CR 86058

BOLT BERANEK AND NEWMAN INC

CONSULTING • DEVELOPMENT • RESEARCH

BBN Report No. 1667
Job No. 11254

A LIMITED SPEECH RECOGNITION SYSTEM

Contract No. NAS 12-138

Final Report

Daniel G. Bobrow
Dennis H. Klatt

15 May 1968

Submitted to:

National Aeronautics and Space Administration
Electronics Research Center
575 Technology Square
Cambridge, Massachusetts 02139

Attention: Mr. Wayne A. Lea

BBN Report No. 1667
Job No. 11254

A LIMITED SPEECH RECOGNITION SYSTEM

Daniel G. Bobrow
Dennis H. Klatt

Prepared under Contract NAS 12-138 by

Bolt Beranek and Newman Inc
Cambridge, Massachusetts

for Electronics Research Center
NATIONAL AERONAUTICS AND SPACE ADMINISTRATION

TABLE OF CONTENTS

ACKNOWLEDGEMENTS

The authors wish to acknowledge the invaluable assistance of a number of people: D. Lucille Darley did most of the programming of LISPER; Kenneth N. Stevens and Gottfried von Bismarck designed and built the filter bank spectrum analyzer; and Wayne A. Lea, the NASA contractor monitor, provided an extensive critique of an earlier version of this report.

A LIMITED SPEECH RECOGNITION SYSTEM

ABSTRACT

LISPER, a program designed for limited speech recognition, learns to identify utterances as one of a limited input message set. Functions of the raw spectral data are transformed into sequences of states (features) in order to normalize for word onset time, speaking rate, and speaking rhythm. State transition thresholds are normalized for speaking level, and a hysteresis region is employed about decision thresholds to reduce the sensitivity of the system to noise. A feature sequence need not be consistently correct because the recognition algorithm makes use of redundancy to identify the most probable message. Features related to tense-lax, sibilant, fricative, continuant, several place-of-articulation features and a stress feature are included in a set of "linguistic" features used for recognition. Another set of non-linguistic features was also tested. Recognition scores approach 97% correct after three rounds of training on a 54 word vocabulary for both sets of features.

I. INTRODUCTION

This report describes the results of a successful study for the National Aeronautics and Space Administration on the feasibility of a voice link with a computer. LISPER, a prototype Limited Speech Recognition system, was built to demonstrate and experiment with the set of techniques for speech recognition described below. The goals of the implementation were to provide a system which would:

1) be able to identify an utterance as one of a limited set of possible input messages. Different message sets used contain approximately 50 to 100 items and consist of words or phrases less than 2 seconds long spoken in isolation.

2) allow an individual user to "train" the system. LISPER learns separately the characteristics of an input message set as spoken by each individual to use the system.

3) allow maximum flexibility for data access and system modification. For this reason LISPER was built in an extended LISP system (McCarthy, 1964) (Bobrow, Murphy and Teitelman, 1968). This decision allowed an easy transfer of the system from a DEC PDP-1 to an SDS 940 computer.

This LISPER system has been used as both a research tool for exploring characteristics of speech, and a prototype of a trainable speech pattern recognition system.

Any pattern recognition system must have three basic components: preprocessing hardware to extract a representation of the input; programs utilizing this raw data to compute properties of the unknown input (data reduction); and a recognition or decision algorithm. Figure 1 shows a block diagram of the organization of the LISPER system. The input speech signal may be obtained from either a microphone or tape recorder. The basic parameters of the input are extracted by a spectrum analyzer which consists of a pre-emphasis network followed by nineteen bandpass filters. Bandpass filter outputs are rectified, low-pass filtered, sampled, converted into logarithic units, and stored on digital tape, or in the memory of the computer.

The spectral representation of a word as derived from our input system consists of 200 spectral samples, corresponding to 2 seconds of speech material. A spectral sample consists of the outputs of the 19 bandpass filters, all sampled at a particular instant of time. A filter output, in log units, is an integer ranging from 0 to 63, covering a 45 decibel range of intensity. Design considerations for the input system are discussed in Section II.

The use of the logarithm of the short-time spectrum is well established as one approach to speech analysis (see, for example, Flanagan, 1965), and has often been used as the basis for recognition programs. The principal contributions of this research have been the development of a set of algorithms for extracting features which characterize speech utterances, and coupling this set with a recognition algorithm capable of high quality word identification in the presence of redundant, inconsistent, or incorrect information from these properties. The recognition algorithm does not directly utilize the spectral data, nor is an

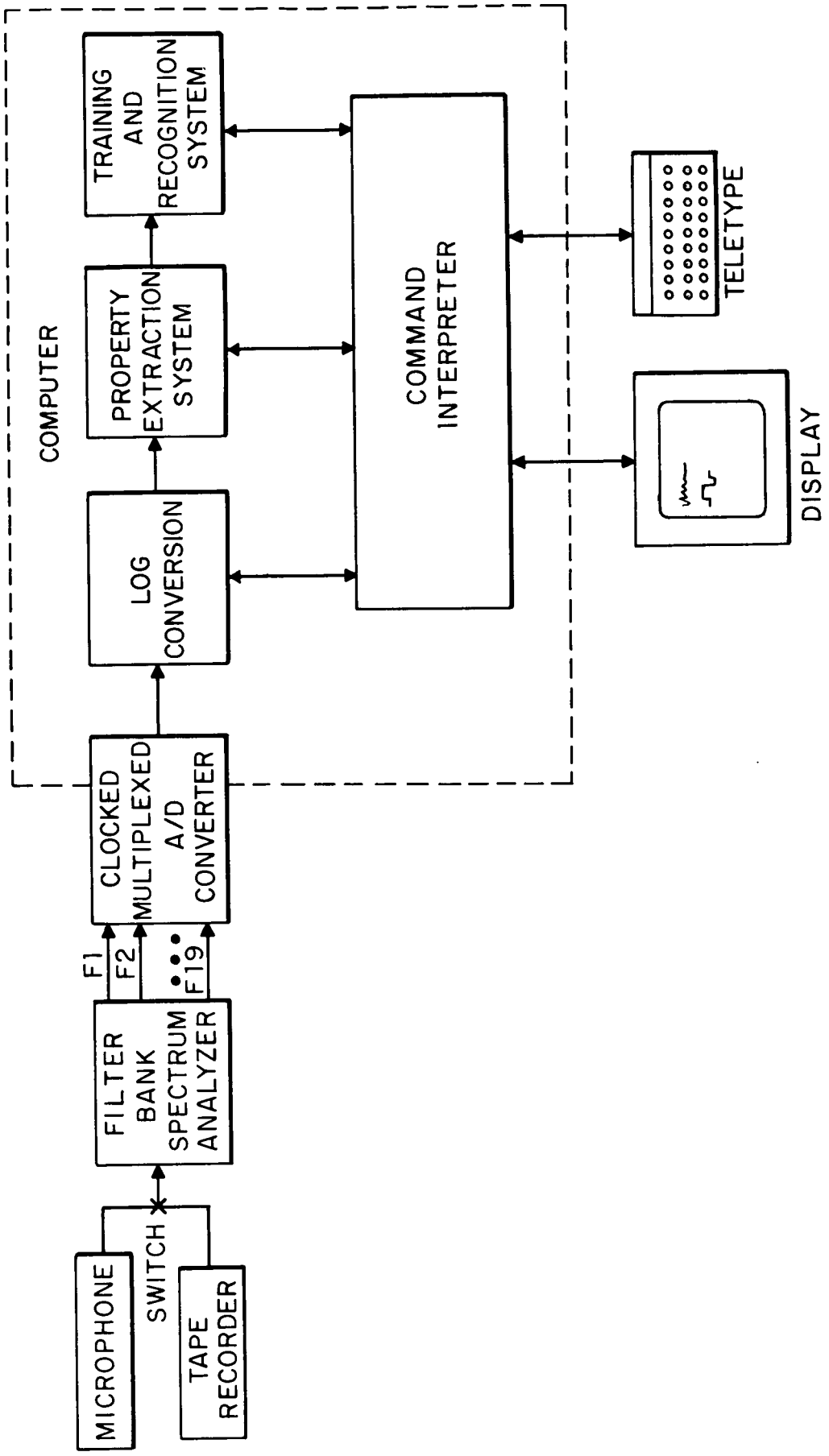


FIG. 1 BLOCK DIAGRAM OF THE LISPER SYSTEM

intermediate phonetic (or phonemic) transcription of the word produced. We discuss below some problems of these two approaches and indicate how a number of these problems have been circumvented by techniques used in LISPER.

Our system has been extensively tested, and recognition scores have been obtained on three vocabularies with three different speakers. Typical of our results is attainment of recognition scores that approach 97% correct on a 54 word vocabulary after three rounds of training for a single speaker. Experiments indicate that our recognition system is very successful in handling a moderate amount of additive noise. Our recognition scores compare favorably with those of other investigators (Gold, 1966; Denes and Mathews, 1964; Dudley and Balashek, 1958; Davis et al., 1952). However, unlike some of these, the LISPER system is not intended to be used simultaneously by multiple speakers, and a new speaker must train the system to the characteristics of his voice.

The techniques used in LISPER can best be appreciated by first exploring some problems involved in identifying a speech utterance on the basis of the 19 x 200 array representing a 2 second sampling of its energy spectrum. Pattern recognition schemes operating on data of this dimensionality are theoretically tenable (Sebestyen, 1960), but only if repetitions of words cluster properly in the resultant vector space. It is by now well-known that speech spectral patterns as initially stored in a computer do not cluster according to the word spoken. Some of the reasons for this are:

1. The range of intensities encountered will vary because the overall recording level is not fixed. Recording level depends on vocal effort and the distance of a speaker from the microphone.

2. An unknown word is difficult to register at the same place within 200 spectral samples because word onset time is not a simple feature to detect reliably. For example, initial voiceless fricatives may be missed, prevoiced stops are hard to treat consistently, etc.

3. The total duration of a word is highly variable. An increase in speaking rate is not manifested by a linear compression of the time dimension. Final syllables are often prolonged. Some transitions (for example, the release of a stop consonant) are not as greatly affected by changes in speaking rate as the steady-state portions of vowels. If shortened enough, vowels are likely to be reduced and consonants may not be carefully articulated, with resultant losses in spectral distinctiveness.

4. A speaker attempts to generate an utterance so that it has a particular set of perceptual attributes (or features). We do not know in detail what the acoustic correlates of these attributes are. There is a great deal of variation allowed in the acoustic properties of the signal that will still give rise to the same utterance. There is also sufficient redundancy in the message to permit a speaker to leave out certain attributes entirely. For example, the degree of stress placed on a syllable will determine the extent to which the vowel <u>may</u> be reduced (Lindblom, 1963). Consonants in unstressed syllables may contain less frication noise, a weak stop burst release, or incomplete stop closure. Vowels may be nasalized in nasal environments. The substitution of one incomplete gesture for a consonant cluster is also common in unstressed syllables of natural speech. None of these effects would

necessarily produce word recognition difficulties if they appeared consistently in the data. Unfortunately, they do not.

5. If a speaker is instructed to speak distinctly and not rapidly, some surprising and unfortunate variability in speaking habits has been experimentally detected. In an attempt to help the system, our speakers released final stops, increased the length of some syllables, and articulated unstressed syllables more carefully than they would normally. Unfortunately, our speakers appear to have found these speaking habits unnatural, and could not remember from repetition to repetition exactly what they had done to help. For example, final voiced stop releases gave trouble by producing short vowel segments that varied greatly in amplitude, and the words "four" and "core" were sometimes pronounced as if they had two syllables.

6. Individual speakers have vocal tracts of different sizes and shapes. It is physically impossible for two speakers to produce identical spectra for a given phone or word. A speaker makes an articulatory gesture that we, as listeners, interpret, possibly with respect to our knowledge of the spectra that he is capable of producing (Gerstman, 1967). The nature and importance of the normalization process of a listener are not well understood.

7. Different speakers have articulatory habits (idiolects) that may be quite distinct. Habits include the timing and dynamics of articulatory movements and the features that a particular speaker employs to manifest a phonemic distinction. Whether recognition difficulties can be

attributed to individual speech habit structures is not known. Very little quantitative data are available on the characteristics that distinguish speakers (Kersta, 1962).

The variability of the spectrum of a word has led to the search for data reduction techniques to eliminate irrelevant information. An approach favored by several investigators has been to develop rules for detecting phonemes (Fry and Denes, 1958; Martin et al., 1966; Reddy, 1967). We believe that phoneme recognition is a much more difficult problem than word recognition because it presupposes a good understanding of the cues that distinguish phonemes in arbitrary phonetic environments. For example, allophones of the phoneme /p/ may be:

1. normally aspirated, as in the word "peak" [p^hik]
2. weakly aspirated, as in the word "supper" [s^p$^{\circlearrowleft}$]
3. non-aspirated, as in the word "spin" [spɪn]
4. non-released, as in the word "top" [t^hap$^{\neg}$]

There is no need for a word recognition program to attempt to group this disparate set of physical signals together into one phoneme. However, to the extent that algorithms for deriving a detailed phonetic-feature description of an utterance can be found, they can be of considerable help to a practical word recognition system. Examples of feature approaches that are relevant include the work of Hughes (1961), Hemdal and Hughes (1964), and Gold (1966).

We have implemented a number of programs which characterize phonetic features of words. These features are defined and compared with the work of other investigators in Section III. Features that have been implemented include (1) voiced, (2) strident,

(3) fricative, (4) stop-burst, (5) nasal, (6) [r]-like, (7) strident-stop, (8) [i]-like, (9) [a]-like, (10) [u]-like. For our purposes, a feature is defined to be a function of sampled time having a range limited to a small number of distinguishable states. State transition thresholds are normalized for speaking level, and a hysterisis region about the threshold reduces the sensitivity to noise. The time dimension is removed by transforming the time function into the sequence of states that is produced by the unknown word. This reduces the information content in the representation of the word, automatically has the effect of normalizing for speaking rate and defines word onset as the first change of state. Several properties are used exclusively to describe the stressed syllable. We have been able to segment words into syllables, primarily on the basis of changes in total signal energy. The stressed syllable has been identified with the aid of a function that computes an approximation to the perceived loudness of a vowel. The features of the stressed syllable are less likely to be affected by the natural variability that characterizes the speech process (Stevens, 1968), and therefore are good properties for use in recognition.

