

# INTEGRATING MULTIMODAL LANGUAGE PROCESSING WITH SPEECH RECOGNITION

Srinivas Bangalore and Michael Johnston

AT&T Labs Research, Shannon Laboratory  
180 Park Ave, Florham Park, NJ 07932, USA

{srini,johnston}@research.att.com

## ABSTRACT

One of the critical challenges facing next-generation human-computer interfaces concerns the development of effective language processing techniques for utterances distributed over multiple input modes such as speech, touch, and gesture. Finite-state models for parsing, understanding, and integration of multimodal input are efficient, enable tight coupling of multimodal language processing with speech recognition, and provide a general probabilistic framework for multimodal ambiguity resolution. We describe an experiment that demonstrates the effectiveness of tight coupling of multimodal language processing in improving speech recognition performance with clean speech and with different levels of background noise. Our approach yields an average 23% relative sentence error reduction on clean speech.

## 1. INTRODUCTION

Multimodal interfaces enable more natural and effective human-computer interaction by supporting input and/or output over multiple channels such as speech, graphics, and gesture (Bolt 1980, Wahlster 1991, Neal and Shapiro 1991, Cohen et al 1997). Our specific concern here is with interfaces supporting input by combinations of speech with pen or touch. These interfaces stand to play a critical role in the migration of interaction from the desktop to small wireless portable computing appliances used for telephony, messaging, roving internet access, and navigation. Since these devices have limited screen real estate and no keyboard, speech and pen will be their primary input modes. Multimodal interfaces require effective methods for parsing, integrating, and understanding commands whose content is distributed over multiple input modes (Johnston et al 1997, Johnston 1998). Finite-state devices have been extensively applied to many aspects of language processing including, speech recognition (Pereira and Riley 1994), phonology (Kaplan and Kay 1994), morphology (Koskenniemi 1984), chunking (Abney 1991, Srinivas 1997), parsing (Roche 1999), and machine translation (Bangalore and Riccardi 1999). In the approach we employ here, parsing, integration, and understanding of multimodal inputs are performed by finite-state transducers which relate speech, gesture, and meaning (Johnston and Bangalore 2000).

The finite-state approach enables multimodal language processing to be tightly coupled with speech recognition so that gestural information directly influences the recognizer's search. In this paper, we describe an experiment which explores the extent to which this tight coupling can improve speech recognition performance. We also examine the relationship between background noise and compensation effects and the effects of tight coupling on the efficiency of speech recognition. Our testbed application is a mobile multimodal messaging and

corporate directory application – a variant of VPQ (Buntschuh et al 1998) with integrated multimodal commands. Section 2 outlines our finite-state approach. Section 3 describes the experiment, Section 4 the results, Section 5 related work, and Section 6 concludes the paper.

## 2. MULTIMODAL FINITE-STATE TRANSDUCERS

Multimodal integration and understanding involves parsing multiple input streams and merging their semantic content to build a combined semantic representation for a multimodal utterance. In the finite-state approach, this is achieved using a finite-state device that reads from multiple input tapes and writes onto a single output tape. In the case of speech and pen input there are three tapes, one for speech, one for pen gesture, and a third for their combined meaning. The structure and interpretation of multimodal commands are expressed in a multimodal context free grammar (MCFG). This is compiled into a multimodal FSA (finite-state automaton) using standard techniques for finite-state approximation of context-free grammars (Pereira and Wright 1997, Nederhof 1997). In the multimodal messaging application used in our experiment, users employ commands such as ‘*email this person and that organization*’ while gesturing on graphical representations of people and organizations. To illustrate the approach we present here only a small fragment (Figure 1) of the multimodal CFG used in the experiment.

