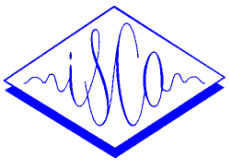

2nd International Conference on
Spoken Language Processing (ICSLP 92)
Banff, Alberta, Canada
October 12-16, 1992

A REAL-TIME SPEAKER-INDEPENDENT CONTINUOUS SPEECH RECOGNITION SYSTEM BASED ON DEMI-SYLLABLE UNITS

Shinji Koga, Ryosuke Isotani, Satoshi Tsukada, Kazunaga Yoshida,
Kaichiro Hatazaki, Takao Watanabe
C & C Information Technology Research Laboratories, NEC Corporation
4-1-1 Miyazaki, Miyamae-ku, Kawasaki, 216 JAPAN

ABSTRACT

This paper describes a real-time speaker-independent continuous speech recognition system. In order to achieve speaker-independent continuous speech recognition with real-time response, demi-syllable speech units, a bundle search algorithm, and multi-processing techniques were used. The use of demi-syllables allows all transitions between phonemes to be represented, thus improving the recognition accuracy. Bundle search is a frame synchronous technique used with Viterbi search to reduce the computational load needed to search the finite state grammar network used. The search process is divided into three pipelined stages : frame-level likelihood calculation, word-level search and sentence level network search. Each of these three pipelined stages is further split into several sub-processes. These processes are run in parallel on a multi-processor machine. Real-time performance was achieved in the evaluation experiments using a 500 word vocabulary. 83.0% sentence accuracy and 95.5% word recognition accuracy were achieved.

10.21437/ICSLP.1992-185

1. INTRODUCTION

Speech, because of its inherent naturalness, is very useful medium to communicate with various systems - for instance a database access system. Achieving natural communication by speech requires the ability to recognize anyone's continuous speech and the ability to respond in real time. Therefore, the authors aim to carry out real-time speaker-independent continuous speech recognition.

A speech recognition method, based on demi-syllable units using hidden Markov models(HMMs), has been developed[1] and proved to be useful for continuous speech recognition, speaker independent recognition, and large vocabulary recognition [2][4][5]. As demi-syllable includes transitional part information, it can efficiently treat contextual variations caused by the co-articulation effect. Each demi-syllable is modeled by a Gaussian mixture density HMM. In continuous speech recognition, a finite state grammar network is used[2].

Real-time response of the recognition system is important for recognition algorithm evaluation in a realistic setting. Furthermore, real-time response is necessary for investigation of speech recognition application as a human interface. For these reasons, a fast search algorithm, by the name of "bundle search", and speech recognition hardware are developed.

Bundle search is a frame-synchronous technique used for Viterbi search (DP matching) to reduce the computational load needed to search the finite state grammar network. To reduce it, a beam search technique was already developed[3]. Bundle search is based on a different idea from the beam search, in that the partial search paths with less likely partial cost are removed. The bundle search examines all the paths in a grammar network in an approximate manner. When a word appears at different positions in the grammar network, the word-level search processing for that word is carried out only once for the most likely occurrence. The result from the processing is used to estimate the network search scores for all other occurrences. Advantages of using the bundle search technique are:

- The removal of the correct path, due to local dis-similarities, is avoided because all the paths are approximately maintained.
- The computational load is almost independent from the grammar network size, increasing only in proportion to the word vocabulary size.

Application specific hardware is developed to achieve efficient speech recognition. Considering the ease for developing the speech recognition evaluation algorithm and the investigation on speech recognition application, there are three important criteria: high speed, which means to be able to accomplish recognition in real time; scalability, which means to be able to increase vocabulary and implement more complex tasks; and flexibility, which means to be able to cope with a change in speech models and recognition algorithms.

