RoboASR: A Dynamic Speech Recognition System for Service Robots

Abdelaziz A.Abdelhamid, Waleed H.Abdulla, and Bruce A.MacDonald

Electrical and Computer Engineering
The University of Auckland, New Zealand
aabd127@aucklanduni.ac.nz,{w.abdulla,b.macdonald}@auckland.ac.nz

Abstract. This paper proposes a new method for building dynamic speech decoding graphs for state based spoken human-robot interaction (HRI). The current robotic speech recognition systems are based on either finite state grammar (FSG) or statistical N-gram models or a dual FSG and N-gram using a multi-pass decoding. The proposed method is based on merging both FSG and N-gram into a single decoding graph by converting the FSG rules into a weighted finite state acceptor (WFSA) then composing it with a large N-gram based weighted finite state transducer (WFST). This results in a tiny decoding graph that can be used in a single pass decoding. The proposed method is applied in our speech recognition system (RoboASR) for controlling service robots with limited resources. There are three advantages of the proposed approach. First, it takes the advantage of both FSG and N-gram decoders by composing both of them into a single tiny decoding graph. Second, it is robust, the resulting tiny decoding graph is highly accurate due to its fitness to the HRI state. Third, it has a fast response time in comparison to the current state of the art speech recognition systems. The proposed system has a large vocabulary containing 64K words with more than 69K entries. Experimental results show that the average response time is 0.05% of the utterance length and the average ratio between the true and false positives is 89% when tested on 15 interaction scenarios using live speech.

Keywords: Human-robot interaction, automatic speech recognition, weighted finite state transducers

1 Introduction

For multi-modal communications in cognitive neuroscience robotic, robot speech recognition is one of the most important and effective means of natural communication between humans and robots [1][2]. Currently, speech recognition systems can be used for HRI, but they impose some restrictions in order to obtain effective speech recognition accuracy. Some of these restrictions are: the limited/small vocabulary, the usage of a headset or the need for noise free environments. However, these ideal conditions are not always suitable for realistic interaction with robots [3]. This research addresses the first restriction in a dynamic and efficient way.

Current state-of-the-art robotic speech recognition systems are usually based on either FSG or statistical models based on N-grams [4]. The FSG based systems have the advantage of achieving high sentence accuracy but at the same time suffer from a high rate of false positives. Besides, these systems are restricted to a limited number of sentences. The N-gram based systems can deal with a large vocabulary in real time, but suffer from low sentence accuracy and a high rate of false positives. In the literature, there are several efforts to exploit the advantages of both FSG based and N-gram based decoders by integrating them together into multi-pass decoders [4] [5].

In [6], key-phrase spotting in longer sentences through the combination of FSG and N-gram decoders is presented. This approach is based on the assumption that the key-phrase of interest is surrounded by a carrier set of words. The N-gram decoder is used to recognize the surrounding words and once it recognizes the first word in the key-phrase of interest, the N-gram decoder works to recognize the remaining words of the key-phrase. Then, the score of the recognized key-phrase is compared to a predefined threshold to decide its correctness. This fine tuning is done on a very low level, that makes it difficult to switch to another FSG or N-gram decoder easily.

In [5], FSG and N-gram decoders are used independently and simultaneously to process the speech data using a set of common acoustic models. The idea of this approach is to compare the best hypothesis resulting from FSG decoder along with the N-best hypotheses resulting from N-gram decoder. The advantage of this approach is that it reduces the ratio of false positives, but the domain was restricted to only 36 words and a command grammar.

In [7], an FSG-decoder is used with another complementary decoder that is also an FSG-decoder but based on N-grams. The objective from this approach was to reduce the false positives through off-line training of the first decoder on sentences with similar meanings. In this approach the result of the first decoder was not rated or rejected afterwards, but the search space is shaped to decrease the false positives.

In [4], the author used the same approach discussed in [5] and investigated its application in different forms of HRI including headset, ceiling boundary, and robot embedded microphones. The author used both FSG and N-gram based decoders in a multi-pass decoding process. This research concluded that the multi-pass decoding approach is effective but only suits domain specific scenarios. Although these multi-pass decoders are running simultaneously, it is expected that the response time of this approach is greater than that of individual decoders.

