1963 IEEE INTERNATIONAL CONVENTION RECORD
Part 7

42

SPEECH FROM STORED DATA

Franklin S. Cooper
Haskins Laboratories
New York, N.Y.

Summary

The retrieval of speech from analog storage (e.g., tape or disc recordings) is a well-established art. The alternative of reconstituting speech from stored data about the speech is being investigated in several laboratories and from various points of view. Practical applications are envisaged to man-machine communication, machine control systems, narrow-band communication, reading aids for the blind, and further research on speech itself. Descriptions (with recordings) are given of some experimental systems. The practical constraints on both storage and output are largely determined by the nature of the data that is stored.

Introduction

People have been generating speech from stored data -- as I am from these notes -- for quite a long time now. When that speech is really worth saving, we have quite adequate techniques for taking it down on magnetic tape or discs or sound film, and then playing it back. This kind of storage for speech data works well for reasonably long utterances, but there are times when we would like to have machines -- not people -- say something that has not been said before in just that way. A familiar example is the telephone time announcement, which you can hear merely by dialing MEridian 7-1212. You remember how it sounds: (Recording supplied by courtesy of Mr. D.P. Fullerton New York Telephone Company.) This is quite satisfactory speech, though a little slow, a little stilted, and the rigid grammatical framework is quite evident. How would the same word recordings sound if they were used in a different grammatical frame? Here are a couple of examples that were made by cutting apart the preceeding recording and reassembling the words in other sequences. (Recording) This illustrates one of the problems of generating speech messages by assembling recordings of short spoken utterances into new arrangements. For convenience, let us refer to this as compiled speech. It has obvious applications -- and is being widely used -- for such things as announcements of aircraft departures, stock quotations, and the like; indeed, it is probably the simplest way to generate new spoken messages on demand whenever the situation allows a restricted inventory of words and phrases, and when the unit recordings can be put into one, or a very few, fixed grammatical frames.

Example: Reading Machines for the Blind

Let us consider another specific example of speech from stored data before we turn to a more systematic analysis. The particular example is the development of reading machines for the blind.[1] The problem, in this case, is to start with ordinary printed books or typewritten materials and to convert the information on the printed page into signals that a blind man can easily understand. A great deal of effort has gone into developing devices that scan the printed line and convert the letter shapes into arbitrary acoustic codes, but this is not the part of the reading machine problem that is relevant for us; rather, let us consider how to design a high-performance type of reading machine[2] that will literally make the printed page talk.

Compiled Speech

One possibility is compiled speech; that is, the reading machine will compile the spoken message from pre-recorded words stored in a random-access memory. An optical character recognizer will also be needed to scan the printed page. Let me assume, though, that the character recognizer and the random-access memory are ready and waiting, so that all we must consider is what words should be recorded and how they should be spoken in order that the output speech from the machine will be fairly rapid and reasonably natural. Since the optical character recognizer can only tell us what letter sequence appeared on the page, we shall have only one recorded version for each

printed spelling of a word, regardless of how that word was used grammatically, or how this may affect its usual pronunciation in connected speech. It is not easy to make these word recordings fit together smoothly when one does not have control of the word order, as you have already heard in the re-arrangement of the time announcement; but the task is not impossible, either; and I should like to play some recordings prepared by Miss Jane Gaitenby and John Wadsworth of our Laboratory. This is connected text spliced together, word-by-word, from isolated word recordings that form part of the "vocabulary" of an experimental word reading machine for the blind. The passage is from Bertrand Russell. (Recording) I think you probably found the compiled speech quite intelligible, though the intonation and stress were a little odd here and there -- not as odd, let me repeat, as they would have been had the recordings simply been lifted from connected speech; such a recording would have been much stranger, and very difficult to understand. The recordings that we did use were made with careful attention to the typical pronunciation and intonation for the most probable form class.

I must admit that the text you heard was not quite a fair test of the method since it evaded one of the very difficult problems that arise in generating compiled speech, namely, the problem of spelling. Obviously, if the recorded vocabulary does not contain a word that the printed text calls for, one must do something to identify the word, and spelling is the evident answer. This problem was evaded in the text you heard, since the original text was paraphrased very slightly to avoid words that we did not have in our inventory of about 7,000 recordings and, also, five common suffixes were added to the storage because most of the recorded vocabulary appeared as root forms. The most necessary suffix was /-s/ which served double duty by making singular nouns into plurals and root verbs into the most common inflected forms. Similarly, the suffix /-d/ was used to form past tenses.

But the need to spell can not always be avoided; one must read texts just as they appear on the printed page and they will indeed have words that were not included in the original set of recordings, however extensive. Perhaps some rough figures on the relationship between the need to spell and the size of the recorded vocabulary would be of interest. I mentioned that we are starting with about 7,000 words; with this vocabulary we expect to have to spell a little less than 5%, or one word in twenty, from running text -- perhaps higher than many readers will tolerate. The spelling rate should drop to about 1% for a recorded vocabulary of 20,000 words, which is about a third of the word count for a desk-size collegiate dictionary, or about one-thirtieth of Webster's Unabridged. Add to this an almost unlimited number of proper names. So, you see, the spelling problem is a very serious one. There may be ways to ameliorate the situation by using special vocabularies for particular topics and, perhaps, by "spelling" on the basis of syllables rather than letters; both of these possibilities deserve additional research.

