PAPER

Efficient Combination of Likelihood Recycling and Batch Calculation for Fast Acoustic Likelihood Calculation

Atsunori OGAWA[†,††a)], **Satoshi TAKAHASHI**[††], ***and*** **Atsushi NAKAMURA**[†], ***Members***

SUMMARY This paper proposes an efficient combination of state likelihood recycling and batch state likelihood calculation for accelerating acoustic likelihood calculation in an HMM-based speech recognizer. Recycling and batch calculation are each based on different technical approaches, i.e. the former is a purely algorithmic technique while the latter fully exploits computer architecture. To accelerate the recognition process further by combining them efficiently, we introduce *conditional fast processing* and *acoustic backing-off*. Conditional fast processing is based on two criteria. The first *potential activity* criterion is used to control not only the recycling of state likelihoods at the current frame but also the precalculation of state likelihoods for several succeeding frames. The second *reliability* criterion and acoustic backing-off are used to control the choice of recycled or batch calculated state likelihoods when they are contradictory in the combination and to prevent word accuracies from degrading. Large vocabulary spontaneous speech recognition experiments using four different CPU machines under two environmental conditions showed that, compared with the baseline recognizer, recycling and batch calculation, our combined acceleration technique further reduced both of the acoustic likelihood calculation time and the total recognition time. We also performed detailed analyses to reveal each technique's acceleration and environmental dependency mechanisms by classifying types of state likelihoods and counting each of them. The analysis results comfirmed the effectiveness of the combined acceleration technique.

key words: *fast acoustic likelihood calculation, state likelihood recycling, batch state likelihood calculation, combined acceleration technique, acoustic backing-off*

1. Introduction

It is well known that acoustic likelihood calculation is the most computationally expensive process in a hidden Markov model (HMM) based speech recognizer. Generally speaking, in the total speech recognition process, more than 50% of the computational time is spent on acoustic likelihood calculation. Thus, to accelerate the speech recognition process, acoustic likelihood computation should be reduced. Many studies have attempted to solve this problem [1]–[10], and they can be roughly classified into the following two technical categories.

The first category consists of purely algorithmic techniques, such as Gaussian reduction [1], model parameter tying [2], scalar quantization of feature vectors [3], Gaussian selection [4], and state selection (or state likelihood recycling) [5], [6]. All of these techniques are based on approximations, i.e. simplifications of detailed model structures and/or detailed likelihood calculations. Thus, these techniques trade a slight degradation in recognition accuracy for process acceleration. However, the acceleration performance is essentially independent of the machine specifications.

On the other hand, the second category consists of techniques based on computer architectures, such as Single Instruction, Multiple Data (SIMD) (e.g. MMX [7] and Streaming SIMD Extensions (SSE) [8]), Graphics Processing Unit (GPU) [9], and batch state likelihood calculation [10]. There is concern that their acceleration performance will depend heavily on the machine specifications. However, since none of these techniques use approximations, process acceleration can be obtained without degrading recognition accuracy.

In this paper, we propose an efficient technique for accelerating acoustic likelihood calculation [11]. The proposed technique is based on a combination of state likelihood recycling [5], [6] and batch state likelihood calculation [10]. As mentioned above, they have different technical characteristics, and their good acceleration performance is reported in [5], [6] and [10], respectively. If we could combine them efficiently, further process acceleration could be expected. Based on the similar idea, the combination of Gaussian selection [4] and batch calculation [10] is proposed in [12], and the complemental process acceleration is reported. However, to the best of our knowledge, there have been no studies investigating the combination of recycling [5], [6] and batch calculation [10]. Therefore, it is not known how much process acceleration could be obtained by using the combined technique.

We introduce *conditional fast processing* and *acoustic backing-off* [13] for combining the two acceleration techniques efficiently [11]. Conditional fast processing is based on two criteria. The first *potential activity* criterion is used to control not only the recycling of state likelihoods at the current frame but also the precalculation of state likelihoods for several succeeding frames. The second *reliability* criterion and acoustic backing-off are used to control the choice of recycled or batch calculated state likelihoods when they are contradictory in the combination, and to prevent word accuracies from degrading.

Large vocabulary spontaneous speech recognition experiments using four machines with different CPUs (Pentium 4, Xeon, Core 2 Duo and Xeon X5570) under two environmental (clean and office noise) conditions showed that,

Manuscript received May 18, 2010.
Manuscript revised October 19, 2010.

[†]The authors are with NTT Communication Science Laboratories, NTT Corporation, Kyoto-fu, 619–0237 Japan.
[††]The authors are with NTT Cyber Space Laboratories, NTT Corporation, Yokosuka-shi, 239–0847 Japan.
a) E-mail: ogawa.atsunori@lab.ntt.co.jp

DOI: 10.1587/transinf.E94.D.648

compared with the baseline recognizer, recycling and batch calculation, our combined acceleration technique further reduced both the acoustic likelihood calculation time and the total recognition time. Most of the previous work on accelerating acoustic likelihood calculations has been conducted using one machine under one (usually clean) environment. In this sense, the experimental results obtained under our wide variety of experimental conditions (four different CPU machines × two environmens = eight conditions for one acceleration technique) are informative.

We also performed detailed analyses to reveal each technique's acceleration and environmental dependency mechanisms by classifying types of state likelihoods and counting each of them. The analysis results confirmed the effectiveness of the combined acceleration technique.

This paper is organized as follows: Section 2 provides further details of the two conventional acceleration techniques; state likelihood recycling [5], [6] and batch state likelihood calculation [10]. Section 3 proposes our combined acceleration technique [11] based on conditional fast processing and acoustic backing-off [13]. Section 4 describes large vocabulary spontaneous speech recognition experiments that were conducted to evaluate the acceleration techniques using four different CPU machines and under two environmental conditions. Section 5 analyzes each technique's acceleration and environmental dependency mechanisms in detail. Section 6 concludes this paper.

2. Conventional Acceleration Techniques

This section details the two conventional fast HMM-state likelihood calculation techniques, namely state likelihood recycling and batch state likelihood calculation. They have different technical characteristics and, by combining them efficiently, further process acceleration could be expected.

2.1 State Likelihood Recycling

The first conventional acceleration technique is state likelihood recycling [5], [6]. It accelerates acoustic likelihood calculations by reducing the number of context-dependent (CD) phoneme-HMM (i.e. biphone and triphone) state likelihood calculations by recycling the corresponding monophone state likelihood calculation results. Hereafter, it is referred to as *recycling*.

Figure 1 (a) is a state-frame likelihood table that shows the recycling procedure. Recycling assumes that monophones are approximated models of CD HMMs and, before decoding, all CD HMM states are linked to the monophone states on the condition that they are in the same phoneme cluster and in the same state position (e.g. S_4, S_7 and S_{10} are linked to S_1, and S_5, S_8 and S_{11} are linked to S_2).

