------------------------
C--                        PROTOTYPE SIZE = ITYP(X)                           --
C--             MINIMUM ACCEPTABLE MAGNITUDE OF CORRELATION = GOOD(X)         --
C-------------------------------------------------------------------------------
C     /.I/
      ITYP(1)=12     $       GOOD(1)=0.157E3
C     /I/
      ITYP(2)=8      $      GOOD(2)=0.126E3
C     />E/
      ITYP(3)=8      $      GOOD(3)=0.128E3
C     /$E/
      ITYP(4)=9      $      GOOD(4)=0.136E3
C     /A/
```

```
      ITYP(5)=12      $      GOOD(5)=0.157E3
C     /C/
      ITYP(6)=12      $      GOOD(6)=0.157E3
C     /O/
      ITYP(7)=12      $      GOOD(7)=0.157E3
C     /U/
      ITYP(8)=12      $      GOOD(8)=0.157E3
C     /OO/
      ITYP(9)=8       $      GOOD(9)=0.128E3
C     /-A/
      ITYP(10)=8      $      GOOD(10)=0.128E3
C     /AE/
      ITYP(11)=15     $      GOOD(11)=0.175E3
C     /EI/
      ITYP(12)=12     $      GOOD(12)=0.157E3
C     /E/
      ITYP(13)=8      $      GOOD(13)=0.128E3
C     /UR/
      ITYP(14)=12     $      GOOD(14)=0.157E3
C     /RI/
      ITYP(15)=12     $      GOOD(15)=0.157E3
C     /CI/
      ITYP(16)=15     $      GOOD(16)=0.175E3
C     /AU/
      ITYP(17)=15     $      GOOD(17)=0.175E3
C     /V/
      ITYP(18)=5      $      GOOD(18)=0.111E3
C     /VI/
      ITYP(19)=5      $      GOOD(19)=0.111E3
C     /VT/
```

```
      ITYP(20)=6      $     GOOD(20)=0.111E3
C     /VM/
      ITYP(21)=6      $     GOOD(21)=0.111E3
C     /VU/
      ITYP(22)=6      $     GOOD(22)=0.111E3
C     /VN/
      ITYP(23)=6      $     GOOD(23)=0.111E3
C     /VS/
      ITYP(24)=6      $     GOOD(24)=0.111E3
C     /T/
      ITYP(25)=6      $     GOOD(25)=0.111E3
C     /TA/
      ITYP(26)=6      $     GOOD(26)=0.111E3
C     /TE/
      ITYP(27)=6      $     GOOD(27)=0.111E3
C     /G/
      ITYP(28)=6      $     GOOD(28)=0.111E3
C     /GU/
      ITYP(29)=6      $     GOOD(29)=0.111E3
C     /GD/
      ITYP(30)=6      $     GOOD(30)=0.111E3
C     /J/
      ITYP(31)=6      $     GOOD(31)=0.111E3
C     /JE/
      ITYP(32)=6      $     GOOD(32)=0.111E3
C     /JL/
      ITYP(33)=6      $     GOOD(33)=0.111E3
C     /OF/
      ITYP(34)=6      $     GOOD(34)=0.111E3
C     /F/
```

```
      ITYP(35)=5        $     GOOD(35)=0.111E3
C     /P/
      ITYP(36)=5        $     GOOD(36)=0.111E3
C     /SE/
      ITYP(37)=5        $     GOOD(37)=0.111E3
C     /CH/
      ITYP(38)=5        $     GOOD(38)=0.111E3
C     /.A/
      ITYP(39)=5        $     GOOD(39)=0.111E3
C     /H/
      ITYP(40)=5        $     GOOD(40)=0.111E3
C     /HE/
      ITYP(41)=5        $     GOOD(41)=0.111E3
C     /R/
      ITYP(42)=5        $     GOOD(42)=0.111E3
C     /RE/
      ITYP(43)=5        $     GOOD(43)=0.111E3
C     /B/
      ITYP(44)=5        $     GOOD(44)=0.111E3
C     /RA/
      ITYP(45)=5        $     GOOD(45)=0.111E3
C     /RK/
      ITYP(46)=5        $     GOOD(46)=0.111E3
C     /RG/
      ITYP(47)=5        $     GOOD(47)=0.111E3
C     /BA/
      ITYP(48)=5        $     GOOD(48)=0.111E3
C     /BE/
      ITYP(49)=5        $     GOOD(49)=0.111E3
C     /BC/
```

```
      ITYP(50)=5      $     GOOD(50)=0.111E3
C     /C/
      ITYP(51)=5      $     GOOD(51)=0.111E3
C     /CP/
      ITYP(52)=5      $     GOOD(52)=0.111E3
C     /CY/
      ITYP(53)=5      $     GOOD(53)=0.111E3
C     /SM/
      ITYP(54)=5      $     GOOD(54)=0.111E3
C     /DB/
      ITYP(55)=5      $     GOOD(55)=0.111E3
C     /AA/
      ITYP(56)=5      $     GOOD(56)=0.111E3
C     /DZ/
      ITYP(57)=5      $     GOOD(57)=0.111E3
C     /NT/
      ITYP(58)=5      $     GOOD(58)=0.111E3
C     /ZH/
      ITYP(59)=5      $     GOOD(59)=0.111E3
C     /W/
      ITYP(60)=5      $     GOOD(60)=0.111E3
C     /Z/
      ITYP(61)=5      $     GOOD(61)=0.111E3
C-----------------------------------------------------------------------
C--                        OUTPUT SENTENCE/WORD TITLE                 --
C-----------------------------------------------------------------------
      PRINT 3
 3    FORMAT(////,1X,"THE WORD/SENTENCE BEING ANALYZED IS :")
      PRINT 4,SYMBOL3,SYMBOL4,SYMBOL5,SYMBOL6,SYMBOL7
 4    FORMAT(1X,5A10)
```