Synthetic Speech

Are there other ways to make a reading machine talk? Why not use synthetic speech as a way to avoid spelling and, at the same time, eliminate the large random-access memory unit? This is, in fact, a possibility that is being carefully considered -- and not only for reading machines for the blind; however, let us continue with the reading machine problem, for the moment.

We are assuming now that an optical character recognizer will be used to provide the identification of successive letters and punctuation marks in the printed text, and that the problem is to generate truly synthetic speech on the basis of this letter-by-letter information. You will see, in Fig.1, that the upper path from character recognizer to synthetic speech involves three successive transformations. The first is from English spelling to a phonemic transcription of the desired speech, the second is from this phonemic transcription to control signals, and the third uses these control signals to operate a synthesizer and thereby generate the actual speech waveform. If we were dealing with certain other languages, or if English spelling were less irrational than it is, we could skip the first step with a clear conscience. I shall skip it anyhow, partly because it is clear that compromise solutions could be worked out, and partly because I want to stay fairly close to speech as a topic. In the worst case, inadequate transformations between spelling and phonemic transcriptions will only result in a higher percentage of bizarre pronunciations.

Let us go on to the next step and consider ways to convert a phonemic transcription into speech. The basic information that makes this possible comes from a long series of experiments on the acoustic cues that listeners use in perceiving the successive phonemes of speech. This work has been summarized[3] as a set of minimal rules for speech synthesis -- minimal in the sense that the rules have been kept as simple as possible, if the speech is to be intelligible -- though not necessarily very natural. These rules specify the formant frequencies and the transitions from one set of frequencies to another. Figure 2 shows the general structure of these rules, and illustrates how they apply to the synthesis of the word "labs". The figure shows the phonemic transcription and the simplified sound spectrogram that we get when the explicit rules entered in the boxes at the top are applied to the four phonemes in succession. I should point out that not only must several sub-rules be used simultaneously for each phoneme, but also that there are interactions between successive sets of sub-rules, as one proceeds from one phoneme to the next. It is, in fact, these interactions that convert the sequence of distinct phonemic elements into the flowing pattern so essential for speech -- which is another way of saying that you can't just string together little unit sounds for the phonemes and get intelligible speech.

When we have the spectrographic patterns, like the one for "labs" at the bottom of the figure, it is easy to generate the audio waveform from it, or from equivalent control voltages. This is how the pattern for "labs" sounds as synthesized on a pattern playback: (Recording) The particular synthesizer that was used in this case converts the pattern directly into sound by means of a tone wheel and photoelectric pickup. Such a playback device is inherently monotone. Here is another example of the same kind, but a longer utterance. Figure 3 shows the pattern that was generated by rule, and this is how it sounds: (Recording) If we use a different kind of pattern playback -- essentially a photoelectrically controlled vocoder -- and if we provide pitch information, we can get a result that is more nearly natural. Here is an example, though I should tell you that the pitch information was not generated by rule and the spectrum information is not strictly "cookbook" either, though the patterns (Fig.4) were drawn by Pierre Delattre without any reference whatever to spectrograms of real speech: (Recording)

The rules for synthesis can be phrased in somewhat different ways to adapt them to other types of synthesizers; thus, the transitions from one set of formant frequencies to a following set -- one of the notable characteristics of the spectrographic patterns -- can be achieved in a "hop-along" manner by using the phoneme information to specify target frequencies to which the formants will move from wherever they may have been for the preceeding sound, and at rates of change that are specified. One must also specify steady-state durations and the changes in intensity, voice pitch, and buzz vs. hiss as the sound source. One early device of this kind used RC circuits, set by hand, to determine targets and rates of change, and a stepping switch to move from one acoustic epoch to the next. Here is an example of its speech: (Recording)

The kinds of operations I have just described lend themselves to computer implementation, and John Kelly and Louis Gerstman at the Bell Telephone Laboratories have made a considerable study of computer-generated speech.[4] I am much indebted to our Chairman for several examples of this work. Kelly and Gerstman used punched cards to call out the sequences of control voltages for the individual phonemes, and so could generate any particular message by sorting their cards into the sequence required by the phonemic transcription and then feeding the cards into the computer, which not only calculated the intermediate frequencies between one set of targets and the next, but also computed the output waveform. The information stored on the cards was essentially the kind of formant-frequency target data that I have already mentioned, together with instructions about how, and how fast, the formant frequencies should change in proceeding from target to target. This kind of information corresponds to rules for synthesis and is punched just once for each phoneme. Kelly and Gerstman found it useful, as we did also, to make ad hoc adjustments to the voice pitch and the relative timing of events, in order to control the intonation and rhythmic characteristics of the speech. Here are some examples of computer-generated speech made at the Bell Telephone Laboratories:

(Recordings)

Hybrid Methods for Generating Speech

We have discussed, thus far, two general methods for generating speech from stored data: we can make speech by compiling it from voice recordings of words or phrases, or we can synthesize speech sounds from information about successive linguistic units such as phonemes, though we may have trouble in guessing the phonemes from the conventional spelling. Neither of these methods has been carried to the point of providing working devices, except those that announce time, plane departures, and the like. But there is no reason why general purpose devices could not be developed.

