Efficient Embedded Speech Recognition for Very Large Vocabulary Mandarin Car-Navigation Systems

Yanmin Qian, *Student Member*, IEEE, Jia Liu, *Member*, IEEE,
and Michael T. Johnson, *Senior Member*, IEEE

Abstract — *Automatic speech recognition (ASR) for a very large vocabulary of isolated words is a difficult task on a resource-limited embedded device. This paper presents a novel fast decoding algorithm for a mandarin speech recognition system which can simultaneously process hundreds of thousands of items and maintain high recognition accuracy. The proposed algorithm constructs a semi-tree search network based on mandarin pronunciation rules, to avoid duplicate syllable matching and save redundant memory. Based on a two-stage fixed-width beam-search baseline system, the algorithm employs a variable beam-width pruning strategy and a frame-synchronous word-level pruning strategy to significantly reduce recognition time. This algorithm is aimed at an in-car navigation system in China and simulated on a standard PC workstation. The experimental results show that the proposed method reduces recognition time by nearly 6-fold and memory size nearly 2-fold compared to the baseline system, and causes less than 1% accuracy degradation for a 200,000 word recognition task*[1].

Index Terms — Beam-search, Search network, Speech recognition, Word-level pruning

I. INTRODUCTION

Voice centric interfaces are widely available in modern embedded devices, including low-cost versions. The applications have grown from speaker-dependent name dialing to speaker-independent capabilities [1]. Recently available advances include capabilities like voice-enabled SMS, email, and even navigation with voice. This evolution is enabled by the fast developments in speech recognition robustness, network capabilities, and increased computational power in small devices.

Automatic speech recognition systems obtain good results in laboratory conditions, but they still require large memory and CPU resources. State of the art speech-to-text systems are usually based on acoustic models composed of several million parameters. Moreover, decoding a segment of voice with conventional methods in a very large vocabulary system requires a huge amount of computational power (often more than 5 or 6 times real time on a standard workstation) [2].

Currently, there is an increasing demand to access large databases by voice on embedded devices, e.g., searching a destination Point-of-Interest (POI) in a car navigation system. These embedded speech recognition systems, which do not have the same large memory size and computational power as a standard workstation, need to significantly reduce both decoding complexity and memory demand. This is a challenging problem for a resource-limited device since a much faster and memory efficient decoding algorithm must be developed to compensate for hardware limitations.

The number of placenames for a POI system in China is very large. Unlike other countries, placenames typically consist of a combination of arbitrarily selected mandarin characters, so the number of possibilities is unlimited. Even in a single city, there are hundreds of thousands; e.g., Shanghai has about 165,000 POIs, and Beijing has about 157,000 POIs. Therefore in-car navigation systems using voice in China need to manage a great deal of vocabulary.

The traditional fast search algorithm has been practical for up to tens of thousands of words on mobile devices [3]; however, this approach cannot manage hundreds of thousands of words. Some very large vocabulary systems have been implemented, but these methods either take too much time to recognize or cause a recognition accuracy decline due to over-pruning.

Recently, to address this problem, Hoon Chung et al. [4], [5], have presented a method based on a multi-pass search scheme that separates the search spaces into acoustic and lexical spaces, and achieved improved performance. In this paper, we propose an alternative novel fast decoding algorithm. We first concentrate on constructing a semi-tree search network based on mandarin pronunciation rules, to avoid duplicate syllable matching and save redundant memory. Then a two-stage variable beam-width pruning strategy with a frame-synchronous word-level pruning is employed to significantly reduce recognition time.

The remainder of this paper is organized as follows: In Section II, we briefly introduce our baseline system, and then a detailed description of our fast recognition strategy is proposed. In Section III, we give experimental results simulated on a standard PC workstation about our algorithm. Finally, we summarize our algorithm and give conclusions in Section IV.

[1] This work was supported by the National High Technology Research and Development Program of China (NO. 2006AA010101 and NO. 2007AA04Z223) and National Natural Science Foundation of China and Microsoft Research Asia (NO. 60776800).

Yanmin Qian and Jia Liu are with the Department of Electronic Engineering, Tsinghua University, Beijing 100084, China (e-mail: qianym07@mails.tsinghua.edu.cn).

Michael T. Johnson is with the Department of Electrical Engineering, Marquette University, Milwaukee, Wisconsin 53201, USA. He is now a visiting professor with the Department of Electronic Engineering, Tsinghua University, Beijing 100084, China (e-mail: mike.johnson@marquette.edu).