Many kinds of recognition hardware, which selected important criteria according to the task, were developed. For example, they include a custom designed machine[6], a tree-structured multiprocessor machine[7], a shared memory multiprocessor machine[8], and so on[9][10]. In order that the developed hardware satisfies those three criteria, architecture is non-shared memory multiprocessor with one common bus, each processor is connected with the bus through first-in first-out (FIFO) memories, and messages are efficiently communicated on the bus by a dedicated processor.

The paper is organized as follows. Section 2 describes the recognition method outline. The bundle search is described in Section 3. The hardware structure is explained in Section 4. Section 5 shows the implementation of recognition process on the hardware. Finally, experimental results on speaker-independent continuous speech recognition are presented in Section 6.

2. RECOGNITION METHOD

As a recognition unit, the demi-syllable is used. The demi-syllable is formed by splitting a syllable in half at its nucleus. It has transitional part information, which is important for accurate phoneme recognition; and it can deal with contextual

variation caused by the co-articulatory effect. The number of demi-syllables is moderate.

Each demi-syllables is modeled by a left-to-right HMM, whose vector output probability is defined using a Gaussian mixture density for speaker-independence. These models are trained on task-independent phonetically balanced speech data from 85 speakers.

The recognition parameter is shown in Table 1.

Table 1: Recognition Parameter.

Acoustic Vector	1 Differenced power
	10 Mel-scaled cepstrum coefficients
	10 Differenced mel-scaled cepstrum coefficients
Frame Rate	10 msec
Recognition Unit	Demi-syllable
Unit Number	241
HMM	Left-to-right, no skip
	Gaussian mixture density, mixture number 2
HMM state number	1 : silence, long vowel
	4 : otherwise

In continuous speech recognition, a finite state grammar network, which consists of nodes and arcs, is used. Each arc is assigned to a word. To connect between words smoothly, the demi-syllable defined by the final syllable of a preceding word and the first syllable of a following word is inserted as a word juncture model. A word dictionary which contains demi-syllable sequences as word representations and the finite state grammar network are compiled into a single HMM network in which Viterbi search will be carried out.

3. BUNDLE SEARCH

In an ordinary frame-synchronous search[3][11], when the same word occurs at many different positions in the grammar network, word-level search is used for each occurrence. In the bundle search, the word-level search is done only once for the most likely occurrence. The result from that word-level search is used to estimate the word-level search scores for all other occurrences.

The bundle search algorithm is as follows:

$T(p,0) = 0; (p = 0)$
$T(p,0) = -\infty; (p > 0)$
$for(i = 1; i <= I; i++)\{$
  $T(p,i) = -\infty; (p = 1, \cdots, Q)$
  $for(n = 1; n <= N; n++)\{$
    /* **word-level search** */
    $T^* = \max_{k=1,K^n} T(p_s(k,n), i-1);$
    $f(n, i-1) = T^*;$
    $g^{i-1}(n,0) = T^*;$
    $l^{i-1}(n,0) = i-1;$
    $for(j = 1; j <= J^n; j++)\{$
      $j_{max} = argmax_{j'}\{g^{i-1}(n,j') + a(n,j',j)\};$
      $g^i(n,j) = d^i(s(n,j)) + g^{i-1}(n,j_{max}) + a(n,j_{max},j);$
      $l^i(n,j) = l^{i-1}(n,j_{max});$
    $\}$
    $g^* = g^i(n,J^n);$
    $l^* = l^i(n,J^n);$
    $T^f = g^* - f(n,l^*);$
    /* **sentence-level network search** */
    $for(k = 1; k <= K^n; k++)\{$
      $T' = T^f + T(p_s(k,n), l^*);$
      $if(T' > T(p_e(k,n),i))\{$
        $T(p_e(k,n),i) = T';$
        $N(p_e(k,n),i) = n;$
        $P(p_e(k,n),i) = p;$
        $L(p_e(k,n),i) = l^*;$
      $\}$
    $\}$
  $\}$
$\},$