For each new utterance, the recognition algorithm decides which previously seen message this one is most like. The algorithm is related to a maximum likelihood decision in which the state sequences for each feature independently suggest for recognition those input messages which have been previously characterized by that sequence during training. The details of this decision procedure are described in Section V. The recognition algorithm is capable of high quality word identification in the presence of incorrect or inconsistent characterizations from many of the features.

It should be emphasized that all of our feature definitions are relatively simple-minded. They do not produce results that correspond to a phonetic transcription of a linguist. Some features work reliably in some phonetic environments, and not at all in others. The features have the single purpose of providing sufficient information about the invariant characteristics of the input message for the recognition algorithm to operate successfully.

It should be clear that we believe a pattern recognition algorithm cannot succeed with data of the type provided by the input system unless the algorithm deals explicitly with some of the problems mentioned previously. In particular, amplitude normalization, word onset time, and speaking rate problems are likely to produce more variability than any naive pattern learning program can successfully treat. Pattern recognition programs that attempt to overcome these difficulties have been described in the literature (Schroeder, 1968; Shearme and Leach, 1968; Denes and Mathews, 1961; Dudley and Balashek, 1958; Davis et al., 1952).

The feature approach that we have taken provides a good working solution to these three problems. In the latter stages of the research, we formulated a set of non-linguistic features to be used as a basis for comparison with the features described in Section IV. The non-linguistic properties involved simple sums and differences of filter outputs. The results of the comparison show that, for a single speaker, pattern detecting properties that are unmotivated by linguistic considerations (but carefully chosen) work very well, approaching 97% correct in a 54 word vocabulary after 3 training rounds. These scores are roughly equivalent to our previous results. The reasons for this success are discussed in Section VI.

II. THE INPUT SYSTEM

A block diagram of the hardware that is employed to convert acoustic waveforms into a digital representation is shown in Fig. 2. Up to two seconds of speech is entered into the computer from a live microphone or tape recorder. A pre-emphasis filter having a frequency response that increases by 6 db/octave (3 db up at 740 Hz) is introduced so that the full dynamic range of each bandpass filter can be utilized. This network is similar to the inverse of the long-term average spectra for a male speaker.

The 19 channel filter bank of Fig. 2 approximates the short-time power spectrum of the speech signal. The filter characteristics have been selected to preserve the significant acoustic properties of an adult male speaker. In the frequency range up to 3000 Hz, 15 filters are arranged to be spaced uniformly and to have equal bandwidths. This is the approximate frequency range encompassed by the first three resonances of the vocal tract (Peterson and Barney, 1952). An even spacing of filters on a linear frequency scale was chosen to enable the hardware to share poles from adjacent filters.

During normal utterances of male speakers, the fundamental frequency rarely exceeds 180 Hz, and is usually in the range 80-150 Hz (Fairbanks, 1940). For voiced sounds, the spectrum analyzer should provide a representation of the spectral envelope, and this representation should be relatively uninfluenced by changes in fundamental frequency. These considerations led to the selection of 360 Hz as an appropriate filter bandwidth in the range up to 3000 Hz. With such a bandwidth there are always at least two harmonics of the fundamental within a given filter, and no more than four such harmonics. Thus changes in the pattern of

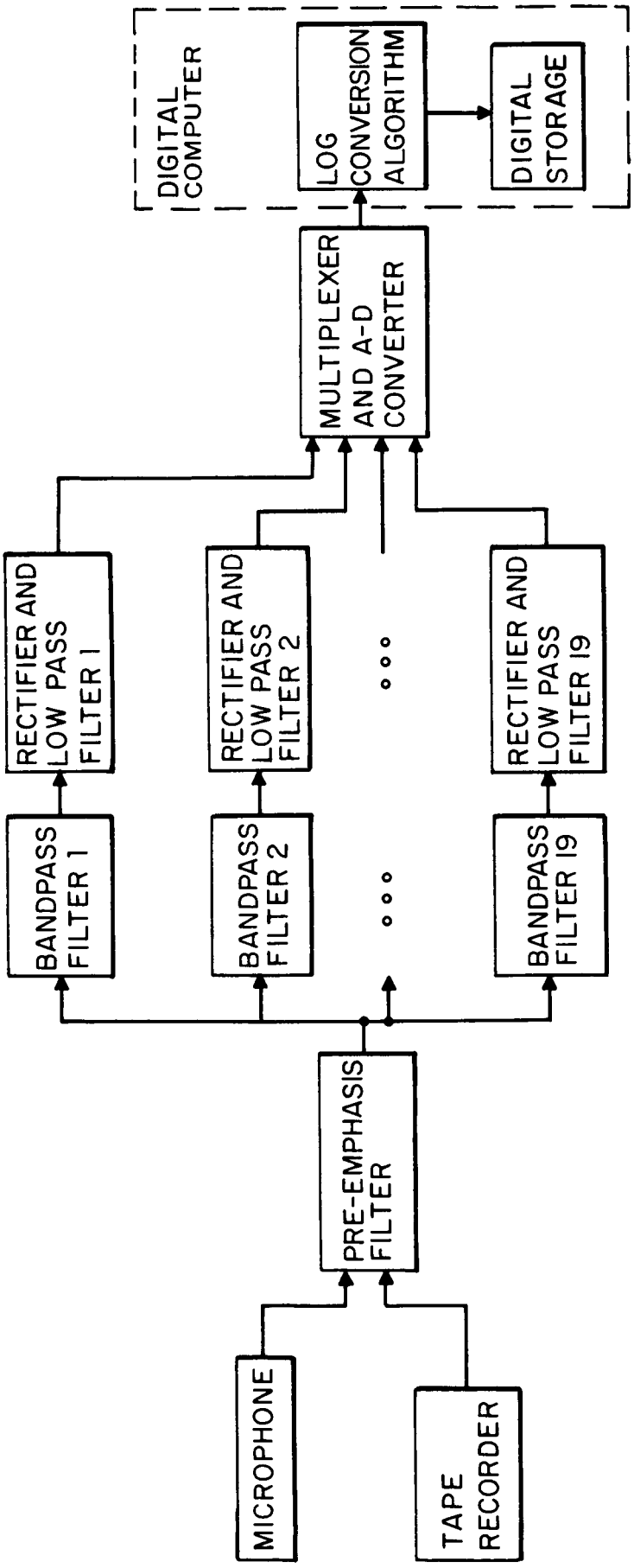


FIG. 2 A BLOCK DIAGRAM OF THE INPUT SYSTEM

filter outputs for a voiced sound due to changes in fundamental frequency are small.

In order to provide an adequate representation of the spectrum envelope, it was decided to space the filters every 180 Hz in the frequency range up to 3000 Hz. Once the desirability of filters with 360 Hz bandwidths has been established, it becomes clear that relatively few filters are required to cover the speech band. If filters were placed every 90 Hz instead of every 180 Hz, it would not be possible to locate the centroid of an energy concentration with significantly greater precision because outputs from adjacent filters have energy from several harmonics in common.

The frequency range above 3000 Hz is of importance primarily for carrying information about the noise-like components of speech--the frication noise for stop and fricative consonants. Fine frequency resolution is not as important (Heinz and Stevens, 1961). The frequency range from 3000 to 6500 Hz is covered by four wide bandwidth filters. Filter center frequencies and bandwidths are listed in Table 1.

Four-pole Lerner filters (Lerner, 1963) are used to realize the bandpass characteristics because (1) adjacent filters can share poles, (2) the phase response is linear over the passband of the filter, and (3) the slope of the phase response, i.e., the delay time, is the same for each of the filters so that spectral components of a signal appear simultaneously at the outputs of the filter channels. It has been shown by Stevens (1968) that the relative timing of energy onsets in different frequency regions may be an important cue for the differentiation of place-of-articulation for stops. The set of Lerner filters will preserve this type of information. Fig. 3 plots the measured frequency response of each bandpass filter.

Filter No.	Center Freq. cps	Lower Cut-Off cps	Higher Cut-Off cps	Band Width cps
1	260	80	440	360
2	440	260	620	360
3	620	440	800	360
4	800	620	980	360
5	980	800	1160	360
6	1160	980	1340	360
7	1340	1160	1520	360
8	1520	1340	1700	360
9	1700	1520	1880	360
10	1880	1700	2060	360
11	2060	1880	2240	360
12	2240	2060	2420	360
13	2420	2240	2600	360
14	2600	2420	2780	360
15	2780	2600	2960	360
16	3260	2960	3560	600
17	3980	3560	4400	840
18	4940	4400	5480	1080
19	6020	5480	6560	1080

