UDC 534.7:681.32

Advances in Speech Recognition Technologies

●Shinta Kimura

(Manuscript received July 9, 1999)

This paper describes the research and development activities for speech recognition conducted in the 1990s at Fujitsu Laboratories Limited. Our interests have been focused on extending the functions and performance of speech recognition technologies developed in the 1980s. Advances in small implementations of speech recognition, recognition of continuous speech, and recognition of speech in noisy environments are described.

1. Introduction

Ever since the invention of the computer, scientists and engineers have dreamt of speech recognition and text-to-speech conversion technologies that would enable humans and computers to communicate with each other through spoken language. In the early stage of computer development, everyone thought that these technologies would be relatively easy to achieve; but engineers met with many difficulties. Around 1980, special hardware systems for speech recognition and text-to-speech conversion that had very restricted functions appeared in the market. However, since 1990, their performance has greatly improved, and because of advances in microprocessors and digital signal processors (DSPs) they no longer require special hardware.

This paper describes our activities in this area at Fujitsu Laboratories Limited. First, Chapter 2 introduces a small implementation of speech recognition and provides background information for Chapters 3 and 4. Then, Chapters 3 and 4 describe some advances in continuous speech recognition and noise processing for speech recognition, respectively. In the references, Ref. 1) gives a wide range of basic knowledge about speech recognition technology and Ref. 2) provides some useful, up-to-date information.

2. Small-size speech recognition

2.1 Voice dialing for cellular phones

In the 1980s, speech recognition required a rack that was 6 feet high, 19 inches wide, and full of equipment that included a mini-computer and special purpose hardware. But in the 1990s, advances in recognition algorithms and semiconductor devices drastically reduced the size of the equipment. **Figure 1** shows the first cellular phone in Japan to feature a speech recognition function. A very compact algorithm for isolated-word recognition based on template matching is used to realize voice dialing in this phone. Section 2.2 describes this algorithm to give the reader a basic understanding of speech recognition.

2.2 Algorithm for isolated-word recognition

2.2.1 Structure

Figure 2 shows the structure of isolated-word recognition based on template matching. The system has two phases: registration and recogni-

Figure 1
NTT DoCoMo F206: cellular phone with voice dialing.

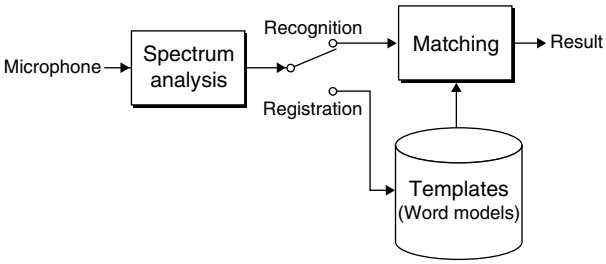


Figure 2
Structure of isolated-word recognition based on template matching.

tion. In both phases, the spectrum of the input speech is analyzed using a frame-by-frame method of FFT (fast Fourier transform) or LPC (linear predicative coefficients) analysis. The length of each frame is 20 to 30 ms, and the spacing between frames is 10 to 20 ms. In the registration phase, the analysis result for a spoken word is stored as a word model. For speaker-dependent systems, word models are generated from utterances by the speaker. For speaker-independent systems, they are generated from a large set of utterances by many speakers. In the recognition phase, by using non-linear time axes alignment, the input speech is compared with each of the registered word models to obtain similarity values. The word of the most similar word model is then selected as the recognition result. The merit of this method is that the algorithm for registration

