

Mutual Disambiguation of Recognition Errors in a Multimodal Architecture*

Sharon Oviatt**

Center for Human-Computer Communication
Oregon Graduate Institute of Science and Technology
oviatt@cse.ogi.edu; <http://www.cse.ogi.edu/~oviatt/>

ABSTRACT

As a new generation of multimodal/media systems begins to define itself, researchers are attempting to learn how to combine different modes into strategically integrated whole systems. In theory, well designed multimodal systems should be able to integrate complementary modalities in a manner that supports mutual disambiguation (MD) of errors and leads to more robust performance. In this study, over 2,000 multimodal utterances by both native and accented speakers of English were processed by a multimodal system, and then logged and analyzed. The results confirmed that multimodal systems can indeed support significant levels of MD, and also higher levels of MD for the more challenging accented users. As a result, although speech recognition as a stand-alone performed far more poorly for accented speakers, their multimodal recognition rates did not differ from those of native speakers. Implications are discussed for the development of future multimodal architectures that can perform in a more robust and stable manner than individual recognition technologies. Also discussed is the design of interfaces that support diversity in tangible ways, and that function well under challenging real-world usage conditions.

Keywords

multimodal architecture, speech and pen input, recognition errors, mutual disambiguation, robust performance, diverse users

INTRODUCTION

Multimodal systems process combined natural input modes—such as speech, pen, touch, manual gestures, gaze, and head and body movements—in a coordinated manner with multimedia system output. These systems represent a new direction for computing that draws from novel input and output technologies currently becoming available. They also represent a research-level paradigm shift away from conventional WIMP interfaces toward providing users with greater expressive power, naturalness, flexibility and portability.

Since the appearance of Bolt's [1] "Put That There" demonstration system, which processed speech in parallel with manual pointing, a variety of multimodal systems has emerged. Some rudimentary ones process speech combined with mouse pointing, such as the early CUBRICON system

[8]. Others recognize speech while determining the location of pointing from users' manual gestures or gaze [7]. Recent multimodal systems now recognize a broader range of signal integrations, which no longer are limited to the simple point-and-speak combinations handled by earlier systems. For example, the Quickset system integrates speech with pen input that includes drawn graphics, symbols, gestures and pointing [5]. It uses a semantic unification process to combine the meaningful multimodal information carried by two input signals, both of which are rich and multidimensional.

Complementarity of Modalities

One major challenge for the design of multimodal systems involves learning how to combine different modes into a strategically integrated whole system. In theory, well designed multimodal systems should be able to integrate complementary modalities to yield a highly synergistic blend in which the strengths of each mode are capitalized upon and used to overcome weaknesses in the other [4, 11]. This approach promotes the philosophy of using component technologies to their natural advantage, and of combining them in a manner that permits mutual compensation. One implication is that the resulting multimodal interface may be capable of functioning more robustly than individual recognition-based technologies, which are inherently error-prone [9]. However, empirical research is needed to examine the possibility of a performance advantage in error handling, to assess its specific nature, and to explore the usage contexts in which it may occur.

Error Handling in Multimodal Interfaces

There are several reasons why a multimodal interface potentially can support better error handling than a unimodal recognition-based one, such as a spoken language interface. The following factors all are capable of leading to better error avoidance and more rapid recovery:

First, users will select the input mode that they judge to be less error prone for particular lexical content, which leads to error avoidance. That is, when free to interact multimodally, they exercise good intuitions about the accuracy of a modality for conveying particular content. For example, they are more likely to write rather than speak a foreign

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, to republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.

CHI '99 Pittsburgh PA USA

Copyright ACM 1999 0-201-48559-1/99/05...\$5.00

*This research was supported in part by Grant No. IRI-9530666 from the National Science Foundation, Contract No. DABT63-95-C-007 from DARPA, and donations from Intel and Microsoft.