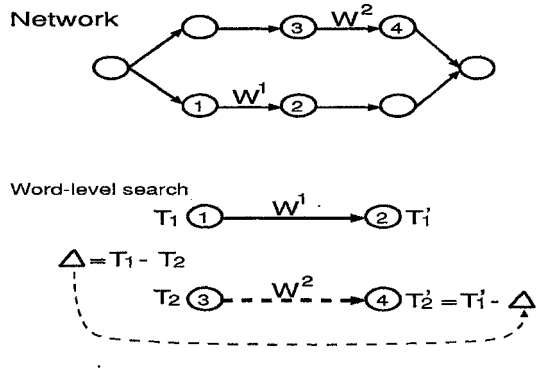

Figure 1: Bundle Search.

where Q is the number of network nodes, in which the initial node number is 1 and the final node number is Q, N is the vocabulary size, K^n is the times word n occurs in the network, $p_s(k,n)$ and $p_e(k,n)$ are the start node and the end node for the arc representing word n of occurrence k, respectively, I is the input speech length, J^n is the length of a word n, and

$f(n,i)$	the initial likelihood for a word n at time i,
$T(p,i)$	the maximum likelihood on a node p at time i,
$N(p,i)$	the preceding word of a node p at time i,
$P(p,i)$	the preceding node of a node p at time i,
$L(p,i)$	the preceding word boundary of a node p at time i,
$s(n,j)$	the state number for a state j of a word n,
$d^i(s)$	the vector output likelihood on a state s at time i,
$g^i(n,j)$	the maximum likelihood on a state j of a word n at time i,
$l^i(n,j)$	the preceding word boundary of a state j of a word n at time i.

In word-level search, the maximum likelihood T^* among the initial likelihoods is used as the search's initial value for all occurrences. For rest of the occurrences except the maximum, the word-level search is not done; instead, the likelihoods after the word-level search are approximately calculated, using the differences between the maximum and others. At time i, the difference $h_i(k,n)$ for the occurrence k of the word n is calculated by subtracting the maximum $f(n,i)$ from the initial likelihood $T^i(k,n)$. On the final state of the word n, the likelihood after the word-level search and $h_i(k,n)$ are added, and the added likelihood is used as the approximate likelihood. Actually f values are saved instead of h_i, in order to reduce memory size.

In the bundle search, though the theoretically optimum likelihood is not assured, the quasi-optimal likelihood is obtained, as all paths are preserved approximately.

4. HARDWARE STRUCTURE

A special hardware was developed to realize real-time response.

The search process is divided into three stages: frame-level likelihood calculation, word-level search, and sentence-level network search. They are repeated per frame in pipeline and in parallel. Each stage is further split into parallel processes by units (a demi-syllable HMM state, a word, and a network node).

For the parallel process, the multi-processor hardware architecture is used. All processors are connected to a data bus to achieve efficient transmission between these processors. For more efficient transmission, a processor which controls data transmission is used.

Figure 2 shows the hardware structure. The processor is a 32bit floating point arithmetic DSP processor. Each has its own local memories, 2-Mbyte SRAM and 32-Kbyte EPROM, and two 8-Kbyte FIFOs.

Since the recognition process is implemented as a program on the DSP board, the recognition algorithm is very easy to modify. All the processor's architectures are the same. Therefore, process assignment is flexible. Data among individual processor are buffered by first-in first-out (FIFO) memories, The bus controller, which controls data transmission, transmits them between FIFOs, depending only on the FIFOs' conditions. Therefore, the processor can transmit data using the FIFOs independently from the bus condition, and data transmission between FIFOs is done in parallel to the processors. Moreover, the processor can broadcast the data.

5. IMPLEMENTATION

This section describes the recognition process implementation on the above hardware.

The processors are divided into three groups. Frame-level likelihood calculation, word-level search, and sentence-level network search are assigned to a particular processor (called F-processor, W-processor, and S-processor depending on the task assigned to it).