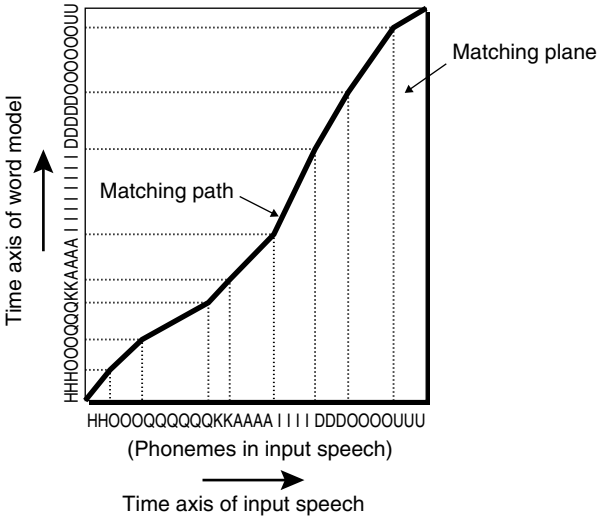


Figure 3
Non-linear time axes alignment.

and recognition is quite compact and does not require much resources for computation and memory. The demerit is that all the words to be recognized must be uttered and registered before recognition becomes possible.

2.2.2 Non-linear time axes alignment

Multiple utterances of the same word, even by the same speaker, will have different durations. Furthermore, the component phonemes of such utterances will be spoken at differing speeds. For example, one utterance might be lengthened at the beginning, while another might be lengthened at the end. This type of elasticity can be compensated for by using non-linear time axes alignment.

Figure 3 shows how this method works. The horizontal axis shows the progression in time of phonemes in a spoken word (Hokkaido), and the vertical axis shows this progression in time for a word model being compared with the word (in this case, the word model for "Hokkaido"). The rectangular area defined by these two axes is called the "matching plane," and the path from the lower left corner to the upper right corner is called the "matching path." This path represents the correspondence of non-linearity between the spoken word and the word model; it can be calculated

using dynamic programming at a relatively low computational overhead. The similarity between an input utterance and a word template is calculated by accumulating the similarity values between the input speech and a word model along the matching path.

3. Continuous speech recognition

3.1 Address recognition

3.1.1 Problem

The task of Japanese address recognition is used here as an example to explain continuous speech recognition. Japanese street addresses have the following three-layer structure: prefecture (ken) – city (shi) – ward (ku/cho). All Japanese addresses fall into one of about 80 000 of these three-layer groups.

The simplest approach for applying speech recognition to Japanese addresses is to treat each address as an isolated word; that is, to register all of the addresses as isolated-word recognition models. Now, the amount of computation required for isolated-word recognition is proportional to the area of the matching plane and the number of registered words. However, for Japanese addresses, the area of the matching plane is nine times larger than that of ordinary isolated-word recognition because the speech input and the word model each contain three words. Also, the number of registered words (80 000) is 800 times bigger than in a 100-word recognition task (100-word recognition can be easily implemented by realtime software for an MPU using a straightforward algorithm and is a commonly used benchmark recognition task). In total, this approach requires 7200 (9×800) more computations than an ordinary 100-word recognition task. Therefore, one of the problems to be solved for continuous speech recognition for Japanese addresses is how to reduce the computational amount.

3.1.2 Approaches

- **Tree structure of word models**

Since Japanese addresses always proceed in the order of prefecture, city, ward, it is only necessary to compare the word models for the prefectures with the first part of the spoken input. In this way, the 80 000 word models for Japanese addresses can be converted into a tree structure, part of which is shown in **Figure 4**. The matching paths now go through multiple matching planes that correspond to individual words. Such a tree structure can reduce the area of matching planes and, therefore, the computational amount by two thirds. This, however, still leaves too much computation for real-time recognition.

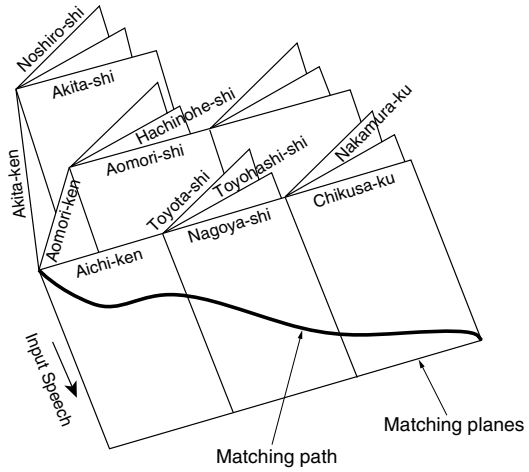


Figure 4
Matching between input speech and tree structure of words in Japanese addresses.