During the frame by frame decoding, we calculate the likelihoods of all the monophone states before calculating the likelihoods of the active CD HMM states (S_1 and S_2 at frames $1, 2, \cdots$). The computational costs of these precalculations are not very high because the number of monophone states is small compared with the number of CD HMM states. Then, the likelihood of the corresponding monophone state is referred in the likelihood calculation of each active CD HMM state (at frame 2, S_7 and S_{10} refer to S_1, and S_5 refers to S_2). If it is higher than the recycling threshold (S_2 at frame 2), the CD HMM state likelihood is regarded as worth calculating and is calculated normally (S_5 at frame

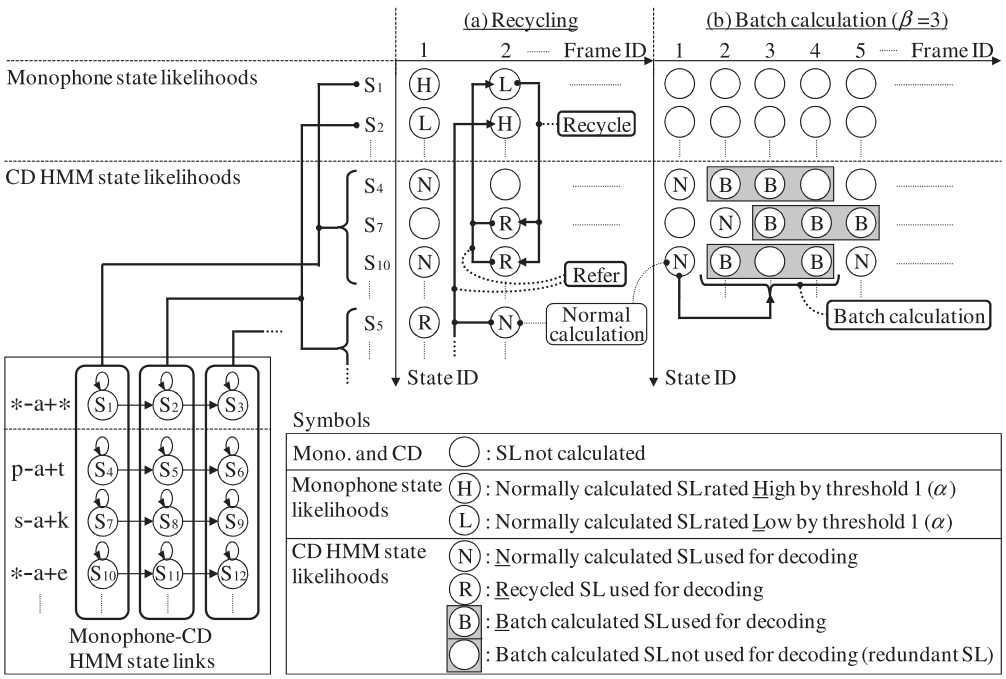

Fig. 1 Procedures of recycling and batch calculation in state-frame likelihood tables (SL: state likelihood).

2). Conversely, if the monophone state likelihood is lower than the recycling threshold (S_1 at frame 2), the CD HMM state likelihood is not regarded as worth calculating. And the monophone state likelihood is *recycled* as the approximated likelihood of the CD HMM state (S_7 and S_{10} at frame 2).

At every frame, the recycling threshold is given by multiplying the maximum monophone state likelihood by recycling coefficient α ($-\infty < \alpha < 1.0$). As α becomes larger, the number of likelihood recycling operations increases, thus the acoustic likelihood calculation is accelerated but with a risk of degraded recognition accuracy. As α becomes smaller, the opposite effect is obtained. Since recycling is a purely algorithmic technique, its acceleration performance is essentially independent of machine specifications.

2.2 Batch State Likelihood Calculation

The second conventional acceleration technique is batch state likelihood calculation [10]. It accelerates acoustic likelihood calculations by reducing the number of time-consuming state parameter fetching processes. Hereafter, it is referred to as *batch calculation*.

Batch calculation is based on the following two experimental analyses.

(i) Profiling shows that, in a state likelihood calculation, much of the time is spent not on floating-point operations, but in fetching the state parameters (i.e. the mean and diagonal covariance vectors and weighting factors of each Gaussian pdf in the state) from the main memory to the cache.
(ii) Speech has several frame intervals that could be regarded as stationary. Therefore, if a state is activated at a frame, it tends to be activated for several succeeding frames.

The batch calculation procedure, which exploits the above two characteristics, is shown in Fig. 1 (b). If a CD HMM state is activated at a frame t, the state likelihoods are calculated and stored in the state-frame likelihood table not only for the current frame t (e.g. S_{10} at frame 1) but also for succeeding β frames (S_{10} at frames 2, 3 and 4, since β is set at 3 in Fig. 1 (b)). Then, for these look-ahead frames, $t+1, \cdots, t+\beta$, if the state likelihoods are required, they are looked up in the table (S_{10} at frames 2 and 4).

In batch calculation, the number of time-consuming state parameter fetching processes described in (i) is reduced, thus, we can expect the acoustic likelihood calculation to be accelerated. If the batch calculated state likelihoods are not used, they become redundant calculations (S_{10} at frame 3). However, because of the property of speech described in (ii), there are not so many of these redundant calculations. There is concern that the acceleration performance will depend heavily on the machine specifications. But, there is no degradation in recognition accuracy because there is no approximation in batch calculation.

3. Combined Acceleration Technique

As described in Sect. 2, recycling and batch calculation have different technical characteristics, and their good acceleration performance is reported in [5], [6] and [10], respectively. To further accelerate the recognition process by combining them efficiently, we introduce the *conditional fast processing* strategy, which is based on two criteria, namely *potential activity* and *reliability* criteria (thresholds or coefficients in the implementation) for our combination algorithm. In addition, we introduce the *acoustic backing-off* [13] strategy to prevent word accuracies from degrading.