```
C-------------------------------------------------------------------------
C--                        TRANSFER CONTROL TO SUBROUTINE                --
C-------------------------------------------------------------------------
      CALL XCORR
C
      STOP
      END


C*************************************************************************
C*************************************************************************
C**     THIS SUBROUTINE USES FFT TECHNIQUES TO CROSSCORRELATE PROTOTYPES  **
C**    WITH SPEECH DATA.THE OUTPUT IS A PHONEMIC REPRESENTATION OF THE INPUT.  **
C**    ALSO INCLUDED AS AN OUTPUT IS ALL THE CORRELATION COEFFICIENTS FOR  **
C**    EACH PROTOTYPE  BY RANK,IN TIME OCCURRANCE ORDER.                   **
C*************************************************************************
C*************************************************************************
      SUBROUTINE XCORR
      COMPLEX SENT(64,32),CPROTO(64,32),CONPRO(64,32),CORR(64,32)
      REAL MARR
      DIMENSION NN(2),B(500,16),PROTO(16,16),C(64,16),D(64,16)
      DIMENSION ACORR(64,64),ICORR(64,64),ACOPAR(64),ICOPAR(64)
      DIMENSION IPRO(64,64),PRO(64,64)
      DIMENSION SYMBOL1(62),SYMBOL2(61),SUMM(64),EPROTO(16,16)
      DIMENSION INDEX(64),ARRAY(64),ARRAYR(64),INDEXR(64)
      DIMENSION IARRAY(64),IHOLD(64),GOOD(64),ITYP(64),IEQUAL(10)
      DIMENSION SYMBOL3(1),SYMBOL4(1),SYMBOL5(1),SYMBOL6(1),SYMBOL7(1),
     1SAVE(64)
```

```
      COMMON NSTART,NN2,NN3,NN5,ISJBLN,IOVLAP,NORMAL,NORMAR,ATOL,BTOL,
     1INHIB,LOOK,IDECID,GOOD,ITYP,ILIM,ILIN
      EQUIVALENCE (CPROTO,CORR)
C     PHONEME SYMBOL SET
      DATA SYMBOL1/4H(.I),4H( I),4H(>E),4H($E),4H( A),4H( C),4H( O),
     14H( U),4H(OO),4H(-A),4H(AE),4H(EI),4H( E),4H(UR),4H($I),4H(CI),
     14H(AU),4H( V),4H(VI),4H(VT),4H(VM),4H(VU),4H(VN),4H(VS),4H( T),
     14H(TA),4H(TE),4H( G),4H(GJ),4H(GD),4H( J),4H(JE),4H(JL),4H(OF),
     14H( F),4H( P),4H(SE),4H(CH),4H(.A),4H( H),4H(HE),4H( R),4H(RE),
     14H( B),4H(BA),4H(BK),4H(BG),4H(BA),4H(BE),4H(BC),4H( C),4H(CP),
     14H(CY),4H(SN),4H(OR),4H(AA),4H(DZ),4H(NT),4H(ZH),4H( W),4H( Z),
     14H     /
C     PHONEME-WORD SET
      DATA SYMBOL2/6H   EVE,6H    IT,6H   MET,6H    AT,6H   NOT,
     16H   ALL,6H  OBEY,6H   PUT,6H  BOOT,6H    UP,6H   ATE,6H  CAME,
     16H   ATE,6HCHURCH,6H     I,6H   BOY,6H   OUT,6H  VITA,6H  VITA,
     16H  VITA,6H  VITA,6H  VITA,6H  VITA,6H  VITA,6H TASTE,6H TASTE,
     16H TASTE,6H  GOOD,6H  GOOD,6H  GOOD,6H JOURN,6H JOURN,6H JOURN,
     16H    OF,6H     F,6HSPEECH,6HSPEECH,6HSPEECH,6HSPEECH,6H  HEAR,
     16H  HEAR,6H RESER,6H RESER,6HBREEDS,6H RECOG,6H RECOG,6H RECOG,
     16H BASIC,6H BASIC,6H BASIC,6H COMPJ,6H COMPU,6H COMPU,6H COMPU,
     16H  OBEY,6H  OBEY,6H JUDGE,6H   NOT,6H CLOSE,6H  WILL,6H  CAJZ/
C-----------------------------------------------------------------------
C--                      VARIABLES PERTINENT TO TRANSFORM                --
C-----------------------------------------------------------------------
C     THE MAXIMUM ARRAY SIZE THAT CAN BE TRANSFORMED IS
      NN(1)=64
      NN(2)=32
C     SIZE OF REDUCED ARRAY
      NN4=16
```

```
C      SIZE OF EXPANDED ARRAY
       NN10=32
C      STARTING POINT IN J DIRECTION TO IMBED ARRAY IN ZEROS
       NN11=NN4+1
C      NUMBER OF PROTOTYPES
       NPRO=61
C      A VALUE ONE MORE THAN THE NUMBER OF SYMBOLS PROVIDED
       NZERO=62
C      LENGTH OF ARRAY TO BE SORTED
       N=64
C-------------------------------------------------------------------------------
C--               READ COMPRESSED SENTENCE DATA FROM PERMANENT FILE           --
C-------------------------------------------------------------------------------
       DO 20 I=1,NN2
  5    READ(1,10)(B(I,J),J=1,NN4)
 10    FORMAT(16F6.3)
       IF(EOF(1))30,20
 20    CONTINUE
 30    CONTINUE
       INEND=I-1
       PRINT 22,NN5,INEND
 22    FORMAT(/,1X,"THE LENGTH OF THE SENTENCE #",I2,1X,"IS",I4)
C
C
C
C-------------------------------------------------------------------------------
C--             REDUCE SENTENCE TO SUB-SENTENCES OF LENGTH "ISUBLN"           --
C-------------------------------------------------------------------------------
       ISCLIM=((INEND-NSTART)/(ISUBLN-IOV_AP))+1
       PRINT 25,ISCLIM
```