CMD → V NP ε:ε:]	ENTRY → ε:e1:e1 ε:ε:]
CONJ → and:ε,	ENTRY → ε:e2:e2 ε:ε:]
V → email:ε:email([	ENTRY → ...
V → page:ε:page([	NP → DET N CONJ NP
N → person:Gp:person(	ENTRY NP → DET N
N → organization:Go:org(	ENTRY DET → this:ε:ε
N → department:Gd:dept(	ENTRY DET → that:ε:ε

Figure 1 MCFG fragment

The multimodal aspects of the grammar become apparent in the terminals. Each contains three components  $W:G:M$ , where  $W$  is for the spoken language stream,  $G$  is the gesture stream, and  $M$  is the combined meaning. The epsilon symbol  $\epsilon$  indicates when one of these is empty. The symbols in  $W$  are words. The symbols in  $G$  are of two types. Symbols like  $Go$  indicate the presence of a particular kind of gesture in the gesture stream, while symbols like  $e1$  are used as references to entities referred to by the gesture.  $Go$  represents a gestural reference to an organization on the display,  $Gp$  to a person,  $Gd$  to a department, and so on. When concatenated together, the symbols in  $M$  yield the semantic representation for the multimodal command. We employ a simple logical representation with predicates  $pred(\dots)$  and lists  $[a,b,\dots]$ , but many other kinds of semantic representation could be generated.

The use of variable references such as  $e1, e2, e3...$  abstracts over the specific content of gestures and avoids the need to represent all possible gesture contents in the grammar and network. For example, if a set of gestured coordinates was needed in the output meaning then all possible coordination sequences would have to be encoded in the network. Instead when gestures are processed, specific content such as coordinates or the ID of an object are stored in a finite set of variables:  $e1, e2, e3...$  and references to these are written onto the gesture input tape. The *ENTRY* productions in the MCFG are used to transfer variable references into the semantic representation (See Johnston and Bangalore 2000 for more details). Our sample MCFG is compiled into the three-tape FSA in Figure 2. The transition symbols of the approximated FSA correspond to the terminals of the MCFG.

While a three-tape finite-state automaton is feasible in principle (Rosenburg 1964) currently available tools for finite-state language processing (Mohri et al 2000) only support finite-state transducers (FSTs), which have two tapes. In order to implement our approach, we convert the three-tape FSA (Figure 2) into an FST, by decomposing the transition symbols into an input component ( $G \times W$ ) and output component  $M$ , thus resulting in a function,  $\mathfrak{I} : (G \times W) \rightarrow M$  (Figure 3) which we use to build the combined meaning representation. The domain of this function  $\mathfrak{I}$  can be further curried resulting in a transducer that maps from gesture to speech  $\mathfrak{R} : G \rightarrow W$  (Figure 4). This transducer captures the mutual constraints among speech and gesture inputs and provides a language model for speech recognition.

For our example command,  $\mathfrak{I}$  and  $\mathfrak{R}$  are used as follows. The gesture stream (Figure 5) is composed with  $\mathfrak{R}$  resulting in a transducer *GestLang* which encapsulates all of the possible word strings that could appear with those gestures. A projection on the output of *GestLang* is used as the language model for speech recognition. The result of speech recognition ‘*email this person and that organization*’ is re-integrated with the gesture information by composition with *GestLang*. The resulting FST

has its input and output ( $G \rightarrow W$ ) factored onto one tape ( $G \times W$ ) and this FSM is composed with  $\mathfrak{I}$  yielding a result transducer  $R$ . The meaning representation,  $\text{email}([\text{person}(e1), \text{org}(e2)])$  is read off the output tape of  $R$ .

### 3. EXPERIMENTAL EVALUATION

One of our goals in applying finite-state techniques to multimodal language processing is to explore tight coupling of constraints from gesture and other modes with speech recognition. In order to evaluate our approach, we designed an experiment which examines the extent to which the early integration of gestural information it enables can improve speech recognition performance.