3.1 Conditional Fast Processing

Figure 2 (a) shows our combination algorithm; *conditional fast processing*. As shown in this figure, monophone state likelihoods are rated high or low by using a recycling threshold (threshold 1) based on a recycling coefficient α as with recycling. And there are two possibilities for the corre-

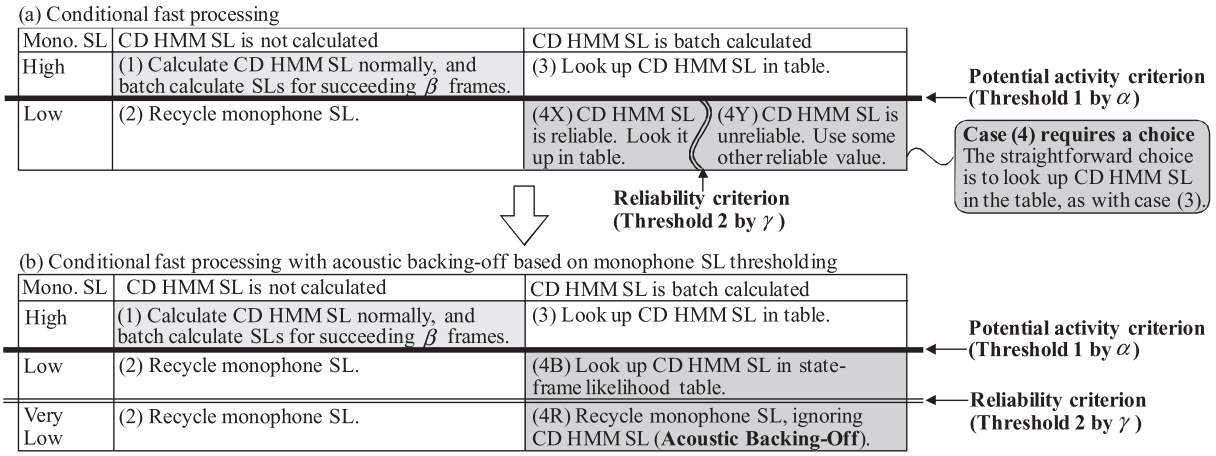

Fig. 2 Conditional fast processing with acoustic backing-off (SL: state likelihood).

sponding CD HMM state likelihoods, one is not calculated and the other is batch calculated. Therefore, the combination basically consists of four cases, (1)–(4).

In case (1), as with recycling, we calculate the CD HMM state likelihoods normally. At the same time, as with batch calculation, we calculate the CD HMM state likelihoods for succeeding β frames. In case (2), as with recycling, we recycle the monophone state likelihoods as the approximated likelihoods of the corresponding CD HMM states. In case (3), as with batch calculation, we look up the batch calculated CD HMM state likelihoods in the state-frame likelihood table.

The function of recycling threshold (threshold 1) is strengthened with this combination algorithm. In recycling, as described in Sect. 2.1, it controls whether we calculate the CD HMM state likelihoods normally or approximate them with corresponding monophone state likelihoods only at the current frame. However, in the combined technique, as described in case (1), it also controls whether we calculate the CD HMM state likelihoods for succeeding β frames *in advance* with the estimation that these states would be activated in the future frames. Thus, in the combined technique, threshold 1 based on coefficient α is not just a recycling threshold, and we refer to it as a *potential activity* threshold.

In case (4), we must choose the state likelihood calculation results obtained with either of the two techniques. In this case, the monophone state likelihoods are rated low. Thus, the recycling estimates that the corresponding CD HMM state likelihoods are not worth calculating and could be approximated. On the other hand, several frame earlier, based on the continuity of the state activation, the batch calculation estimated that the CD HMM state likelihoods would be worth calculating and calculated them in advance. That is, in this case, the state likelihood calculation results of recycling and batch calculation are *contradictory*. The straightforward choice in case (4) would be to look up the batch calculated CD HMM state likelihoods in the state-frame likelihood table as with case (3). This is because, in general, CD HMM state likelihoods are more precise than those of the corresponding monophone states.

In contrast to the straightforward choice, we adopt a more efficient method for preventing word accuracy degradation. It is based on the *reliabilities* of the state likelihoods and includes the straightforward choice as a special case. This reliability is a sort of frame level confidence [14] and is estimated frame by frame. Our method divides case (4) into two cases with a *reliability* threshold. If the CD HMM state likelihoods are regarded as reliable (case (4X)), we look up them in the state-frame likelihood table as with case (3). On the other hand, if the CD HMM state likelihoods are regarded as unreliable (case (4Y)), we use some other reliable value in place of the unreliable CD HMM state likelihoods.

3.2 Unreliable CD HMM State Likelihood

A monophone state is a representative version of the corresponding CD HMM states. Conversely, a CD HMM state is a detailed version of the corresponding monophone state. There should be a certain degree of correlation between monophone state likelihoods and the corresponding CD HMM state likelihoods.

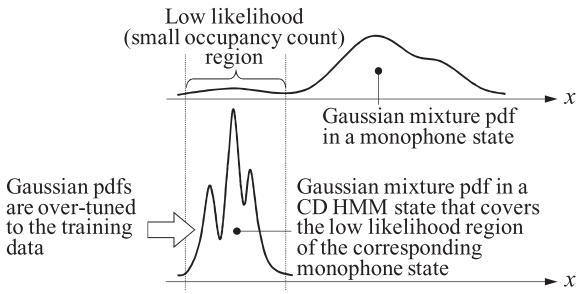

Fig. 3 Example of unreliable CD HMM state likelihood.

A monophone state is trained to cover all the data of a part (i.e. beginning, middle or end) of a phoneme segment. Since the occupancy counts of the monophone states are large, the parameters of the Gaussian pdfs in the monophone states are robustly estimated, and the reliabilities of the state likelihoods obtained from the monophone states are expected to be high. The training data of a monophone state are divided into parts according to the preceding and succeeding phoneme dependencies determined by the tree-based state clustering result. And each of the corresponding CD HMM states is individually trained using part of the divided data.

As shown in Fig. 3, it is difficult to obtain a robust estimate of the Gaussian pdf parameters of a CD HMM state that covers the *low likelihood region* of the corresponding monophone state. In this *small occupancy count region*, Gaussian pdfs in the CD HMM state are over-tuned to the training data. Consequently, their covariances tend to be small. That is, if the monophone state likelihood is *very low* for an input feature vector, the state likelihoods of the corresponding CD HMM states for the input feature vector are unreliable. In some cases, even if the monophone state likelihood is very low for an input feature vector, the state likelihoods of the corresponding CD HMM states for the feature vector may be extremely high because of small covariances.

3.3 Acoustic Backing-Off

Based on the consideration described in Sect. 3.2, as shown in Fig. 2 (b), we divide case (4) into two cases, (4B) and (4R), by introducing a new coefficient, i.e. the *reliability* coefficient γ, in addition to the potential activity coefficient α ($-\infty < \gamma \leq \alpha < 1.0$). Figure 4 corresponds to Fig. 2 (b) and shows the procedure of our combined acceleration technique in the state-frame likelihood table. γ gives threshold 2 that divides monophone state likelihoods into low or *very low* ranks (threshold 1 $\geq$ threshold 2). In case (4B), the monophone state likelihoods are rated low (in Fig. 4, S_1 at frame 3). Here, as with the straightforward choice, we look up the CD HMM state likelihoods in the state-frame likelihood ta-