Contributed Paper
Manuscript received June 11, 2009

II. Fast Speech Recognition Strategy

A. Baseline System Review

The sampling frequency of the baseline system is 8KHz, using 16-bit linear quantization, with a frame length of 32ms with 16ms shift. Each speech frame is parameterized as a 27-dimensional feature vector containing 12 Mel Frequency Cepstral Coefficients (MFCCs), 12 first order MFCC deltas, the C0 energy and its first and second order deltas [6]. This system utilizes CDHMM as the basic identification framework, using a phonetic system of initial/final (including 21 initials and 38 finals) to build the HMM models to achieve mandarin recognition.

The baseline system utilizes a two-stage fixed-width beam-search strategy, and first finds N-best candidates by a coarse match using monophone models (including 208 states, one Gaussian mixture per state). Then a detailed match is executed using biphone models (including 358 states, three Gaussian mixtures per state) to obtain the final recognition result [7], [8].

Using a Viterbi algorithm decoder, the calculation of the speech recognition likelihood mainly contains two aspects: probability calculation and network search [8]. When accessing a thousand word vocabulary, this is relatively simple. Even if constructing a traditional parallel network, as illustrated in Figure 1, the consumption of network search is very low, with the calculation probability occupying most of the recognition time. In such a case, the two-stage fixed-width beam-search strategy of the baseline system effectively compresses probability calculation. But when scaling to hundreds of thousands of words, the search time of the original strategy increases dramatically, as shown in Table I, far beyond the scope of real-time requirements. Therefore, in order to reduce recognition time, we need to find a more effective recognition strategy.

TABLE I
THE TIME CONSUMPTION TIME OF NETWORK SEARCH AND PROBABILITY CALCULATION IN BASELINE SYSTEM

Size of vocabulary	Network search	Probability calculation
1000	12.1%	87.9%
150000	95.4%	4.6%

B. A New Semi-Tree Search Network Construction

For a medium size isolated word recognition task, the traditional search network uses the parallel network construction framework shown in Figure 1. Every time a new recognition task begins, Viterbi search algorithm is applied frame by frame, and we retain the highest scores as candidates for the final recognition result. When the recognition task increases to tens of thousands or even hundreds of thousands, a large number of syllables are duplicated. Continuing to construct a network using the traditional method causes substantial duplicate syllable matching, which results in wasted recognition resources, comprising memory size and computational time. This is particularly true in mandarin, due to the large number of homonyms.

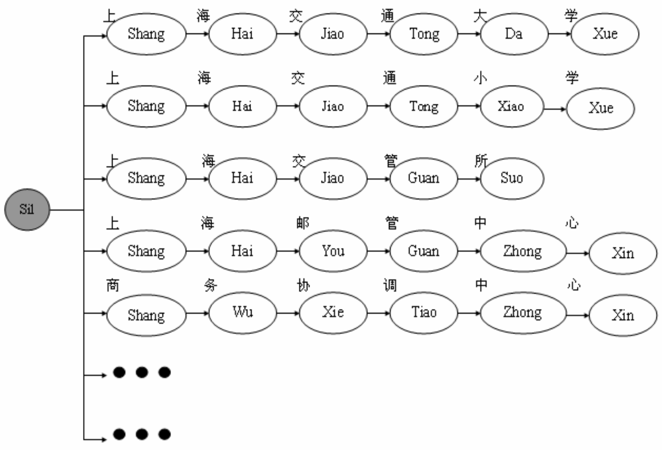


Fig. 1. Traditional parallel network used in embedded baseline system.

In order to overcome these shortcomings, it is possible a full-tree network can be used, as shown in Figure 2. As we can see from Figure 2, when there are duplicate syllables between the entities, the match of these syllables is only executed once. This full-tree network allows fast decoding and saves computation time. However, implementation of this form needs a large number of link lists and pointers for bookkeeping, so the decline in the memory usage is not significant.

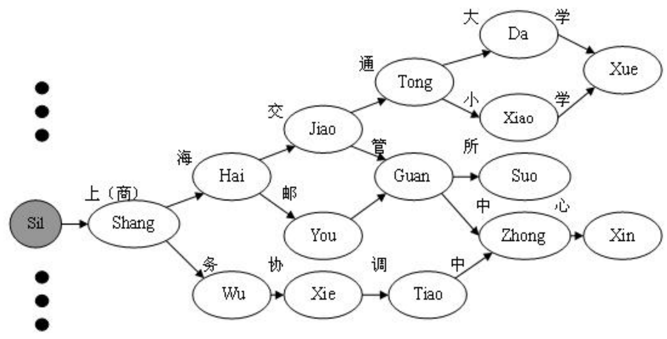


Fig. 2. The full-tree network.