TABLE 1. List of Filter Center Frequencies and Bandwidths

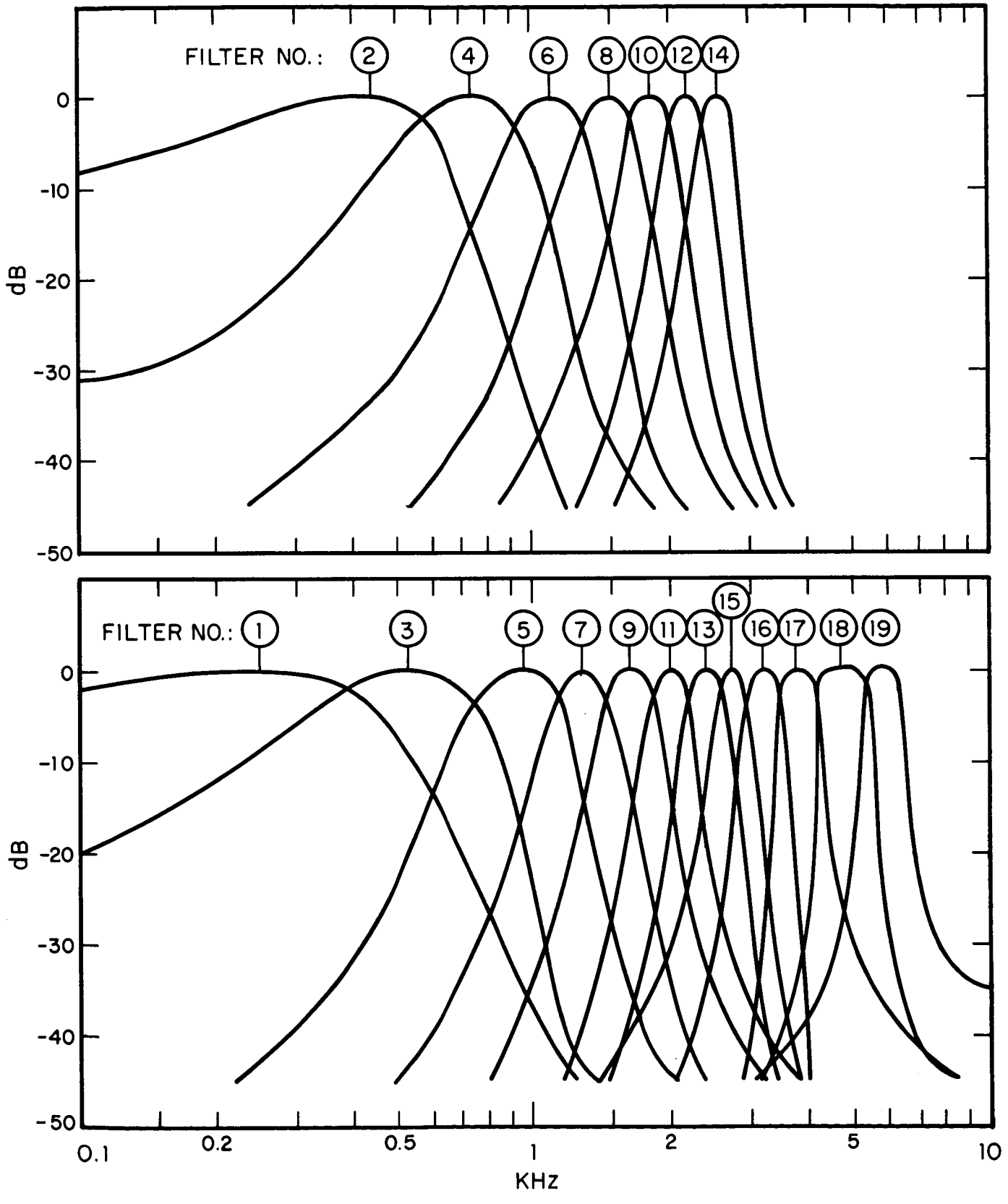


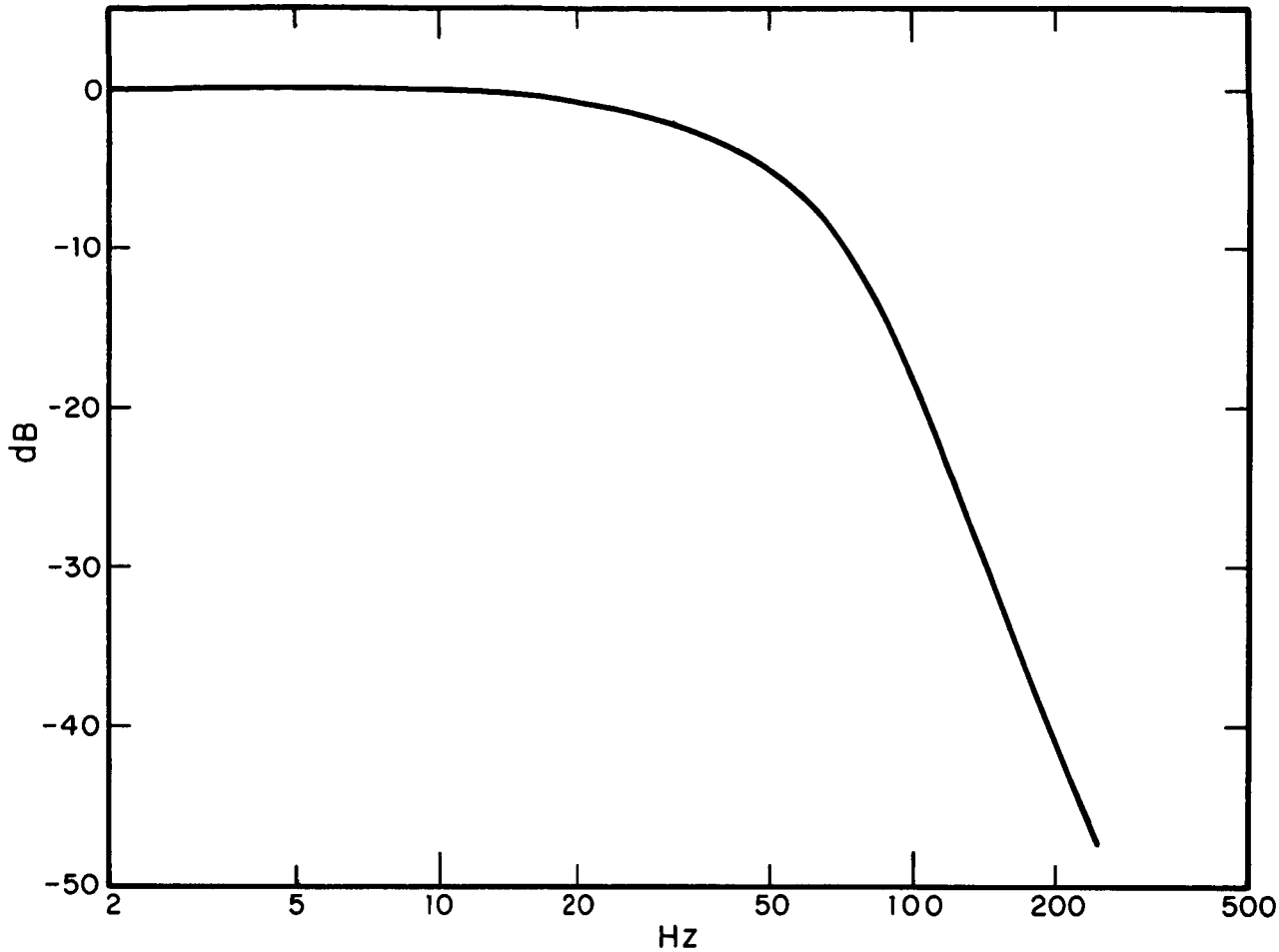
FIG. 3 MEASURED FREQUENCY RESPONSE OF FILTER BANK. RATIO OF DC OUTPUT TO AC INPUT IN dB

The output of a bandpass filter is first full-wave rectified and then low-pass filtered. The averaging time for the low-pass filters should be sufficiently long that there are no appreciable fluctuations in the outputs of the channels within a period of the fundamental frequency because fluctuations are not easy to distinguish from significant spectral changes. On the other hand, the averaging time should be sufficiently short so that rapid changes in level and spectral shape associated with stop and nasal consonants can be detected. These considerations led to the selection of a compromise averaging time on the order of 10-20 msec.

The weighting function (i.e., the impulse response) associated with the low-pass filter should have as short a tail as possible so that only events over the 10-20 msec interval are averaged. The Bessel filter provides a non-ringing impulse response with a tail that is relatively short. A four-pole Bessel low-pass filter with the characteristics indicated in Fig. 4 terminates each of the 19 filter channels.

The output of a filter is sampled every 10 msec, using the multiplexer and analog-to-digital converter of the computer. Thus, each 10 milliseconds the computer extracts 19 numbers (integers ranging from 0 to 1023) which represent the average energy in 19 frequency bands for the previous 10 milliseconds. In the computer, the logarithm of this energy is computed. Each sample is transformed to an integer between 0 and 63, covering a 45-decibel range of intensity.

An example of the spectrum of the word "store", as derived by the input system, is shown in Fig. 5. Note that the resolution of individual formant peaks is poor with these filters (e.g., the first and second formants of [o^{w}] in "store" do not produce



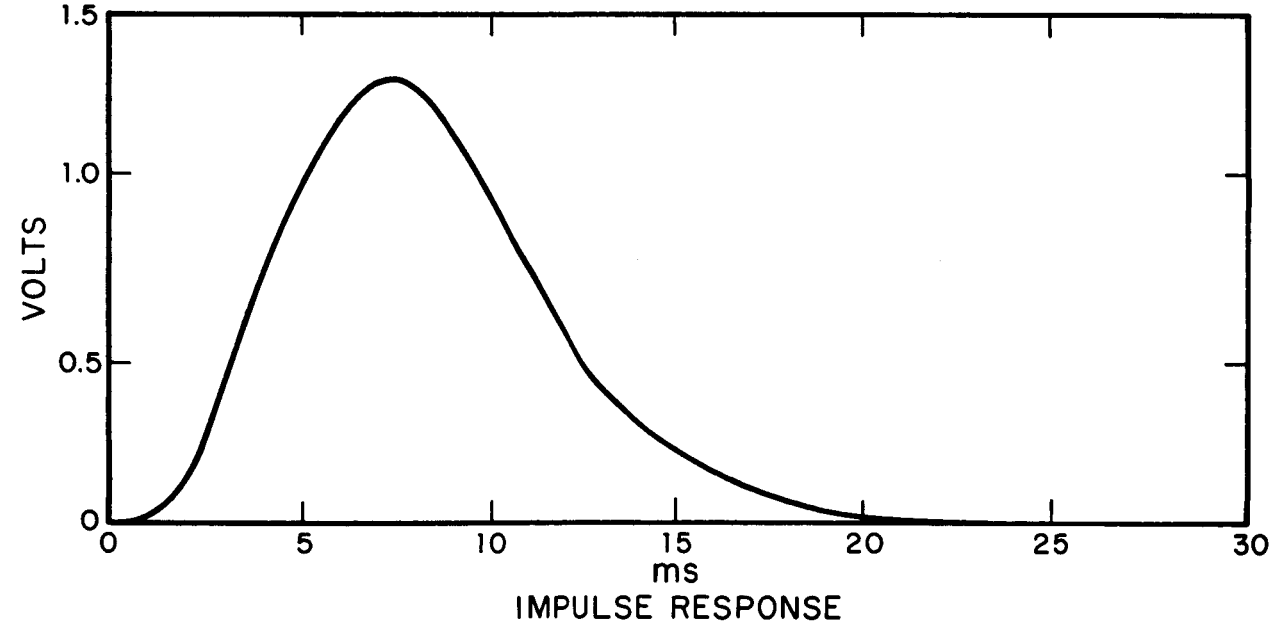


FIG. 4 CHARACTERISTICS OF THE FOUR-POLE BESSEL LOW-PASS FILTER

word 12 speaker 2 repetition 1

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	total
1	12	3	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	16
2	11	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	11
3	7	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	8
4	10	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	10
5	17	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	17
6	12	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	2	16
7	9	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	4	2	8	24
8	16	2	0	0	0	0	0	0	1	0	0	0	0	0	0	2	10	7	15	53
9	15	0	0	0	0	0	0	1	0	0	0	0	1	0	0	6	15	12	19	69
10	12	0	0	0	0	0	0	1	1	0	0	0	2	0	0	6	17	20	24	83
11	18	3	2	0	1	1	1	2	2	0	2	1	7	0	2	11	28	25	36	142
12	16	0	1	1	1	1	0	2	1	1	2	0	2	0	3	17	28	27	32	135
13	17	5	2	0	1	0	0	1	1	0	1	1	4	0	3	26	29	29	39	159
14	13	2	2	1	2	0	1	2	2	1	3	2	7	2	6	30	32	36	43	187
15	12	2	2	1	2	0	1	2	3	1	4	5	7	3	7	33	32	34	42	193
16	22	3	2	1	2	0	1	2	2	1	4	4	7	3	6	29	29	32	39	189
17	15	1	1	0	1	0	1	1	2	1	2	1	4	0	5	26	25	27	35	148
18	10	3	0	0	0	0	0	0	0	0	1	0	2	0	2	21	20	23	29	111
19	21	0	0	0	0	0	0	0	0	0	0	0	0	0	1	17	14	17	24	94
20	14	1	0	0	0	0	0	0	0	0	0	0	0	0	0	12	9	13	19	68
21	20	2	0	0	0	0	0	0	0	0	0	0	0	0	0	8	6	9	14	59
22	21	2	0	0	0	0	0	0	0	0	0	0	0	0	0	6	4	5	9	47
23	16	3	0	0	0	0	0	0	0	0	0	0	0	0	0	2	2	2	5	30
24	14	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1	2	18
25	18	4	0	0	0	0	0	0	0	0	0	2	3	0	3	3	1	1	2	37
26	22	14	15	12	10	7	6	10	6	3	8	18	22	20	18	19	7	6	19	242
27	19	11	9	7	6	4	3	8	5	3	8	14	17	15	14	19	8	9	17	196
28	32	26	18	8	5	5	4	6	2	1	4	6	9	5	8	15	5	5	11	175
29	48	46	39	28	19	18	9	6	3	5	6	9	8	5	6	10	3	1	7	276
30	47	48	43	33	29	27	18	10	2	2	2	3	4	0	3	8	2	1	4	286
31	48	50	46	36	35	32	20	11	6	5	4	2	3	0	2	6	1	0	3	310
32	48	52	48	42	40	36	22	12	6	7	5	3	3	0	1	5	1	0	2	333
33	48	53	50	44	40	34	20	11	7	7	5	1	2	0	0	3	1	0	1	327
34	48	53	51	46	40	33	19	12	7	9	5	0	1	0	0	2	0	0	0	326
35	48	54	52	47	42	33	18	14	9	11	5	1	1	0	0	1	0	0	0	336
36	49	55	53	49	44	33	17	15	12	14	7	2	1	0	0	0	0	0	0	351
37	48	55	54	50	45	33	17	17	13	15	9	2	2	0	0	1	0	0	1	362
38	47	55	54	51	45	33	17	17	14	17	10	5	2	0	0	0	0	0	0	367
39	49	56	55	51	46	34	18	19	16	19	12	5	2	0	0	0	0	0	0	382
40	50	57	56	52	47	35	19	21	18	20	14	5	2	0	0	0	0	0	0	396

(Continued)

41	51	58	58	53	47	35	21	23	20	22	17	6	3	0	0	0	0	0	2	416
42	50	58	58	52	45	35	21	22	18	21	15	5	3	0	0	0	0	0	2	405
43	49	57	56	52	46	36	21	21	18	21	16	5	3	0	0	0	0	0	2	403
44	49	56	56	52	46	36	20	21	18	21	17	7	4	0	0	0	0	0	3	406
45	48	56	56	52	47	38	22	23	19	22	19	8	4	0	0	0	0	0	1	415
46	49	56	56	52	47	39	22	23	20	22	18	8	6	0	0	1	0	0	1	420
47	47	56	56	52	49	42	24	25	22	23	20	13	8	0	1	1	0	0	1	440
48	47	54	54	52	49	43	28	26	23	24	19	13	8	1	1	2	1	0	3	448
49	46	53	54	52	50	46	30	27	24	24	19	14	9	3	2	1	1	0	1	456
50	45	52	52	51	50	46	33	28	24	23	16	11	7	2	2	0	0	0	0	442
51	44	51	51	50	49	45	33	28	24	22	17	14	9	4	3	1	0	0	0	445
52	42	50	50	49	48	45	34	30	25	22	16	14	9	4	4	2	0	0	1	445
53	40	49	50	48	47	45	35	31	26	21	16	14	9	4	5	1	0	0	1	442
54	39	49	50	48	47	45	36	33	27	21	14	12	8	3	3	1	0	0	0	436
55	36	48	49	47	46	45	38	35	29	20	12	11	7	2	3	1	0	0	0	429
56	40	49	49	46	45	45	39	36	28	19	10	9	6	0	2	1	0	0	0	424
57	42	50	50	46	44	44	40	36	28	18	10	9	7	1	2	2	1	0	0	430
58	42	50	50	46	43	43	40	36	28	19	9	9	7	1	3	3	1	0	0	430
59	42	48	48	44	40	42	38	36	28	17	7	8	4	0	1	1	0	0	0	404
60	40	46	46	41	36	38	36	34	24	13	4	2	2	0	0	0	0	0	0	362
61	40	45	45	40	35	36	35	34	24	14	3	1	1	0	0	0	0	0	0	353
62	40	43	43	36	32	33	33	32	23	12	2	0	1	0	0	0	0	0	0	330
63	39	43	42	36	29	31	32	32	23	13	2	0	0	0	0	0	0	0	0	322
64	38	42	42	35	27	29	30	30	22	10	2	0	0	0	0	0	0	0	0	307
65	39	41	39	33	24	26	27	28	19	10	1	0	0	0	0	0	0	0	0	287
66	39	36	33	28	20	21	24	25	17	9	1	0	0	0	0	0	0	0	0	253
67	32	32	29	24	17	17	20	23	14	8	1	0	0	0	0	1	0	0	1	219
68	30	29	25	20	13	12	17	19	12	7	2	0	0	0	0	1	0	0	2	189
69	32	24	21	17	10	8	13	17	10	5	1	0	0	0	0	1	0	0	0	159
70	31	22	18	14	9	9	12	17	9	4	0	0	0	0	0	0	0	0	0	145
71	30	24	14	10	7	5	8	12	6	2	0	0	0	0	0	0	0	0	0	118
72	25	15	8	6	5	3	5	7	3	0	0	0	0	0	0	0	0	0	0	77
73	16	13	5	5	5	3	4	6	2	0	0	0	0	0	0	0	0	0	0	59
74	18	10	5	3	3	2	3	6	1	0	0	0	0	0	0	0	0	0	0	51
75	14	6	3	3	3	2	2	3	1	0	0	0	0	0	0	0	0	0	0	37
76	13	6	2	2	2	1	1	2	0	0	0	0	0	0	0	0	0	0	0	29
77	12	2	1	1	2	3	1	1	0	0	0	0	0	0	0	0	0	0	0	23
78	13	4	2	2	2	1	1	1	0	0	0	0	0	0	0	0	0	0	1	27
79	17	3	2	1	2	1	1	1	1	0	0	0	0	0	0	1	0	0	1	31
80	17	3	2	1	2	1	0	1	0	0	0	0	0	0	0	0	0	0	0	27

FIG.5. THE SPECTRUM OF THE WORD "STORE" AS DERIVED BY THE INPUT SYSTEM.

individual peaks). The choice of filters represents a commitment to a non-formant-tracking approach to speech processing. It is based on the hypothesis that the human perceptual apparatus does not attempt to resolve individual formant tracks, but rather is sensitive to gross changes in energy concentrations. Thus, when two formant frequencies come close, as in a back vowel, only one broad energy concentration is seen by our input system and, presumably, by the peripheral auditory system as well. An approximation to the frequency locations of formants could be recovered from the spectral data through an analysis-by-synthesis procedure (Bell, et al., 1963) if desired.