We first developed a sizeable MCFG (417 rules and 2388 different words) for a realistic and useful multimodal messaging and corporate directory application. Users can find directory information and initiate calls, email, and faxes using combinations of speech and gesture. The grammar supports deictic expressions, proper names (400), conjunction, possessives, and numeral expressions. Sample commands are: ‘*email these two departments and this person*’, ‘*could you please send a fax to this person’s manager*’, ‘*his mobile phone number please*’, and ‘*send email to these three labs*’.

We then collected a set of one hundred sample commands from each of ten volunteer subjects. The subject pool contained six men and four women, five native and five non-native speakers of English. There were 891 words in the test corpus and an average sentence length of approximately 9 words. We set up a data collection environment in which each subject was prompted with the sequence of one hundred sample commands. They issued each command in turn and their speech signal was recorded. We recorded speech at 8000Hz using a Sennheiser noise-canceling microphone. In our multimodal messaging application, gesture input is limited to deictic pointing events, so there is little room for gesture recognition error. For the purposes of this initial experiment, we pre-computed the gesture stream appropriate for each command. This data collection

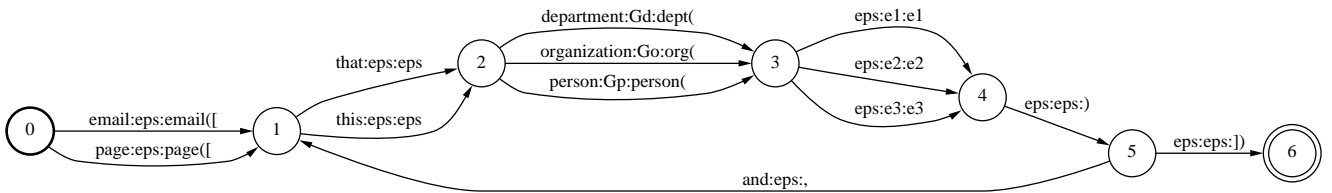


Figure 2 Three-tape FSA  $W:G:M$

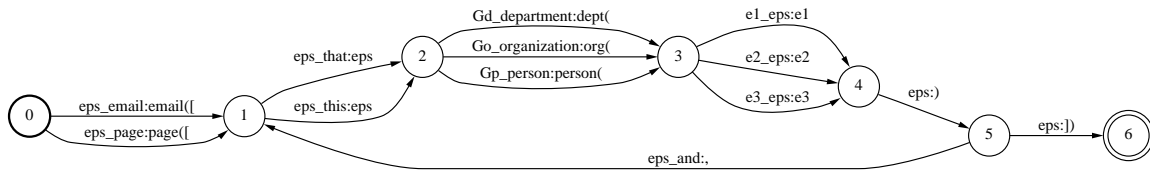


Figure 3  $\mathfrak{I} : (G \times W) \rightarrow M$  relating gesture and speech to meaning

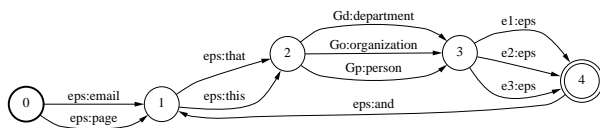


Figure 4  $\mathfrak{R} : G \rightarrow W$  relating gesture and speech

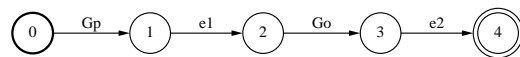


Figure 5 Gesture stream FSM

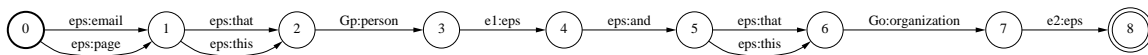


Figure 6 *GestLang* transducer

provided us with a corpus of 1000 spoken utterances and their associated gestures. This speech is closer in character to read speech. It is likely to be more consistent in timing and less disfluent than interactive speech to a multimodal system. However, since it is in fact easier to recognize read speech we believe this data provides us with a reasonable baseline for examination of compensation effects.