In this paper we present a speech recognition system that merges both FSG rules and N-gram models together into a single pass decoder without sacrificing the accuracy or the vocabulary size. The proposed system is designed using multi-threads and multi-buffers and can achieve a high recognition accuracy in a very short real time factor (RTF). The single pass decoder runs on a tiny decoding graph that is extracted from a pre-compiled large decoding graph using the composition operation of weighted finite state transducers (WFSTs) [8]. The pre-compiled large decoding graph is based on N-grams and a vocabulary containing 64K words.

The rest of this paper is structured as follows; section 2 discusses the tiny decoding graph extraction method. The general structure of the developed speech recognition system (RoboASR) is presented in section 3. Then, the preliminary experiments are discussed in section 4, followed by the conclusion and possible future directions in section 5.

2 The Proposed Approach

Before discussing the proposed single pass decoder in some details, we first present some relevant basics of the current state-of-the-art approaches in speech recognition systems.

2.1 Acoustic and Language Modelling

Modelling the speech signal could be approached through developing acoustic and language models. Currently, most speech recognition systems use statistical models to represent all the speech units. The most widely used and successful modelling approach is hidden Markov models (HMM) [9]. Another component in speech modelling is the language modelling that is used to model the constraints imposed on the spoken sentence by the grammar (or syntax) to determine the optimal sentence in the language. There are two methods to represent the language models namely: FSGs and statistical N-grams. Both of these methods are based on a dictionary that defines the sequence of phones constituting each word. The main difference between FSG and N-grams is that an FSG is an automaton of a predefined set of transitions between words, while N-grams are statistically trained based on the measured frequency of each word. The name N-grams comes from the dependency between the word and its preceding (N-1) words. The training of higher order N-grams usually requires a huge amount of training data, so that Bi-grams and Tri-grams are usually used in current state-of-the-art speech recognition systems.

The actual decoding of the speech utterance is based on searching the acoustic and language models to find out the best fitting hypothesis. There are many approaches for doing this search, the most common approach that is currently used is Viterbi beam search that searches for the best decoding hypothesis with the possibility to prune away the hypotheses with small scores.

2.2 Weighted finite state transducers (WFSTs)

WFSTs are the current state-of-the-art method used for building speech decoding graphs through the integration of the speech knowledge sources together into a single decoding graph [10]. The common speech knowledge sources are the acoustic knowledge represented by HMMs, syntax knowledge represented by N-grams, and lexical knowledge represented by the pronunciation dictionary. The integrated decoding graph can be constructed through the application of a series of WFST operations such as composition, determinization, epsilon removal, and weight pushing [11].

In this research, we constructed a large decoding graph that will be used in the extraction of the tiny decoding graph that will be used later in the single pass decoding process. The sizes of the knowledge sources used in this research along with the series of WFST operations that are used to generate the large decoding graph are shown in table 1. This table presents the sequence of WFST operations applied to the speech knowledge sources to generate the large decoding graph along with the size of the intermediate graphs resulting from each operation.

Table 1. The operations and intermediate graphs used for building the large WFST decoding graph.

Operation	#States	#Transitions
C	1,681	84,080
L	523,083	592,837
G	595,765	1,327,969
T	63,999	191,997
$det(L)$	209,919	279,673
$C \circ det(L)$	346,452	550,709
$G \circ T$	886,099	1,932,311
$(C \circ det(L)).(G \circ T)$	5,579,208	8,082,205

C : Context dependency WFST, L : Lexicon WFST, G : Tri-gram WFST, T : Silence WFST, $\circ$: The composition operation, det : The determinization operation.

2.3 Tiny decoding graph extraction

The basic idea of this research is to dynamically generate a tiny decoding graph for each state in the state-based HRI. In the state-based HRI the interaction with the robot is defined through a sequence of HRI states and each state has a set of allowable commands (these commands can be words or sentences) and actions as shown in Fig. 1 [12]. In order to generate a tiny decoding graph for certain HRI state, firstly, the FSG rules defining this HRI state are converted to a weighted finite state acceptor (WFSA) $A(y, y)$ where each transition in this acceptor has the same word as an input and output. Secondly, the generated WFSA is composed with the large WFST decoding graph $T_1(x, y)$ to get the tiny WFST $T_2(x, y)$ for that HRI state as follows:

$$T_2(x, y) = [T_1 \circ A] = \bigoplus_{y} [T_1(x, y)] \otimes [A(y, y)] \tag{1}$$