```
 25   FORMAT(/,1X,"THE NUMBER OF SUB-SENTENCES REQUIRED IS",I3)
      K=1
      MSTART=0
      MSTOP=0
      DO 500 ISECTN=1,ISCLIM
      IF(MSTOP.GE.INEND) GO TO 500
      IF(ISECTN.EQ.1) GO TO 28
      REWIND 2
 28   CONTINUE
C     INITIAL VALUE FOR FINAL CORRELATION VECTOR LENGTH
      IEND =0
      IF(ISECTN.NE.1) GO TO 31
      MSTART=NSTART
      GO TO 32
 31   CONTINUE
      MSTART=(MSTOP+1)-IOVLAP
 32   CONTINUE
      MSTOP=MSTART+(ISUBLN-1)
      IF(MSTOP.LE.INEND) GO TO 37
      MSTOP=INEND
 37   CONTINUE
      I=1
      DO 35 K=MSTART,MSTOP
      DO 34 J=1,NN4
      C(I,J)=B(K,J)
 34   CONTINUE
      I=I+1
 35   CONTINUE
      LEN=I-1
      PRINT 33,ISECTN,LEN
```

```
 33   FORMAT(////,1X,"THE LENGTH OF SUB-SENTENCE #",I2,1X,"IS",I4)
      IF(LEN.LT.22) GO TO 706
C
C-----------------------------------------------------------------------------
      IF(NORMAL.NE.1) GO TO 123
C--                          ENERGY NORMALIZE SENTENCE                      --
C-----------------------------------------------------------------------------
      IASIZE=64
      CALL NORM(C,D,LEN,NN4,IASIZE)
      GO TO 128
 123  CONTINUE
      DO 127 II=1,LEN
      DO 127 JJ=1,NN4
      D(II,JJ)=C(II,JJ)
 127  CONTINUE
 128  CONTINUE
C-----------------------------------------------------------------------------
C--                MAKE SENTENCE COMPLEX AND APPEND TO ZEROS                --
C-----------------------------------------------------------------------------
      IP=N-LEN
      PRINT 183 ,IP
 183  FORMAT(/,1X,"THE NUMBER OF ZEROS ADDED TO THE SUB-SENTENCE",
     1I4,/)
      DO 210 NK=1,IP
      DO 210 JJ=1,NN10
      SENT(NK,JJ)=(0.,0.)
 210  CONTINUE
      IP1=IP+1
      II=1
      DO 220 NK=IP1,N
```

```
      DO 215 JJ=1,NN4
      SENT(NK,JJ)=D(II,JJ)
  215 CONTINUE
      II=II+1
  220 CONTINUE
      DO 211 NK=IP1,N
      DO 211 JJ=NN11,NN10
      SENT(NK,JJ)=(0.,0.)
  211 CONTINUE
C-----------------------------------------------------------------------
C--                                  FFT SENTENCE                     --
C-----------------------------------------------------------------------
      CALL FOURT(SENT,NN,2,-1,0,0)
C
C
C#######################################################################
C##                          CROSSCORRELATION SEQUENCE                  ##
C#######################################################################
      DO 400 JP=1,NPRO
C-----------------------------------------------------------------------
C--                  READ PROTOTYPE FROM CARDS/PERMANENT FILE          --
C-----------------------------------------------------------------------
      IF(ISECTN.GT.1) GO TO 870
      DO 150 K=1,NN3
      READ(6,140)(PROTO(K,L),L=1,NN4)
  140 FORMAT(16F6.3)
      IF(EOF(6).NE.0) GO TO 151
  150 CONTINUE
  151 CONTINUE
      GO TO 875
```

```
870   CONTINUE
      DO 874 K=1,NN3
      READ(2,871)(PROTO(K,L),L=1,NN4)
871   FORMAT(16F6.3)
      IF(EOF(2).NE.0) GO TO 875
874   CONTINUE
875   CONTINUE
      NUM=K-1
      IF(INHIB.EQ.0) GO TO 147
      PRINT 153,JP,NUM
153   FORMAT(//,1X,"THE LENGTH OF PROTOTYPE #",I2,1X,"IS",I3)
      PRINT 144,SYMBOL1(JP),SYMBOL2(JP)
144   FORMAT(/,1X,"THE PROTOTYPE REPRESENTS",1X,A4,1X,"AS IN(",A5,")")
147   CONTINUE
      IF(ISECTN.GT.1) GO TO 800
      DO 152 K=1,NUM
      IF(INHIB.EQ.0) GO TO 148
      WRITE(9,146)(PROTO(K,L),L=1,NN4)
146   FORMAT(1X,16F6.3)
148   CONTINUE
      IF(ISECTN.GT.1) GO TO 152
      WRITE(2,145)(PROTO(K,L),L=1,NN4)
145   FORMAT(16F6.3)
152   CONTINUE
      IF(ISECTN.GT.1) GO TO 800
      ENDFILE2
800   CONTINUE
      IF(NORMAL.NE.1) GO TO 159
C-----------------------------------------------------------------------
C--                    ENERGY NORMALIZE PROTOTYPE                     --
```

```
C-----------------------------------------------------------------------
 140  CONTINUE
      IASIZE=15
      CALL NORM(PROTO,EPROTO,NUM,NN4,IASIZE)
      IF(INHIB.EQ.0) GO TO 969
      IF(ISECTN.GT.1) GO TO 969
      PRINT 966
 966  FORMAT(/,1X,"VECTOR NORMALIZED PROTOTYPE")
      DO 967 K=1,NUM
      WRITE(9,155)(EPROTO(K,L),L=1,NN4)
 155  FORMAT(1X,16F6.3)
 967  CONTINUE
 969  CONTINUE
 154  CONTINUE
 801  CONTINUE
      GO TO 161
 150  CONTINUE
      DO 157 II=1,NUM
      DO 157 JJ=1,NN4
      EPROTO(II,JJ)=PROTO(II,JJ)
 157  CONTINUE
 161  CONTINUE
C-----------------------------------------------------------------------
C--              DETERMINE NUMBER OF ZEROS REQUIRED TO PREVENT "END EFFECT"    --
C-----------------------------------------------------------------------
      IZ=1
      ZEROS=NUM+LEN
      MARR=ZEROS
 160  MARR=MARR/2
      IF(MARR.LT.2) GO TO 170
```