In working with the input system, it has become evident that two qualifications should be made concerning its design. A finer frequency resolution would be of some help in the frequency region from about 300 Hz to 1200 Hz. Additional filters (or a modification in the spacing of the maximum of 19 filters to which we are limited by the multiplexer hardware) would hopefully improve the ability of the system to detect sudden movements in energy concentrations more reliably. The second design improvement that we would include in a second-generation input system would be a shortening of the averaging time of the low-pass filters by about 30-40 percent. At present, sudden changes in energy concentrations are indicated with insufficient temporal resolution. This latter improvement might be accompanied by an increase in the digital sampling rate from 100 spectra per second to about 200 spectra per second.

A detailed description of the electronic circuits of our input system is contained in Appendix A. This system was designed and built by K. N. Stevens and G. von Bismarck (1967).

III. PROPERTY EXTRACTION SYSTEM

An unknown word is represented in the computer by a matrix of integers (filter number vs. sampled time vs. log-amplitude). The purpose of a property or feature extraction system is to reduce the information content of the input signal to a level that makes it possible for the decision or recognition algorithm to operate reasonably. The critical thing is that the information eliminated should be information which is irrelevant to the decision that the recognition algorithm must make.

The principal result of this research has been the development of a set of feature definitions which characterize speech spectra in a way that makes recognition possible. The feature definitions have evolved through the course of the research and represent a gradual learning by the researchers of the relevant characteristics of speech as seen by the input system. The approach has been to make use of the present day body of knowledge about acoustic phonetics and the distinctive feature approach to phonological description (Jakobson, et al., 1963; Chomsky and Halle, 1968). A similar orientation is exemplified by the work of Hughes and Hemdal (1965).

The properties of speech which are used by the recognition program are based on the energy measures derived from the input system. The output of each filter is an elementary function of the speech input signal. We use the notation $F_n(i)$ for the output of filter n at sample interval i; that is F1(i), F2(i), ... , F19(i) are used for the output of filters 1 through 19 at sample interval i. The filter number n ranges from 1 for the low frequency filter to 19 for the high frequency filter; and i=1 for the first sample interval to i=200 for the last time sample of a two second utterance.

More complicated functions of the speech input signal can then be defined in terms of these elementary (base) functions, F1, ... F19, in the LISP system. For example, the following function has been found to correlate with perceived loudness, independent of vowel quality.

$$Loud(i) = F1(i) + F2(i) + F3(i) + F4(i) + F12(i) - F7(i)$$

The output of the seventh filter is subtracted from the sum of the first four filters and filter 12 to compensate for the fact that low vowels are inherently more intense when produced with the same vocal effort. We will indicate later how a modified version of this function Loud(i) can be used to correct for differences in recording level between repetitions of the same word.

Loud(i) is useful in reducing the information that must be processed by the decision algorithm, since it helps to normalize the input with respect to a variable which is not important to recognition, namely, the recording level of the input signal. We describe in this section a number of techniques which are used to reduce the information content of the incoming signal in ways that preserve the invariance of message identity over ranges of parameters which are irrelevant to the decision about the identity of the message.

One important way we have found to reduce the information content of a function is to reduce its range of values. Thus, for example, we may define a reduced information property Amp2(i) based only on the output of filter F2.

$$\text{Amp2}(i) = \begin{cases} +1 & \text{if } F(2) > 40 \\ 0 & \text{if } F(2) > 20 \\ -1 & \text{otherwise} \end{cases}$$

Amp2(i) is thus a 3 valued function, where we ascribe no significance to the names of the three values. As described later, a set of non-linguistic threshold properties similar to Amp2 can be used as a basis for recognition. However, we note the following problem. If the signal in F2 were varying around either of the thresholds, 20 or 40, then there would be little significance to the changes of state between -1 to 0, or 0 to 1; they probably would be caused more by noise than a real (significant) variation in the input signal.

To make such properties less sensitive to noise, we introduce a hysteresis region around the thresholds. This helps to insure that a change of state is significant. A revised definition of Amp2(i) is

$$\text{Amp2}(i) = \begin{cases} +1 & \text{if } [F2(i) > 40] \text{ or } [F2(i) > 37 \text{ and } F2(i-1) = 1] \\ 0 & \text{if } [F2(i) > 20] \text{ or } [F2(i) > 17 \text{ and } F2(i-1) \neq -1] \\ -1 & \text{otherwise} \end{cases}$$

Most of the features we use are functions of sampled time having a range limited to a small number of distinguishable states. These functions are very sensitive to slight changes in time scale and origin; however, these variations in the data are irrelevant for recognition. Therefore, time dimension is removed from a feature by transforming the time function into a sequence of transitions of states, as is illustrated in the following example:

```
i          1 2 3 4 5 6 7 8 9 ...
Voice(i)   0 0 1 1 1 1 1 1 0 ...
Voice =    0 1 0 ...
```

This transformation reduces the amount of data that must be manipulated by the program. Due to the nature of spoken language, the exact time when features change value will vary from repetition to repetition of the same word, but the essence of the word remains in the sequence of state transitions of an appropriate set of features.

No information about the word onset time, speaking rate, and speaking rhythm can be recovered from the sequence unless these parameters have an effect on the actual states that are reached. To this extent, recognition will be unperturbed by variations in word onset time, speaking rate, and speaking rhythm.

A problem that arises from collapsing the time dimension is an inability to tell whether two features were in specific states at the same time. Time removal assumes that features independently characterize a word. This is obviously false. A clear example is provided by the words "sue, zoo" which can only be distinguished by knowing whether the strident is simultaneously voiced, or not. It is possible to retain some timing information by the inclusion of features containing states which are entered only upon simultaneous satisfaction of two conditions, or features that count the number of time samples between temporal landmarks in the data and change state if the count exceeds a threshold.

For example, the state "-1" corresponds to voiceless segments in the definition of the spectral quality [r]-like(i). This is used to indicate in which voiced segment an [r]-like phone occurred.

The sequence produced for the word "ratio" should be -1, 1, 0, -1, 0, -1. A more detailed localization of the [r] is probably not necessary and would certainly be more difficult. We found that very gross timing information is satisfactory for recognition. Two features are used to make a preliminary division of a word or message into one or several segments. One feature divides a word into voiced and voiceless segments and the other divides it into syllables. Voice(i) and Syllable(i) are then used to introduce time markers in the definitions of other features.

Time removal achieves the most significant reduction in the information content of the raw data. Information content is also reduced when the intensity and filter number (frequency) dimensions are transformed into changes of state of selected functions. The information retained by these transformations can be increased by (1) an increase in the number of distinguishable states, or (2) an increase in the number of feature detectors.

Information manipulable by the recognition algorithm can also be obtained by mapping the value of a function at a particular time interval into a small range of values. For example, features which classify the vowel quality of the stressed syllable of an utterance into one of 10 categories have been found to be very useful in recognition. The stressed syllable is identified with the aid of a function Loud(i) that computes an approximation to the perceived loudness of a vowel. These features of a stressed syllable are less likely to be affected by the natural variability that characterizes the speech process (Stevens, 1968).

Three types of information are available for evaluating and redefining features. The primary data source is a set of computer print-outs of the spectra produced by the 54 word vocabulary of Gold (1967) repeated once by 2 speakers. The word list appears

in Table 2; an example of the computer printouts is shown in Fig. 5 earlier.

A feature definition that appears to work for this data is tested on 9 other recordings of the same word list by the two speakers. One result of this test is a set of recognition scores: X percent correct identification for each repetition of the word list. This is usually not a sufficiently sensitive measure of feature performance, so we also obtain a printout of the feature characterization of each word in each repetition of the vocabulary for all feature definitions. Examples of feature characterization for the Gold vocabulary are contained in Appendix B.

From this information we might find that, for example, the feature Voice(i) produces a sequence 01010 for some repetitions of the word "divide". Each "1" represents a voiced segment and each "0" represents an unvoiced segment. Reference to the original spectral printout for "divide" suggests that the problem is a weakly voiced [v]; sometimes the voicing in [v] falls below our threshold setting, sometimes not.

At this point we would lower the voicing threshold and observe the results. We might find that a vocabulary item with two voiced segments (e.g. "compare") now sometimes produces the incorrect sequence 010, indicating only one voiced segment. A simple solution, threshold adjustment, is incapable of making the voicing definition conform to our linguistic intuition. More importantly, the simple adjustment does not produce consistent results.

However, our recognition algorithm is powerful enough to deal with ambiguity in the voicing assignment of some vocabulary items, and thus we did not develop a more sophisticated voicing algorithm.

We made no attempt to use rate of voicing onset, offset, time between burst and energy build-up for stops, etc., as additional cues to a better feature definition. In all the features used, the ambiguity remains at a level tolerable to the recognition algorithm. The redundancy of the feature definitions allows recognition of variations of the same message.

IV. RECOGNITION ALGORITHM

Our recognition algorithm is a program that learns to identify a word by associating the outputs of various property extractors with that word. During learning, the vocabulary of words is presented a number of times, and information is accumulated about the different ways that the speaker may pronounce each word. For example, the result of the training procedure applied to the feature sequence Voice after 5 presentations of the 4-word vocabulary "one, two, subtract, multiply" might be:

Word	Sequence	Number of times sequence occurred
one	010	5
two	010	5
subtract	01010	5
multiply	0101010	3
multiply	01010	2

The two versions of the word "multiply" exemplify a common problem. No matter where we place threshold boundaries, there are some words that are treated inconsistently by the feature detectors. Another way of putting it is to say that there appear to be no absolute boundaries along the dimensions we have chosen. The recognition algorithm takes this fundamental limitation into account and makes a best guess given the imperfect nature of the properties.

The program first reorganizes the training data for each property into a list of these sequences (or other outputs) that have occurred. The list for Voice of our example is shown below:

Sequence	(Word, frequency)
010	one 5, two 5
01010	subtract 5, multiply 2
0101010	multiply 3

If a new utterance is presented to the program for recognition and the feature Voice(i) produces the sequence 01010, then Voice will register one vote for the word "subtract" and one vote for the word "multiply". The unknown word is identified as the vocabulary word eliciting votes from the most features. Typically, the identified word received a vote, signifying a perfect match, from only about 80 percent of the features.

In case of ties, the program makes use of information concerning the number of times a word appears at a node. Thus, if "subtract" and "multiply" tie for first place in the voting from all the features and Voice=01010, then Voice will register 5 votes for "subtract" and 2 votes for "multiply" in the run-off between these two candidates.

The final decision procedure is an attempt to find the message from the set of possible inputs which is most similar to the current input message. Because there is a wide variation in the way people say things, the decision procedure does not insist that the current input must be like one of the prototype input strings in all ways that it was categorized; that is, it need not be suggested by all properties. In this sense, the decision procedure allows a generalization of the original training learning by looking for a best fit without putting a bound on the goodness of this fit.

The property sequences have been considered as independent characterizations of the input message. In this case, by independent we do not mean that the computations themselves are necessarily independent, but that these descriptors of the input message are treated independently in the decision process. With the assumption of independence, the voting procedure is similar to a maximum likelihood estimate.

The a priori probabilities of all messages are the same. Therefore, the most likely message is the one for which the product of the a posteriori probabilities (or the sum of their logarithms) is a maximum. For each property, every message, M_i, previously associated with the current input sequences S_i is given a (scaled) log probability of $100+N_i$, where $N_i \geq 1$ is the frequency with which M_i was seen for S_i. Messages not associated with S_i in training are given log-probability $\emptyset$ and can therefore be ignored in the summation. These assignments of probabilities achieve in one step a score which allows a single search to determine the most likely message.

It has been experimentally determined that this voting scheme works as well or better than a number of other measures that make use of the same information. For example, we tried the adjusted weight used by Teitelman (1964) in ARGUS, a hand printed character recognizer (a similarly structured pattern recognizer); and a weight of $\log(1+N_i)$ which is a good small sample probability estimate of the a posteriori probability of S_i.

V. FUNCTIONS AND PROPERTIES USED IN RECOGNITION

The principal results of this research have been the development of properties which characterize speech input signals in a way to make recognition possible. These properties have evolved over the course of the research, and represent a gradual learning by the researchers of characteristics of speech as seen through the "ear" of our input spectrum analyzer filter bank. Two different sets of recognition properties are included here. The first is a set of properties and functions which tend to describe the speech signal in more linguistic terms. Transitions of these properties describe features of the input message which can be understood in terms of the ordinary linguistic descriptions of such messages. In this way, they tend to be more speaker independent than the second set of properties which are described.

The second set of properties used for recognition described below extract features of the spectral shape of a message. Their extreme simplicity makes them seem ideal for hardware implementation, and their high accuracy in recognition testifies to the uniform way that people tend to say a particular message. At least there seems to be a fair uniformity as seen through our spectrum analyzer.

A. Linguistic Features

We distinguish functions, such as Loud(i), which have a domain equal to the input spectral representation of an unknown word; features, such as Voice(i), which are functions that are used in recognition and have a range limited to a small number of states; and feature sequences, such as Voice, which are the result of removing the time dimension from a feature. The following functions are used in the definitions of a set of 15 features. All functions

have a single variable, i, which represents an index over time. The notation F6(i) will be used to denote the output of filter 6 at time sample i.

Functions

1. Loud(i) =F1(1) + F2(i) + F3(i) + MAX [F2(i), F4(i)]
   + MAX [2(F3(i) - F4(i)), 0].

   This function is intended to provide a measure of the perceptual loudness of vowels. Perceptual loudness is related to vocal effort whereas the acoustic energy in a vowel depends in part on the vocal tract configuration as well as vocal effort. For example, the low vowel, [a], for which the vocal tract is relatively open will have a greater natural intensity than the high vowel, [i], for which the vocal tract is more closed (Lehiste and Peterson, 1959; Fant, 1960). The function is intended to compensate for the reduced level associated with a low first formant.