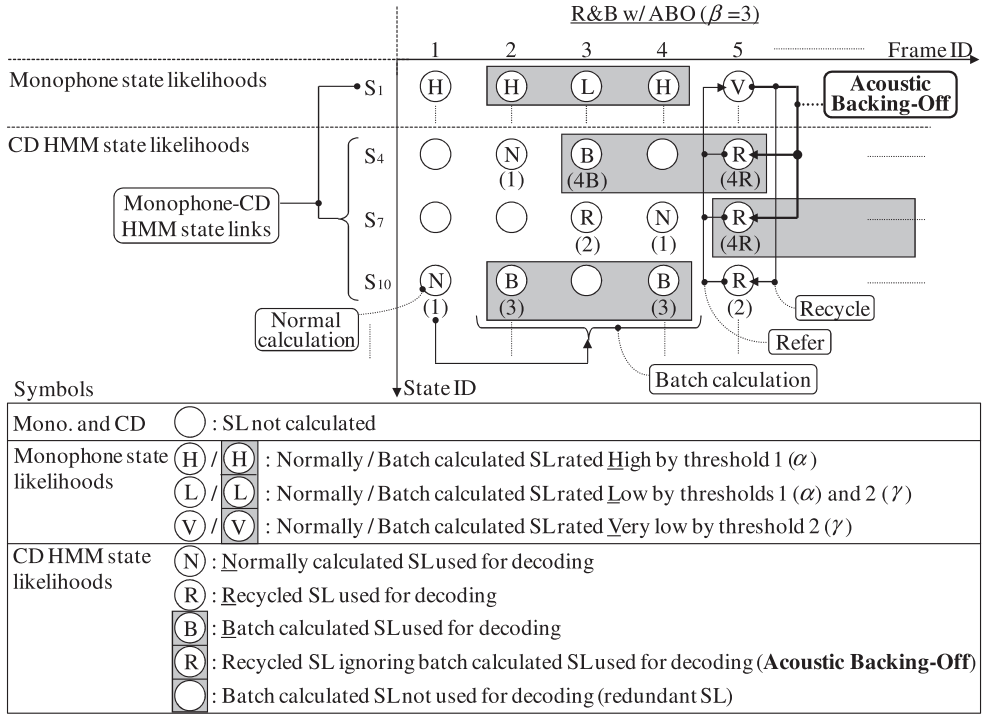

Fig. 4 Procedures of the combined acceleration technique (R&B w/ ABO) in state-frame likelihood tables (SL: state likelihood).

ble (S_4 at frame 3. (4B) means that the Batch calculation results are chosen). In case (4R), the monophone state likelihoods are rated very low (S_1 at frame 5). In this case, we estimate that the corresponding batch calculated CD HMM state likelihoods are unreliable and, as with recycling, we recycle the more reliable monophone state likelihoods as the approximated likelihoods of the CD HMM states (S_4 and S_7 at frame 5. (4R) means that the Recycling results are chosen).

Recycling in case (4R) is a sort of *acoustic backing-off* [13] i.e. we postpone the detailed local frame scoring of the active hypotheses by giving them certain low but reliable state likelihoods. If γ is set at $-\infty$, case (4R) (i.e. acoustic backing-off) disappears, and our method becomes equivalent to the straightforward method. If γ is set equal at α, case (4B) disappears, and in case (4) (i.e. (4R)), CD HMM state likelihoods are approximated by the corresponding monophone state likelihoods according to the acoustic backing-off strategy. It should be noted that, in the combined technique, the monophone state likelihood calculations are also accelerated by batch calculation (S_1 at frames 2, 3 and 4). Hereafter, we refer to our combined acceleration technique shown in Figs. 2 (b) and 4 as "R&B w/ ABO" (i.e. a combination of Recycling and Batch calculation with Acoustic Backing-Off).

4. Speech Recognition Experiments

To evaluate the proposed combined acceleration technique in comparison with the two conventional acceleration techniques and the baseline speech recognizer, we conducted large vocabulary spontaneous speech recognition experiments using four different CPU machines under two environmental conditions. Under all of the experimental conditions, our combined acceleration technique further reduced both the acoustic likelihood calculation time and the total recognition time compared with the other techniques.

Table 1 Acoustic analysis conditions.

Sampling frequency	16 kHz
Window type	Hamming
Frame length/shift	20 ms/10 ms
Pre-emphasis	$1 - 0.97z^{-1}$
Feature vector	12MFCC+12ΔMFCC+Δ logPow
Feature normalization	Moving average CMN

4.1 Experimental Conditions

We built a speech database that consisted of conversations between agents and customers in simulated call-center situations. We gave anwering manuals to the speakers who played call-center agents (all of them worked as agents in real call centers). And we provided the speakers who played the customers with goals (i.e. their reasons for contacting the call center). Then they conversed according to the agent's guide with the aim of dealing with the customer's reason for calling. Since, there were no constraints other than the agents' manuals and the customers' aims, their conversations were quite spontaneous. The utterances were recorded at a high signal-to-noise ratio (SNR) of more than 50 dB.

The acoustic analysis conditions are shown in Table 1. An HMM-based female acoustic model was trained using

Table 2 Specifications of the four machines.

CPU					# of CPUs	Memory	
Type	# of cores	Clock speed	Cache size	FSB speed		Chip	Size
Intel Pentium 4 Extreme Edition (Gallatin)	1	3.40 GHz	2 MB	800 MHz	1	DDR400	2 GB
Intel Xeon (Irwindale)	1	3.60 GHz	2 MB	800 MHz	2	DDR2-400	4 GB
Intel Core 2 Duo (Conroe) E6600	2	2.40 GHz	4 MB	1066 MHz	1	DDR2-533	4 GB
Intel Xeon (Nehalem-EP) X5570	4	2.93 GHz	8 MB	1333 MHz	2	DDR3-1333	24 GB

100 hours of speech consisting of 120k utterances by 55 female agents from our database described above. The acoustic model had 2000 states (consisting of 90 monophone states and 1910 CD HMM states) and each state had 16-mixture Gaussian components with diagonal covariance parameters (thus, the total number of Gaussian mixture components was 32000). A trigram language model was trained using 1.1M words of text data from a transcription of our speech database, World Wide WEB texts and newspaper articles, with Witten-Bell smoothing. The vocabulary size of the word pronunciation dictionary was set at 30k. The baseline speech recognizer was VoiceRex [15], [16], which employs a standard Viterbi beam search with a two-pass decoding strategy. The three acceleration techniques described in Sects. 2 and 3 were implemented on VoiceRex.

As described in Sect. 2.2, there is concern that the acceleration performance of batch calculation (and also the combined techniques) depends on the machine specifications. Thus, we conducted the experiments using four machines with different specifications as shown in Table 2. Hereafter, we identify them by CPU types, i.e. Pentium 4, Xeon, Core 2 Duo and Xeon X5570. Pentium 4 is an old CPU. However, we employed it since many call centers (as described above, one of our target fields) still use the machines with such old CPUs. Xeon X5570 is a state-of-the-art CPU based on Intel microarchitecture of codename Nehalem as with Core i7. 32-bit CentOS Linux 5.2 operating system was installed in all of the machines.