Before we talk about the practical problems of doing so, however, let us look at some hybrid methods that combine storage with synthesis. Figure 5 shows (at top and bottom) the kinds of systems we have been discussing, and (in the center) some intermediate methods. One of these solves the problem of converting spelling into phonemic transcriptions by a look-up operation in a stored dictionary that contains the pronunciation -- in phonemic form -- for each spelled word in the language.

Another method shown in the figure is to look-up stored tables of control voltages that would operate a speech synthesizer. These could be stored in either analog or digital form, and of course the exact nature of the stored data would depend on the kind of synthesizer that one is using. If the synthesizer were so good that one could not tell its output from real recorded speech, then the synthetic speech generated by this procedure should sound very much like compiled speech -- even better, since one could avoid some of the pitch and intensity jumps inherent in a recorded vocabulary. Indeed, it is possible to make synthetic speech that is almost indistinguishable from the real speech on which it is modelled, provided one does a very careful job of preparing the control signals. I should like to play a tape for you that was made by John Holmes of the Joint Speech Research Unit while he was a guest in Gunnar Fant's laboratory at the Royal Institute of Technology in Stockholm. The recording is one that Fant presented to the Acoustical Society about two years ago.[5] You will hear first the spoken sentence, then the synthetic one prepared on the OVE II synthesizer. (Recording) The significant point for us is, of course, that it is possible -- though not easy -- to generate very high quality synthetic speech[6]. You would not expect, of course, such smoothly flowing speech as you have just heard if the sentence had had to be composed of "standardized" sets of control signals for the successive words.

Comparison of Methods

What comparisons can we make among these various ways of generating speech? For one thing, the method of pure synthesis -- the top path in Fig. 5 -- avoids spelling completely and requires rather little storage, only enough for the rules for synthesis. It does require an analog synthesizer, a thorough understanding of speech, and some rather sophisticated logic; that is to say, not very much hardware but a lot of know-how. At the other extreme, -- the bottom line in the figure -- the only major requirement is a memory, though a very big one. Of course, the device will have to spell some words and it will put odd intonations on others. The hybrid routes trade memory capacity for design sophistication in increasing degree. Here are some rough estimates of the memory capacities likely to be required by the four methods, when they have about equally large vocabularies (20 thousand words, or a spelling rate of about one percent): stored recordings of words, about 400 million bits; stored control voltages, about 10 million bits; stored phoneme equivalents, about 1.5 million bits; rules for synthesis, about 30-50 thousand bits. I have put the memory requirements in digital terms because it may well be that digital memories will be chosen simply because they are available; roughly the same relative sizes would apply to analog memories, though they could almost certainly be smaller than their digital counterparts.

We have dealt thus far with a single application -- reading machines for the blind. Let me summarize briefly the main points: first, we assumed one particular kind of input, namely, the spelled words that appear on a printed page; second, we have considered two principal kinds of speech output -- compiled or synthetic -- though there are hybrid methods that may have important practical advantages. The choice of method will inevitably be a

compromise among such factors as the quality of the final speech, the allowable spelling rate, the size of the memory unit, and the amount of development work required to substitute sophistication of design for mere quantity of hardware.

I think it is worth noting that we did not pick an easy example. The reading machine problem is a difficult one for at least three separate reasons: one is the fact that the device must deal with an open-ended and very large vocabulary -- essentially, anything that one is likely to find on a printed page. This is the origin of the spelling problem and of the practical need to store control signals, or even phonemic transcriptions, in order to keep the storage requirements within reasonable bounds. A second reason is that the grammatical frame is just as variable as the language will allow. A third difficulty is that the input information -- the word spellings and punctuation -- are only loosely related to the sounds that must be generated.

General Case: Speech from Stored Data

Let us turn now to some other problems and see what kinds of input signals they may involve, and what other methods might be used to generate the required speech. Figure 6 suggests some of the possibilities; clearly, there are too many to deal with systematically, which is one reason I have limited the discussion thus far to a specific example. The block diagram across the top of the figure stresses one central point: we must somehow convert signals that are organized in machine terms into other signals that are structured in speech terms. The box in the center labelled "Converter" has this function of translating from a machine-organized world to a speech-organized one. Its operation may involve several stages of processing the signal information, or of using input signals to locate stored speech data, or a combination of processing and retrieval. One way to characterize what goes on in the Converter is to specify the kind of language unit that it employs in controlling the speech output device. Often, the Converter is simply a random-access memory; in these cases, we are concerned with the kind of unit that is stored in the memory. This might range from a phrase, or even a complete message, down through smaller and smaller units such as words, syllables, phonemes, or even brief time segments of speech. The choice of units obviously affects the speech generating operations toward the right of the diagram; just as obviously, the choice of units is determined by the characteristics of the input devices to the left.

I have indicated some examples of these input devices in the lefthand column, and some characteristics of the signals they generate, in the second column. The signal units may be either alphanumeric characters or voltage pulses (or levels) on control lines. The message units will ordinarily be letters or words, as in the case of the reading machine, or codes for complete messages, as in the case of a process controller that is warning of excessive temperatures. The message set may be quite small for the process controller, or fairly large for a teletype communication link, or completely open-ended in the case of a reading machine. Likewise, the format or grammatical framework may be quite restricted or totally unrestricted.