- **Best first search**

To further reduce the amount of computation required to find the best matching path, we introduce a heuristic method called the "Best First Search." This method is based on the rule that whenever the matching process reaches a branching point in the tree, the matching in the most promising branch is calculated first. This method can find the best-matched path with considerably less computation than a method which checks all the branches in the tree. By using this method alone, the computational amount can be reduced by a factor of 100. By combining it with the tree structure, therefore, we can reduce the computation amount to 1/300 of the amount required for an exhaustive search and thereby realize a practical real-time system for address recognition.

Table 1
Modify and modified grammar.

Phrase group	Example member(s) of phrase group	Modifiers of phrase group
Iku (go) verb phrase	Iki-masu (go)	Person does To place
Person does	Watashi wa (I)	–
To place	Kouen e (to park) Gakkou e (to school)	Adjective
Adjective	Chikaku no (nearby)	–

3.2 Recognition of natural sentences

3.2.1 Problem

Japanese sentences can theoretically be represented by a tree. However, the tree required for natural sentences is too big to be built in memory. Therefore, the recognition should be done by building a partial tree using a grammar. Some examples of the grammars used in speech recognition are context free grammar (CFG), modify and modified grammar, and N-gram grammar. We give an outline of modify and modified grammar and N-gram grammar below (these grammars are also called "sentence models").

3.2.2 Approaches

- **Modify and modified grammar (case-flag propagation)**

We will now describe a method for building a tree of Japanese sentences using the modify and modified grammar, using the example Japanese sentence "Watashi wa chikaku no kouen e iki masu" (I go to a nearby park). In this sentence "watashi wa" is "I," "chikaku no" is "nearby," "kouen e" is "to park," and "iki-masu" is "go." "Watashi wa" and "kouen e" modify "iki-masu," and "chikaku no" modifies "kouen e."

The modify and modified grammar represents the modification relations between short Japanese phrases called "bunsetsu." **Table 1** shows the grammar for four phrase groups. The left column shows the example phrase groups, the center column gives one or two members of each group, and the right column shows the modifiers of the example phrase groups. For example, Kouen e (To park) and Gakkou e (To school) belong to the group To place.

The case-flag propagation method is introduced for generating sentences using the modify and modified grammar. In Japanese, the order of phrases in a sentence is relatively unrestricted. For example, "Watashi wa gakkou e iki masu" and "Gakkou e watashi wa iki masu" are both grammatically correct and have the same meaning. The case-flag propagation method can treat this kind of phenomenon within the framework of continuous speech recognition. A case-flag is a flag set for each of the short phrase groups, for example, To place and From place. The case-flag propagation method generates sentences from end to beginning while manipulating case flags. **Figure 5** shows an example generation of a sentence using case-flag propagation and the grammar shown in Table 1. Figure 5 indicates that Japanese sentences can also be represented with a tree structure.

Next, the method used to manipulate case flags is explained. This method proceeds from the sentence end to the sentence beginning. Because the verb phrase Iki-masu (go) can be modified by the Person does and To place phrase groups, the case flags of these two phrase groups are set (①). If a To place member comes before the Iki-masu verb phrase, the flag for the To place group is canceled. Since Table 1 shows that To place is modified by an adjective, the flag for Adjective is set. In this case, the flag for Person is not affected (② and ③). In this way, this method decides which phrases can come before the current phrase by setting and canceling the flags. The flags are canceled according to the following rules:

1) A verb phrase is not modified by multiple phrases of the same group.
2) If there are more than two modification relations in a sentence, no two of them can cross each other on the flag propagation chart.