The evaluation speech data consisted of 850 utterances by 17 female agent speakers (50 utterances per speaker) from our speech database, who were different from the 55 female speakers who provided the acoustic model training data. The total number of words was 9488 and the average utterance length was 3.28 sec. The test set perplexity with the trigram language model was 107.3 and the out-of-vocabulary rate was 0.8%. As described in Sect. 3.3, the combined acceleration technique uses acoustic backing-off that was originally proposed for recovering degraded speech recognition accuracy under nonstationary noisy conditions [13]. Therefore, in addition to the clean environmental condition, we also conducted experiments using the 850 utterances described above contaminated with office noise data with a 15 dB SNR.

We compared the following five techniques: 1) a baseline speech recognizer that did not employ any special techniques for acoustic likelihood calculation acceleration, 2) recycling, 3) batch calculation, 4) the proposed acceleration technique *without* acoustic backing-off, i.e. the combined technique that employed the straightforward choice in case (4) in Fig. 2 (a) or set γ at $-\infty$ in case (4) in Fig. 2 (b) (hereafter, referred to as R&B), and 5) the combined acceleration technique *with* acoustic backing-off (R&B w/ ABO) shown in Figs. 2 (b) and 4. Basic decoding parameters such as beam width for hypothesis pruning, language weight and word insertion penalty were common to all the techniques.

Many different experimental setups were realized by changing the five techniques, their parameters (excepting the basic decoding parameters), the four machines, and the two environmental conditions. In each experimental setup, we ran a single-thread speech recognition program while automatically monitoring that there were no other running programs. We measured the raw acoustic likelihood calculation time (hereafter, referred to as raw ALCT) and the raw total recognition time (raw TRT) based on the CPU time obtained by using the clock function of the ANSI C programming language. With each setup, we measured the CPU time five times and averaged the results to reduce measurement errors. We also calculated the normalized ALCTs and TRTs on the basis of the raw ALCTs and TRTs of the baseline speech recognizer.

4.2 Experimental Results in Clean Environment

Table 3 shows raw ALCTs, TRTs and their normalized versions obtained with the five techniques on the four different CPU machines in a clean environment. The word accuracies (WACCs) and parameters for the five techniques are also shown.

According to the requirements of the application systems, we allowed a relative word accuracy degradation of up to 2.00% from the baseline word accuracy (75.89%), i.e. absolute degradation of 0.49%. Allowing this degradation of word accuracy, with recycling, the recycling coefficient α was increased from 0.600 to 0.950 in 0.025 steps, and was finally fixed at 0.750. We can confirm that, as described in Sect. 2.1, the word accuracy degrades slightly (0.35% in absolute) from the baseline but similar reductions of the ALCTs and TRTs are obtained for all of the machines. The reduction rates of the normalized ALCTs and TRTs are up to 37% (Core 2 Duo) and 20% (Xeon and Core 2 Duo), respectively.

With batch calculation, the number of look-ahead frames β was fixed at 7 according to our preliminary experiments and [10]. We can confirm that, as described in Sect. 2.2, there is no degradation in word accuracy but the ALCT and TRT reductions depend heavily on the machine specifications. The reduction rates of the normalized ALCT range from 2% (Xeon X5570) to 42% (Xeon) and

Table 3 Raw ALCT, TRT and their normalized versions (in parentheses) obtained with the five techniques on the four machines in a clean environment. Memory bandwidth of the four machines measured by the STREAM benchmark with TRIAD performance are also shown.

Technique	Parameter			WACC	Raw ALCT [sec] (Normalized ALCT) / Raw TRT [sec] (Normalized TRT)			
	α	β	γ	[%]	Pentium 4	Xeon	Core 2 Duo	Xeon X5570
Baseline	—	—	—	75.89	0.69 (1.00) / 1.21 (1.00)	0.82 (1.00) / 1.46 (1.00)	0.45 (1.00) / 0.81 (1.00)	0.23 (1.00) / 0.46 (1.00)
Recycling	0.750	—	—	75.54	0.48 (0.70) / 1.00 (0.83)	0.53 (0.64) / 1.17 (0.80)	0.28 (0.63) / 0.64 (0.80)	0.15 (0.67) / 0.39 (0.84)
Batch calc.	—	7	—	75.89	0.47 (0.68) / 0.99 (0.82)	0.47 (0.58) / 1.12 (0.77)	0.34 (0.76) / 0.70 (0.87)	0.22 (0.98) / 0.46 (0.99)
R&B	0.825	7	$-\infty$	75.82	0.29 (0.42) / 0.82 (0.68)	0.25 (**0.31**) / 0.90 (**0.62**)	0.20 (0.45) / 0.57 (0.70)	0.15 (0.65) / 0.38 (0.84)
R&B w/ ABO	0.825	7	0.375	**75.94**	0.29 (0.43) / 0.82 (0.68)	0.25 (**0.31**) / 0.90 (**0.62**)	0.21 (0.46) / 0.57 (0.70)	0.15 (0.65) / 0.38 (0.83)
Memory bandwidth [MB/s]					3165.79	2844.80	3596.31	6857.25

Table 4 Raw ALCT, TRT and their normalized versions (in parentheses) obtained with the five techniques on the four machines in a noisy environment.

Technique	Parameter			WACC	Raw ALCT [sec] (Normalized ALCT) / Raw TRT [sec] (Normalized TRT)			
	α	β	γ	[%]	Pentium 4	Xeon	Core 2 Duo	Xeon X5570
Baseline	—	—	—	69.06	0.80 (1.00) / 1.40 (1.00)	0.92 (1.00) / 1.68 (1.00)	0.52 (1.00) / 0.94 (1.00)	0.26 (1.00) / 0.54 (1.00)
Recycling	0.750	—	—	68.94	0.64 (0.80) / 1.24 (0.89)	0.71 (0.77) / 1.47 (0.88)	0.39 (0.76) / 0.82 (0.87)	0.20 (0.79) / 0.48 (0.90)
Batch calc.	—	7	—	69.06	0.52 (0.66) / 1.13 (0.81)	0.52 (0.56) / 1.28 (0.76)	0.37 (0.72) / 0.80 (0.85)	0.24 (0.94) / 0.52 (0.97)
R&B	0.800	7	$-\infty$	68.77	0.42 (0.53) / 1.03 (0.73)	0.38 (0.41) / 1.14 (**0.68**)	0.28 (0.55) / 0.71 (0.76)	0.19 (0.76) / 0.48 (0.89)
R&B w/ ABO	0.800	7	0.575	**69.20**	0.42 (0.52) / 1.02 (0.73)	0.37 (**0.40**) / 1.13 (**0.68**)	0.29 (0.56) / 0.72 (0.76)	0.19 (0.76) / 0.48 (0.89)

the normalized TRT range from 1% (Xeon X5570) to 23% (Xeon). One reason for obtaining these results could be attributable to the memory bandwidths of the four machines. We measured them by using the STREAM benchmark with the TRIAD performance [17], [18] as shown in the bottom of Table 3. We can confirm that the effect of batch calculation in a machine is *inversely* proportional to the memory bandwidth of the machine. Namely, in the case of Xeon, the effect of batch calculation becomes relatively large because of the low memory bandwidth of the machine[†]. In contrast, in the case of Xeon X5570, the effect of batch calculation is very small because of the very high memory bandwidth of the machine.