The figure is intended to suggest some of the possibilities at each stage between input and output, though obviously not all combinations are plausible; each type of input device will weave its own preferred path among the options. Some of the routes for reading machines, for example, would appear as in Fig. 7. The signal unit will obviously be the letter. Within the converter, these may either be grouped into words and then used to look-up stored recordings for sequential playback, or the letter units may be used to get phonemes and these in turn to generate control signals for synthesis.

Example: Computer that Talks

Another example might be an application in which we wish a computer to talk back to us as we de-bug a program. This may require only a modest inventory of words, which could perhaps be stored as voice recordings. On the other hand, we may wish to be free from vocabulary constraints and also to use the computer's own memory, provided the storage requirements are not excessive. Figure 8 shows a possible approach. The use of half-syllables or sub-word segments of roughly this size, allows a rather large vocabulary to be put together from a much more restricted number of storage units. By storing the segments in terms of control signals for a formant-type synthesizer, one gains two important advantages over the use of stored voice

reccrdings: first, the number of bits is very much less, by a factor of about forty, and second, the resulting synthetic speech is quite satisfactory whereas it is extremely difficult to get a satisfactory output from real-speech segments that are shorter than whole words. A rough estimate puts the number of half-syllable units at perhaps a thousand, each requiring about two hundred bits for the storage of control signals. Thus, a total storage of about a quarter of a million bits would let a computer talk ad lib -- if that is desired. Disc files would provide adequate access and could accommodate the additional bits without undue strain.

A development along these general lines was described by Robert Walker and his group at IBM Research (San Jose) in a paper before the Seattle meeting of the Acoustical Society of America last November.[7] The sub-word segment that they used was not the half-syllable, but rather a two-phoneme sequence -- a diphone, in their terminology -- that begins with the middle (or steady-state portion) of one phone, includes the transition to the next, and ends in the middle of the second phone. These units are stored as control signals for synthesis and can be assembled by the computer into speech sequences. It will, of course, be desirable to add some further control signals to make the stress and intonation as natural as possible. Something over 800 diphones at about 400 bits each puts their present storage requirement, exclusive of stress and intonation, at about a third of a million bits, though this can probably be reduced by a factor of 2 to 3.

The diphone as a unit is the synthetic counterpart of the dyad, proposed by Gordon Peterson several years ago as a means of obtaining short spoken segments that could be assembled into speech sequences. The great practical difficulty with dyads is that one must have so many of them, since both formants and voice harmonics must match at the mid-phone junction. Peterson, Wang and Sivertsen estimated that about 8000 dyads would be needed, if stress and intonation are to be preserved.[8]

Example: Bandwidth Compression System

Still another example of the interplay of requirements in a practical situation is shown in Fig. 9. We are dealing here with the receiving end of a bandwidth compression system applied to a voice-communication link. The distinguishing characteristic of this compression system is that it does not, like so many other systems, send a continuously varying description of the input speech, but rather a sequence of discrete, categorical codes. These are, in fact, the addresses of speech segments that are already in storage at the receiving end. Thus, the digital addresses call from storage a sequence of brief time segments of the acoustic spectrum and send them through a channel vocoder to generate the output speech. An obvious advantage is the high compression factor, better even than the type of control signals that we have discussed thus far. The major unknown is the number of segments that must be stored in order to get acceptable speech, because this will determine the size and cost of the terminal equipment. This system[9] is under active development by Caldwell Smith at the Air Force Cambridge Research Laboratory, and I am indebted to him for Fig. 10 and a recording illustrating one of the many modes in which his Voice Data Processor can operate. (Recording)

Constraints and Possibilities

We have not exhausted, by any means, the system configurations that people are working on, but let us turn to a more restricted view of the right-hand side of the basic diagram of Fig. 6. We have already had examples of the constraints that exist among the units of the conversion operation, shown in the center column. For one thing, the available input may not be related in a simple way to the best choice of unit for the output, so that complicated conversions may be required for this reason. Then, too, the number of units to be stored will vary enormously, depending on the choice of unit. A detailed study of this relationship was made by Gordon Peterson and Eva Sivertsen.[10] One of their major conclusions was that the number of units becomes very large whenever the choice of unit differs from one of the "natural" linguistic units (e.g., words, syllables, and phonemes).

The two columns on the right, dealing with speech data and output devices, look deceptively simple. I am sure you realize that there are a number of subentries to be made in each column. I should like to mention two or three entries under "Control Signals" and "Speech Synthesizers".

Synthesis by Analogs of the Vocal Tract

We have talked, thus far, as if formant information, and a formant-type synthesizer, were the only possibilities. They have had the most attention, to be sure, but some very significant work is being done on synthesizers that are modeled on the vocal tract, in the sense that the control signals cause the synthesizer to change its transmission properties in the same way that the human vocal tract changes its transmission when it changes shape, due to movements of the tongue, jaw and lips. An intensive program of research along these lines is underway at MIT on a dynamic vocal tract analog[11] (DAVO); here is a tape of a little ditty that it sings: (Recording) My thanks to Kenneth Stevens, Arthur House, and the rest of the MIT group for the use of these recordings.