- **N-gram grammar**

The major problem in a system that uses a rule-based grammar, for example, the modify and

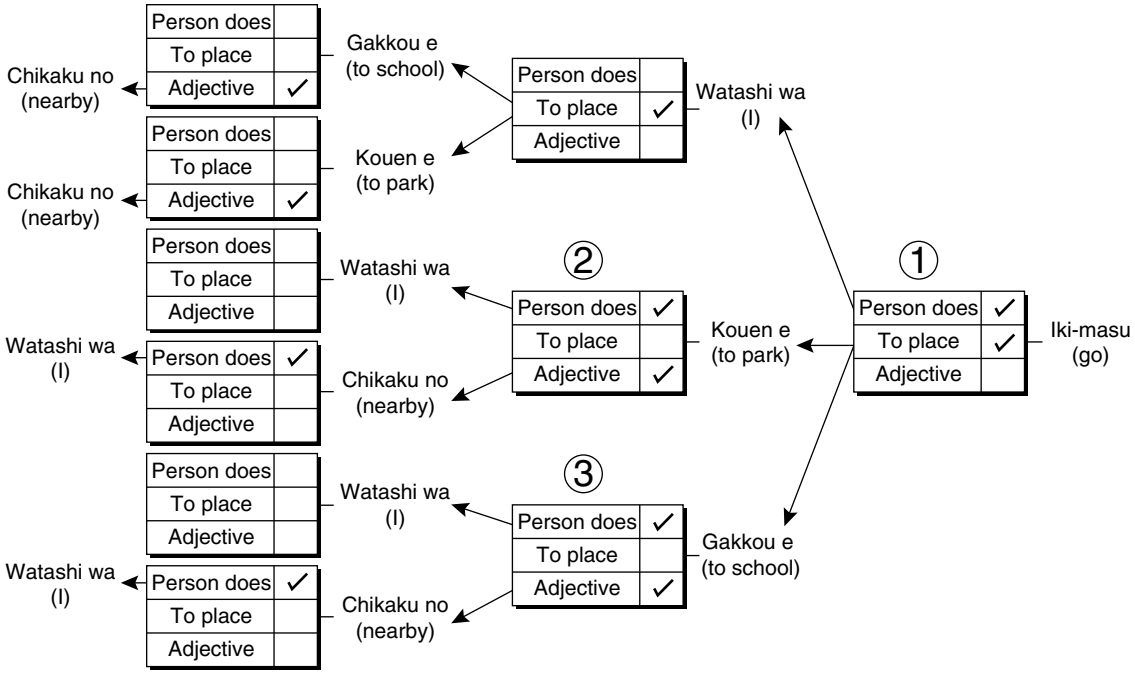


Figure 5
Right-to-left sentence generation by case-flag propagation.

modified grammar and context free grammar, is that engineers must write all the rules manually. If the system treats a very small portion of the language, for example, the language needed for a hotel reservation service, engineers might be able to write all the rules for the service. However, a dictation system must treat the entire written language, and engineers cannot write all of the rules for the entire written language. To solve this problem, a framework for automatic grammar generation is necessary. One such framework, N-gram grammar, is based on statistical values for N-word sequences in a very large set of sentences called a "corpus."

The main feature of N-gram grammar is that it can be automatically extracted from a corpus. If N is 1, the model is called a uni-gram (uni-grams are single words, so the set of uni-grams is simply the vocabulary of the system). Each uni-gram represents the appearance probability of a word in the corpus. A bi-gram (N = 2) represents the probability that a word appears after a specific word. A tri-gram (N = 3) represents the probability that a word appears after a specific two-word sequence. In the recognition process, the system also generates a left-to-right sentence tree in which each branch is given a language score according to the rules of N-gram grammar. The Best First Search can be also used in this tree search for realizing real-time recognition.

3.3 Evaluation results

Table 2 shows the isolated-word recognition results for 100-word and 1000-word vocabularies and the results of some continuous speech recognition experiments. The word perplexity is the average number of candidate words the grammar will predict for the given type of word recognition and size of vocabulary; it is one of the measures used for representing the complexity of a task.