In order to improve both recognition time and memory size of the decoding network, we construct a semi-tree search network according to mandarin pronunciation rules, as shown in Figure 3. As illustrated in Figure 3, in this form, syllables are merged only if they have the same parent node, leading to a strict tree data structure that is easier to trace. This data structure does not require as much booking, and can be realized by a simple index table. There are only 404 toneless syllables in mandarin [9], so there are 404 semi-trees and the branching factor and matching time will be restricted to these, no matter how large the vocabulary is. When accessing a very large vocabulary, such repeated syllable phenomenon is more prominent, and this search network structure can help to save not only a great deal of search time but also memory requirements.

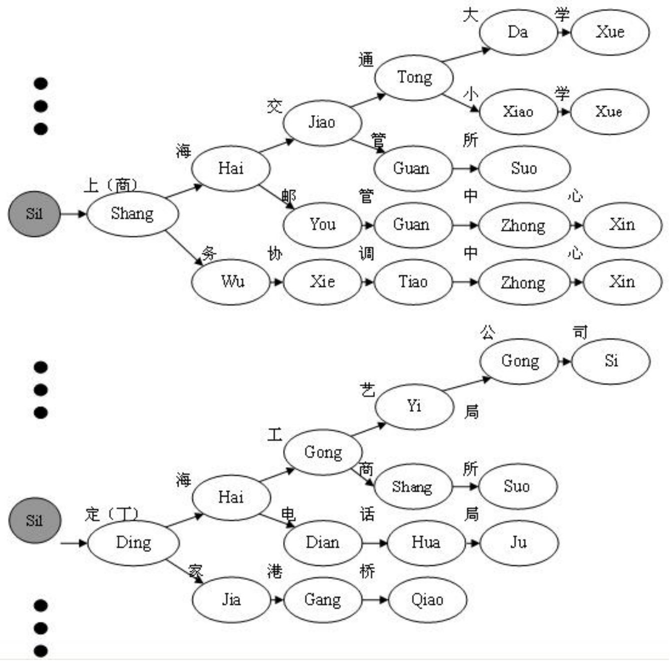


Fig. 3. The semi-tree search network proposed in this paper.

C. Fast Decoding Algorithm

1) Variable Beam-Width Pruning

Viterbi search is a breadth first search algorithm, and the time-synchronous nature of viterbi search implies a 2-D left-to-right traversal space. Since even a medium vocabulary system consists of thousands of HMM states, the state-time matrix quickly becomes too large to compute in real time. So a conventional fixed-width beam-search strategy is proposed, in which only the N-best state are considered to be active and to propagate to the next frame, to keep the computation within manageable limits.

Experimentally, the narrower the beam-width, the shorter the recognition time would be, but also the lower the recognition accuracy. One possible method to reduce computation time and also maintain high recognition accuracy is using time decaying widths. As the beam search continues, the possible paths become more confirmed, and the search beam-width can be narrower with less recognition accuracy degradation. We have derived a simple but effective time-decaying beam-width function to adjust the beam-width dynamically:

$$B_r = \max\left\{-n \cdot f_a + B_{MAX}, B_{MIN}\right\} \tag{1}$$

Where n is the frame number, f_a is the attenuation factor of the beam-width, and B_{MAX} , B_{MIN} , and B_r denote the upper bound, lower bound and final beam-width, respectively.

2) Frame-Synchronous Word-Level Pruning

From Table I, we know that network search time is the main issue when the recognition vocabulary is very large. It is important to find a method which can reduce the number of candidate terms, while constraining recognition accuracy degradation.

Based on perplexity theory we have developed a frame-synchronous word-level pruning strategy. Without pruning, at the beginning the perplexity of the search is just 404, for the 404 toneless syllables in mandarin. The perplexity increases exponentially with network growth according to this same branching factor. Accordingly we apply an additional pruning method at the word-level, which can reduce the size of the search space as well as limiting recognition accuracy degradation.