1. F-processor (Frame-level likelihood calculation)
   This processor inputs analyzed speech data for each frame, and calculate the vector output likelihoods d^i(in logarithms) at each demi-syllable HMM state. The d^i values are transmitted to W-processors.

2. W-processor (Word-level search)
   The word-level search is carried out in this processor, after receiving d^i from F-processors and initial likelihoods T from S-processors. When the calculation at the final state in each word is accomplished, T^f, l^*, k and n for that word are transmitted to S-processors.

3. S-processor (Sentence-level network search)
   This processor receives data from W-processors, and the sentence-level network search is carried out. After the search, the maximum likelihood T values are transmitted to W-processors, for use as the initial likelihood for the next word-level search.

Figure 3 shows the timing for the above processes. The process in each processor is accomplished in parallel, and the data transmission from W-processors to S-processors is also carried out in parallel.

6. EXPERIMENTS

Continuous speech recognition experiments were carried out on a concert ticket reservation task (Japanese) with 500-word vocabulary and 5.5 word perplexity. The recognition test was implemented on a set of 30 sentences, each of which includes 7.1 words on the average, from 10 speakers (7 males and 3 females). The bundle search reduced compiled HMM network states to one third and the search time to 30 percent. Sentence and word recognition accuracy were 83.0% and 95.5%, respectively.

19 processors comprising 3 F-processors, 12 W-processors and 4 S-processors achieved real-time performance.

7. CONCLUSION

This paper presented a real-time speaker-independent continuous speech recognition system. To achieve speaker-independence, continuous speech recognition, and real-time response; the demi-syllable was used as a recognition unit, the bundle search algorithm, a frame synchronous technique used with Viterbi search, was developed, and the special hardware, used multi-processor processing techniques, was developed. For a concert ticket reservation task with 500-word vocabulary, the bundle search reduced search time to 30 percent. 19 processors achieved real-time performance. Sentence accuracy and word recognition accuracy of 83.0% and 95.5%, respectively, were achieved.

ACKNOWLEDGMENTS

The authors wish to thank Mr. Masao Watari for his helpful comments and encouragement and other members of the Media Technology Research Laboratory for their continuous support. They also thank members of NEC Scientific Information System Development, Ltd. for their support.

References

[1] K.Yoshida, T.Watanabe, and S.Koga, "Large Vocabulary Word Recognition Based on Demi-syllable Hidden Markov Model Using Small Amount of Training Data," *Proc. ICASSP-89*, pp.1-4, Glasgow, 1989.

[2] H.Hattori, S.Tsukada, K.Yoshida, and T.Watanabe, "Continuous Speech Recognition Based on Demi-syllable Hidden Markov Models," *IEICE Technical Report*, pp.23-28, 1989(in Japanese).

[3] H.Sakoe, and H.Fujii, "A High Speed DP-matching Algorithm based on Beam Search and Vector Quantization," *IEICE Technical Report*, pp.33-40, 1987(in Japanese).

[4] R.Isotani, K.Hatazaki, T.Watanabe, H.Ohtsubo, and M.Mizuno, "Speaker-independent Speech Recognition Based on Demi-syllable Hidden Markov Models," *Proc. of ASJ Autumn Meeting*, October 1990(in Japanese).

[5] S.Koga, K.Yoshida, and T.Watanabe, "Evaluation of Large Vocabulary Speech Recognition Based on Demi-syllable HMM," *Proc. of ASJ Autumn Meeting*, October 1989(in Japanese).