**Author: Center for Human-Computer Communication, Department of Computer Science, Oregon Graduate Institute of Science & Technology, P.O. Box 91000, Portland, OR, 97291.

surname when addressing a computer [16]. In this respect, a well-designed multimodal interface that gives users flexibility can leverage from their natural ability to use modes accurately and efficiently. Furthermore, the degree of error avoidance possible when using a flexible multimodal interface can be substantial.

Secondly, *users' language is simplified when interacting multimodally, which reduces the complexity of natural language processing and avoids errors.* For example, when users are interacting multimodally they speak fewer words, briefer utterances, fewer referring expressions, less anaphora and linguistic indirection, fewer complex spatial descriptions, and fewer disfluencies than when interacting with unimodal spoken language [10, 15]. In short, there is evidence that multimodal language can be simpler and less ambiguous than unimodal speech, and these altered linguistic features generally would be associated with a reduction in system recognition errors.

Thirdly, *users tend to switch modes after system errors, which facilitates error recovery.* That is, people's natural predilection during multimodal interaction is to switch input modes when they encounter a system recognition error. In fact, their likelihood of mode switching following a system error is 3-fold higher than during baseline periods when recognition is error-free [12]. Since the confusion matrices differ for the same lexical content when delivered via different modes, a mode shift could effectively shortcut a string of repeated system failures (i.e., spiral errors), thereby facilitating error recovery.

The fourth reason why multimodal systems support more graceful error handling is that *users report less subjective frustration with errors when interacting multimodally, even when errors are as frequent as in a unimodal interface* [12]. This reduction in users' level of frustration may result from a greater sense of control when they can switch actively between modes. All of the factors outlined above essentially are user-centered reasons why multimodal systems support improved error avoidance, recovery, and user satisfaction with error handling.

Goals and Predictions of the Study

The present study aimed to investigate a fifth possible basis for superior error handling in multimodal systems, but in this case a by-product of the multimodal architecture's design. This study explored whether a multimodal architecture can support *mutual disambiguation* (MD) of input signals. Mutual disambiguation involves recovery from unimodal recognition errors within a multimodal architecture, which leads to more stable and robust performance [9]. For example, a speech recognizer might misrecognize "ditches" and instead rank the singular "ditch" as first choice on its n-best list, although parallel recognition of several ink lines could result in recovery of the correct plural during multimodal interpretation.

A second goal was to explore whether the relative advantage of multimodal over unimodal processing would be more pronounced in some usage contexts than others. The study investigated whether input from users defined as "challenging" (i.e., accented speakers) could be processed more successfully by a multimodal system than a traditional unimodal one, such as a spoken language system. In particular, it assessed whether the MD rates supported by the

architecture would be higher for accented speakers of English, in comparison with native ones. It was expected that speech recognition would be degraded for accented speakers and, as a result, that a higher percentage of their utterances with MD would involve retrieval of incorrect speech interpretations rather than gestural ones. That is, with respect to mutual disambiguation of input modes within a multimodal architecture, it was predicted that gestural input would disambiguate error-prone speech more often for accented speakers, whereas speech input would disambiguate faulty gesture recognition more often for native speakers. To pursue these goals, the Quickset pen/voice multimodal system was adapted for testing. In addition, a novel metric of mutual disambiguation was developed, and an automated tool was created for logging and analyzing MD during multimodal system processing.

Apart from analyzing MD, recognition rates also were assessed for the two component input signals, speech and gesture, and for multimodal processing after these signals were integrated. If MD is supported by multimodal processing, then it was predicted that spoken language recognition rates would be higher when processing occurs within a multimodal architecture than when conducted as a stand-alone recognition process. Finally, although it was anticipated that speech recognition rates would be poorer for accented speakers than native ones, if the system indeed supports higher MD rates for this group, then their multimodal recognition rates should more closely resemble those of native speakers—yielding a closing of the performance gap between the two groups.