Why should one go back to something as clumsy as the human vocal tract? The point, of course, is that the sounds a person can make are severely constrained by the possible shapes of his vocal machinery; so, by building these constraints into the device, one can hope for simpler control signals and, quite possibly, for more smoothly flowing speech as well.

The constraints could, of course, be incorporated into a computer program just as well as into hardware. This is a line of investigation[12] that John Kelly at the Bell Telephone Laboratories has been following; that is, he starts with a phoneme transcription and, by taking account of the articulatory shapes that correspond to it, uses a computer to generate the audio waveform. This is an ambitious undertaking, and the results should be judged accordingly. My thanks again to Ed David for this recording of a computer trying to mimic our own inimitable way of shaping sounds: (Recording)

I should like to mention one more possibility, though it is a long way from realization. There is a substantial amount of evidence that the significant aspect of speech is not so much the shape of the articulatory tract as the sequence of muscle contractions which produce that shape -- or, even earlier, the neural motor commands that activate the muscles of the articulatory apparatus. If it does indeed turn out that this is a particularly simple way to describe speech, one would certainly wish to build a synthesizer analogous to the vocal tract, and controlled by signals that mirror the motor commands.[13]

In Conclusion

In conclusion, we have looked at a number of the interrelations that exist between the kinds of signal units involved in going from machine language to human language, and we have seen how these are related to the characteristics of the final output speech. Here is the main crossroad of the diverse ways of generating speech from stored data.

Teaching machines to talk is a very lively area of research and one in which the study of practical problems feeds information back into our body of knowledge about speech fundamentals. It is a research area in which computers will play an increasingly significant role, sometimes simulating the total speech process, though more commonly leaving the actual waveform generation to analog devices -- including, even, the two-legged variety.

Notes and References

1. The research reported here on reading machines for the blind was supported by contracts with the Research and Development Division of the Prosthetic and Sensory Aids Service, Veterans Administration.

2. F.S. Cooper, "Toward a High-performance Reading Machine for the Blind" in Human Factors in Technology, Ed. E. Bennett, J. Degan, and J. Spiegel, McGraw-Hill, New York, 1963.

3. A.M. Liberman, Frances Ingemann, Leigh Lisker, Pierre Delattre, and F.S. Cooper, "Minimal Rules for Synthesizing Speech, J. Acoust. Soc. Amer. 31, No. 11, 1490-1499, 1959. (This paper contains a full account of the procedures for synthesis by rules and references to earlier publications on research tools and results.)

4. J.L. Kelly, Jr. and L.J. Gerstman, "An Artificial Talker Driven from a Phonetic Input", J. Acoust. Soc. Amer. 33, 835(A), 1961; "Digital Computer Synthesizes Human Speech", Bell Lab. Rec. 40, 216-218, 1962.

5. C.G.M. Fant, J. Martony, U. Rengman,

A. Risberg, and J.N. Holmes, "Recent Progress in Formant Synthesis of Connected Speech", J. Acoust. Am. 33, 834-5(A), 1961.

6. C.G.M. Fant, J. Martony, U. Rengman, and A. Risberg, "OVE II Synthesis Strategy". Proc. Speech Communication Seminar, Royal Institute of Technology, Stockholm, 1962 (In press).

7. S.E. Estes, H.R. Kerby, H.D. Maxey, and R.M. Walker, "Speech Synthesis from Stored Data", J. Acoust. Soc. Am. 34, 2003(A), 1962.

8. G.E. Peterson, W.S-Y. Wang, and Eva Sivertsen, "Segmentation Techniques in Speech Synthesis", J. Acoust. Soc. Am. 30, 739-742, 1958.

9. L.P. Schoene, H.A. Straight, O.C. King, "Design and Development of a Digital Voice Data Processing System". Report No. AFCRL-62-314 by Melpar, Inc., 1962.

10. G.E. Peterson and Eva Sivertsen, "Objectives and Techniques of Speech Synthesis", Language and Speech, 3, 84-95, 1960;

Eva Sivertsen, "Segment Inventories for Speech Synthesis" Language and Speech, 4, 27-61, 1961.

11. J.B. Dennis, "Computer Control of an Analog Vocal Tract", Proc. Speech Communications Seminar, Royal Institute of Technology, Stockholm, 1962 (In press).

12. J.L. Kelly, Jr. and Carol Lochbaum, "Speech Synthesis", Proc. Speech Communication Seminar, Royal Institute of Technology, Stockholm, 1962 (In press).

13. A.M. Liberman, F.S. Cooper, K.S. Harris, and P.F. MacNeilage, "A Motor Theory of Speech Perception", Proc. Speech Communications Seminar, Royal Institute of Technology, Stockholm, 1962 (In press);
F.S. Cooper, A.M. Liberman, L. Lisker, J.H. Gaitenby, "Speech Synthesis By Rules", Speech Communication Seminar, Royal Institute of Technology, Stockholm, 1962 (In press).