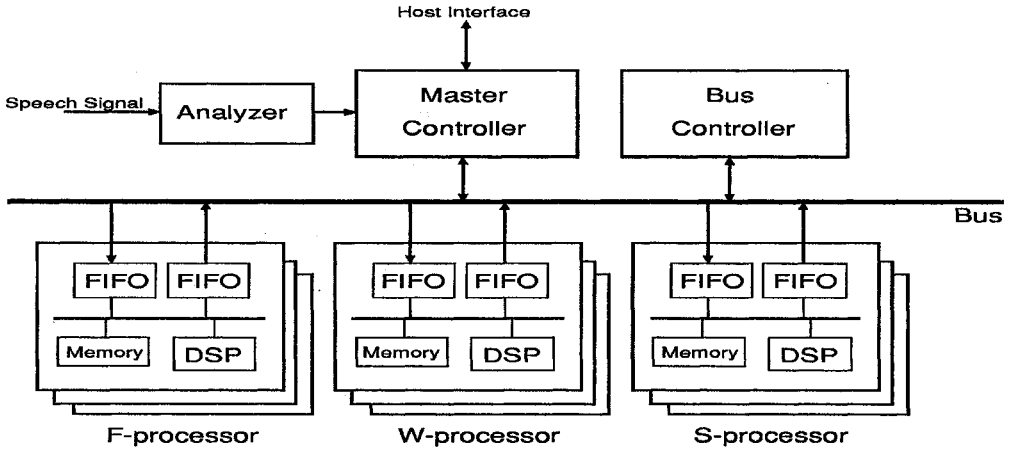

Figure 2: Hardware Structure.

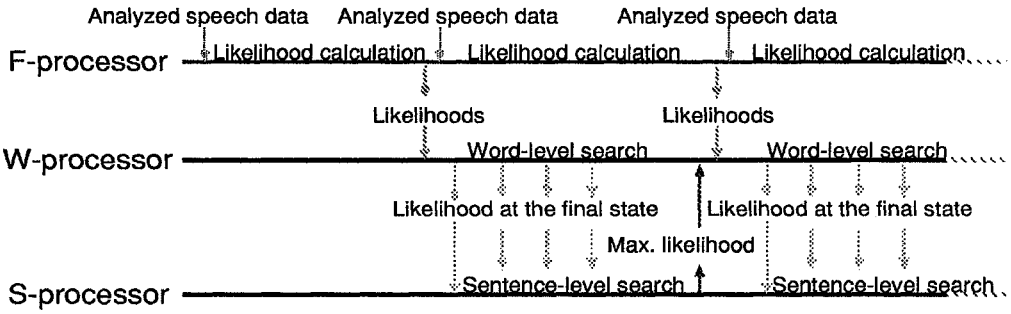

Figure 3: Process Timing.

[6] H.Murveit, J.Mankoski, J.Rabaey, R.Brodersen, T.Stoelzle, D.Chen, S.Narayanaswamy, R.Yu, P.Schrupp, R.Schwartz and A.Santos, "A Large-Vocabulary Real-Time Continuous-Speech Recognition System," *Proc. ICASSP-89*, pp.789-792, Glasgow, 1989.

[7] D.B.Roe, A.L.Gorin, and P.Ramesh, "Incorporating Syntax Into The Level Building Algorithm on a Tree-structured Parallel Computer", *Proc. ICASSP-89*, pp.778-781, Glasgow, 1989.

[8] R.Bisiani, T.Anantharaman, and L.Butcher, "BEAM: An Accelerator for Speech Recognition", *Proc. ICASSP-89*, pp.782-784, Glasgow, 1989.

[9] A.Nagai, K.Kita, T.Hanazawa, T.Suzuki, T.Iwasaki, T.Kawabata, K.Nakajima, K,Shikano, T.Morimoto, S.Sagayama, and A.Kurematsu, "Hardware Development for HMM-LR Continuous Speech Recognition System," *Proc. of ASJ Autumn Meeting*, October 1991(in Japanese).

[10] Y.Suzuki, and A.Imamura, "A Parallel Processor for HMM-Based Word Spotting," *IEICE Technical Report*, pp.9-16, 1990(in Japanese)

[11] J.S.Bridle, M.D.Brown, and R.M.Chamberlain, "An Algorithm for Connected Word Recognition,", *Proc. ICASSP-82*, pp.899-902, 1982.