where $\oplus$ and $\otimes$ are the *semiring-add* and *semiring-multiply* operations respectively, for more details about the WFST operations please refer to [10]. The proposed composition algorithm used to generate the tiny WFST T_2 from the large WFST T_1 is shown in algorithm 1. This algorithm describes the composition operation used to generate the tiny decoding graph. The basic idea of this algorithm is to merge the transitions of both large WFST and WFSA where the output labels of the transitions in the first graph coincide with the input labels of the transitions in the second graph. The notations used in this algorithm are defined as follows. Q is a temporary set used to hold set of states, , $\wp$ is a queue used to hold the set of pairs of states yet to be examined, I_G and F_G are the set of initial and final states of the graph G respectively, q is a state in the decoding graph, e is a transition, $E[q]$ is the set of all transitions getting out from the state q, and λ_G is the weight of the final state of the graph G. The default weights of the constructed WFSA are all zeros. Then, the resulting tiny WFST is stored on disk and named with the same HRI state number (to be easily recalled while running the HRI scenarios). While running the live interaction with the robot, the tiny decoding graph is loaded automatically for each interaction state and the single pass decoder runs on the loaded tiny WFST.

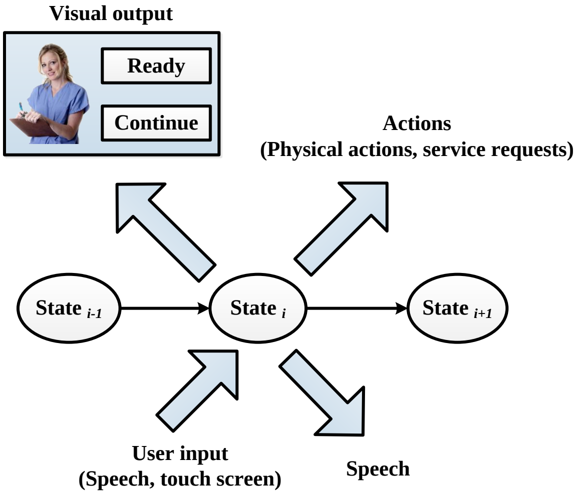

Fig. 1. The spoken HRI state.

To clarify the idea of the proposed approach, sample grammar rules used to describe a potential HRI state is shown in Fig. 2. These grammar rules are parsed and processed to generate the corresponding WFSA graph shown in Fig. 3. Then, the generated WFSA is composed with the large WFST using the proposed algorithm and a tiny WFST graph is produced for that HRI state.

Algorithm 1: Weighted Composition

```
Input: $Transducer\ T_1, Acceptor\ A$
Output: $Transducer\ T_2 = [T_1 \circ A]$
begin
    $Q \longleftarrow I_{T_1} \times I_A$
    $\wp \longleftarrow I_{T_1} \times I_A$
    while $\wp \neq \phi$ do
        $q = (q_1, q_2) \longleftarrow Head(\wp)$
        $Dequeue(\wp)$
        if $q \in I_{T_1} \times I_A$ then
            $I_{T_2} \longleftarrow I_{T_2} \cup \{q\}$
            $\lambda_{T_2}(q) \longleftarrow \lambda_{T_1}(q_1) \otimes \lambda_A(q_2)$
        end if
        if $q \in F_{T_1} \times F_A$ then
            $F_{T_2} \longleftarrow F_{T_2} \cup \{q\}$
            $\rho_{T_2}(q) \longleftarrow \rho_{T_1}(q_1) \otimes \rho_A(q_2)$
        end if
        for $each\ (e_1, e_2) \in E[q_1] \times E[q_2]$ do
            if $o[e_1] = i[e_2]\ and\ i[e_1] \neq o[e_2]$ then
                if $(\acute{q} = (n[e_1], n[e_2])) \notin Q$ then
                    $Q \longleftarrow Q \cup \{\acute{q}\}$
                    $Enqueue(\wp, \acute{q})$
                end if
                $E_{T_2} \longleftarrow E_{T_2} \uplus \{(q, i[e_1], o[e_2], w[e_1] \otimes w[e_2], \acute{q})\}$
            end if
        end for
    end while
end
```

```
<utterance>    = <confirmation>|<communication>
<confirmation> = yes | no | correct | wrong
<communication>= <command> | <action>
<command>      = abort | help | stop
<action>      = (go to <location>)|(come here)
<location>     = desk | chair | sofa
```

Fig. 2. Sample finite state grammar for a spoken HRI state.