```
      IZ=IZ+1
      GO TO 160
  170 IZ=IZ+1
      IDIN=2**IZ
      IF(INHIB.EQ.0) GO TO 171
      PRINT 173,IDIN
  171 CONTINUE
      IF(IDIN.GT.N) GO TO 704
  173 FORMAT(/,1X,"THE LENGTH OF SUPPLEMENTED PROTOTYPE & SENTENCE VECTO,
     1RS ARE",I4)
      IF (IDIN.GT.64) GO TO 702
C-----------------------------------------------------------------------
C--            MAKE PROTOTYPE COMPLEX AND APPEND NECESSARY ZEROS      --
C-----------------------------------------------------------------------
      DO 176 K=1,NUM
      DO 176 L=1,NN4
      CPROTO(K,L)=EPROTO(K,L)
  176 CONTINUE
      DO 177 K=1,NUM
      DO 177 L=NN11,NN10
      CPROTO(K,L)=(0.,0.)
  177 CONTINUE
      NUM1=NUM+1
      DO 180 K=NUM1,IDIN
      DO 180 L=1,NN10
      CPROTO(K,L)=(0.,0.)
  180 CONTINUE
C-----------------------------------------------------------------------
C--                          FFT PROTOTYPE                            --
C-----------------------------------------------------------------------
```

```
      CALL FOURT(CPROTO,NN,2,-1,0,0)
C-----------------------------------------------------------------------
C--                  FIND  COMPLEX CONJUGATE OF PROTOTYPE              --
C-----------------------------------------------------------------------
      DO 200 K=1,IDIN
      DO 200 L=1,NN10
      CONPRO(K,L)=CONJG(CPROTO(K,L))
  200 CONTINUE
      IF(INHIB.EQ.0) GO TO 201
      PRINT*,ENG
  201 CONTINUE
C-----------------------------------------------------------------------
C--              PROTOTYPE  UNIT  NORMILIZATION                       --
C-----------------------------------------------------------------------
      SUME=0.0
      DO 996 I=1,64
      DO 996 J=1,32
      E=REAL(CONPRO(I,J))
      F=AIMAG(CONPRO(I,J))
      G=E**2+F**2
      SUME = SUME + G
  996 CONTINUE
      ENERGY = SQRT(SUME)
      DO 997 I=1,64
      DO 997 J=1,32
      CONPRO(I,J) = CONPRO(I,J)/ENERGY
  997 CONTINUE
C-----------------------------------------------------------------------
C-----------------------------------------------------------------------
C          FREQUENCY SELECTION FILTER  WITH VARIABLES WIDTH AND MENGTH
```

```
C-----------------------------------------------------------------------
      MIDTH=25
      MENGTH=45
      MM=MIDTH/2+1
      N4=NN10-MIDTH/2
      J=IDIN-(MENGTH/2-1)
      II=MENGTH/2+1
      DO 990 K1=1,IDIN
      DO 990 K2=1,NN10
      IF(K2.GT.MM.AND.K2.LE.N4) CONPRO(K1,K2)=(0.0,0.0)
      IF(K1.GT.II.AND.K1.LT.J) CONPRO(K1,K2) =(0.0,0.0)
  990 CONTINUE
C-----------------------------------------------------------------------
C--                 CACULATE CORRELATION IN FREQUENCY DOMAIN          --
C-----------------------------------------------------------------------
      DO 250 K=1,IDIN
      DO 250 L=1,NN10
      CORR(K,L)=CONPRO(K,L)*SENT(K,L)
  250 CONTINUE
C-----------------------------------------------------------------------
C--                          TAKE INVERSE TRANSFORM                   --
C-----------------------------------------------------------------------
      CALL FOURT(CORR,NN,2,+1,+1,0)
      DO 290 IK=1,IDIN
      SUMM(IK)=CORR(IK,1)
  290 CONTINUE
C-----------------------------------------------------------------------
C--          USE SUBROUTINE TO SORT CORRELATION COEFFICIENT ARRAY      --
C-----------------------------------------------------------------------
      DO 297 IN=1,N
```

```
  297  ARRAY(IN)=SUMM(IN)
       CALL SORT(N,ARRAY)
C
C-------------------------------------------------------------------------------
C--              DETERMINE SENTENCE POSITION OF ARRAY ELEMENT                 --
C-------------------------------------------------------------------------------
       DO 450 IN=1,IDIN
       DO 420 IK=1,IDIN
       IF(SUMM(IK).EQ.ARRAY(IN)) GO TO 430
  420  CONTINUE
       GO TO 431
  430  INDEX(IN)=IK
  431  CONTINUE
  450  CONTINUE
C-------------------------------------------------------------------------------
C--              REARRANGE ARRAY AND INDEX IN DESENDING ORDER                 --
C-------------------------------------------------------------------------------
       DO 470 IN=1,IDIN
       IM=IDIN+1-IN
       INDEXR(IM)=INDEX(IN)
       ARRAYR(IM)=ARRAY(IN)
  470  CONTINUE
C-------------------------------------------------------------------------------
C--                    USE SUBROUTINE TO SORT INDEX ARRAY                     --
C-------------------------------------------------------------------------------
       DO 505 IN=1,N
  505  IARRAY(IN)=INDEXR(IN)
       CALL SORT(N,IARRAY)
       DO 520 IN=1,IDIN
       DO 510 IK=1,IDIN
```