Empirically, we use a step-linear function to approximate the non-linear perplexity change. When the perplexity is small, we adopt a steep linear function to rapidly increase pruning, and as the perplexity grows, we utilize a smooth linear function to slow the rate of increase:

$$W_r = \max\left\{-f_i \cdot (n - N_i) + W_i, W_{MIN}\right\}, i = 1, 2, 3, ... \tag{2}$$

Here n is the frame number, W_{MIN} and W_r are the lower bound and the final number of allowed candidates, N_i is the frame number at which the pruning is altered, and f_i is the parameter of each linear function for word-level pruning. The word-level pruning process is illustrated in Figure 4.

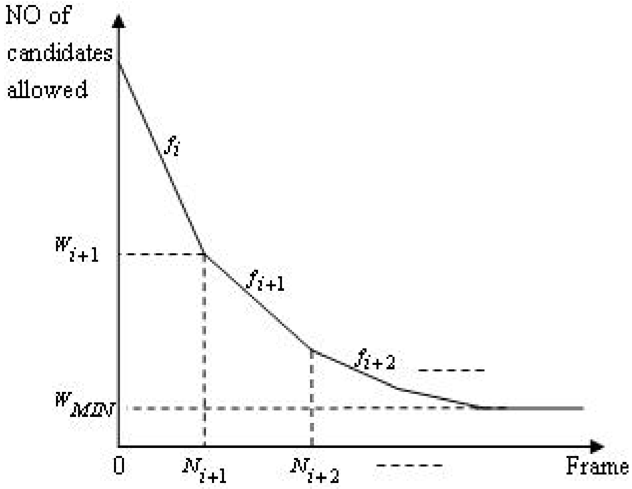


Fig. 4. The process of frame-synchronous word-level pruning.

III. Experimental Results

A. Experimental Data Presentation

The Chinese POI domain is very large, and we select the Shanghai POI as our isolated word recognition task. The vocabulary size is 165,176 and there are nearly 200,000 entries when taking into account multiple pronunciations.

The training set is the Chinese speech database of DB863, which is a continuous speech recognition corpus including 54 hours of male speech and 57 of female speech, with a total of 166 speakers. The test set has 9000 utterances, including 10 males and 5 females, and has been recorded by the author's laboratory. All the data used both in training and test is recorded in quiet office environment, with 8KHz sampling and 16-bit linear quantization.

All the experiments are conducted on a standard PC workstation with a processor operating at 1600MHz. The computer programs use fixed-point arithmetic to simulate embedded system resource constraints.

B. Evaluation of the Baseline System

We conducted experiments on the baseline system as described above. Recognition accuracy was measured by word accuracy and search time by real-time factor, defined as the total recognition time divided by the total time of the speech utterances.

For baseline results, we use several different fixed beam-widths, including 0.5, 1.0, 1.5, 2.0 and 2.5 multiples of the initial beam-width. Table II shows the recognition accuracy and recognition time using these fixed beam-widths.

TABLE II
FIXED-WIDTH BEAM-SEARCH RESULTS

Beam width ratio	0.5	1.0	1.5	2.0	2.5
1-best Accuracy (%)	72.21	85.60	90.32	91.10	91.27
5-best Accuracy (%)	76.50	93.32	96.53	96.90	96.98
10-best Accuracy (%)	78.09	94.02	97.15	97.82	97.88
Real-time factor	1.080	1.404	2.268	2.744	3.261

Baseline performance is considered to be a beam width ratio of 2.0, giving a real-time factor of 2.7 and 1-best accuracy of 91%. This shows that the baseline system is much too slow for applications on embedded devices.

C. Evaluation of the Algorithm Proposed In This Paper

1) Evaluation of the new semi- tree search network

To demonstrate the effectiveness of the new search network, we first evaluate the memory size used among different networks. Table III shows that the semi-tree network has lower memory size, about an improvement of 43.4% compared to the baseline system which uses a parallel network.

TABLE III
MEMORY SIZE COMPARISON BETWEEN THREE DIFFERENT NETWORKS

Network type	Parallel	Full-tree	Semi-tree
Memory size	16.07MB	11.10MB	9.09MB

To compare real-time factor, the semi-tree and baseline parallel network were implemented using identical fixed beam-width pruning. The results in Table IV show that the semi-tree search network is effective in reducing recognition time, giving an improvement of 36.4% in real-time factor.