2. ah(i) = F3(i) + F4(i) - F1(i) - F2(i).

   This function and the following two functions are used to characterize the spectral quality of the vowel nuclei of a word. The vowels [i, a, u] represent limiting articulatory positions for the tongue body. The vowel [a], as in "pot", is produced by placing the tongue as far back and as low as possible without producing a constricted vocal tract (and therefore a consonant). An acoustic correlate of this articulation is a high first formant, resulting in a greater output in filters 3 and 4 than in filters 1 and 2. The function ah(i) will therefore be a maximum when vowels with a high first formant similar to [a] are produced.

3. ee(i) = F9(i) + F10(i) + F11(i) + F12(i) - 2F6(i) - 2F7(i).

The vowel [i] as in "beet" is produced by positioning the tongue body as high and as forward in the mouth as possible. An acoustic correlate of this articulation is a low first formant and a high second and third formant, resulting in an energy minimum in filters 6 and 7, and an energy concentration between filters 9 and 12. The function ee(i) will therefore be a maximum for vowels similar to [i].

4. oo(i) = F3(i) + F4(i) + F5(i) - F8(i) - F10(i) - F12(i) -F14(i) - F16(i).

The vowel [u] as in "boot" is produced by positioning the tongue body as high and as far back in the mouth as possible. An acoustic correlate of this articulation is a low first and second formant, resulting in a major energy concentration below F7 and reduced energy above F7. The function oo(i) will therefore be a maximum for vowels similar to [u].

5. er(i) = F7(i) + F8(i) - F4(i) - F12(i).

This function is similar to the vowel functions in form, but has the task of detecting spectra characteristic of the consonantal and syllabic allophones of [r]. The low third formant characteristic of [r] and [ɚ] (as the vowel in Bert) produces a distinct spectral shape with an energy concentration centered in filters 7 and 8, a dip in energy at filter 4 between the first and second formants, and an absence of energy above F10 due to the low third formant. The function er(i) will therefore be a maximum when phones similar to [r, ɚ] are produced.

6. str(i) = F16(i) + F17(i) + F18(i) + F19(i).

This function is a maximum when high frequency frication energy is present. The strident phones produce intense high frequency energy, whereas a non-strident fricative produces a small energy peak in filter 19. The function str(i) will therefore be a maximum for many allophones of [s, z, š, ž, č, ǰ, t, k], which are the first consonants in [sin, zen, shin, azure, chin, gin, tin, kin] respectively.

7. dspect(i) = |ah(i) - ah(i-1)| + |ee(i) - ee(i)|
+ |oo(i) -oo(i-1)|

This function is a computationally inexpensive approximation to a spectral derivative. The function will be a maximum when the spectrum is changing rapidly, as for example in a consonantal transition. Dspect(i) is used as an aid in delimiting syllable boundaries. It is also used to distinguish ee-like vowels from some consonant-vowel transitions that produce a momentary peak in the function ee(i).

8. cv(i) = sum of F1(i) through F16(i).

This function tends to be greater for vowels than for adjacent consonants. This is because a constricted or closed vocal tract configuration results in a reduced acoustic output in the frequency range spanned by these filters. The function is used to detect syllable boundaries.

9. damp(i) = sum of F1(i) through F19(i) minus sum of F1(i-1) through F19(i-1).

This function indicates a sudden increase in energy in all filters. It will be a maximum for stop releases and sudden voicing onsets. It is used to detect stop bursts.

Features

The preceding functions are used in the computation of the following features. The usual interpretation of the three feature values (states) is: "-1" implies that the property is irrelevant for this time interval; "0" means that the property is relevant but not present; and "1" means that the property is relevant and present.

1. Voice(i) = 1 if [F2(i)>t1] or [F2(i)>t1-5 and Voice(i-1) = 1]
0 otherwise
t1 = 34

A hysteresis region of 5 units is employed about the threshold, t1, to ensure that a change of state is significant and not due to random fluctuations in level. The effect of the second conditional part of the definition is to keep Voice(i) in its current state for values of F2(i) between 30 and 34, the hysteresis region.

An unknown word may vary considerably in overall recording level if the speaker moves with respect to the microphone or changes his vocal effort. It is much easier to compensate for recording level by modifying the threshold, t1, than by attempting to modify the raw data. The maximum value attained by the time function Loud(i) is used for this normalization:

$$\text{Maxloud} = \text{Max}_i\ [\text{Loud}(i)]$$

Loud(i) attains its maximum value near the midpoint of the stressed vowel nucleus. For a speaker positioned correctly in front of a microphone and speaking at a comfortable level, a typical value for Maxloud is 200. Threshold normalization is accomplished by computing a new threshold, tl', for each unknown word according to the following formulae:

$$tl' = tl * (200/Maxloud)$$

The method fails if the recording level increases to saturation or decreases below background noise. The effective range for recording level sensitivity is somewhat less than the full 45 decibel range of the input system.

The fundamental voice frequency of a male speaker will vary from about 90 to 180 Hz. The greatest energy due to voicing will appear in the low frequency filters because energy in the harmonics of the laryngeal source falls off at about 12 db per octave. Voice(i) looks at the energy in filter 2. If this energy exceeds a threshold, voicing is present, otherwise not. This simple definition has worked surprisingly well. The greatest difficulty has been in treating final voiced stop releases consistently, and in detecting voicing energy in some voiced fricatives.

2. Syllable(i) = 1 when the function Cv(i) is significantly increasing
   0 when the function Cv(i) is significantly decreasing
   otherwise, set to same value as in previous time sample.

The precise computation is most easily stated in terms of a program segment. Peak and Valley are variables indicating the local maximum and minimum in the function Cv(i). Time is a variable that counts the number of time samples during which Syllable(i) has remained in the same state.

```
        Syllable(0) = 0, Valley = 0, Peak = 0,
        Time = 6, Cv(0) = 0
        Do for i = 1 in steps of 1 to 200:
        Time = Time + 1
        if [Syllable(i-1) = 0], Go To 1
        if [Cv(i-1)>Peak], Peak = Cv(i-1)
        Syllable(i) = 1
        if [[Time<6] or [[Cv(i)>Peak - 70] and [Dspect(i)<35]] or
        [Cv(i)>Peak - 20]], Go To 2
        Syllable(i) = 0, Valley = 1000, Time = 0
        Go To 2
    1   if [Cv(i-1)<Valley], Valley = Cv(i-1)
        Syllable(i) = 0
        if [[Time<6] or [Voice(i) = 0] or [[Dspect(i)<35] and
        [Cv(i)<Valley + 40]] or [Cv(i)<Valley + 20]],
        Go To 2
        Syllable(i) = 1, Peak = 0, Time = 0
    2   Continue
```

The times at which this feature changes state do not delimit precise syllable boundaries. The feature usually gives an accurate count of the number of syllables in a word and provides a rough segmentation. Examples of difficulties with Syllable(i) include a frequently missed third syllable in "binary" and the segmentation into two syllables of some repetitions of the words "four" and "core".

3. Stress(i) = 1 if Loud(i) = Maxloud for the first time.
0 if Syllable(i) = 1
-1 otherwise

$$\text{Maxloud} = \text{Max}_i\ [\text{Loud}(i)]$$

This feature is designed to indicate which syllable of a word is the one which has the maximum loudness and is therefore the stressed syllable. Stress assignment has worked very well. The only word for which stress assignment has not consistently agreed with the expected stress assignment is "overflow", and this may be due to variations in the actual stress given the word by our speakers.

4. [a]-like(i) = -1 if Voice(i) = 0
1 if [ah(i)>1] or [ah(i)>-2 and
[a]-like(i-1)= 1]
0 otherwise

This feature indicates the presence of the vowels [a, o$^{\omega}$, ɔ, æ, ʌ] in many phonetic environments, the vowels in [pot, boat, bought, bat, but] respectively.

5. [i]-like(i) = -1 if Voice(i) = 0
1 if[[ee(i)>16]or [[i]-like(i-1) = 1 and ee(i)>8]]
and F19(i)<25
and Dspect(i)<30
0 otherwise

This feature indicates the presence of the vowels [i, ɪ, e^{y}, ɛ] (the vowels in [beat, bit, bait, bet) in many phonetic environments. The two additional clauses involving F19(i) and Dspect(i) are

utilized to eliminate false responses to stridents and consonantal transitions which are momentarily [i]-like.

6. [u]-like(i) = -1 if [Voice(i) = 0]
1 if [oo(i)>t2 or [[u]-like(i-1) = 1
and oo(i)>t2-7]] and F2(i)>F1(i)
0 otherwise

t2 = 32

This feature indicates the presence of the vowels [u, ɝ, o$^{\omega}$, ɔ] (the vowels in [boot, book, boat, bought]) in many phonetic environments. The additional clause involving F1(i) and F2(i) is intended to eliminate responses to consonants such as [m, n, l, w]. It has been found necessary to ignore any transitions to the "1" state in the features [i]-like(i) and [u]-like(i) if they last less than 6 time samples. Even with this additional constraint, a number of consonants are assigned one of these vowel-like qualities. In addition, there appear to be no natural threshold boundaries in the vowel-quality space so that a certain percentage of the vocabulary is treated inconsistently by each feature.

7. [r]-like(i) = -1 if [Voice(i) = 0]
1 if [[er(i)>18 and F3(i)-F4(i)>5]
or [er(i)>14 and [r]-like(i-1) = 1]]
0 otherwise

This feature indicates the presence of [r, ɚ] in many phonetic environments. The clause involving F3(i) and F4(i) is intended to ensure that the first formant frequency is not high. The low

third formant of these phones produces a spectrum that is distinct and relatively easily differentiated from all other speech spectra. The feature works consistently except following voiceless stops and in some unstressed intervocalic positions.

8. Strident(i) is most easily defined in terms of a program segment:

```
        Maxstr = 0      t3 = 70
        Do for i = 1 in steps of 1 to 200
        if not [[[str(i)>t3] or [[str(i)>t3-20] and
               [Strident(i-1) = 1]]] and [str(i)>Maxstr-30] and
               [[str(i)>str(i-1)] or [Strident(i-1)= 1]]]
               Go To 1
        Maxstr = Max [Maxstr, str(i)]
        Strident(i) = 1
               Go To 3
    1   Maxstr = 0
        if Voice(i) = 1  Go to 2
        Strident(i) = 0
               Go To 3
    2   Strident(i) = -1
    3   Continue
```

This feature indicates the presence of [s, z, š, ž,] and also [t, d, k, g] in certain phonetic environments (generally in stressed position and followed by a front vowel). The floating threshold, Maxstr, is employed to divide a sequence of two successive stridents (e.g., [st]) into two strident segments separated by the state "-1".

9. Strid-stop(i) = 0 if Lstrident(i) = 1
1 if Strident(i) = 1
-1 otherwise

where Lstrident(i) is the same as Strident(i) except that the state "+1" must occur for more than 6 time samples or it is transformed to the state -1. This property computes the length of a strident segment and places it in one of two categories. In general, a long strident is a continuant and a short strident is a stop. However, in many phonetic environments, voiceless stops may have an aspiration duration that is longer than the frication duration of an intervocalic fricative.

10. Fricative(i) = 1 if [[F19(i)>t4] or [[F19(i)>t4-4]
and [Fricative(i-1) = 1]]
and [Loud(i)<120]]
0 if Voice(i) = 0
-1 otherwise

t4 = 6

This feature looks for the high frequency energy characteristic of [f, v, θ, ð], (the initial consonants of [fin, vend, thin, then]). Strident phones generally exceed the threshold and are also detected by Fricative(i). The upper bound on Loud(i) is used to prevent a response to a loud vocalic sound in which energy is spread over the entire spectrum. A feature like Fricative(i) or Strident(i) sends a great deal of information about a word to the recognition algorithm, but neither feature is as consistent in its analysis of an unknown word as a simple feature such as Voice(i).

11. Stop-burst(i) = 1 if [[damp(i)>75] and [F19(i-1)<30] and
[[F2(i)<20] or [F2(i)-F2(i-1)<1]]]
0 if Voice(i) = 1
-1 otherwise

This feature detects the sudden onset of energy in all filters that is characteristic of the release of a stop. The stop burst is distinguished from a rapid voicing onset by requiring the energy in F2(i) to either be low or not increasing. The "-1" state following a burst is removed if it is less than 3 time samples long. This transformation is an only partially successful attempt to signal voiced stops by the sequence "1,0" and voiceless stops by the sequence "1,-1,0."

12. Nasal(i) = -1 if Voice(i-2) = 0
1 if [Loud(i)-Loud(i-1)>5
or [Nasal(i-1) = 1
and Loud(i)-Loud(i-1)>1]]
and [ah(i) -ah(i-1)>5
or ah(i) = ah(i-2)]
0 otherwise

This feature looks for a voiced segment followed by a sudden increase in loudness and a rising first formant frequency. A nasal followed by [i] does not satisfy this criterion, but nasals in many other environments are detected. Initial [l] is frequently also detected by the feature. Features that look for dynamic properties of the input spectra are not easy to define efficiently. This relatively simple algorithm does not give a very satisfactory definition of nasality.

13-15. There are three other features of a somewhat different type. These features are intended to give a fairly precise characterization of the stressed vowel in the unknown word. The features, called [a]-stress, [i]-stress, and [u]-stress respectively, characterize the spectrum at a single time sample, that point during the stressed vowel when Loud(i) reaches a maximum for the word.

The ranges of the functions ah(i), ee(i), and oo(i) have been divided into 10 regions corresponding to the 10 possible states of the three special features:

UPPER BOUND ON REGION

State:	1	2	3	4	5	6	7	8	9	10
ah(i)	none	14	9	4	0	-3	-6	-11	-16	-27
ee(i)	none	80	60	40	20	0	-20	-40	-60	-80
oo(i)	none	80	60	40	20	0	-20	-40	-60	-80

A typical stressed [i] phone might be characterized by the states [a]-stress=9, [i]-stress=1, and [u]-stress=6. It has been found that this characterization is relatively stable for a given word. Most words fall into two or at most three states for each of the features.