```
      IF(INDEXR(IK).EQ.IARRAY(IN)) GO TO 515
  510 CONTINUE
      GO TO 516
  515 INDEX(IN)=IK
  516 CONTINUE
  520 CONTINUE
C
C
C-----------------------------------------------------------------------
C--     DETERMINE IF CORRELATION IS SUFFICIENT TO WARRENT CONSIDERATION --
C-----------------------------------------------------------------------
      IF(IDECID.EQ.1) GO TO 528
      IF(LOOK.EQ.1) GO TO 527
      IF(ARRAYR(1).LT.GOOD(JP)*BTOL)  GO TO 338
      GO TO 527
  528 IF(ARRAYR(1).LT.GOOD(JP)*BTOL) GO TO 522
  527 AMAXX=ARRAYR(1)
      DMAXX=(AMAXX/GOOD(JP))*100
      PRINT 521,SYMBOL1(JP),AMAXX,DMAXX
  521 FORMAT(/,1X,"MAX CORRELATION FOUND FOR PROTOTYPE",1X,A4,1X," =",
     1E9.3,"(",F5.1,"%)")
      IF(ISECTN.GT.1) GO TO 525
      ASAVE(JP)=ARRAYR(1)
  525 CONTINUE
      IF(ARRAYR(1).LT.ASAVE(JP)) GO TO 529
      ASAVE(JP)=ARRAYR(1)
  529 CONTINUE
C-----------------------------------------------------------------------
C--       DETERMINE NUMBER & LOCATION OF "POSSIBLE" PHONEME OCCURANCES --
C-----------------------------------------------------------------------
```

```
      IFIRST=0
      ID=1
      DO 660 I=1,IDIN
      IF(ARRAYR(I).LT.ATOL*AMAXX) GO TO 522
      IF(IFIRST.EQ.0) GO TO 611
      IF(NUM.GT.6) GO TO 607
      IDIFF=ILIL
      GO TO 609
  607 IF(NUM.GT.9) GO TO 608
      IDIFF=ILIM
      GO TO 609
  608 IDIFF=ILIN
  609 CONTINUE
      DO 610 IE=1,ID
      IF(ID.GT.60) GO TO 703
      IF(INDEXR(I).LT.(IHOLD(IE)+NJM-(1+IDIFF))) GO TO 613
      GO TO 610
  613 CONTINUE
      IF((INDEXR(I)+(NUM-1)).GT.(IHOLD(IE)+IDIFF)) GO TO 660
  610 CONTINUE
      GO TO 612
  611 IFIRST=1
  612 IHOLD(ID)=INDEXR(I)
      IPOS=INDEXR(I)
      IB=IPOS+(NUM-1)
      ID=ID+1
      DO 659 IA=IPOS,IB
      ACORR(IA,JP)=(ARRAYR(I)/GOOD(JP))
      ICORR(IA,JP)=JP
  659 CONTINUE
```

```
  550  CONTINUE
  522  CONTINUE
C-------------------------------------------------------------------------------
C--       STORE THE CORRELATION VECTOR AND THE COMPUTED SENTENCE RANK          --
C-------------------------------------------------------------------------------
       ISTORE=0
       IDEN=IP+1
       LP=1
  401  DO 402 IL=IDEN,IDIN
       IPRO(LP,JP)=INDEX(IL)
       PRO(LP,JP)=SUMM(IL)/GOOD(JP)
       LP=LP+1
  402  CONTINUE
       ISTORE=LP-1
       IF(ARRAYR(1).LT.GOOD(JP)*BTOL) GO TO 397
       IF(ISTORE.LT.IEND) GO TO 400
       IEND=ISTORE
       GO TO 400
  397  CONTINUE
       IF(ISTORE.LT.IEND) GO TO 398
       IEND=ISTORE
  398  CONTINUE
       DMAX=ARRAYR(1)
       DO 399 I=1,N
       ACORR(I,JP)=0.
       ICORR(I,JP)=0
       IF(IDECID.EQ.1) GO TO 399
       PRO(I,JP)=0.
       IPRO(I,JP)=0
  399  CONTINUE
```

```
      DCENT=(DMAX/GOOD(JP))*100
      PRINT 411,SYMBOL1(JP),DMAX,DCENT
  411 FORMAT(1X,"THE MAXIMUM CORRELATION COEFFICIENT LOCATED FOR",1X,A4,
     11X,"WAS",E9.3,1X,"WHICH IS",1X,F5.1,"% OF GOOD(X)")
  400 CONTINUE
      PRINT*,IEND
  391 CONTINUE
C-----------------------------------------------------------------------
C--                          OUTPUT CORRELATION DATA                  --
C-----------------------------------------------------------------------
      PRINT 394
  394 FORMAT(1X,"PROTOTYPE LOCATION")
      DO 396 I=IDEN,IDIN
      WRITE(9,392)(ICORR(I,J),J=1,NPRO)
  392 FORMAT(1X,9I1,52I2)
  396 CONTINUE
      PRINT 395
  395 FORMAT(1X,"PROTOTYPE LOCATION - MAX-CORRELATION COEFFICIENT")
      WRITE(9,490)(SYMBOL1(I),I=1,12)
  490 FORMAT(1X,12(A4,7X))
      DO 491 I=IDEN,IDIN
      WRITE(9,492)(ACORR(I,J),J=1,12)
  492 FORMAT(1X,12(E9.3,2X))
  491 CONTINUE
      IF(NPRO.LE.12) GO TO 497
      WRITE(9,490)(SYMBOL1(I),I=13,24)
      DO 493 I=IDEN,IDIN
      WRITE(9,492)(ACORR(I,J),J=13,24)
  493 CONTINUE
      IF(NPRO.LE.24) GO TO 497
```