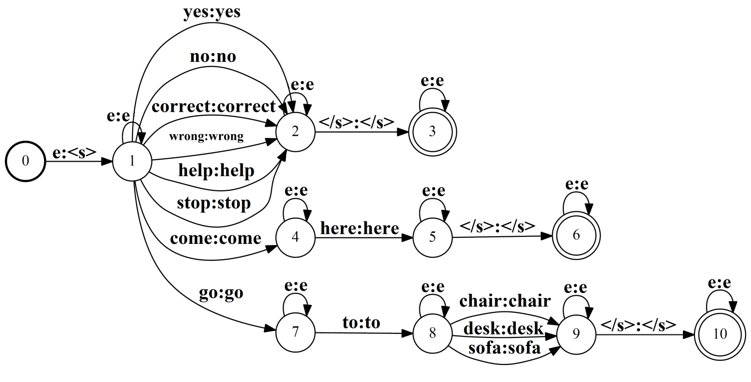

Fig. 3. WFSA graph corresponding to the sample grammar.

The resulting tiny WFST has hundreds of states and transitions, so it did not fit to be included in this paper.

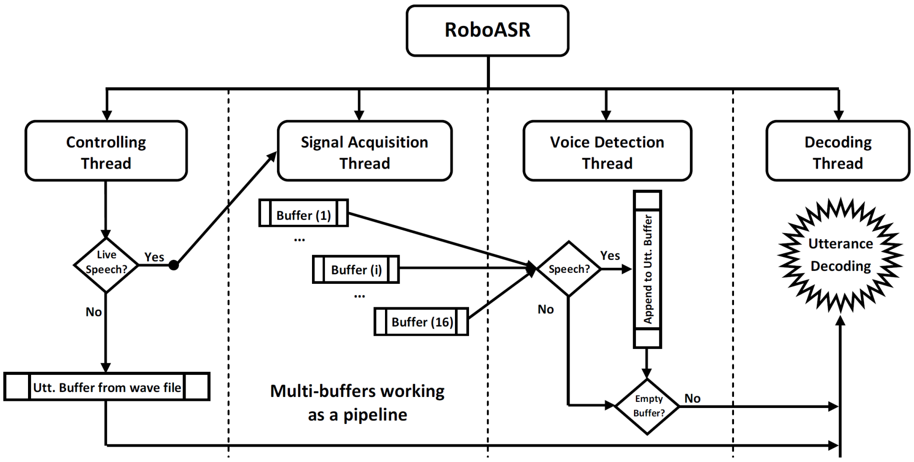

Fig. 4. The proposed ASR system based on multi-threads and multi-buffers.

3 RoboASR: The Developed System

In this research, we used the developed speech decoder presented in [13]. In comparison to other promising systems [14][15] the proposed speech recognition system has the advantage of being applicable to devices and robots with limited resources. The developed speech recognition system is fully implemented in C++. The next sections discuss the internal structure of the developed system.

3.1 The structure of the speech recognition system

The proposed speech recognition system is based on multi-threads to achieve a harmony in the processing of various operations that take place on the captured speech signal and to allow for a continuous signal capturing and decoding. Besides, a set of multi-buffers is used to efficiently handle the continuous capturing of the incoming audio stream. The structure of RoboASR is shown in Fig. 4, which is based on the following four threads:

1. **Controlling thread**: which controls the overall system and is responsible for loading the acoustic models and the tiny WFST graphs (only on demand) and initiating the signal acquisition thread.
2. **Signal acquisition thread**: this thread is responsible for continuous capturing of the speech signal from the sound card. The signal acquisition is done based on a series of multi-buffers working together as a pipeline.
3. **Voice detection thread**: while capturing the continuous audio stream, this thread is responsible for detecting which buffers contain a speech signal, then accumulate these buffers into another large buffer for decoding.
4. **Decoding thread**: once the large buffer is filled with the speech utterance from the signal acquisition thread, the decoding thread is activated to decode the speech signal in this large buffer.

The arrows in Fig. 4 indicate a process flow, while the dotted lines are used just to visually separate the tasks of each thread.