With R&B, β was fixed at 7 as with batch calculation, and the *potential activity* coefficient α was adjusted with the same procedure employed for the recycling coefficient α as described above and was finally fixed at 0.825. We can confirm that the word accuracy degrades slightly (0.07% in absolute) from the baseline but further ALCT and TRT reductions are obtained for all of the machines. The slight degradation in the word accuracy is derived from the recycling property, and the machine dependence of the ALCT and TRT reductions derives from the batch calculation property. Normalized ALCT and TRT reduction rates are the largest for Xeon at 69% and 38%, respectively.

With R&B w/ ABO, α and β were fixed at the same value with R&B, namely 0.825 and 7, respectively. Then the *reliability* coefficient γ was increased from 0.025 to 0.825 in 0.025 steps and was finally fixed at 0.375. Compared with R&B, the word accuracy of R&B w/ ABO is recovered 0.12% in absolute. This value itself is small, however, compared with 0.07%, i.e. the absolute degradation of the word accuracy from the baseline to R&B, this recovery value is sufficiently large. As a result, the word accuracy of R&B w/ ABO becomes slightly, 0.05% in absolute, higher than the baseline. From this result, we can confirm that acoustic backing-off is effective in preventing the word accuracy from degrading even in a clean environment as designed in Sect. 3.3. The ALCT and TRT reductions of R&B w/ ABO are almost the same with R&B for all of the machines.

4.3 Experimental Results in Noisy Environment

Table 4 is the noisy environment version of Table 3. The parameters, α, β and γ, were adjusted with the same procedure employed for the clean environment case. With a baseline word accuracy of 69.06%, a 2.00% relative degradation correponded to a 0.63% absolute degradation. Allowing this degradation of word accuracy, we adjusted the recycling or *potential activity* coefficient α and finally fixed it at 0.750 (the same as in the clean environment) for recycling and at 0.800 (one step smaller than for the clean environment value of 0.825) for R&B and R&B w/ ABO. The number of look-ahead frames β was fixed at 7 for batch calculation, R&B and R&B w/ ABO. The *reliability* coefficient γ was finally fixed at 0.575 (larger than the clean case value of 0.375) for R&B w/ ABO.

Because of the acoustic mismatch between the acoustic model training data and the evaluation data, compared with the clean environment case, performance degradations are observed. The word accuracies degrade about 7% with all of the techniques. Also the raw ALCT and TRT increase with all of the techniques. However, the increases with batch calculation are relatively small. And in contrast to the other techniques, with batch calculation, the normalized ALCT and TRT reduction rates slightly improves.

Despite the performance degradation described above, also in a noisy environment, R&B and R&B w/ ABO show their advantages as in a clean environment. Their normalized ALCT and TRT reduction rates are the largest for Xeon

[†]As shown in Tables 3 and 4, with all of the techniques, *raw* ALCT and TRT of Xeon are larger than those of the older machine; Pentium 4. We guess that one reason of these results could also be attributable to the low memory bandwidth of Xeon.

at 60% and 32%, respectively. With R&B w/ ABO, larger reliability coefficient γ caused acoustic backing-off to occur more frequently. The word accuracy of R&B w/ ABO recovered 0.43% in absolute compared with R&B. This recovery value is larger than 0.29%, which is the absolute degradation of the word accuracy from the baseline to R&B. As a result, the word accuracy of R&B w/ ABO is slightly higher than the baseline (0.14% in absolute). From these results, we can confirm that acoustic backing-off is effective in a noisy environment as originally reported in [13].

5. Detailed Analyses

In Sect. 4, we showed acceleration performance and environmental dependence of the five techniques. In this section, we performed detailed analyses to reveal their mechanisms. All of the acceleration techniques in this paper focus on state likelihood calculations, and therefore, the analyses should also be performed by focusing on them. We classified types of state likelihoods and counted each of them. The analysis results confirmed the effectiveness of our combined acceleration technique.

5.1 Analysis Conditions

We classified types of state likelihoods into three main categories and further classified each of them into some subcategories as follows.

The first main category consisted of state likelihoods that were *actually calculated* with floating-point operations during the decoding for an evaluation utterance (hereafter, referred to as Act-SLs). Act-SLs consisted of *normally calculated* state likelihoods (Nml-SLs) and *batch calculated* state likelihoods (Bat-SLs), i.e. #Act-SLs = #Nml-SLs + #Bat-SLs. Once calculated, the state likelihoods for the evaluation utterance were stored in a state-frame likelihood table as shown in Figs. 1 and 4. We also computed the state likelihood *calculated rate* in the state-frame likelihood table (SLC-Rate), i.e. the ratio of #Act-SLs divided by the size of the state-frame likelihood table. The table size is obtained by multypling the number of states in the acoustic model by the number of frames in the evaluation utterance.

The second main category consisted of *required* state likelihoods during the decoding for an evaluation utterance (Req-SLs). The Req-SLs consisted of the Nml-SLs described above, state likelihoods that were *looked up* in the state-frame likelihood table (Lup-SLs), *recycled* state likelihoods (Rcy-SLs) and state likelihoods given by *acoustic backing-off* (ABO-SLs), i.e. #Req-SLs = #Nml-SLs + #Lup-SLs + #Rcy-SLs + #ABO-SLs.

The third main category consisted of *redundant* state likelihoods (Rdn-SLs) for the decoding of an evaluation utterance. As described in Sect. 2.2, with batch calculation, also with R&B and R&B w/ ABO, some batch calculated state likelihoods are not used in the decoding. In such cases, they become redundant state likelihoods.

For each of the two environmental conditions and the five techniques, all the types of state likelihoods described above were accumulated for all of the 850 evaluation utterances and averaged by the number of evaluation utterances (850). As described in Sect. 4.1 and shown in Table 1, there were 2000 states in the acoustic model, the average length of the evaluation utterances was 3.28 sec and the frame shift was 10 ms. Therefore, the average number of frames in the evaluation utterances was 328 and the average size of the state-frame likelihood tables was 656000. The parameter (α, β, γ) settings of all the techniques are the same as in Sects. 4.2 and 4.3.

5.2 Analysis Results in Clean Environment

Table 5 shows the state likelihood counting results obtained with the five techniques in a clean environment. It also shows the word accuracies, raw ALCTs, TRTs and their normalized versions (in parentheses) for the five techniques on Xeon (the machine that showed the best acceleration performance in Sects. 4.2 and 4.3).