```
      WRITE(9,490)(SYMBOL1(I),I=25,36)
      DO 494 I=IDEN,IDIN
      WRITE(9,492)(ACORR(I,J),J=25,36)
  494 CONTINUE
      IF(NPRO.LE.36) GO TO 497
      WRITE(9,490)(SYMBOL1(I),I=37,48)
      DO 496 I=IDEN,IDIN
      WRITE(9,492)(ACORR(I,J),J=37,48)
  496 CONTINUE
      IF(NPRO.LE.48) GO TO 497
      WRITE(9,490)(SYMBOL1(I),I=49,60)
      DO 495 I=IDEN,IDIN
      WRITE(9,492)(ACORR(I,J),J=49,60)
  495 CONTINUE
      IF(NPRO.LE.60) GO TO 497
      WRITE(9,498)(SYMBOL1(61))
  498 FORMAT (1X,A4)
      DO 900 I=IDEN,IDIN
      WRITE(9,901)(ACORR(I,61))
  901 FORMAT(1X,E8.3)
  900 CONTINUE
  497 CONTINUE
C
C
C
C
C-----------------------------------------------------------------------
C--                          SELECT PHONEMES                          --
C-----------------------------------------------------------------------
      DO 673 I=1,IDIN
```

```
      IZ=1
      JJ=2
      COMPAR=ACORR(I,1)
      IOMPAR=1
      DO 670 J=JJ,NPRO
      IF(COMPAR.LT.ACORR(I,J)) GO TO 668
      IF(COMPAR.LE.0) GO TO 668
      IF(COMPAR.NE.ACORR(I,J)) GO TO 670
      IEQUAL(IZ)=J
      IZ=IZ+1
      IF(IZ.LT.10) GO TO 670
      PRINT 679,ACORR(I,J)
  679 FORMAT(1X,"DANGER-SCREWY ARRAY ENCOUNTERED,ALLVALUES EQUAL",F6.3)
      GO TO 705
  668 CONTINUE
      COMPAR=ACORR(I,J)
      IOMPAR=J
      GO TO 670
  668 CONTINUE
      IOMPAR=0
  670 CONTINUE
      IF(IZ.EQ.1) GO TO 672
      PRINT 665,I
  665 FORMAT(1X,I4,3X,"#CAUTION# EQUAL CORRELATION COEFFICIENTS FOUND")
      WRITE(9,666)(IEQUAL(J),J=1,IZ)
  666 FORMAT(1X,10(I4))
      DO 667 J=1,10
  667 IEQUAL(J)=0
  672 CONTINUE
      ACOPAR(I)=COMPAR
```

```
      ICOPAR(I)=IOMPAR
  673 CONTINUE
C-----------------------------------------------------------------------
C--                        OUTPUT DATA BEFORE FINAL DECISION           --
C-----------------------------------------------------------------------
      PRINT 675
  675 FORMAT(1X,"DATA BEFORE FINAL DECISION SCHEME")
      WRITE(9,674)(ICOPAR(I),I=IDEN,IDIN)
  674 FORMAT(1X,45(I3))
C-----------------------------------------------------------------------
C--                        COMPLETE FINAL DECISION SCHEME              --
C-----------------------------------------------------------------------
      J=1
      K=1
  671 IF(ICOPAR(J).EQ.0) GO TO 691
      ITYPE=ITYP(ICOPAR(J))
      LK=J
      KTYPE=J+(ITYPE-1)
      K=0
      DO 680 I=LK,KTYPE
      K=K+1
      IF(ACOPAR(I).NE.ACOPAR(I+1)) GO TO 681
  680 CONTINUE
      GO TO 692
  681 XTOL=(0.75)*FLOAT(ITYPE+1)
      ITOL=IFIX(XTOL)
      IF(K.GE.ITOL) GO TO 692
      IF(K.NE.1) GO TO 682
      JL=J
      GO TO 683
```

```
  582   JL=J+K
  583   CONTINUE
        DO 690 IF=J,JL
        ACOPAR(IF)=0.
        ICOPAR(IF)=0
  690   CONTINUE
        GO TO 682
  681   K=1
  682   J=J+K
        IF(J.LT.IDIN) GO TO 671
        DO 695 IN=1,IDIN
        IF(ICOPAR(IN).NE.0) GO TO 695
        ICOPAR(IN)=NZERO
  695   CONTINUE
C-------------------------------------------------------------------------------
C--                  OUTPUT PHONEMIC SENTENCE REPRESENTATION                  --
C-------------------------------------------------------------------------------
        PRINT 696
  696   FORMAT(1X,"PHONEMIC REPRESENTATION OF SENTENCE")
        DO 751 I=IDEN,IDIN
        M=I+(MSTART-(IP+1))
        WRITE(9,750)M,SYMBOL1(ICOPAR(I))
  750   FORMAT(1X,I4,3X,A4)
  751   CONTINUE
C-------------------------------------------------------------------------------
C--                       OUTPUT CORRELATION COEFFICIENTS                     --
C-------------------------------------------------------------------------------
        PRINT*,IEND
        WRITE(9,551)(SYMBOL1(I),I=1,8)
  551   FORMAT(1X,"SENT",5X,A4,7(12X,A4))
```

```
      PRINT 552
552   FORMAT(6X,8(1X,"------------",3X))
      DO 600 I=1,IEND
      J=I+(MSTART-1)
      WRITE(9,550)J,(PRO(I,J),IPRO(I,J),J=1,8)
550   FORMAT(1X,I3,2X,8(E9.3,1X,I3,3X))
      IF(IEND.GT.200) GO TO 600
600   CONTINUE
      IF(NPRO.LE. 8) GO TO 557
      WRITE(9,551)(SYMBOL1(I),I=9,16)
      PRINT 552
      DO 756 I=1,IEND
      J=I+(MSTART-1)
      WRITE(9,550)J,(PRO(I,J),IPRO(I,J),J=9,16)
756   CONTINUE
      IF(NPRO.LE.16) GO TO 557
      WRITE(9,551)(SYMBOL1(I),I=17,24)
      PRINT 552
      DO 752 I=1,IEND
      J=I+(MSTART-1)
      WRITE(9,550)J,(PRO(I,J),IPRO(I,J),J=17,24)
752   CONTINUE
      IF(NPRO.LE.24) GO TO 557
      WRITE(9,551)(SYMBOL1(I),I=25,32)
      PRINT 552
      DO 753 I=1,IEND
      J=I+(MSTART-1)
      WRITE(9,550)J,(PRO(I,J),IPRO(I,J),J=25,32)
753   CONTINUE
      IF(NPRO.LE.32) GO TO 557
```