In isolated-word recognition with a 100-word vocabulary, 100 words are predicted before recognition. For the 60 000-word dictation task, the number of predicted words is 60 000. But, because there is a wide variation in the probabilities of individual words predicted by the tri-gram mod-

Table 2
Comparison of recognition results for several tasks.

Task		Grammar	Size of vocabulary (words)	Word perplexity	Word recognition accuracy (%)
Isolated-word recognition	Place name 1	–	100	100	97.3
	Place name 2	–	1000	1000	91.3
Continuous speech recognition	3-layer address	Pre-compiled tree	52 854	47	92.5
	Inquiring sentence	Modify and modified grammar	150	9	81.0
	Dictation	N-gram (N = 3)	60 000	87	91.4

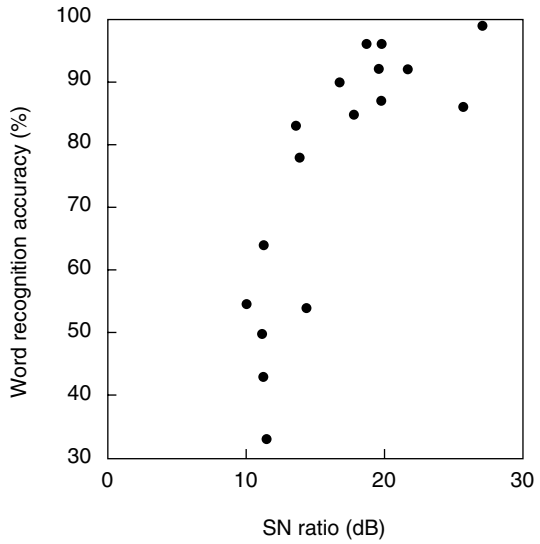


Figure 6
Performance degradation by stationary noise.

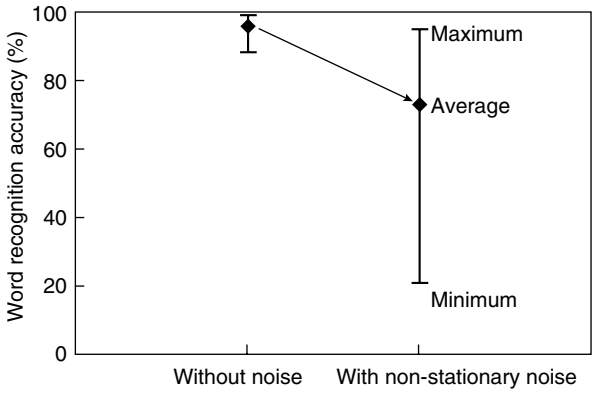


Figure 7
Performance degradation by non-stationary noise.

el, the effective number of predicted words, i.e., the perplexity, is considerably smaller than 60 000.

The vocabulary needed to recognize inquiring Japanese sentences includes the two words "wa" and "ga," which have similar pronunciations and similar grammatical functions. Confusion between these two words degraded the recognition accuracy even though their word perplexities are quite small. If the confusion between these two words is ignored, the accuracy is almost 98%.

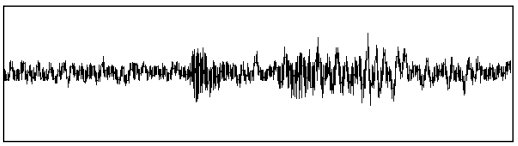
4. Reliability in noisy environments

4.1 Necessity of noise processing

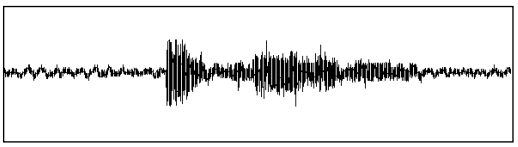
Figure 6 shows the 100-word recognition results at several signal-to-noise ratios. In this evaluation, the speech recognizer did not perform noise processing, so the recognition accuracy goes down sharply when the signal-to-noise ratio becomes worse. **Figure 7** shows the 1000-word recognition results for a noise-free signal and a signal corrupted by a non-stationary noise such as the clicking of a car's indicator unit. The figure shows that the recognition accuracy is drastically degraded by non-stationary noise. Processing for noise is therefore necessary for recognition performed in everyday environments.