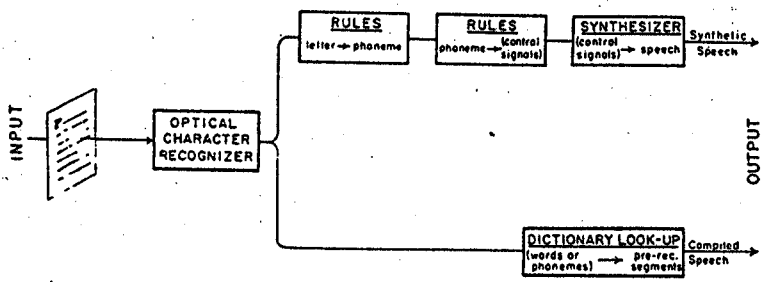

Fig. 1. Block diagram for high-performance reading machine for the blind that generates synthetic speech by rules or compiled speech from stored recordings.

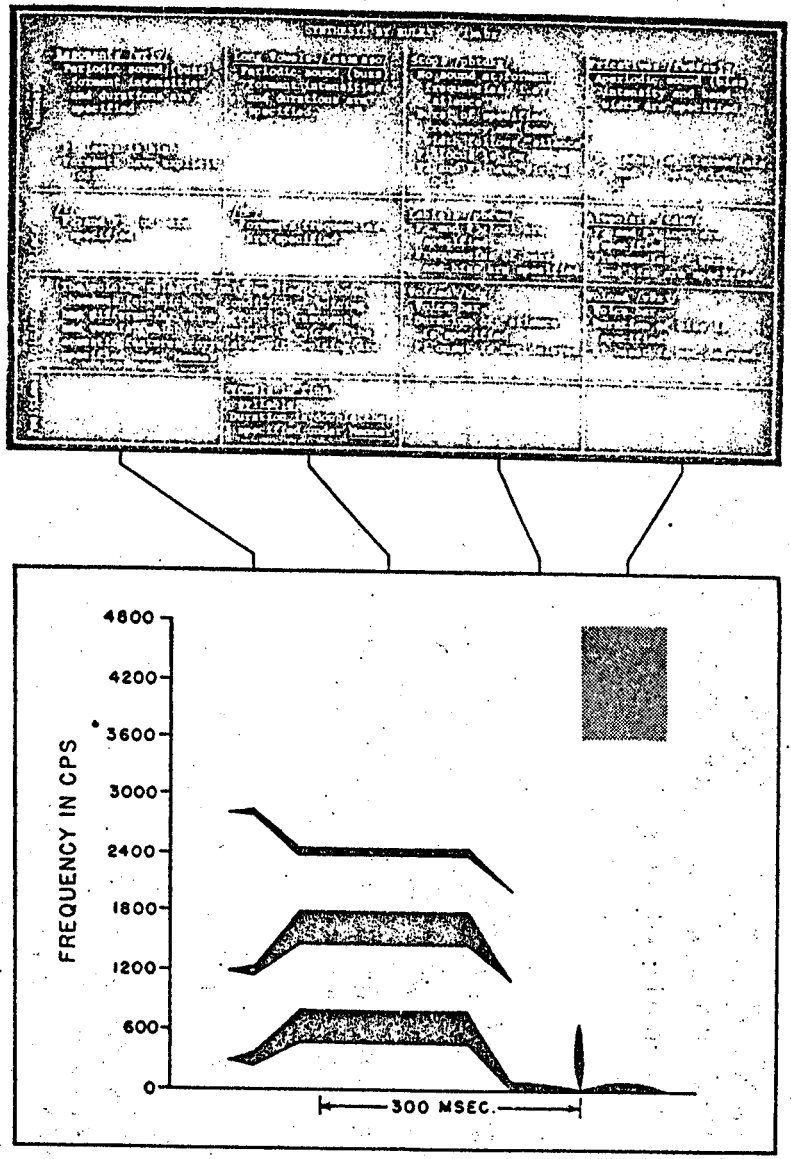

Fig. 2. Table illustrating the rules for synthesizing the word "labs", and the spectrographic pattern derived therefrom. (Reproduced by courtesy of the Journal of the Acoustical Society of America)

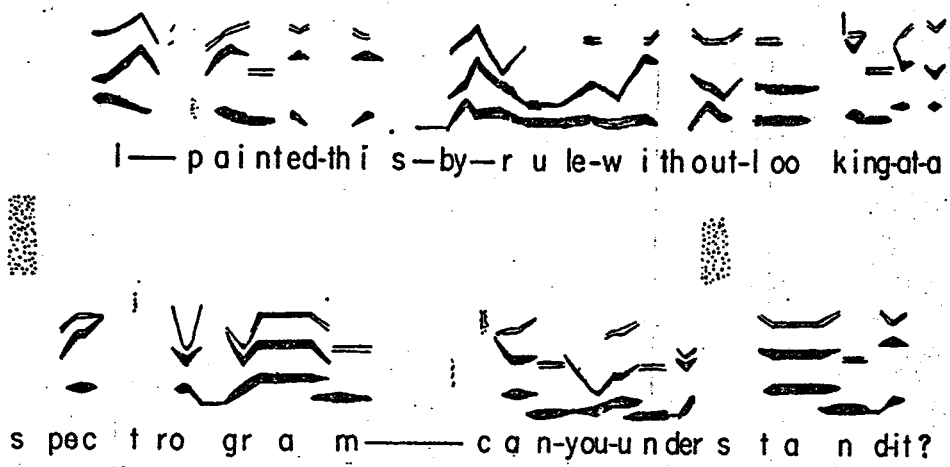

Fig. 3. Spectrographic patterns made by the minimal-rules-for-synthesis procedure. (The sound recording presented with the paper was generated from this set of patterns.)