```
      WRITE(9,551)(SYMBOL1(I),I=33,40)
      PRINT 552
      DO 754 I=1,IEND
      J=I+(MSTART-1)
      WRITE(9,550)J,(PRO(I,J),IPRO(I,J),J=33,40)
  754 CONTINUE
      IF(NPRO.LE.40) GO TO 557
      WRITE(9,551)(SYMBOL1(I),I=41,48)
      PRINT 552
      DO 755 I=1,IEND
      J=I+(MSTART-1)
      WRITE(9,550)J,(PRO(I,J),IPRO(I,J),J=41,48)
  755 CONTINUE
      IF(NPRO.LE.48) GO TO 557
      WRITE(9,551)(SYMBOL1(I),I=49,56)
      PRINT 552
      DO 556 I=1,IEND
      J=I+(MSTART-1)
      WRITE(9,550)J,(PRO(I,J),IPRO(I,J),J=49,56)
  556 CONTINUE
      IF(NPRO.LE.56) GO TO 557
      WRITE(9,902)(SYMBOL1(I),I=57,61)
  902 FORMAT(1X,"SENT",3X,A4,5(12X,A4))
      PRINT 552
      DO 903 I=1,IEND
      J=I+(MSTART-1)
      WRITE(9,904)J,(PRO(I,J),IPRO(I,J),J=57,61)
  904 FORMAT(1X,I3,2X,5(E9.3,1X,I3,3X))
  903 CONTINUE
  557 CONTINUE
```

```
      DO 599 I=1,N
      DO 599 J=1,NPRO
      ACORR(I, J)=0.
      ICORR(I, J)=0
      PRO(I, J)=0.
      IPRO(I, J)=0
  599 CONTINUE
  600 CONTINUE
C-----------------------------------------------------------------------------
C--   OUTPUT THE MAXIMUM CORRELATION COEFFICIENTS FOUND FOR EACH PROTOTYPE    --
C-----------------------------------------------------------------------------
      DO 501 I=1,NPRO
      BSAVE=ATO_*ASAVE(I)
      WRITE(9,502)I,SYMBOL1(I),SYMBOL2(I),ASAVE(I),BSAVE
  502 FORMAT(//,1X,I2,2X,"THE PROTOTYPE",1X,A4,1X,"AS IN",1X,A5,1X,"HAS
     1A MAX CORRELATION OF",1X,E9.3,1X,"(CUTOFF=",E9.3)
  501 CONTINUE
      RETURN
  702 STOP"ARRAY EXCEEDS DIMENSIONS"
  703 STOP"ID EXCEEDS LIMIT"
  704 STOP"IDIN NOT EQUAL TO N"
  705 STOP
  706 STOP"REMAINDER OF DATA OF INSUFFICIENT LENGTH"
      END


C*****************************************************************************
C         SUBROUTINE USED TO NORMALIZE DATA AT EACH TIME INCREMENT
```