TABLE IV
COMPARISON BETWEEN THE SEMI-TREE SEARCH NETWORK AND THE TRADITIONAL PARALLEL NETWORK WITH BASELINE DECODING

Network type	Semi-tree network	Parallel network
Real-time factor	1.746	2.744

2) Evaluation of the fast decoding algorithm

We first measure the performance of the variable beam-width pruning strategy. Figure 5 shows the tradeoff between speed and accuracy controlled by the factor f_a with a semi-tree network, compared to using different fixed beam-widths. The solid curve is the 1-best accuracy of the variable beam-width pruning, and the dashed curve is the 1-best accuracy of baseline system with different but fixed beam-widths. The point labeled with a * shows the selected performance tradeoff, where $f_a = 0.33, B_{MAX} = 20$ and $B_{MIN} = 5$, with accuracy of 90.81% and real-time factor of 1.316.

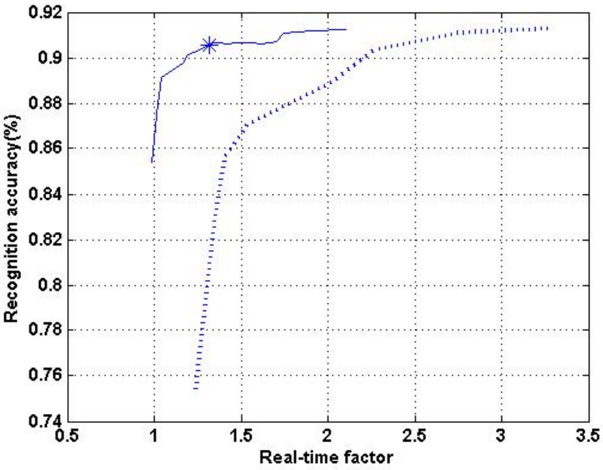


Fig. 5. Dependency of the accuracy and speed, variable beam-width pruning versus baseline system with different fixed beam-widths.

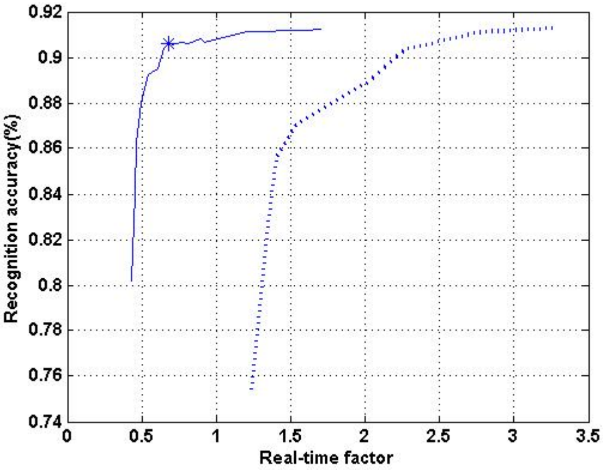


Fig. 6. Dependency of the accuracy and speed, frame-synchronous word-level pruning versus baseline system with different fixed beam-widths.

The performance of the frame-synchronous word-level pruning strategy is illustrated in Figure 6. This shows the tradeoff between speed and accuracy controlled by the factor f_i and N_i with a semi-tree network. The solid curve is the 1-best accuracy of the frame-synchronous word-level pruning, and the dashed curve is the 1-best accuracy of baseline system with different but fixed beam-widths. The point labeled with a * shows the selected performance tradeoff, where $W_{MIN} = 3000$, $f_1 = 7500, f_{i+1} = \frac{1}{2} f_i$, $N_1 = 15$, and $N_i = iN_1$, resulting in accuracy of 90.65% and real-time factor of 0.676.

As illustrated above, these two pruning strategies bring substantial benefits in the reduction of recognition time.

3) Evaluation of the overall performance

Table V shows the overall performance of the algorithm proposed in this paper, combining the semi-tree search network and both pruning strategies. The proposed algorithm achieves nearly a 6-fold speed improvement compared to the baseline system at a similar level of recognition accuracy, decreasing real-time factor from 2.744 to 0.474 with an accuracy reduction of only 0.5%.

TABLE V
THE OVERALL PERFORMANCE OF THE PROPOSED ALGORITHM

System	1-best Accuracy	5-best Accuracy	10-best Accuracy	Real-time factor
The proposed system	90.54%	96.50%	97.27%	0.4743
The baseline system	91.10%	96.90%	97.82%	2.7444

IV. CONCLUSION

In this paper, we propose a novel fast decoding algorithm for an embedded speech recognition system when accessing a very large vocabulary. Since computing time and memory size are the main constraints for realization in embedded systems, we construct a new semi-tree search network to avoid duplicate syllable matching as well as save redundant memory, and then employ a variable beam-width pruning strategy and a frame-synchronous word-level pruning strategy to perform the searching process more efficiently. The proposed recognition strategy improves recognition speed by 6-fold and memory size nearly 2-fold compared to the baseline system and causes very little accuracy degradation. Experimental results indicate that this new algorithm is suitable for a very large vocabulary recognition task in embedded systems.