B. Non-Linguistic Properties

A set of abstract non-linguistically oriented properties was defined near the end of the research project in order to test the performance of the linguistically oriented features just described. The properties are intended to take advantage of some of the constraints that apply to speech spectra, but to be unaware of the detailed phonetic content of the words.

There are two distinct types of spectral shape properties that were used in this set. The first nineteen properties crudely categorize the shape over time of the output of each of the nineteen filters, individually, with no attention paid to correlation between filters. There are three regions of "interest" in the output. The spectral property is given the value "-1" if the output is in the lowest region, "0" in the middle region, and "1" in the highest region. A hysteresis region around the transition threshold is used to prevent many transitions when the filter output is near the border of these regions.

Formally, we define a set of 19 spectral amplitude properties, Sn(i), as follows, where the notation Fn(i) denotes the output of filter n at the i^{th} sample interval:

$$\begin{aligned} Sn(i) = {} & 1 \text{ if } [Fn(i)>B] \text{ or } [[Fn(i)>B-8] \text{ and } [Sn(i-1)=1]] \\ & 0 \text{ if } [Fn(i)>A] \text{ or } [[Fn(i)>A-8] \text{ and } [Sn(i-1)\neq -1]] \\ & -1 \text{ otherwise} \end{aligned}$$

A and B, as a function of the filter number, are given in the following table:

n	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19
A	37	34	30	25	20	30	25	20	15	25	21	18	15	20	15	17	20	17	13
B	50	47	44	40	35	45	40	35	30	38	34	31	27	35	30	30	30	25	30

The threshold values in this table should be modified if there is a change in recording level. This can be done in the same way as previously described for thresholds involving linguistic features.

Another ten properties are used which roughly describe the correlation between adjacent filters for the lower ten filters. These properties categorize the differences between energies in adjacent filters. The three alternatives are: the next higher band has significantly more energy, significantly less energy, or about the same energy, as its lower neighbor. These properties are given values "1", "-1", and "0" respectively in these cases. Formally, we define a set of 10 spectral difference properties, Dn(i), as follows, where n=1, 2, ... , 10 and m=n+1:

```
Dn(i) = 1 if [[Fm(i)-Fn(i)>4]
           or [[Fm(i)-Fn(i)>-1]
          and [Dn(i-1)=1]]
       -1 if [[Fm(i)-Fn(i)<-4]
           or [[Fm(i)-Fn(i)<1]
          and [Dn(i-1) = -1]]
        0 otherwise
```

A change of state for one of these properties indicates that the spectral shape of the current time sample differs from the spectral shape of the previous time sample. A change of state is not produced by a simple change in level. This is because the large hysteresis region around the thresholds prevents a property from changing state for the equal energy case.

The thresholds for the two types of properties were selected so that each state could be reached by at least some of the vocabulary items. The 29 properties produce sequences of states used in recognition. We can compare directly these recognition scores with those produced by state sequences from the 15 linguistic features.

VI. RESULTS

The results of this research are best described in three parts. We first describe the experimental procedure and the results of the recognition system in an optimal (quiet) environment. We then describe an experiment which indicates the behavior of the system when degraded by added noise. Finally, we discuss a technique by which we evaluate individual properties when the recognition rate gets so high that improvements can only be measured indirectly.

A. Recognition Experiments

As benchmark experimental data, we recorded a word list, Table 2, used by Ben Gold in other speech recognition experiments. This list was recorded in a quiet room (S/N ratio > 35 db) as spoken 10 times by two speakers, KS and CW. Both had been used as subjects by Gold. KS and CW were also recorded 5 times each on two other lists, an augmented International Word Spelling alphabet, Table 3, and a message list typical of one that might be used in a NASA mission context, Table 4.

The messages were recorded on high quality magnetic tape at 7 1/2 ips, with approximately two seconds gap between words. For almost all experiments we used a digital tape containing the 6 bit logarithms of the filter bank output sampled every 10 milliseconds. Only in the noise degradation experiment was the analog input used directly.

insert	name
delete	end
replace	scale
move	cycle
read	skip
binary	jump
save	address
core	overflow
directive	point
list	control
load	register
store	word
add	exchange
subtract	input
zero	output
one	make
two	intersect
three	compare
four	accumulate
five	memory
six	bite
seven	quarter
eight	half
nine	whole
multiply	unite
divide	decimal
number	octal

TABLE 2. Word List From Ben Gold

zero	hotel
one	India
two	Juliet
three	kilo
four	Lima
five	Mike
six	November
seven	Oscar
eight	papa
niner	Quebec
affirmative	Romeo
negative	Sierra
alpha	tango
bravo	uniform
Charlie	Victor
delta	whiskey
echo	x-ray
foxtrot	yankee
golf	zulu

TABLE 3. International Word Spelling Alphabet and Numbers

one	distance to dock
two	fuel tank content
three	time to sunrise
four	time to sunset
five	orbit apogee
six	orbit perigee
eight	revolution time
nine	closing rate to dock
point	midcourse correction time
plus	micrometeoroid density
stop	radiation count
zero	what is attitude
seven	remaining control pulses
minus	alternate splashdown point
pressure	weather at splashdown point
negative	sea and wind at splashdown point
what is yaw	visibility at splashdown point
what is pitch	temperature at splashdown point
end repeat	skin temperature
affirmative	power consumption
inclination	fuel cell capacity
distance to earth	repeat at intervals

TABLE 4. Message List From NASA Context

A training round consists of one repetition of the entire word list in which the computer makes an identification attempt, is told the correct response, and stores the characterization of this message for use in future identification. With four rounds of training utilizing the linguistic properties, the system recognized correctly (gave as unambiguous first choice) 51 and 52 out of 54 words of the Ben Gold list for KS and CW respectively. Early experiments indicated that additional training would not increase the recognition scores after reaching its asymptote, and these 95 +% correct scores were available after only 3 rounds of training. Further experiments gave speaker KS 38 out of 38 correct on the International Word Spelling Alphabet and 43 out of 44 correct on the NASA message set, indicating the usefulness of these linguistic properties on other message sets. Typical recognition rates for the Gold list as a function of training rounds are shown in Table 5. Note the lack of improvement (and oscillation of scores) with more than 4 rounds of training. Part of the difficulty is that these later rounds were recorded at a different time. This indicates that some degradation in performance may be expected under less well controlled conditions.

Some incorrect responses are listed in Table 6 (for KS on the Gold list after 4 training rounds). These confusions give an indication of the types of phonetic information that is not well represented by the non-linguistic features. For example, the "four-core" confusion suggests that frication noise often falls below the lower thresholds of the filters, and that formant transitions are not sufficiently distinct to trigger different state transitions for this word pair. It is expected that some improvements could be made in the threshold settings by studying the data of Table 6.

TABLE 5.

Typical Recognition Scores (KS)

Previous Training	1	2	3	4	5	6	7	8	9	Perfect
Ben Gold List	43	44	49	51	47	45	52	49	51	54
Alphabet	30	36	34	38	—	—	—	—	—	44
NASA List	35	37	43	43	—	—	—	—	—	38

TABLE 6.

Typical Mistakes

Guess	Correct Answer
Alternate Splashdown Point	Weather at Splashdown Point
Core, or Whole	Four
Memory	Binary
One	Load

These asymptotic recognition scores also indicate the separability of this 54 word vocabulary with these linguistic properties. The three vocabularies were combined into a single vocabulary of 114 distinct messages (duplications between vocabularies were considered only one message). On this larger vocabulary, the recognition score was a remarkable 110 out of 114 forks. All those messages identified correctly in the individual lists were also identified in this larger context after the same four training rounds. This indicates very little interference between words on these different lists; and implies that fairly large non-interfering (phonetically balanced) vocabularies might be constructed while maintaining this high recognition rate. It should be emphasized that while messages were repeated in training and testing, the same speech utterance was never seen more than once by the system.

To test the consistency of speakers uttering messages, we devised the spectral threshold properties described earlier. We tested these properties for recognition only on the Ben Gold word list. After four rounds of training, the system was able to identify correctly 52 out of the 54 input messages presented to it for KS; and 51 out of 54 for CW. This rate is as good as that discussed for the linguistic features; however, the cross speaker correlation was higher for the linguistic property set. For the second speaker (using the first speaker's training) 26 out of 54 messages were identified correctly on the first round, whereas using the spectral properties only 15 out of 54 were correctly identified.

B. Performance of the System with Added Noise

Speech-shaped random noise, (i.e., white random noise filtered by a bandfilter whose frequency response is identical to the long-term RMS speech spectrum) was added to the speech signal to test the degradation of the system under adverse conditions. It was prefiltered in order to (1) obtain approximately equal noise levels in all filter channels and (2) avoid overloading of the filter-networks. This would occur for a white noise input at low S/N ratios, due to the high frequency pre-emphasis in the filter bank. The set up used to add noise to the speech input is shown in Fig. 6.

The S/N ratio at the input to the filter networks (B) is determined by the constant S/N ratio of the speech input and by the noise level as set on the calibrated attenuator. The noise level of the speech input was measured at points A, B, C and found to be 35 db, 36 db* and 34 db respectively, below the speech peaks with speaker DB. (speech peaks measured with a ballantine-true-RMS-meter, VU-METER damping) The noise level of the speech-noise generator was set so that the overall level was 0 VU at point D. (attenuator in ODB position) With the attenuator set at -36 db and a speech S/N ratio of 36 db, the effective S/N ratio at point B would be 33 db.

*The noise level at point (B) is 1db lower than at point (A) because of the low frequency cut off (120 HZ) in the input amplifier.

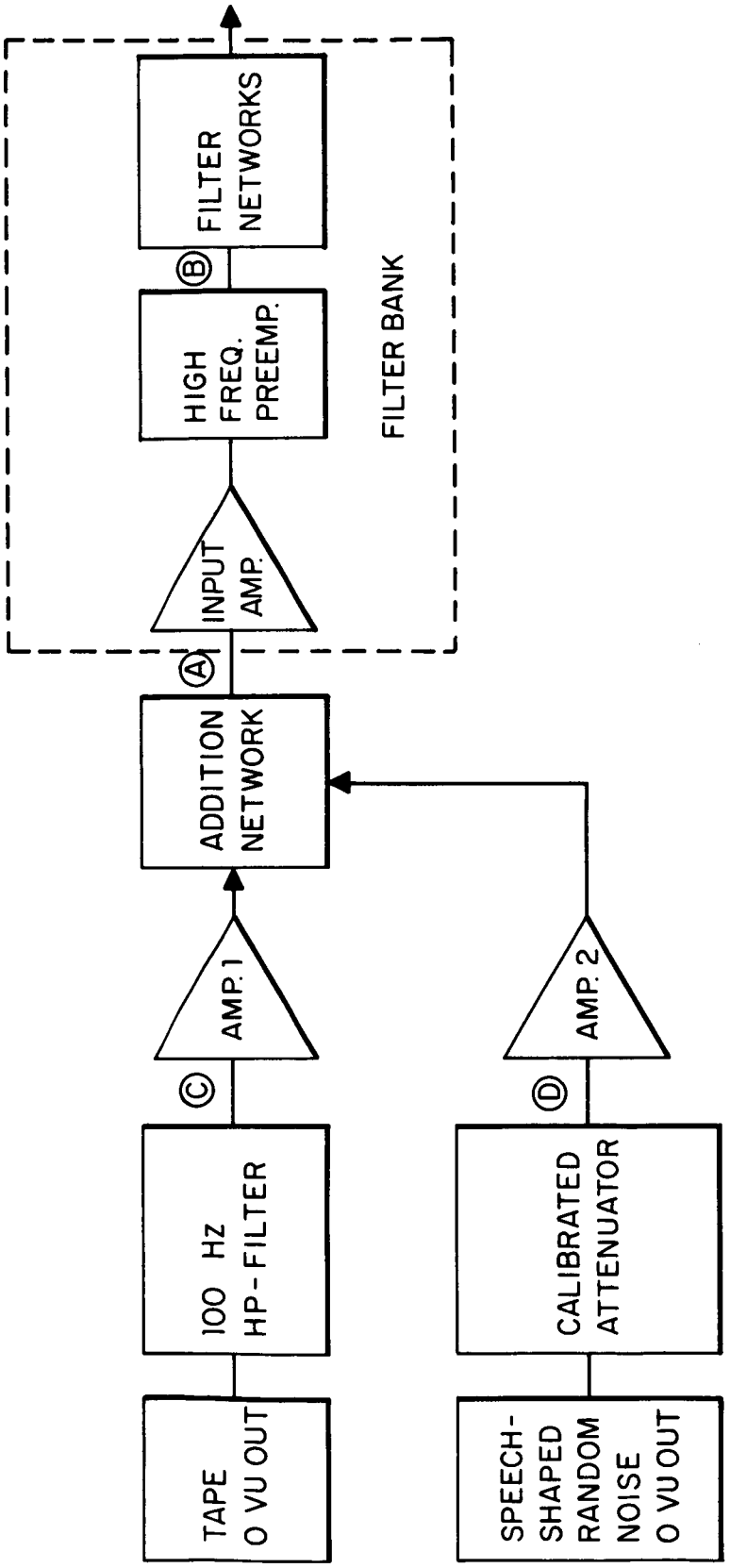


FIG. 6 SET UP USED TO ADD NOISE TO THE SPEECH INPUT

The system was trained for four rounds with no noise added. The same set of utterances with different levels of noise added were used to test the system. The recognition rates are shown in Fig. 7. Note that DB is a less consistent speaker than the other two, or the properties need to be tuned for him, because the asymptotic recognition rate is only about 80%. The shape of the rate of recognition curve under noise degradation is identical to that for humans, falling off almost completely in 20 db. However, the break for humans is 10 to 20 db lower, and the asymptotic rate much higher.

C. Evaluation of Properties

As recognition scores of the system get better and better, it becomes more difficult to evaluate the individual properties. A measure of information content has been used as an aid in the evaluation of individual features. This measure of the information contained in a feature tree has provided some insight into the desirable and undesirable properties of a feature. The measure will be defined, applied to some examples, and then discussed.