```
C*************************************************************************
      SUBROUTINE NORM(DATA,RDATA,IX,IY,IZ)
      DIMENSION DATA(IZ,16),RDATA(IZ,16)
      DO 25 II=1,IX
      SUME=0
      DO 20 JJ=1,IY
      SUME=SUME+DATA(II,JJ)**2
 20   CONTINUE
      ENERGY=SQRT(SUME)
C-------------------------------------------------------------------------
C--              DETERMINE NON-INFORMATION AREAS IN SENTENCE            --
C-------------------------------------------------------------------------
 30   CONTINUE
      DO 25 JJ=1,IY
      RDATA(II,JJ)=DATA(II,JJ)/ENERGY
 25   CONTINUE
      RETURN
      END
```

A P P E N D I X

C

GLOSSARY OF TECHNICAL TERMS

1. Aliasing: The term "aliasing" refers to the fact that high-frequency components of a time function can impersonate low frequencies if the sampling rate is too low.

2. Allophone: The variant forms of a phoneme as conditioned by position or adjoining sounds.

3. Amplitude Normalization: The removal of speech amplitude as a parameter in speech sound similarity measurement. This ensures that a sound that varies in energy but not in spectral composition is still interpreted as the same sound (Ref 28:51).

4. Dipthong: A combination of two vowels in the same syllable, in which the speaker glides continuously from one vowel to another.

5. Dynamic Range Normalization: The determination of the energy variations of speech in order to adjust thresholds to allow energy to be used in segmentation (Ref 28:51).

6. Frame: A single time increment of the digital spectrogram.

7. Fricatives: Sounds produced by partial constriction along the vocal tract which results in turbulence. The sounds can be further subdivided into voiced and unvoiced categories. The voiceless fricatives are produced as a result of frictional modulation. The voiced fricatives combine frictional with vocal cord and cavity modulation.

8. Leakage: The term "leakage" refers to the discrepancy between the continuous and discrete Fourier transforms caused by the required time domain truncation.

9. Morpheme: Any of the minimum meaningful elements in a language, not further divisible into smaller meaningful elements, usually recurring in various contexts with relatively constant meaning, such as a word.

10. Nasals: Sounds that are produced by allowing the air to flow through the nasal cavities. Coupling the nasal cavities to the resonance system of the vocal tract

results in nasalized vowels. If the air flow is restricted to only flowing through the nasal cavities, nasal consonants are produced.

11. Noise Subtraction Normalization: The determination of the energy of ambient noise and the subtraction of that energy from the input signal so that only the speech signal is left (Ref 28:51).

12. Phone: An individual speech sound.

13. Phoneme: The smallest distinctive group or class of phones in a language. In a very general sense, the phonemes that make up a speech sound can be compared to the letters that make up a written word.

14. Pitch: The pitch of a sound with a periodic wave form - i.e., a voiced sound - is determined by its fundamental frequency, or rate of repetition of the cycles of air pressure.

15. Plosives: Sounds that are produced by a sudden release of built up air pressure. The sounds can be further distinguished by the presence of absence of voicing. A voiceless stop occurs when the stop is combined with fricative modulation. A voiced stop occurs when vocal cord modulation is combined with stop and fricative modulation.

16. Pragmatic Knowledge: A record of changes in the listener's world model occurring in the course of a conversation.

17. Prosodic Knowledge: Imputes meaning to the variation in pitch or stress in phrases.

18. Semantic Knowledge: General knowledge about the domain of discourse.

19. Speaker Spectra Normalization: The transformation of the power spectral density function in order to remove the effects of differing vocal tract lengths (Ref 28:51).

20. Syntactic Knowledge: A set of rules specifying legal sequencies of words or similar units.

21. Time Normalization: The stretching or shrinking of the length of time elapsed between given speech segments (Ref 28:51).

22. Velocity Normalization: Shortening of steady state speech segments to remove artificial variations in sound duration due to variations in speaking rate (Ref 28:51).

23. Vowels: Sounds whose source of excitation is the glottis. During vowel production, the vocal tract is relatively open and the air flows over the center of the tongue, causing a minimum of turbulence. The phonetic value of the vowel is determined by the resonances of the vocal tract, which are in turn determined by the shape and position of the tongue and lips.

Vita

William R. Hensley was born on 15 February 1942 in Marion, North Carolina. He graduated from high school in 1960, and enlisted in the USAF in July 1961. After serving 7 years he was commissioned through the Airman Education and Commissioning Program in September, 1968. He completed the Navigator Training and Electronic Warfare Training Schools at Mather AFB, California. While assigned to the 42nd Tactical Electronic Warfare Squadron, Korat Royal Thai AFB, Korat, Thailand, he flew 102 combat missions as an electronic warfare officer on the EB-66 aircraft. In September, 1972, he was assigned to the 453rd Electronic Warfare Training Squadron as an electronic warfare instructor. From May, 1974, until May, 1975, he was an instructor for a NATO Staff Officer Course. In May, 1975, he was assigned to the Air Force Institute of Technology in the graduate electrical engineering program.

Permanent Address: P.O. Box 789
Marion, North Carolina 28752

This thesis was typed by Annette Marchand.

Unclassified

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

(14) **REPORT DOCUMENTATION PAGE**

READ INSTRUCTIONS BEFORE COMPLETING FORM

1. REPORT NUMBER: GE/EE/76-24	2. GOVT ACCESSION NO.
3. RECIPIENT'S CATALOG NUMBER	
4. TITLE (and Subtitle): (6) COMPUTER IDENTIFICATION OF PHONEMES IN CONTINUOUS SPEECH.	5. TYPE OF REPORT & PERIOD COVERED: MS Thesis
	6. PERFORMING ORG. REPORT NUMBER
7. AUTHOR(s): (10) William R. Hensley	8. CONTRACT OR GRANT NUMBER(s): (9) Master's thesis,
9. PERFORMING ORGANIZATION NAME AND ADDRESS: Air Force Institute of Technology (AFIT-EN) Wright-Patterson AFB, Ohio 45433	10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS
11. CONTROLLING OFFICE NAME AND ADDRESS: Air Force Institute of Technology (AFIT-EN) Wright-Patterson AFB, Ohio 45433	(11) 12. REPORT DATE: Dec 1976
	13. NUMBER OF PAGES: 138 (12) 139p.
14. MONITORING AGENCY NAME & ADDRESS (if different from Controlling Office)	15. SECURITY CLASS. (of this report): Unclassified
	15a. DECLASSIFICATION/DOWNGRADING SCHEDULE

16. DISTRIBUTION STATEMENT (of this Report)
Approved for public release; distribution unlimited

17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)

18. SUPPLEMENTARY NOTES
Approved for public release; IAW AFR 190-17
Jerral F. Guess, Capt
Director of Information

19. KEY WORDS (Continue on reverse side if necessary and identify by block number)
Phoneme recognition by prototype matching
Phoneme recognition by crosscorrelation
Speech recognition
Continuous speech recognition

20. ABSTRACT (Continue on reverse side if necessary and identify by block number)
The purpose of this investigation was to identify phoneme segments as they appeared in continuous speech. The input device was an audio tape recorder from which the analog speech signal was digitized and fast Fourier transformed. The amplitudes of this transformed signal were combined in a logarithmic manner and printed out in a 16 channel digitized spectrogram. Sixty-one

DD FORM 1 JAN 73 1473 EDITION OF 1 NOV 65 IS OBSOLETE

012225

Unclassified

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

prototypes were selected to represent the phonemes of the English language. These prototypes were stored and used in a running crosscorrelation with the unknown speech signal. The amplitude values resulting from the correlation process were used to predict phoneme locations and the values were compared in order to identify the correct phoneme.

The phonemes were selected from Speaker A's speech signal and tests were conducted to analyze utterances from Speaker A and Speaker B. For Speaker A, location was rated at 81 percent while identification was rated at 45 percent. For Speaker B, location was found to be 70 percent with identification at 40 percent.

Spatial filtering techniques, uniform length prototypes, and various normalization procedures were investigated next with the result of improving location for Speaker B.

prototypes were selected to represent the phonemes of the English language. These prototypes were stored and used in a running crosscorrelation with the unknown speech signal. The amplitude values resulting from the correlation process were used to predict phoneme locations and the values were compared in order to identify the correct phoneme.

The phonemes were selected from Speaker A's speech signal and tests were conducted to analyze utterances from Speaker A and Speaker B. For Speaker A, location was rated at 81 percent while identification was rated at 45 percent. For Speaker B, location was found to be 70 percent with identification at 40 percent.

Spatial filtering techniques, uniform length prototypes, and various normalization procedures were investigated next with the result of improving location for Speaker B.