3.2 The structure of the single pass decoder

The developed decoder [13] contains several classes for loading the acoustic models (HMM), and the WFST decoding graph along with a memory pool that is used through the decoding process to handle the competing hypotheses. Additionally, this decoder contains special structures for holding the best token and for holding the set of competing hypotheses during the decoding process. The single pass decoding mechanism is based on the Viterbi algorithm though the implementation of a token passing technique with beam pruning to speed up the decoding process [16].

4 Experimental Results

4.1 Experimental Setup

The preliminary experiments established in research were based on two steps. First, a large WFST decoding graph was built based on N-grams (Tri-grams) and using a large vocabulary containing 64K words. Second, we described each spoken HRI state as a set of FSG rules, then these FSG rules for each scenario were converted to a WFSA and composed with the large WFST decoding graph to extract the tiny decoding graph. These two steps result in a set of a tiny decoding graphs one for each spoken HRI state.

To investigate the effectiveness of the proposed approach, we measured the overall rate of true positives (Tp) and false positives (Fp) for each HRI scenario from the single-pass decoder using the large N-gram based decoding graph, and an FSG based decoding graph [17], and compared these two decoders with the proposed decoder using the extracted tiny decoding graph. Then, the average ratio between the true and false positives was recorded.

All the experiments used live speech recorded in 16 bits format and a sampling rate of 16K Hz, and captured using head mounted microphone. The number of spoken HRI states involved in these experiments is 15 HRI states.

4.2 Results and discussion

To validate the appropriateness of the proposed approach for robots with limited resources, we compared the resources required by the traditional decoding graph and the proposed approach as shown in table 2. It is shown that the performance of the proposed tiny decoding graph is much better than that of the large one in terms of the Tp/Fp ratio and the average RTF. The proposed approach can achieve 89% Tp/Fp in an average RTF of 0.05 of the utterance length, that makes the proposed approach more appropriate for the resource limited devices and robots. Since the tiny decoding graphs are extracted to fit the HRI states, if the HRI state received an unexpected utterance, it will be approximated to the best decoding hypothesis, then a threshold is used to decide whether to keep or prune this best decoding hypothesis.

Table 2. Comparison between the large WFST and tiny WFST decoding graphs.

Resources	Large WFST	Tiny WFST
Num. States	5,579,208	300 (avg)
Num. Transitions	9,082,205	500 (avg)
Avg. RTF (xRT)	1.5	0.05
Memory (MB)	900	350
Tp/Fp ratio.	64%	89%

The significant evaluation of the proposed approach compared to the FSG and large N-gram based methods is shown in table 3. The results indicate that the proposed approach outperforms the other methods and achieves a Tp/Fp ratio of 89%.

The method we followed to measure the evaluation criteria (i.e. Tp and Fp) is shown in Fig. 4. In this figure, the correctly recognized commands are coloured with blue colour and, the miss-recognized commands are coloured with red colour. However, if the decoder recognized the command with low probability, in this case, the recognition result is pruned, and the command is rejected as shown in yellow coloured cells in this table.

Table 3. The performance of different decoders.

Decoder	Tp	Fp	Tp/Fp ratio
FSG decoder	77%	18%	81%
large N-gram decoder	64%	36%	64%
Proposed decoder	80%	10%	89%

Tp/Fp ratio = Tp / (Tp + Fp) * 100

Table 4. Sample commands and their recognition results.

Command	FSG	Large WFST	Tiny WFST
go to chair	🟦 go to chair	🟥 got chair	🟦 go to chair
go to sofa	🟦 go to sofa	🟥 got sofa	🟦 go to sofa
come here	🟥 go to chair	🟨	🟦 come here
correct	🟦 correct	🟦 correct	🟦 correct
wrong	🟥 no	🟨	🟥 no
help	🟨	🟥 held	🟦 help
yes	🟦 yes	🟦 yes	🟦 yes
no	🟦 no	🟦 no	🟦 no

Blue color : denotes *Tp* command, *Yellow color* : denotes Rejected command and *Red color* : denotes *Fp* command.

5 Conclusion

In this paper we present a new method for building the speech decoding graphs for state based spoken HRI. The proposed method is based on merging an FSG with N-grams decoding graphs to produce a more efficient tiny decoding graph. Also, we presented the structure of our speech recognition system (RoboASR) along with the structure of the decoding engine. The experimental results show the effectiveness of the proposed approach over the traditional N-gram based large decoding graphs for handling large vocabulary tasks.