We used an HMM-based speech recognizer with an off-the-shelf acoustic model. Each speech file in the test corpus was recognized under a **without-gesture** condition and **with-gesture** condition at a range of beam widths from one to sixteen. The speech recognizer accepts a finite-state machine as its language model. For the **without-gesture** condition we used the output tape projection of the  $\mathcal{R}:G \rightarrow W$  transducer as the language model. As a result it contains all strings in the application grammar. For the **with-gesture** condition, we first encoded the gesture sequence for the current sample command into an FSM and composed it with the  $\mathcal{R}:G \rightarrow W$  transducer. We then used the output tape projection of the result as the language model. This restricted the language model for the **with-gesture** condition to only those strings compatible with the gesture stream for the sample command to be recognized.

We repeated these recognition experiments under a number of simulated noise conditions. We added two kinds of noise to the clean recorded signal: babble noise and car noise, each at three different signal-to-noise ratios (30 dB, 20 dB, 10 dB). For each recognition, we compared the result string to the reference string for that command, and used the standard ‘score’ mechanism from NIST to compute word and sentence accuracy. Sentence accuracy is more pertinent to the interface task that we describe, since it provides an indication of how many commands would have succeeded, but we also present word accuracy in keeping with practice in speech recognition research.

#### 4. RESULTS

In order to determine the effects of tight-coupling finite-state multimodal language processing with speech recognition, we compared word and sentence accuracy for the **with-gesture** condition to the **without-gesture** condition. The graph in Figure 6 shows word and sentence accuracy for the **with-gesture** and **without-gesture** conditions for clean speech at beam widths from 3 to 16 averaged over all 1000 utterances in the corpus. Both word and sentence accuracy were found to be higher for the **with-gesture** condition at all beam widths. A series of within-subject ANOVA tests showed these differences to be statistically significant at beam widths of 3 and above (all  $p < 0.05$ ). Both word and sentence accuracy for both conditions start to even off around beam width 11 or 12. The rest of the results we will present are for beam width 11. At beam width 11, average word accuracy for clean speech is 89.8% **without-gesture** and 91.8% **with-gesture**; an absolute error reduction of 2% (relative error reduction of 20%). A within-subject ANOVA showed this increase in performance to be statistically significant ( $F(1,9) = 49, p < 0.001$ ). At beam width 11, average sentence accuracy is 62.7% in the **without-gesture** condition and 71.4% in the **with-gesture** condition; an absolute error reduction of 8.7% (relative error reduction of 23%). A within-subject ANOVA showed this increase in performance to be statistically significant ( $F(1,9) = 63, P < 0.001$ ).

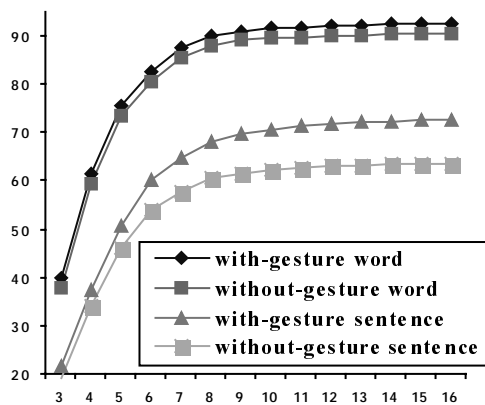


Figure 6 Accuracy across various beamwidths

To examine the relationship between background noise level and multimodal compensation, we compared word and sentence accuracy for the **with-gesture** and **without-gesture** conditions with different levels of noise at beam width 11 (Figure 7).

	word accuracy		sent. accuracy	
	w/o G	w/ G	w/o G	w/ G
clean	89.8	91.8	62.7	71.4
babble 30 dB	88.3	90.8	59.3	68.6
babble 20 dB	74.7	78.1	33.7	41.6
babble 10 dB	26.8	30	2.1	3.5
car 30 dB	84.8	87.4	51.5	60.9
car 20 dB	52.1	55	10.7	14.2
car 10 dB	5.7	7.4	0.1	0.1