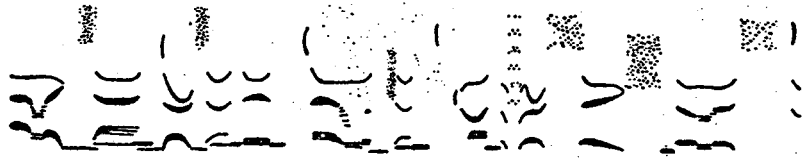

Fig. 4. Spectrographic patterns drawn by hand on the basis of a knowledge of the acoustic cues for speech perception. Pitch is controlled (during the conversion of pattern to sound) by the hill-and-dale patterns along the top; buzz-hiss switching is determined by the on-off channel just beneath the pitch patterns.

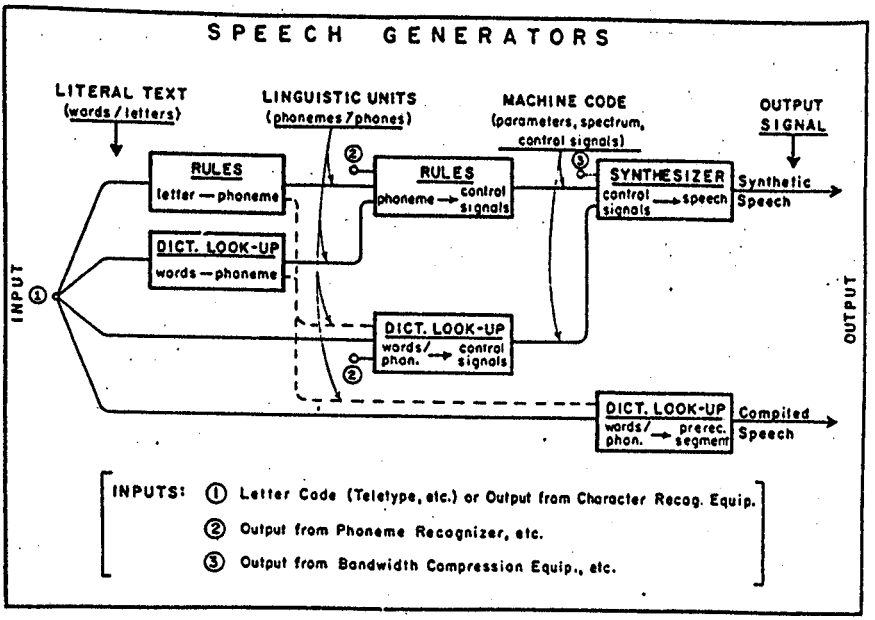

Fig. 5. Block diagram showing various hybrid methods of generating synthetic and compiled speech in a reading machine for the blind.

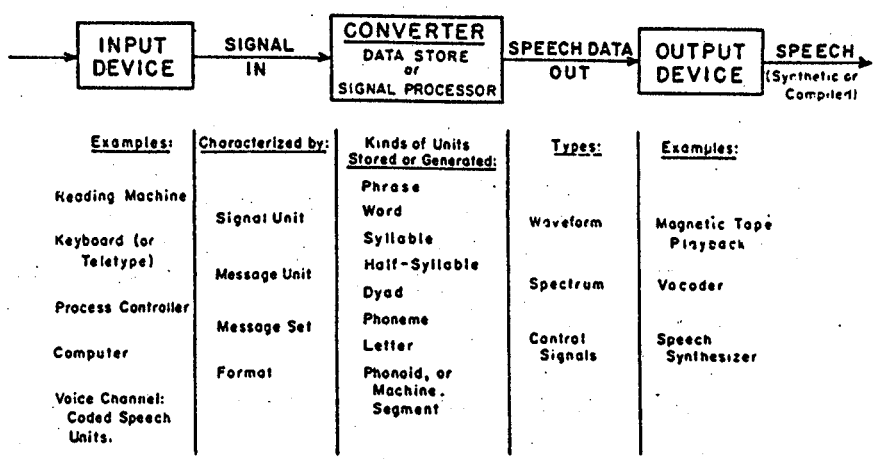

Fig. 6. The general processes of converting from machine-organized data to speech-organized data.