The number of Act-SLs with recycling is small compared with the other techniques (#Act-SLs = #Nml-SLs). And by recycling them (Rcy-SLs, these are monophone state

Table 5 State likelihood counting results obtained with the five techniques in a clean environment.

# state likelihoods	Technique and parameter (α,β,γ) setting				
	Baseline (—,—,—)	Recycling (0.750,—,—)	Batch calc. (—,7,—)	R&B (0.825,7,−∞)	R&B w/ ABO (0.825,7,0.375)
#Act-SLs	228947	162871	289516	198526	198529
#Nml-SLs	228947	162871	42792	31614	31613
#Bat-SLs	0	0	246724	166912	166916
SLC-Rate [%]	35	25	44	30	30
#Req-SLs	851704	894988	851704	895346	896548
#Nml-SLs	228947	162871	42792	31614	31613
#Lup-SLs	622757	560284	808912	696156	693855
#Rcy-SLs	0	171833	0	167576	167731
#ABO-SLs	0	0	0	0	3349
#Rdn-SLs	0	0	60569	33811	35555
WACC [%]	75.89	75.54	75.89	75.82	75.94
Raw ALCT [sec] (Norm. ALCT)	0.82 (1.00)	0.53 (0.64)	0.47 (0.58)	0.25 (0.31)	0.25 (0.31)
Raw TRT [sec] (Norm. TRT)	1.46 (1.00)	1.17 (0.77)	1.12 (0.78)	0.90 (0.62)	0.90 (0.62)

Table 6 State likelihood counting results obtained with the five techniques in a noisy environment.

# state likelihoods	Technique and parameter (α,β,γ) setting				
	Baseline (—,—,—)	Recycling (0.750,—,—)	Batch calc. (—,7,—)	R&B (0.800,7,−∞)	R&B w/ ABO (0.800,7,0.575)
#Act-SLs	254264	207530	313953	262438	262520
#Nml-SLs	254264	207530	46199	40066	40063
#Bat-SLs	0	0	267755	222372	222457
SLC-Rate [%]	39	32	48	40	40
#Req-SLs	959375	988602	959375	992989	991090
#Nml-SLs	254264	207530	46200	40066	40063
#Lup-SLs	705111	657125	913175	856846	846414
#Rcy-SLs	0	123947	0	96078	96514
#ABO-SLs	0	0	0	0	8099
#Rdn-SLs	0	0	59689	42903	47065
WACC [%]	69.06	68.94	69.06	68.77	69.20
Raw ALCT [sec] (Norm. ALCT)	0.92 (1.00)	0.71 (0.77)	0.52 (0.56)	0.38 (0.41)	0.37 (0.40)
Raw TRT [sec] (Norm. TRT)	1.68 (1.00)	1.47 (0.88)	1.28 (0.76)	1.14 (0.68)	1.13 (0.68)

likelihoods), it accelerates the acoustic likelihood calculations.

#Act-SLs is large with batch calculation compared with the other techniques. However, breaking down it, we can see that #Nml-SLs is small compared with the number of low-cost batch state likelihood calculations (#Bat-SLs). Based on this mechanism, although there are Rdn-SLs, batch calculation can accelerate the acoustic likelihood calculations as described in Sect. 2.2.

We can confirm that R&B and R&B w/ ABO inherit properties from recycling and batch calculation. For example, their #Rcy-SLs are close to that of recycling, their #Nml-SLs are smaller than those of recycling and batch calculation, and their #Act-SLs (and therefore the SLC-Rate) take values between those of recycling and batch calculation.

With R&B w/ ABO, #ABO-SLs are very small. We analyzed these results in greater detail. Acoustic backing-off occurred during the decoding for all of the 850 evaluation utterances. Comparing the recognition results of R&B and R&B w/ ABO utterance by utterance, we found they are same for 833 utterances and only different for the remaining 17 utterances. However, for 14 of the remaining 17 utterances, R&B w/ ABO provided more accurate recognition results than those given by R&B. From these results, we can again confirm the stable effect of acoustic backing-off as regards preventing the word accuracy from degrading.

5.3 Analysis Results in Noisy Environment

Table 6 is the noisy environment version of Table 5. In Sect. 4.3, it was confirmed that, compared with the clean environment case, raw ALCTs and TRTs increase with all of the techniques in a noisy environment. We can confirm that these increases are caused by the increases in the #Act-SLs (and therefore the SLC-Rate) and #Req-SLs.

With recycling, despite the #Req-SLs increases compared with the clean environment case, #Rcy-SLs decreases. In addition to the increases of #Act-SLs and #Req-SLs, this decrease of #Rcy-SLs also degrades the acceleration performance of recycling. In a clean environment, i.e. under the acoustically matching condition between training and evaluation, it is expected that a few monophone states that match the feature parameter of the current frame will have high likelihods and be well divided from the other monophone states in frame by frame monophone state likelihood calculations. On the other hand, in a noisy environment, the discrimination abilities of the monophone states trained using clean speech data are degraded, and then all of the monophone states tend to have similar low likelihoods. Therefore, if we employ the same recycling threshold for both the clean end noisy environments (as described in Sects. 4.2 and 4.3, the recycling coefficient α was finally set at 0.750 for both the clean and noisy environments), #Rcy-SLs in a noisy environment tends to be smaller than that in a clean environment as shown in Tables 5 and 6. These results indicate that, to improve the robustness of recycling in a noisy environment, we should include recycling criteria other than the thresholding of the monophone state likelihoods.

As with the other techniques, also with batch calculation, #Act-SLs increases compared with the clean environment case. However, the increase in #Nml-SLs is small. And the increase in #Act-SLs is mainly accomplished by the increase in the low-cost batch state likelihood calculations (#Bat-SLs). Therefore, even in a noisy environment, with batch calculation, increases in ALCTs and TRTs are small compared with the other techniques as described in Sect. 4.3.

R&B and R&B w/ ABO inherit properties from recycling and batch calculation as in a clean environment. Their performance degradation is mainly attributable to that of recycling as described above.

With R&B w/ ABO, #ABO-SLs increases compared with that in a clean environment according to the increase in the reliability coefficient γ from 0.375 to 0.575. Acoustic backing-off occurred during the decoding for all of the 850 evaluation utterances as in a clean environment case. For 801 utterances, R&B and R&B w/ ABO provided the same recognition results. For 39 of the remaining 49 utterances, R&B w/ ABO gave more accurate recognition results

than R&B. From these results, we can again confirm the stable effect of acoustic backing-off in preventing the word accuracy from degrading in a noisy environment as originally reported [13] and as in a clean environment. However, acoustic backing-off in R&B w/ ABO is also based on the thresholding of monophone state likelihoods as with recycling. Thus, if some other reliability criteria could be included, we could expect to obtain better performance for acoustic backing-off.