4.2 Speech waveforms in noise

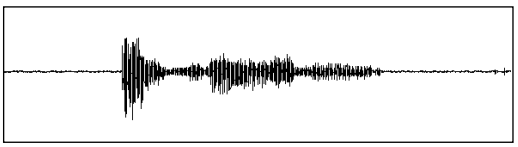
Some examples of speech waveforms in different levels of stationary noise are shown in **Figure 8**. These are for the word "Asahi" after processing by a commonly used method for speech recognition called pre-emphasis. The signal-to-noise (SN) ratio indicates the ratio of speech power to noise power and is used as a measure of how



(a) SN Ratio: 8 dB



(b) SN Ratio: 16 dB



(c) SN Ratio: 27 dB

Figure 8
Examples of speech waveforms in stationary noise.

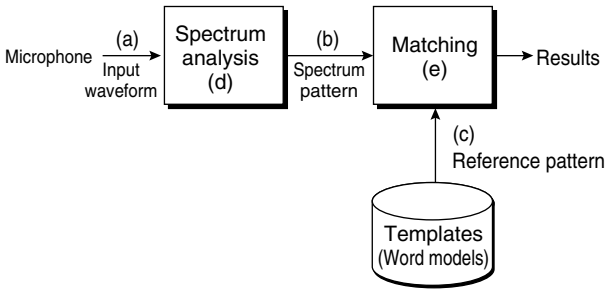


Figure 9
Noise processing in speech recognition system.

much a speech signal is damaged by noise. In actual cases, it is difficult to calculate the speech power because the speech portion also contains noise. So, in this paper, the SN ratio is defined as the ratio of the power of the speech portion (S+N) to the power of the noise (N). In Figure 8, the SN ratio of (a) is 8 dB, and those of (b) and (c) are 16 dB and 27 dB, respectively. In (b) and (c), the speech waveform can be clearly seen, but it is difficult to see in (a).

4.3 Basic ideas for noise processing

In the typical system for speech recognition shown in **Figure 9**, two types of noise processing can be implemented. In one method, a new process for noise is inserted at points (a), (b), or (c); in the other method, processes (d) and (e) are modified. The basic concepts and some examples of these methods are described below.

(a) For the input waveform

For the input waveform at point (a), the idea is to reduce the noise component in the speech portion or enhance the speech signal. To do this, a directional microphone system having multiple microphones is used. Directional microphones are effective for both stationary and non-stationary noise.

(b) For the speech spectrum pattern

A method called "spectrum subtraction"[3] that was invented over 20 years ago is still powerful for stationary noise. This method subtracts an estimate of the noise power-spectrum from the input power-spectrum. This spectrum subtraction method is widely used because its algorithm is not only effective but also very simple and inexpensive to implement.

(c) For reference patterns

Spectrum addition to template patterns is a typical method for stationary noise. This method adds an estimate of the noise power-spectrum to the template patterns created from noise-free speech. In contrast to spectrum subtraction, which is executed only once for each input pattern, spectrum addition must be done for all the templates when the noise pattern changes. So spectrum addition requires more computation.

(d) Spectrum analysis

Spectrum analysis methods, which output identical results whether noise exits or not, are desirable. Filters can be very effective when the component frequencies of the noise spectrum differ from those of the speech signal. The delta parameters, which are the differential values of the time series of the original feature parameters, are not affected by stationary noise.