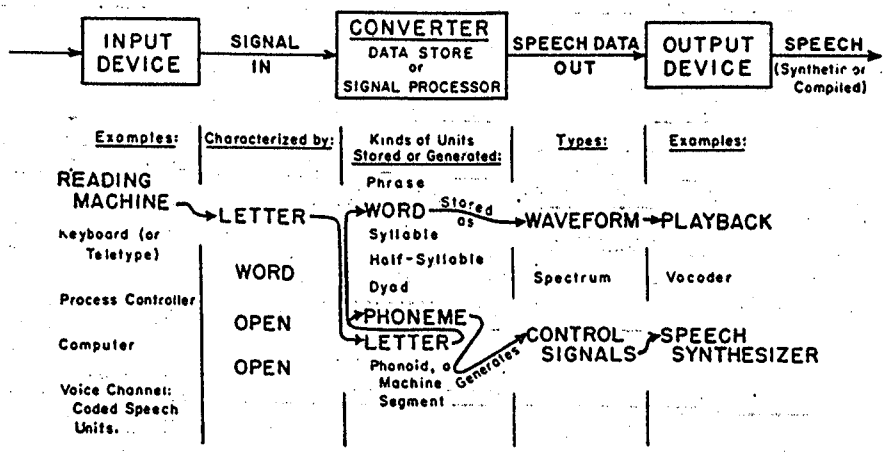

Fig. 7. Specialization of the general processes shown in Fig. 6 to reading machines.

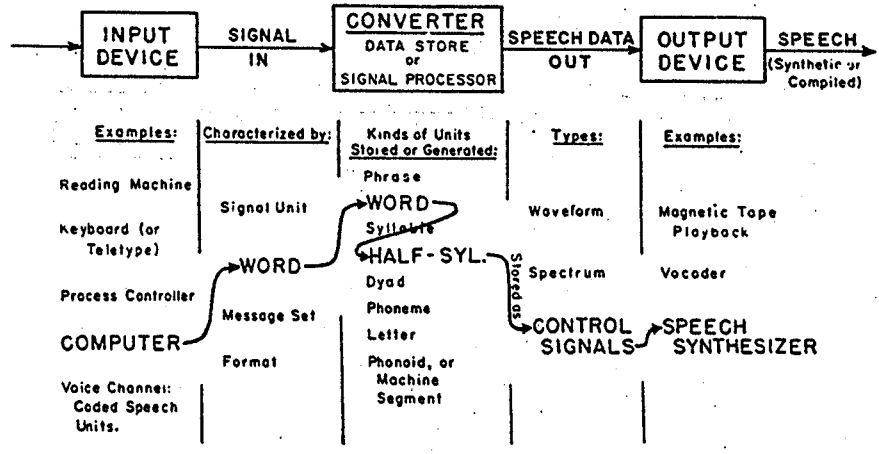

Fig. 8. Specialization to computer-controlled speech.

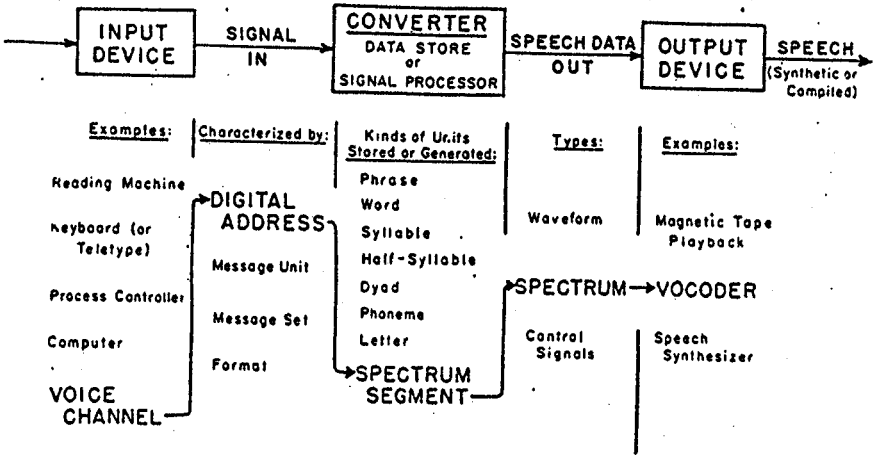

Fig. 9. Specialization to bandwidth compression in a voice channel.

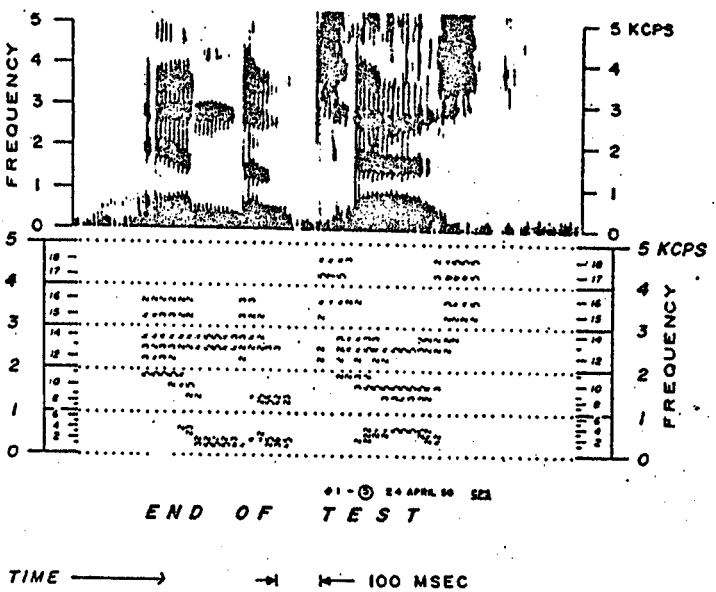

Fig. 10. Analog and digital spectrograms that form the basis for a band-width compression system using addresses of brief spectrum "events."