6. Conclusion and Future Work

We proposed an efficient combination of state likelihood recycling and batch state likelihood calculation based on conditional fast processing and acoustic backing-off for accelerating acoustic likelihood calculation in an HMM-based speech recognizer. We conducted large vocabulary spontaneous speech recognition experiments using the four different CPU machines under two different environmental conditions. Our combined acceleration technique showed the best performance under all experimental conditions. Compared with the baseline speech recognizer, the combined acceleration technique achived reductions in the acoustic likelihood calculation time and total recognition time respectively of up to 69% and 38% in a clean environment and up to 60% and 32% in a noisy environment. Detailed analyses based on state likelihood type clustering clearly revealed the acceleration and environmental dependency mechanisms of each technique and confirmed the effectiveness of our combined acceleration technique.

In future work, to improve the robustness of state likelihood recycling and the combined technique in noisy environments, in accordance with the detailed analysis results, we should include some recycling, potential activity and reliability criteria other than the thresholding of monophone state likelihoods, e.g. confidence based criteria [14]. Moreover, we can combine other accelration techniques, e.g. Gaussian reduction [1], SSE [8] and GPU [9], with our technique for a further acceleration of the acoustic likelihood calculations. The acceleration techniques used in this paper have up to three parameters, α, β and γ. As described in Sects. 4.2 and 4.3, we manually adjusted them while allowing slight degradation of recognition accuracy. We have to make a guideline to ease these adjustment processes. And for that, we have to reveal the sensitivities or stabilities of the parameters against the changing of the conditions such as speakers, noise types, acoustic/language models and decoder settings.

References

[1] K. Shinoda and K. Iso, "Efficient reduction of Gaussian components using MDL criterion for HMM-based speech recognition," Proc. ICASSP, pp.869–872, 2002.
[2] S. Takahashi and S. Sagayama, "Four-level tied-structure for efficient representation of acoustic modeling," Proc. ICASSP, pp.520–523, 1995.
[3] S. Sagayama and S. Takahashi, "On the use of scalar quantization for fast HMM computation," Proc. ICASSP, pp.213–216, 1995.
[4] E. Bocchieri, "Vector quantization for the efficient computation of continuous density likelihoods," Proc. ICASSP, pp.692–695, 1993.
[5] Y. Komori, M. Yamada, H. Yamamoto, and Y. Ohora, "An efficient output probability computation for continuous HMM using rough and detail models," Proc. EUROSPEECH, pp.1087–1090, 1995.
[6] A. Lee, T. Kawahara, and K. Shikano, "Gaussian mixture selection using context-independent HMM," Proc. ICASSP, pp.69–72, 2001.
[7] S. Kanthak, K. Schutz, and H. Ney, "Using SIMD instructions for fast likelihood calculation in LVCSR," Proc. ICASSP, pp.1531–1534, 2000.
[8] M. Afify, F. Liu, H. Jiang, and O. Siohan, "A new verification-based fast-match for large vocabulary continuous speech recognition," IEEE Trans. SAAP, vol.13, no.4, pp.546–553, July 2005.
[9] P.R. Dixon, T. Oonishi, and S. Furui, "Harnessing graphics processors for the fast computation of acoustic likelihoods in speech recognition," Comput. Speech Lang., vol.23, pp.510–526, 2009.
[10] M. Saraclar, M. Riley, E. Bocchieri, and V. Goffin, "Towards automatic closed captioning: Low latency real time broadcast news transcription," Proc. ICSLP, pp.1741–1744, 2002.
[11] A. Ogawa, S. Takahashi, and A. Nakamura, "Efficient combination of likelihood recycling and batch calculation based on conditional fast processing and acoustic back-off," Proc. ICASSP, pp.4161–4164, 2009.
[12] G. Saon, D. Povey, and G. Zweig, "Anatomy of an extremely fast LVCSR decoder," Proc. Interspeech, pp.549–552, 2005.
[13] J. de Veth, B. Cranen, and L. Boves, "Acoustic backing-off as an implementation of missing feature theory," Speech Commun., vol.34, pp.247–265, 2001.
[14] H. Jiang, "Confidence measures for speech recognition: A survey," Speech Commun., vol.45, pp.455–470, 2005.
[15] T. Hori, Y. Noda, and S. Matsunaga, "Improved phoneme-history-dependent search for large-vocabulary continuous speech recognition," Proc. Interspeech, pp.1809–1813, 2001.
[16] A. Ogawa, Y. Noda, and S. Matsunaga, "Novel two-pass search strategy using time-asynchronous shortest-first second-pass beam search," Proc. Interspeech, pp.290–293, 2000.
[17] J.D. McCalpin, "Memory bandwidth and machine balance in current high performance computers," IEEE Computer Society Technical Committee on Computer Architecture (TCCA) Newsletter, pp.19–25, Dec. 1995.
[18] J.D. McCalpin, STREAM: Sustainable memory bandwidth in high performance computers, http://www.cs.virginia.edu/stream/

Atsunori Ogawa received the B.E. and M.E. degrees in information engineering, and the Ph.D. degree in information science from Nagoya University, Aichi, in 1996, 1998, and 2008, respectively. Since joining NTT Laboratories in 1998, he has been engaged in research on speech recognition. He is a member of the IEEE, ISCA and Acoustical Society of Japan (ASJ). He received the ASJ Best Poster Presentation Award in 2003 and 2006, respectively.

Satoshi Takahashi received the B.E., M.E., and Ph.D. degrees in information science from Waseda University, Tokyo, in 1987, 1989, and 2002, respectively. Since joining NTT in 1989, he has been engaded in speech recognition and pattern recognition. He is a member of the Acoustical Society of Japan (ASJ).

Atsushi Nakamura received the B.E., M.E., and Dr.Eng. degrees from Kyushu University, Fukuoka, Japan, in 1985, 1987 and 2001, respectively. In 1987, he joined Nippon Telegraph and Telephone Corporation (NTT), where he engaged in the research and development of network service platforms, including studies on application of speech processing technologies into network services, at Musashino Electrical Communication Laboratories, Tokyo, Japan. From 1994 to 2000, he was with Advanced Telecommunications Research (ATR) Institute, Kyoto, Japan, as a Senior Researcher, working on the research of spontaneous speech recognition, construction of spoken language database and development of speech translation systems. Since April, 2000, he has been with NTT Communication Science Laboratories, Kyoto, Japan. His research interests include acoustic modeling of speech, speech recognition and synthesis, spoken language processing systems, speech production and perception, computational phonetics and phonology, and application of learning theories to signal analysis and modeling. Dr. Nakamura is a senior member of the IEEE and a member of the Acoustical Society of Japan (ASJ). Also he serve as a Vice Chair of the IEEE Signal Processing Society Kansai Chapter. He received the IEICE Paper Award, and the Telecom-technology Award of The Telecommunications Advancement Foundation, in 2004 and 2006, respectively.