Figure 7 Average word and sentence accuracy at beam 11

Adding car noise had a more pronounced effect on performance than babble noise. Tight coupling of multimodal language processing and speech recognition reduced both word and sentence error at all noise levels except for sentence accuracy at car 10 dB which was close to zero both with and without gesture. A series of within-subject ANOVA tests confirmed the differences in word and sentence accuracy between the **without-gesture** and **with-gesture** conditions to be statistically significant under all of the different noise conditions (all  $p < 0.05$ ), except for sentence accuracy at car 10 dB. Our tightly coupled approach continues to be effective as the amount of noise added to the signal increases – the relative accuracy improvement goes up for both word and sentence accuracy.

As a rough estimate of the effect of our approach on the efficiency and speed of speech recognition, we calculated the average time taken to recognize an utterance over all the utterances in the corpus at the 16 different beam widths for both the **with-gesture** and **without-gesture** condition. The average time climbs much more sharply in the **without-gesture** condition. At beam width 11, the average was 18 secs **without-gesture** and 10 secs **with-gesture** – a 44% speed-up. The size of the language model for recognition is reduced through composition with gestural information leading to a significant reduction in processing time for the **with-gesture** condition.

#### 5. RELATED WORK

Most previous work on compensation among input modes has focused on enhancing speech recognition using visual

information from the lips (e.g. Petajan et al 1988). Oviatt 1999 reports on a study of compensation among speech and pen input for an approach in which  $n$ -best lists of speech and gesture interpretations are re-ranked through unification-based multimodal integration (Johnston et al 1997). The ASR performance results are of comparable magnitude to those we found. Averaging over the reported results for native and non-native speakers, the average sentence accuracy in Oviatt's study was 67.9% without and 74.5% with multimodal re-ranking (6.6% absolute, 21% relative). An overall error reduction of 13.3% (41.3% relative) is also reported, which includes resolution of semantic ambiguities in addition to resolution of ASR errors. However, the performance figures are hard to compare as the task is very different from that described here. It supports only 200 unique multimodal utterances, each of 1-7 syllables, and only combinations of speech with a single gesture. Wu et al 1999 present a statistical overlay to the unification-based approach which significantly improves the multimodal recognition rate when trained on the data from the Oviatt 1999 task. However, it is not clear how the approach will scale to more complex verbal language and combination of speech with multiple gestures.

## 6. CONCLUSION

In the finite-state approach to multimodal language processing (Johnston and Bangalore 2000), multiple modes are parsed, integrated, and assigned a combined semantic representation by finite-state transducers. This enables multimodal language processing to be tightly coupled with speech recognition, allowing gestural information to directly influence speech recognition search. We performed experiments to evaluate the effectiveness of our approach in improving speech recognition performance. For clean speech, our approach yielded an average 23% relative error reduction – about a quarter of commands that would have failed from recognition errors succeed using our approach. Compensation was found across a broad range of background noise conditions. The relative accuracy improvement was found to increase as the level of background noise increased. Although background noise significantly reduced speech recognition accuracy, there continued to be a high level of error reduction. Our approach also yielded a significant reduction in the processing time needed for speech recognition, since pre-composition with gesture reduces the size of the recognition language model. In addition to improving speech recognition performance, the finite-state approach is computationally more efficient than unification-based multimodal parsing (Johnston 1998) and provides a well-understood probabilistic framework for multimodal ambiguity resolution.

## 7. REFERENCES

Abney, S. 1991. Parsing by chunks. In R. Berwick, S. Abney, and C. Tenny (eds.), *Principle-based parsing*. Kluwer Academic Publishers, Dordrecht.

Bangalore, S. and G. Riccardi. 2000. Stochastic finite-state models for spoken language machine translation. *NAACL Workshop on Embedded Machine Translation Systems*.

Bolt, R. A. 1980. "Put-That-There": Voice and gesture at the graphics interface. *Computer Graphics*, 14.3, p. 262-270.