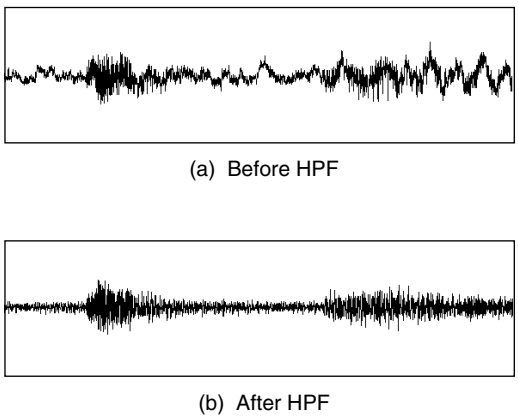


Figure 10
Effect of highpass filter (HPF).

(e) Matching

Noise increases the legacy Euclidean distance between feature vectors of an input and a model. To reduce this, a new measure called a "projection distance" is introduced. The projection distance is calculated as an inner product of the feature vectors. Its value is very stable against noise. Also, high-level stationary noise masks the weaker portions at the beginning and end of an utterance. To eliminate this masking, masked templates are used. For non-stationary noise, a smoothing filter and limiter for the phoneme recognition results are effective. The smoothing filter and limiter avoid extreme mis-recognitions of phonemes due to non-stationary noise.

4.4 Evaluation results

4.4.1 Highpass filter and spectrum subtraction for stationary noise

- **Evaluation data**

Speech data recorded in a noisy environment was used to evaluate various noise processing methods. In the evaluations, 100 words were output from 4 loudspeakers twice. The evaluation data set contained 800 words.

- **Recognition method**

The method of using acoustic-segment networks[4] was used. In this method, orthographic strings of words are used as entries of vocabulary; it is therefore suited to recognition of a large vocabulary. Acoustic templates that represent the spectra of the 28 phonemes of the Japanese language are used as the reference templates described above. Speaker-independent acoustic templates are trained with a large speech data set of many speakers.

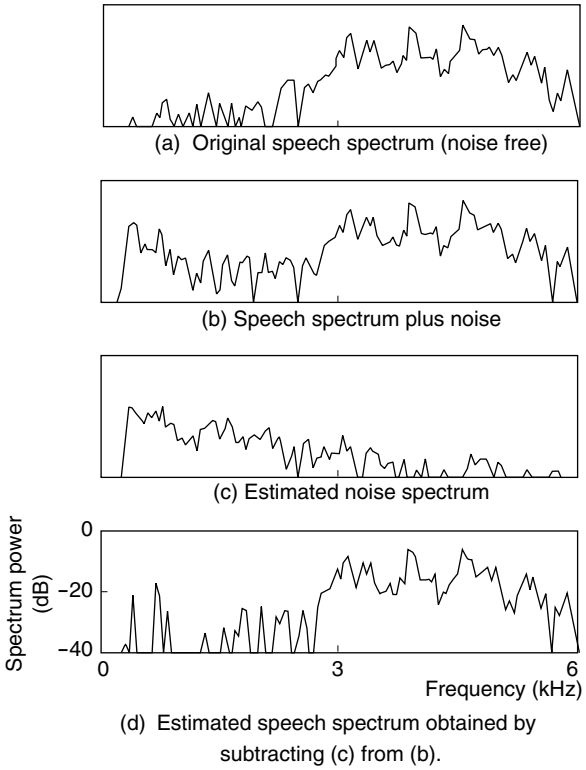


Figure 11
Spectrum subtraction.

- **Highpass filter**

Noise included in speech data recorded in a running car mainly consists of low-frequency components. To reduce its influence, a highpass filter (HPF) with a cut-off frequency of around 300 Hz is very effective. **Figure 10** shows an example waveform before and after it is passed through an HPF. Clearly, the HPF improves the waveform.

- **Spectrum subtraction**

A noise spectrum is estimated using the average spectra of the input when the speaker is quiet. By subtracting the estimated noise spectrum from the input, the noise in the input is reduced. **Figure 11** illustrates the method. In the figure, (a) shows the spectrum of a palatal-

ized /k/ in noise-free speech, (b) is the spectrum of the same utterance but corrupted by noise, (c) is an estimate of the noise spectrum, and (d) is the result after (c) is subtracted from (b). As can be seen, (d) is a good estimation of (a).