METHOD

Subjects, Tasks & Procedure

Sixteen people participated as paid volunteers, eight native speakers of English and eight accented non-native speakers. The non-native speakers represented a range of different languages from the Asian, Indian, European, and African continents—including Mandarin, Cantonese, Tamil, Hindi, Spanish, Turkish, and Yoruba. All of the non-native speakers of English still were active speakers of their native language part of the day. In terms of duration of experience speaking English while resident in the U.S., the non-native speakers varied widely from 1.5 weeks to 23 years, with the strength of their accents varying from mild to strong.

Within the native and non-native speaker groups, half were male and half female. Participants' ages ranged from mid-20s through mid-50s. Their professional backgrounds were broad-spectrum white collar, ranging from scientists and business personnel to facilities support staff.

During the study, volunteers were given an orientation to the Quickset system, including its map-based interface and capabilities. Using the system, they were shown how to set up simulations involving community fire and flood control activities. After this orientation, they also practiced using all of the system's basic capabilities.

While engaged in simulation activities, users could issue commands to the system to: (1) Scroll, zoom, or otherwise control the system's map display (e.g., "Scroll here" [draws arrow upward]; "Zoom in" [circles school]), (2) Automatically locate objects on the map (e.g., "Show me the hospital" [points to map]), (3) Add objects to the map as individual entities or subsets (e.g., "Backburn zone" [draws ir-

regular rectangular area]; “Volunteers... here” [marks point], “here” [marks point], “here” [marks point]), (4) Orient or otherwise define objects on the map (e.g., “Wind-speed 40 miles an hour” [draws arrow toward northeast]; “Airstrips... facing this way” [draws arrow toward northeast], “facing this way” [draws arrow north]), (5) Specify the movement of entities (e.g., “Remove residents” [delete mark over municipal building]; “Helicopter follow this route” [draws line]), (6) Ask questions about map objects (e.g., “Show number of gallons” [places question mark over water tower]), and (7) Regulate general map capabilities (e.g., “Print map” [check mark]). Figure 1 illustrates the Quickset interface during a simulated fire control scenario. In this example, the user said “pan” and drew an arrow down to indicate the area they wanted to see.

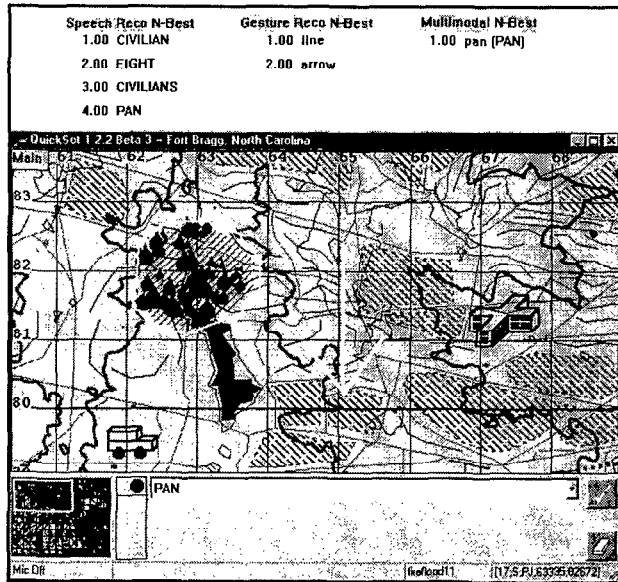


Figure 1: Quickset user interface during a multimodal command to “pan” the map, which illustrates MD occurring such that speech and gesture choices were pulled up on their n-best lists to produce a correct multimodal interpretation by the system

During testing, participants sat in a quiet office environment in front of a Wacom flat-panel LCD display with digitizer that presented the color Quickset map. They communicated multimodally¹ with the system using pen input and by speaking their information. They spoke to the Quickset system with an Andrea noise-canceling microphone headset. A separate audio recording of the users’ speech was captured using a table-top Crown microphone, and fed into the STAMP multimodal data logger (described below). In addition, a record of all pen input and system responding on the map interface was videorecorded for use in conjunction with the STAMP data logger.