Let:
- J be an index over the nodes of a tree
- I be an index over the words in the vocabulary
- n_{IJ} be the number of times word I appears at node J
- NV be the total number of words in the vocabulary
- NT be the number of rounds of training
- R be a measure of the information contained in the tree about the identity of an unknown word.
- P(I) probability of Ith word (assumed equal to 1/NV)

$$R = H(I) - H_J(I)$$

where $$H(I) = -\sum_I p(I) \log [p(I)] = \log NV$$

and $$H_J(I) = -\sum_{IJ}\sum p(I,J) \log [p_J(I)] = -\sum_{IJ}\sum \frac{n_{IJ}}{(NV)(NT)} \log \left[\frac{n_{IJ}}{\sum_I n_{IJ}}\right].$$

Effective S/N Ratio	34	31	25	20	15	10
Number Correct	42	44	43	38	31	23
Percent Correct	78	81	80	70	59	44
Log (%-Correct)	1.89	1.91	1.90	1.85	1.77	1.65

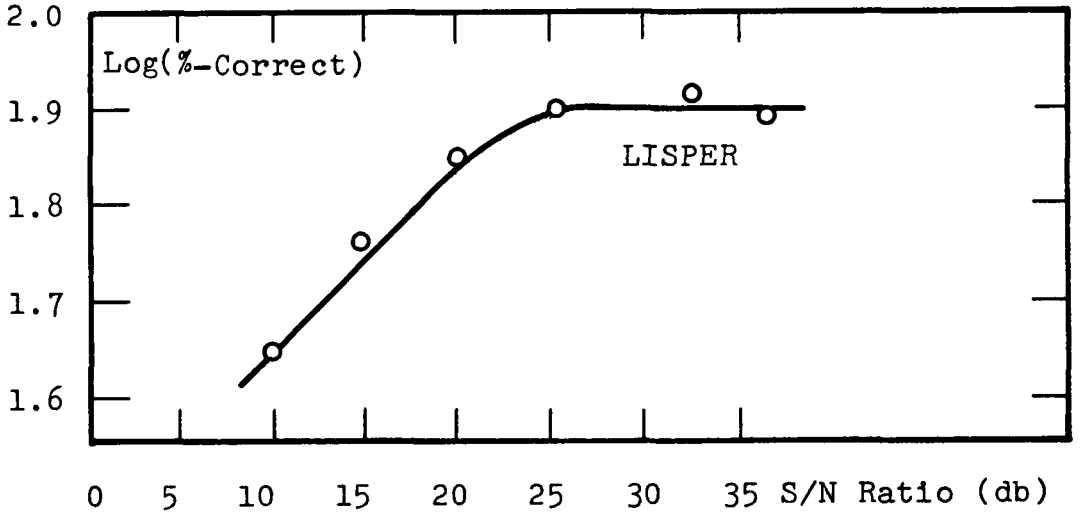
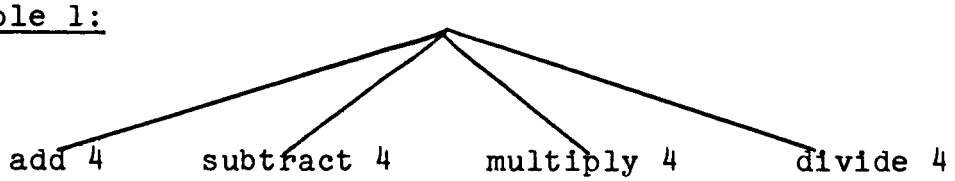


Fig. 7 Recognition Rate With Added Noise

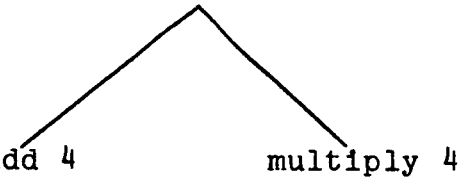
Example 1:



R = H(I) = 2 bits

This is an example of the perfect tree, one that assigns a unique node to each vocabulary word.

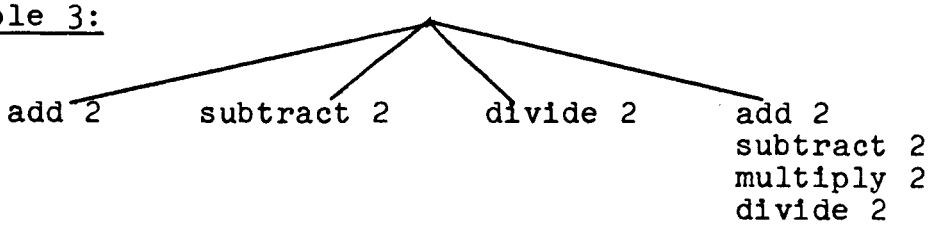
Example 2:



R = 2 - 1 = 1 bit

This is an example of a tree that consistently assigns a word to the same node, but does not completely separate the vocabulary words from one another.

Example 3:



R = 2 - 1 = 1 bit

This is an example of a tree with more nodes (greater possible separability), but with an inconsistent assignment of words to nodes.

There is always the possibility that the unknown word will be assigned to a new node of the tree, one that was not present during training. The tree can provide no information about the identity of the word in this case. The measure of information, R, does not account for this possibility because it assumes incorrectly that the frequencies, n_{IJ}, are accurate estimates of the underlying probabilities.

R may be a biased measure because of this fact. When training grows a large tree, there is, initially, a greater chance of a word producing a novel node than when training grows a smaller tree. The difference between large and small trees in growth of novel nodes after about three rounds of training is not very great. Therefore, we have not attempted to correct for the potential bias in the information measure R.

There are some intuitive measures of the goodness of a feature we have found interesting. The number of nodes N (different input sequences found in training) gives a measure of the range of separation that can be expected of the property. The number of messages M, or average number of variations per input message M', gives a measure of the consistency of the property. The ratio N/M', of separability to consistency, seems intuitively to be a quantity one would want to maximize.

The intuitive measures and the computed information provided by each of the linguistic features is tabulated in Table 7. There seems to be a reasonable correlation between R and N/M'. It should be obvious that the features are highly redundant and that there may be pairs of words that we are unable to differentiate with this feature set. The latter question can only be answered by testing additional repetitions of the vocabulary and noting both error frequencies and the degree of similarity of feature assignments for pairs of words.

	N=# Nodes	M=# Messages	Average # of variations M'=M/54	N/M'	R Information Measure
1. Voice	3	64	1.19	2.5	1.4
2. Syllable	4	70	1.30	3.1	1.5
3. Stress	7	72	1.33	5.3	2.1
4. [a]-like	33	110	2.02	16.3	3.0
5. [i]-like	37	126	2.34	15.8	3.0
6. [u]-like	36	127	2.35	15.3	2.6
7. [r]-like	21	91	1.68	12.5	2.4
8. Strident	45	110	2.02	22.2	3.2
9. Strid-stop	11	92	1.71	6.4	2.5
10. Fricative	55	139	2.57	21.4	3.2
11. Stopburst	50	167	3.09	16.2	2.7
12. Nasal	19	110	2.02	9.4	1.8
13. [a]-stress	10	158	2.92	3.4	2.3
14. [i]-stress	10	166	3.07	3.3	2.5
15. [u]-stress	10	189	3.50	2.9	1.8

TABLE 7. Evaluation of Linguistic Properties After Ten Rounds of Training - Speaker KS

The trees grown by the non-linguistic features are quite large. Most words are assigned to more than one node during training. Even with this inconsistency, the information provided by each feature is relatively high. As a comparison figure, consider the fact that a feature detecting the presence or absence of voicing, functioning with no errors, provides less than 2 bits of information to the recognition algorithm. (The vocabulary words contain 1, 2, 3, or 4 voiced segments and this results in a feature tree with only four nodes.)

There are two conclusions to be drawn from these results. One is obvious, that there is a trade-off between the separability of words provided by a large tree and the consistency with which a word is analyzed; see Example 2 vs. Example 3. The information measure, R, gives a good idea of what that trade-off is.

Another, non-obvious, conclusion is that the information measure accounts for the comparable recognition scores using non-linguistic and linguistic features. The linguistic feature set was derived with a strong emphasis on consistency. The linguistic features condense a number of acoustic measurements to a binary decision, the presence or absence of some attribute. Both of these strategies are bad given the nature of the recognition algorithm. This algorithm is designed to make efficient use of all of the clues provided by the features. The emphasis on consistency and minimal binary distinctions reduces the information flowing to the recognition algorithm. Errors in feature assignment are then more likely to be fatal. A general observation is that more features with greater redundancy almost invariably led to higher recognition scores.

VII. DISCUSSION

We have outlined the reasons why word recognition based on spectral input data is not a straightforward problem in pattern recognition. The invariants in the speech code appear to be relatively complex functions of the spectral input. We have also outlined the reasons why phoneme recognition is not an appropriate intermediate step in a word recognition algorithm. Unsolved problems include the segmentation of speech spectra into phoneme sized chunks and the derivation of rules for recognizing all of the allophones of a given phoneme. Our approach is intermediate between pattern recognition and phoneme recognition; it combines some of the advantages of both.

A set of acoustic feature detectors have been defined. The time dimension is removed from each feature so that the learning program deals only with the sequences of states produced by the feature definitions. The feature sequence characterization of an input message is sufficiently redundant so that the recognition algorithm is able to deal successfully with errors and inconsistencies in some fraction of the features.

The advantages of this approach are:

(1) No precise segmentation of the utterance is required; features can change state in arbitrary time segments because the time dimension is removed from a feature in a way that is independent of the changes of state of other features.

(2) A word need not be registered at exactly the same place in the 2 second speech sample; state sequences are independent of time origin.

(3) Feature definitions can be made insensitive to the recording level by adjusting threshold settings according to a measure of the loudness of the unknown word.

(4) Feature definitions can be made less sensitive to noise by employing hysteresis regions about thresholds.

(5) State transition sequences weight changes in spectra as the important cues to word recognition. (simple pattern-matching schemes will give undue weight to a sustained vowel).

(6) Features may be defined to take advantage of natural acoustic boundaries between phones (Stevens, 1968b) and thereby minimize the variability in state transition sequences produced by a feature.

(7) Features may be added to the system to provide redundancy for those decisions that are difficult. There is a direct control over the amount of redundancy sent to the recognition algorithm.

(8) The feature approach permits the introduction and testing of linguistic hypotheses, such as placing greater emphasis on the properties of the stressed syllable.

The disadvantages of this approach are:

(1) Removal of the time dimension discards all information concerning the simultaneous occurrence of specific states for two or more features. We have used the voicing feature to reintroduce some timing information.

There exist pairs of English words for which this "time with respect to the voiced segments" is not enough to disambiguate the pair. Special features can always be defined to treat special cases, but our approach is not easily generalized to yield segmentation of an utterance into phone-sized chunks.

(2) The features currently implemented are not speaker independent. Each speaker will have to train the system and this requires approximately 3 or 4 repetitions of the vocabulary.

(3) Our system will degrade in performance as the length of the vocabulary is increased or as the number of speakers that it can simultaneously recognize is increased. This property is of course true of any recognition program; however, it should be noted that, with our current simple-minded set of features, there is a high error rate in any feature characterization and we rely heavily on redundancy to select the most likely input message.

(4) For our limited objectives, the current implementation is computationally fast and gives satisfactory results. However, more sophisticated and more reliable features would be desirable. The exponent in the function relating computation time and feature performance is not known, but may be restrictively large.

A set of simple sum-and-difference properties was compared with the linguistically motivated features in the latter stages of research. The comparable performance of this set of properties has led to several conclusions concerning the desirable attributes of features that operate within this framework.

(1) A good feature includes a maximum amount of information in the sequences that it produces. Factors that influence information content are consistency of characterization and the number of vocabulary items that can be differentiated on the basis of the feature. A voicing feature that works perfectly does not contain as much information about the unknown word as any of the relatively inconsistent spectral properties. In other words, a moderate amount of inconsistency is tolerable if accompanied by increased word separability (and additional features to provide redundancy).

(2) Another reason why the spectral properties compared favorably on the basis of recognition scores is that the linguistic features are fewer in number and computationally similar in form to the spectral properties. We believe that, even for limited vocabulary word recognition, a set of phonetically oriented features exist which are better in some sense than simple pattern features. The reasons for this faith consist of arguments:

(a) that there exist natural acoustic boundaries between phones (Stevens, 1968b),

(b) and that there exist invariant attributes in acoustic waveforms from different speakers; these invariants, when incorporated into feature definitions, will produce recognition scores that are less sensitive to the individual speaker.

(3) The arguments in favor of phonetically oriented features are offset by the computational simplicity of the spectral properties. Not enough is currently known in acoustic phonetics to take advantage of these theoretical benefits without additional basic research. We argue that a carefully selected set of properties like our spectral properties represent a practical word recognition solution that may not be superseded for some time to come.

The research described in this report does not consistute a completed investigation. There are many open questions that can profitably be pursued. The list of linguistic features described in this report is biased in favor of features that seemed easiest to define and implement. A glaring deficiency is in the area of place-of-articulation features for stop and nasal consonants. The primary reason for not including features of this type is that the feature definition will depend in part on the specified vowel following the stop. The result must be a fairly complex definition in terms of parameters that are not well understood.

We hope to study this and related problems of context dependent features by implementation of recent proposals of K. N. Stevens that place-of-articulation for stops may be identified by the timing of energy build-up in high and mid-frequency regions relative to the time of build-up in the low-frequency region. Other features to be developed are Aspiration, Tense-lax, Front-glide, and Back-glide. Several of the current features can be substantially improved by context-dependent definitions (e.g., nasal).

We wish to investigate further the reasons for the success of the non-linguistic features. Of help would be a measure of the information about the identity of a word that one feature gives

conditional to the fact that we know the sequence of states generated by another feature, i.e., independent additional information that a new feature provides. Such a measure can be defined in the same general way as was described in section VI. With this measure and the results obtained from the development of new linguistic features, we may be able to modify the spectral property set to give even better results.

APPENDIX A CIRCUITS OF FILTER BANK

Input Stage

The circuit diagram of the input stage is shown in Fig. 7. The output from a conventional tape recorder is fed into an input amplifier (Type BBN-DE-200) whose variable feedback resistor provides the common gain adjustment for all filter channels. The gain adjustment procedure is facilitated by a VU-meter connected to the output of the input amplifier. By means of a separate potentiometer this VU-meter may be adjusted such that speech peaks exceeding +3VU will indicate clipping in the filter channels and/or overloading of the A/D converter.