Figure 12 shows the recognition results with and without spectrum subtraction for a vocabulary of 100 words. The figure shows that spectrum subtraction can improve the recognition accuracy, especially in the low SN ratio area.

4.4.2 Smoothing filter and limiter for non-stationary noise

Speech data for evaluation was recorded from 24 people talking in a noisy car-environment. An improved version of our recognition algorithm using acoustic-segment networks was used. **Figure 13** shows the results for a 1000-word recognition task with and without a smoothing filter and phoneme limiter in the absence and presence of non-stationary noise. The average and worst recognition accuracies for the 24 speakers in the presence of non-stationary noise and with the smoothing filter and limiter connected are 92% and 85%, respectively. This evaluation showed that the new algorithm significantly improves recognition.

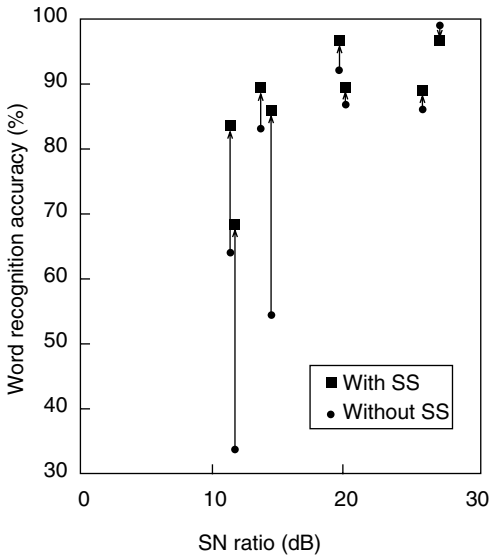


Figure 12
Improvement obtained by spectrum subtraction (SS).

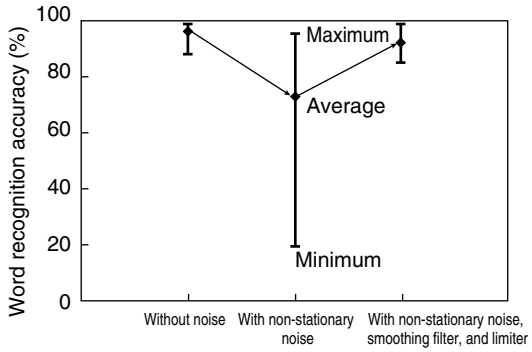


Figure 13
Effect of noise processing for non-stationary noise.

5. Summary

We have described the speech recognition work we conducted in the 1990s. Our focuses have included a small implementation of speech recognition, the combination of speech processing and language processing for continuous speech recognition, and noise processing for speech recognition in noisy environments. Currently, we are investigating spontaneous speech recognition for speech dialog systems.

References

1) Rabiner, B. H. Juang: Fundamentals of Speech Recognition. Prentice-Hall, 1993.
2) comp. speech frequently asked questions web page: http://www.speech.cs.cmu.edu/comp.speech/ or http://www.itl.atr.co.jp/comp.speech/
3) F. Boll: Suppression of Acoustic Noise in Speech Using Spectral Subtraction. *IEEE Trans. Acoustic, Speech Signal Processing*, **ASSP-27**, 2, pp.113-120 (1979).
4) S. Kimura: Boosting Accuracy of a 100,000-word Japanese Recognition System. IEEE ICASSP91, 1991.

Shinta Kimura received the B.E. and M.E. degrees in Electrical and Electronic Engineering from Kobe University, Japan in 1978 and 1980, respectively. Since joining Fujitsu Laboratories Ltd. in 1980, he has been investigating speech recognition, text-to-speech conversion, and speech-signal processing. He received the OHM Award in 1998 for research and development of text-to-speech technology. He is a member of the Institute of Electrical and Electronics Engineers (IEEE), the Institute of Electronics, Information and Communication Engineers (IEICE) of Japan, the Information Processing Society of Japan (IPSJ), and the Acoustic Society of Japan (ASJ).