The overall performance of the developed system can be improved by capturing the live speech signal using an array of microphones to handle the ambient noise. In addition, We may confirm the effectiveness of our approach in a real environment with reverberation and in a dynamically changing environment.

6 Acknowledgement

This work is supported by the R&D program of the Korea Ministry of Knowledge and Economy (MKE) and Korea Evaluation Institute of Industrial Technology (KEIT). [KI001836, Development of Mediated Interface Technology for HRI].

We acknowledge the support of the New Zealand Ministry for Science and Innovation. We thank ETRI for their contributions and help with the work.

References

1. T. Kanda, M. Shiomi, Z. Miyashita, H. Ishiguro, and N. Hagita, "Communication robot in a shopping mall," *IEEE Transactions on Robotics*, pp. 897–913, 2010.
2. K. K. Paliwal and K. Yao, "Robust speech recognition under noisy ambient conditions," in *Human-centric interfaces for ambient intelligence.* Academic Press, Elsevier, 2009.
3. F. Alonso-Martin and M. A.Salichs, "Integration of a voice recognition system in a social robot," *IEEE Transactions on Cybernetics and Systems*, pp. 215–245, 2011.
4. S. Heinrich and S. Wermter, "Towards robust speech recognition for human-robot interaction," in *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2011, pp. 468–473.
5. M. Doostdar, S. Schiffer, and G. Lakemeyer, "A robust speech recognition system for service-robotics applications," in *Proceedings of the International RoboCup Symposium*, 2008, pp. 1–12.
6. Q. Lin, D. Lubensky, M. Picheny, and P. S. Rao, "Key-phrase spotting using an integrated language model of N-grams and finite-state grammar," in *Proceedings of the European Conference on Speech Communication and Technology*, 1997, pp. 255–258.
7. M. Levit, S. Chang, and B. Buntschuh, "Garbage modeling with decoys for a sequential recognition scenario," in *Proceedings of the IEEE Workshop on Automatic Speech Recognition & Understanding*, 2009, pp. 468–473.
8. C. Allauzen and J. Schalkwyk, "Generalized composition algorithm for weighted finite state transducers," in *Proceedings of the International Speech Communication Association*, 2009.
9. L. Rabinar and B.-H. Juang, *Fundamental of speech recognition.* Prentice-Hall, 1993.
10. M. Mohri, F. Pereira, and M. Riley, "Weighted finite state transducers in speech recognition," *Transactions on Computer Speech and Language*, vol. 16, pp. 69–88, 2002.
11. J. R.Novak, N. Minemaysu, and K. Hirose, "Painless WFST cascade construction for LVCSR-Transducersaurus," in *Proceedings of the International Speech Communication Association*, 2011.
12. E. Broadbent, C. Jayawardena, N. Kerse, R. Q. Stafford, and B. A. MacDonald, "Human-robot interaction research to improve quality of life in elder care - An approach and issues," in *Proceedings of the Workshop on Human-Robot Interaction in Elder Care*, 2011, pp. 7–11.
13. A. A.Abdelhamid, W. H.Abdulla, and B. A.MacDonald, "WFST-based large vocabulary continuous speech decoder for service robots," in *Proceedings of the International Conference on Imaging and Signal Processing for Healthcare and Technology*, 2012, pp. 150–154.
14. A. Lee and T. Kawahara, "Recent development of open-source speech recognition engine Julius," in *Proceedings of the APSIPA*, 2009, pp. 131–137.
15. D. Rybach, C. Gollan, G. Heigold, B. Hoffmeister, J. Loof, R. Schluter, and H. Ney, "The RWTH Aachen university open source speech recognition system," in *Proceedings of the International Conference of Speech Communication Association*, 2009, pp. 2111–2114.

16. S. Young, N. Russell, and J. Thornton, "Token passing: A simple conceptual model for connected speech recognition systems," Tech. Rep., 1989.
17. S. Young, G. Evermann, M. Gales, T. Hain, D. Kershaw, X. A. Liu, G. Moore, J. Odell, D. Ollason, D. Povey, V. Valtchev, and P. Woodland, *The HTK book*. Cambridge University, 2009.