REFERENCES

[1] J. Cohen, "Embedded speech recognition application in mobile phone: status, trends, and challenges", Proceedings of the ICASSP. Las Vegas,USA, pp. 5352-5355, 2008.

[2] C. Levy, G. Linares, P. Nocera, and J. F. Bonastre, "Reducing computation and memory cost for cellular phone embedded speech recognition system", Proceedings of the ICASSP. Montreal, Canada, vol. 5, pp. 309 - 312, 2004.

[3] M. Novak, R. Hampl, P. Krbec, V. Bergl, J. Sedivy, "Two-Pass search strategy for large list recognition on embedded speech recognition platforms", Proceedings of the ICASSP. Hong Kong, China, vol. 1, pp. 200 - 203, 2003.

[4] Hoon Chung, Jeon Park, Yun Lee, and Ikjoo Chung, "Fast speech recognition to access a very large list of items on embedded devices", IEEE Trans. on Consumer Electronics, vol. 54, no.2, pp. 803-807, May. 2008.

[5] Hoon Chung, Ikjoo Chung, "Memory efficient and fast speech recognition system for lowresource mobile devices", IEEE Trans. on Consumer Electronics, vol. 52, no. 3, pp. 792-796, Aug. 2006.

[6] Zhu Xuan, Wang Rui, and Chen Yi-ning, "Acoustic model comparison for an embedded phonme-based mandarin name dialing system", Proceedings of International Symposium on Chinese Spoken Language Processing, Taipei, Institule of China Computational Linguistics, pp. 83-86, 2002.

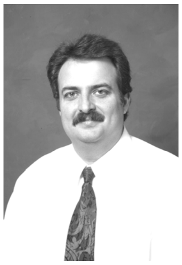
[7] Shi Yuanyuan, Liu Jia, and Liu Runsheng, "Single-Chip speech recognition system based on 8051 microcontroller core", IEEE Trans. on Consumer Electronics, vol. 47, no.1, pp. 149-153, Feb. 2001.

[8] Zhu Xuan, Chen Yi-ning, Liu Jia, and Liu Run-sheng, "Multi-pass decoding algorithm based on a speech recognition chip", Chinese ACTA ELECTRONICA SINICA, vol. 32, no. 1, pp. 150-153, 2004.

[9] Boon Pang Lim, Haizhou Li, and Bin Ma, "Using local & global phonotactic features in Chinese dialect identification", Proceedings of the ICASSP, Philadelphia, USA, vol. 1, pp. 577 - 580, 2005.

Yanmin Qian received his B.S degree in the Department of Electronic and Information Engineering from Huazhong University of Science and Technology, China, in 2007. He is currently a Ph.D candidate in the Department of Electronic Engineering, Tsinghua University, China. His research focuses upon fast decoding, robust speech recognition and large vocabulary speech recognition.

Jia Liu received his B.S, M.S and Ph.D degrees in communication and electronic systems from Tsinghua University, China, in 1983, 1986 and 1990 respectively. After receiving his PhD, he worked at the Remote Sensing Satellite Ground Station, Academic Sinica, and then as a Royal Society visiting scientist at Cambridge University Engineering Department from 1992-1994. He is currently a professor in the Department of Electronic Engineering at Tsinghua University. His research interests include signal processing, speech recognition, speaker identification, speech synthesis, speech coding and multimedia communication.



Michael T. Johnson received a B.S. degree in computer science engineering and a B.S. degree in engineering with electrical concentration, from LeTourneau University, Longview, TX, in 1989 and 1990, respectively. He received the M.S.E.E. degree from the University of Texas, San Antonio, in 1994 and the Ph.D. degree from Purdue University, West Lafayette, IN, in 2000. He worked as a design engineer and engineering manager from 1990 to 1996. Dr. Johnson became a member of IEEE in 1994 and was elevated to Senior Membership in 2002. He is currently an Associate Professor of Electrical and Computer Engineering at Marquette University, Milwaukee, WI. His primary research area is speech and signal processing, with interests in machine learning, statistical pattern recognition, and nonlinear signal processing.