¹ Although the Quickset system can process either unimodal or multimodal utterances, users only were asked to deliver multimodal commands. This generated maximum data on the occurrence of MD during multimodal processing.

Each volunteer entered approximately 100 multimodal commands to Quickset, 50 involving a fire control task and another 50 on a flood control task. The order of completing tasks was counterbalanced. All commands were processed by the Quickset system. After each multimodal command was received, Quickset confirmed its semantic interpretation. For example, if an airstrip was recognized, then an airstrip icon would be added in the correct map location and the system’s verbatim recognition confirmed in the text field beneath the map. Participants were told that if the system was correct, they could simply enter their next command. If for any reason the system did not correctly recognize their command after three attempts, they were told to skip over it to the next one. They also were shown how to correct any system errors that occurred by erasing and reentering their input. After each session, volunteers were interviewed briefly about the system, its performance, and its features.

Research Design

The research design was a completely-crossed factorial with two between-subject factors: (1) Native speaker status— unaccented native vs. accented non-native speech, and (2) Gender— male vs. female. The order of presentation of each subtask, fire and flood control, was counterbalanced within each condition. In total, data were available for analysis from 16 users and over 2,000 multimodal commands.²

Quickset Multimodal System

As described earlier, Quickset is a multimodal pen/voice system that supports map-based interactions, especially simulation scenarios. It is a distributed system that runs on a hand-held PC, and uses a multi-agent architecture for parallel processing of spoken and pen-based input. Different versions of the Quickset system and its interface have been developed for use with different application domains. The sections that follow specify system features relevant to the

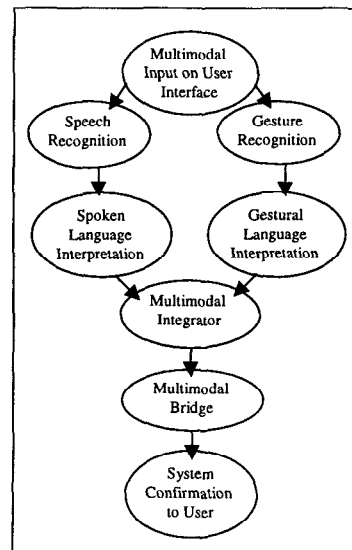


Figure 2: Multimodal architecture for handling signal and language processing of parallel speech and gesture input

² This total includes approximately 1600 original commands and over 400 repair attempts.

Vocabulary and Grammar

A total vocabulary of over 400 spoken words and 9 types of gestures³ were used in conjunction with the fire and flood management tasks in this study. In addition, given the grammatical combinations possible between the speech and gesture vocabulary, a total of over 200 unique multimodal utterances were available within these tasks. The gesture recognition for Quickset was developed at OGI, and the speech recognition was Microsoft's Whisper 3.0. Both recognizers provided n-best lists with probability estimates. The vocabulary and grammar used in this study for speech, gestures, and multimodal constructions were selected to sample broadly from those available within the Quickset system. According to task analysis, Quickset multimodal constructions often were spatial location commands [see 14 for details].

Signal Processing & N-best Recognition

In Quickset, pen-based and spoken input each are time-stamped to mark their beginning and end. For pen-based input, time-stamping occurs for the beginning and end of each stroke, which is an internal data structure that represents all tracking of the pen's x,y coordinates, and this data structure then is sent to the gesture recognizer for signal-level processing. Gestures can be quite ambiguous, and the same stroke can have different legitimate interpretations in different contexts. During processing, the gesture recognizer produces an n-best list of possible meaningful interpretations, each of which is associated with a probability estimate. These signal-level stroke interpretations then are passed on for processing by the natural language agent to create a gestural parse n-best list before being integrated with the parallel speech interpretation.