The RL-network in the feedback path of the first driver amplifier provides the high frequency pre-emphasis. By means of a switch this network may be replaced by a resistor for gain calibration of the filter channels.

Since the bandpass filter networks require inputs of opposite phase, a second inverting driver amplifier is used as shown on the lower right of Fig. 7. A potentiometer in the feedback path provides for adjustable amplitude balance of the two input signals.

Band-Pass Filters

The pole pattern yielding the frequency responses of Fig. 8 is shown in Fig. 9. Poles associated with one filter are connected by converging lines. Each filter shares two poles with each neighboring filter.

According to the Lerner synthesis procedure, poles of an individual filter are spaced at equal intervals (240 Hz) in the pass-band and at half intervals (120 Hz) at the band edges. Since the frequency responses of neighboring filters have their 3 dB crossover points at a frequency mid-way between the shared poles, all 4-pole filters (no. 1-15) have bandwidths of 360 Hz. The number of the poles for the remaining filters are selected such that bandwidths as given in Table 1 resulted.

Each pole of Fig. 9 is realized by a series resonant circuit as shown in Fig.10 for filters no. 1 through 15. The proper pole residue is achieved by resistor weighting of the currents from each resonant circuit into the summing node of an operational amplifier as described by Drouilhet and Goodman(1960). Identical inductor values were used for all resonant circuits and the capacitors were calculated by $C = 1/4\pi^2 Lf^2$, where f is the frequency of a pole. (Approximately 1% of the C value was realized by a trimmer capacitor to compensate for slight inaccuracies and variations of the filter components.) For a chosen value of L = 0.5 mh, R = 1.66 k was obtained from the relation $L/R' = 1/2\pi \cdot 360$, where $R' = 2/3$ R, the parallel resistance of R and 2R.

Filters no. 16 though 19 were realized by the same circuit structure as shown in Fig.10, with poles in the passband and at the band-edges. (Fig. 9) weighted by R and 2R respectively. The highest pole at 6620 Hz was tied over R to ground to simulate the input impedance of an additional neighboring summing amplifier as is also shown for the lowest pole in Fig. 9.

Rectifiers and Low-Pass Filters

The circuitry containing the summing amplifiers, rectifiers, low-pass filters and output amplifiers is shown in Fig. 11. (All operational amplifiers: Union Carbide type H6010). The circuit for filter channel no. 1 is shown in the upper portion of the figure. The summing amplifier is followed by a full wave rectifier circuit. Full wave rectification was necessary for filter channel no. 1 only, in order to reduce the ac-components passed by the low-pass filter of this channel. (At 80 Hz the rms level of the ac-component in the output of filter channel no. 1 is 35 dB below the dc level and decreases with a slope of 22 dB/octave.)

The circuit in the lower portion of Fig.11 is used for filter channels no. 2-19. The first amplifier (1) sums the current components from the band-filter, (2) half-wave rectifies that signal and (3) provides a signal of adjustable amplitude to drive the low-pass filter. The gain of these amplifiers is adjusted such that all dc channel outputs are equal for white noise input to the spectrum analyzer (pre-emphasis network not used).

The output of the low-pass filters is fed into output amplifiers which provide low impedance sources to drive the A/D converter.

APPENDIX B Examples of the sequences produced by selected features for ten repetitions of the Gold vocabulary by speaker KNS.

1. Voice

The sequence 010 indicates the presence of one voiced segment, 01010 indicates two voiced segments, and 0101010 indicates three voiced segments. The initial portion of any sequence is not repeated and is implicit in the columns in the printout. A ten bit (octal) code follows each vocabulary word. This code indicates which repetitions of the word produced the sequence. For example, the first 9 repetitions of the word "bite" produced an indication of one voiced segment, and the 10th repetition produced an indication of two voiced segments; intense prevoicing of the [b] appears to have triggered the extra segment in this case.

Voice

0 1 0 (370 BITE 1776 SEVEN 1336 UNITE 1777 WHOLE 1777 HALF 1777 QUARTER 77 MEMORY 1777 MAKE 1777 EXCHANGE 101 WORD 1777 POINT 1777 JUMP 1777 SKIP 1777 SCALE 1777 END 1777 NAME 1777 NUMBER 377 NINE 1775 EIGHT 1777 SIX 1777 FIVE 1777 FOUR 1777 THREE 1777 TWO 1777 ONE 1777 ZERO 1777 ADD 1777 STORE 1777 LOAD 1777 LIST 1777 CORE 1777 SAVE 1777 BINARY 1777 READ 1777 MOVE 1777 DELETE 1777)

1 0 (222 NUMBER 1400 QUARTER 1700 EXCHANGE 1676 NINE 2 OCTAL 1777 DECIMAL 1777 BITE 1 ACCUMULATE 777 COMPARE 1737 INTERSECT 33 OUTPUT 1777 INPUT 1777 CONTROL 1777 OVERFLOW 35 ADDRESS 1777 CYCLE 1777 DIVIDE 1777 SEVEN 441 SUBTRACT 1777 DIRECTIVE 1777 REPLACE 1777 INSERT 1777)

1 0 (123 ACCUMULATE 1000 COMPARE 40 INTERSECT 1744 OVERFLOW 1742 REGISTER 1777 MULTIPLY 1777)

2. Strident

Strident is an example of a feature that is relatively inconsistent in assigning sequences to vocabulary items. Nevertheless, Strident sends a great deal of information to the recognition algorithm because there is good separation of words in this tree and, after a few rounds of training, much of the possible inconsistency has been learned and is represented in the tree.

STRIDENT

```
  0 -1  (210 END 1400 THREE 300 ADD 500 HALF 1770 WORD 1770
LOAD 1774 WHOLE 1777 QUARTER 77 MEMORY 1777 MAKE 713 NAME 1777
 NUMBER 377 NINE 1775 FIVE 1777 FOUR 1777 ONE 1777 CORE 1777
 BINARY 1777 READ 301 MOVE 1773)
        0 -1  (110 NUMBER 1400 DIVIDE 720 NINE 2 COMPARE 1737
 OVERFLOW 31)
              0 -1  (104 COMPARE 40 OVERFLOW 1742)
              1 -1  (124 REPLACE 1700 ADDRESS 1724 BITE 1 OUTPUT
 1777 INPUT 1777 DIVIDE 1051)
           1 -1  (106 ADDRESS 52 REPLACE 77)
        1 -1  (154 MAKE 1064 MOVE 4 BITE 1776 READ 1476 UNITE
 1777 HALF 7 WORD 7 POINT 1777 END 377 EIGHT 1777 ADD 1277 LOAD
 3 DELETE 701)
              0 -1  (117 DIRECTIVE 720 DECIMAL 210 OVERFLOW
4 OCTAL 1767 CONTROL 1777)
                    0 -1  (104 MULTIPLY 1321)
                          1 -1  (100 ACCUMULATE 1000)
                    1 -1  (104 INSERT 320 ACCUMULATE 451)
                 1 -1  0 -1  1 -1  (100 INTERSECT 400)
                    0 -1  1 -1  (100 INTERSECT 1000)
              1 -1  (100 LIST 100)
                 0 -1  0 -1  (100 MULTIPLY 400)
                       1 -1  (107 EXCHANGE 1676)
           0 -1  (99 OCTAL 10 DIRECTIVE 1)
                 1 -1  (106 ACCUMULATE 326 ADDRESS 1 INSERT
453)
                       0 -1  (101 REGISTER 1200)
                       1 -1  0 -1  (100 REGISTER 400)
              1 -1  0 -1  (101 REGISTER 140)
                          1 -1  (103 INTERSECT 344)
                    1 -1  0 -1  (102 REGISTER 32)
                       0 -1  (101 REGISTER 5)
     1 -1  0 -1  (105 QUARTER 1200 DECIMAL 1526)
                 0 -1  (101 MULTIPLY 42)
                 1 -1  (103 INSERT 1004 INTERSECT 33)
           1 -1  (108 LIST 1677)
        0 -1  (101 QUARTER 500)
              0 -1  (101 MULTIPLY 14)
  1 -1  0 -1  (121 SCALE 600 SAVE 574 SEVEN 206 THREE 1477 TWO
 1777 ZERO 1041)
              0 -1  (108 CYCLE 773 SEVEN 441)
                    1 -1  (101 DIVIDE 6)
              1 -1  (104 SIX 1446)
                    0 -1  (102 DIRECTIVE 40 SUBTRACT 541)
                          1 -1  (105 SUBTRACT 1236)
                 0 -1  (100 DIRECTIVE 4)
                       1 -1  (100 DIRECTIVE 1000)
        1  0 -1  (123 SKIP 1777 SCALE 1177 STORE 1777)
                 1 -1  (101 EXCHANGE 101)
     0 -1  (118 SEVEN 1130 ZERO 736 JUMP 1777 SAVE 1203)
           0 -1  (101 CYCLE 1004)
           1 -1  (108 DELETE 1076 SIX 331)
                 0 -1  (100 DECIMAL 40)
              0 -1  (101 DIRECTIVE 12)
        1 -1  0 -1  (100 DECIMAL 1)
```

REFERENCES

Bell, C. G., H. Fujisaki, J. Heinz, K. N. Stevens, and A. House (1961). "Reduction of speech spectra by analysis-by-synthesis techniques." The Journal of the Acoustical Society of America, 33, pp. 1725-1736.

Bobrow, D. G., D. L. Murphy and W. Teitelman. The BBN LISP System, BBN Report No. 1677, April, 1968.

Chomsky, N., and M. Halle (1968). The Sound Pattern of English. New York: Harper and Row.

Davis, K. H., H. R. Biddolph, and S. Balashek (1952). "Automatic recognition of spoken digits." The Journal of the Acoustical Society of America, 24, pp. 637-642.

Denes, P., and M. Mathews (1960). "Spoken digit recognition using time-frequency pattern matching." The Journal of the Acoustical Society of America, 32, pp. 1450-1455.

Drouilhet, D. R. and L. M. Goodman (1966). "Pole shared linear-phase band-pass filter bank" Proc. IEEE, 54, pp. 701-703.

Dudley, H., and S. Balashek (1958). "Automatic recognition of phonetic patterns in speech." The Journal of the Acoustical Society of America, 30, pp. 721-732.

Fairbanks, G. (1940). "Recent experimental investigations of vocal pitch in speech." The Journal of the Acoustical Society of America, 11, pp. 457-466.

Fant, G. (1960) Acoustic Theory of Speech Production, Mouton and Co., 'S-Gravenhage:

Flanagan, J. L. (1965). Speech Analysis, Synthesis and Perception. New York: Academic Press.

Fry, D. B., and P. Denes (1958). "The solution of some fundamental problems in mechanical speech recognition." Language and Speech, 1, pp. 35-58.

Gerstman, L. J. (1967). "Classification of self-normalized vowels." 1967 Conference on Speech Communication and Processing, 6-8 November, Office of Aerospace Research, United States Air Force, pp. 97-100.

Gold, B. (1966). Word Recognition Computer Program. Technical Report 452, Research Laboratory of Electronics. Cambridge: M. I. T.

Heinz, J. M., and K. N. Stevens (1961). "On the properties of voiceless fricative consonants." The Journal of the Acoustical Society of America, 33, pp. 589-596.

Hemdal, J., and G. Hughes (1964). "A feature based computer recognition program for the modeling of vowel perception." In W. Wathen-Dunn (ed.), Models for the Perception of Speech and Visual Form, pp. 440-453. Cambridge: The M. I. T. Press.

Hughes, G. W. (1961). The Recognition of Speech by Machine. Technical Report 395, Research Laboratory of Electronics. Cambridge: M. I. T.

Jakobson, R., C. G. M. Fant, and M. Halle (1963). Preliminaries to Speech Analysis. Cambridge: The M. I. T. Press.

Kersta, L. G. (1962). "Voiceprint identification." Nature, 196, pp. 1253-1257.

Lehiste and Peterson (1959). "Vowel Amplitude and Phonemic Stress in American English." The Journal of the Acoustical Society of America, 31, pp. 428-435.

Lerner, R. M. (1964). "Band pass filters with linear phase." Proc. IEEE, 52, pp. 249-268.

Lindblom, B. (1963). "Spectrographic study of vowel reduction." The Journal of the Acoustical Society of America, 35, pp. 1773-1781.

Martin, T. B., A. Nelson, H. Zadell, and R. Cox (1966). Continuous Speech by Feature Abstraction. DDC No. AFAL-78-66-189. Camden, N. J.: RCA.

McCarthy, J. et al. (1964). The LISP 1:5 Programmers Manual. M. I. T. Press.

Peterson, G. E., and H. L. Barney (1952). "Control methods used in a study of the vowels." The Journal of the Acoustical Society of America, 24, pp. 175-185.

Reddy, D. R. (1966). An Approach to Computer Speech Recognition by Direct Analysis of the Speech Wave. Technical Report C549. Palo Alto, Cal: Stanford University, Computer Science Department.

Sebestyen, G. S. (1961). "Recognition of membership in classes." IRE Transactions on Information Theory, IT-6, pp. 44-50.

Schroeder, M. R. (1968). "Similarity measure for automatic speech and speaker recognition." The Journal of the Acoustical Society of America, 43, pp. 375-376.

Schearme, J. N., and P. F. Leach (1967). "Some Experiments with a simple word recognition system." 1967 Conference on Speech Communication and Processing, 6-8 November, Office of Aerospace Research, United States Air Force, Paper B17.

Stevens, K. N. (1968). (forthcoming) Study of Acoustic Properties of Speech Sounds. Technical Report, Department of Defense, Advanced Research Projects Agency. Cambridge, Mass.: Bolt Beranek and Newman Inc.

Stevens, K. N. and G. von Bismarck (1967). A Nineteen-Channel Filter Bank Spectrum Analyzer for a Speech Recognition System, NASA Scientific Report No. 2.

Stevens, K. N. (in press). "The quantal nature of speech: Evidence from articulatory-acoustic data." In E. E. David and P. B. Denes (eds.), Human Communication: A Unified View. New York: McGraw-Hill.

Teitelman, W. (1964). "Real Time Recognition of Hand Printed Characters." Proc. FJCC, Spartan Press.