Buntschuh, B., Kamm, C., DiFabrizio, G., Abella, A., Mohri, M., Narayanan, S., Zeljkovic, I., Sharp, R.D., Wright, J., Marcus, S., Shaffer, J., Duncan, R. and J.G. Wilpon, 1998. VPQ: A spoken language interface to large scale directory information. *ICSLP '98*, Sydney, Australia.

Cohen, P. R., Johnston, M., McGee, D., Oviatt, S. L., Pittman, J., Smith, I., Chen, L., & Clow, J. 1998. Multimodal interaction for distributed interactive simulation. In M. Maybury and W. Wahlster (eds.), *Readings in Intelligent Interfaces*, Morgan Kaufmann Publishers. p. 562-569.

Johnston, M. and S. Bangalore. 2000. Finite-state multimodal parsing and understanding. *Proceedings of COLING-2000*.

Johnston, M. 1998. Unification-based multimodal parsing. In *Proceedings of COLING-ACL 98*, p 624-630.

Johnston, M., Cohen, P.R., McGee, D., Oviatt, S.L., Pittman, J.A., Smith, I. 1997. Unification-based multimodal integration. In *Proceedings of the 35th ACL*. Madrid, Spain. p. 281-288.

Kaplan, R.M. and M. Kay, M. 1994. Regular models of phonological rule systems. *Computational Linguistics*, 20:3.

Koskenniemi, K. K. 1984. *Two-level morphology: a general computation model for word-form recognition and production*. PhD thesis, University of Helsinki.

Mohri, M., Pereira, F.C.N. and M.D. Riley. 2000. The design principles of a weighted finite-state transducer library. *Theoretical Computer Science*, 231:17-32.

Neal, J. G., and S. C. Shapiro. 1991. Intelligent multi-media interface technology. In J. W. Sullivan and S. W. Tyler (eds.), *Intelligent User Interfaces*, ACM Press, New York. p. 45-68.

Nederhof, M-J. 1997. Regular approximations of CFLs: A Grammatical View. *Proceedings of the International Workshop on Parsing Technology*, Boston, 1997.

Oviatt, S.L. 1999. Mutual disambiguation of recognition errors in a multimodal architecture. *CHI '99*, ACM Press, New York.

Pereira, F.C.N. and M.D. Riley. 1994. Speech recognition by composition of weighted finite automata. In *Proceedings of the ARPA Workshop on Human Language Technology*.

Pereira, F.C.N. and R. Wright. 1997. Finite-state approximation of phrase structure grammars. In E. Roche and Y. Schabes (eds.), *Finite-state Language Processing*, MIT Press, MA.

Petajan, E.D., Bischoff, B., and D. Bodoff. 1988. An improved automatic lipreading system to enhance speech recognition. *ACM SIGCHI-88*, p. 19-25.

Roche, E. 1999. Finite state transducers: parsing free and frozen sentences. In A. Kornai (ed.), *Extended Finite State Models of Language*, Cambridge University Press, Cambridge, England.

Rosenberg, A.L. 1964. On  $n$ -tape finite state acceptors. *FOCS 5*.

Srinivas, B. 1997. *Complexity of lexical descriptions and its relevance to partial parsing*. PhD Thesis, UPenn.

Wahlster, W. 1991. User and discourse models for multimodal communication. In J. Sullivan and S. Tyler (eds.), *Intelligent User Interfaces*, ACM Press. p. 45-67.

Wu, L., Oviatt, S.L. and P.R. Cohen. 1999. Multimodal integration - a statistical view. *IEEE Transactions on Multimedia*, 1(4):334-341.

**Acknowledgements:** Thanks to Rick Rose, Marilyn Walker, Candace Kamm, Elliot Pinson, Paolo Ruscitti, Giuseppe DiFabrizio, Mazin Rahim, and Giuseppe Riccardi.