For spoken input, time-stamping begins and the speech recognition engine is engaged when an acoustic signal exceeding a minimum energy threshold is picked up. Time-stamping ends when the signal's energy falls below this threshold for a given duration, after which speech processing is completed. Since the interface is a tap-to-speak one,⁴ a pen-down event indicating that the user's input was intentional also is a prerequisite for time-stamping and processing. Like gesture processing, the speech recognizer generates an n-best list of lexical interpretations, each associated with a probability estimate that represents the likelihood that the incoming speech signal matches a particular string of phonemes in the speech recognizer's model. These signal-level interpretations then are filtered by the natural language agent's parser, which forms a spoken language n-best list.

To interpret a whole multimodal command, the time-stamps for speech and gestural input are compared by the integrator agent. Based on results of empirical analysis of the synchronization patterns typical of speech and pen input in a similar domain [14], an integration rule is applied to these time-stamped signals. The integrator will combine speech and pen signals and attempt to process their multimodal

meaning: (1) in all cases for which there is temporal overlap between signals, and (2) in cases involving sequential signals if the speech signal begins within four seconds of the end of gesture. When the architecture's synchronization rules permit joint processing, and one or more successful unifications (see details below) yield candidates for inclusion on the final multimodal n-best list, then these lexical items also are ranked on that list according to their probability estimates. The top-ranked multimodal integration then is sent to the architecture's application bridge agent, at which point this system interpretation is confirmed as the user's intended command.

Semantic Unification

In addition to temporal rules, the multimodal architecture imposes constraints based on authentication and semantic unification before joint processing of signals is permitted for a multimodal command. With respect to semantic unification, typed feature structures are employed to provide a common meaning representation for speech and gesture [2]. Unification is an operation that compares two partial specifications of information and combines them into a single complete semantic interpretation, if they are compatible. Multimodal integration is mediated by a unification operation over feature structures that represent the semantic interpretations of the spoken and gestural components of a multimodal utterance. Each candidate string in the n-best lists for both speech and gesture recognition is parsed by a unification-based parser, and then is assigned a feature structure representation of its semantic interpretation. Each of these representations is underspecified or partial until the modes are integrated during the unification process by the multimodal integration agent, at which point full interpretations are generated. The multimodal integration agent examines the cross-product of the spoken and gestural interpretations, filtering out combined interpretations that do not unify [6]. The remaining "legal" unifications then comprise the final multimodal n-best list, which is rank-ordered by probability estimates that are derived by combining probability estimates from the spoken and gestural components. For further details on Quickset's unification and multimodal integration capabilities, see [6].

STAMP Multimodal Analysis Tool

To support this research, a new multimodal data analysis tool was designed to analyze overall multimodal system performance, including the unimodal pieces of the architecture and their capacity for mutual disambiguation. The videotaped record of each user's multimodal commands during human-computer interaction with the map interface, as well as the system's processing results for each command, was routed to the STAMP multimodal data logger. STAMP was designed to permit researchers to analyze multimodal system performance. It: (1) records data on users' multimodal input and system processing as these events are captured during user testing, (2) organizes this information into a database that supports coordinated replay of the users' multimodal commands and the system's processing results (i.e., in the form of n-best recognition lists for individual modalities and their combined interpretation), and (3) supports the flexible and automated analysis of different indices of multimodal system performance. The STAMP suite of multimodal analysis tools consists of four separate pieces: a data logger, a loader, a marking/analysis tool, and a video controller.

³ Gesture types also could have subtypes (e.g., different map orientations for arrows, such as N, NE, E).

⁴ Quickset is typically used with a tap-to-speak interface, because speech during tap-to-speak interaction is known to be substantially more intelligible than that during open-microphone interaction [13].

For each user utterance directed to the system, STAMP uses side-by-side display screens to permit flexible replay of the user's multimodal command on the map along with the system's synchronized recognition results for each of its components. System processing is summarized as a collection of four to five n-best lists, as illustrated in Figure 3, for: (1) speech signal recognition, (2) gesture signal recognition, (3) interpretation of parsed spoken language, (4) interpretation of parsed gestural language, and (5) final semantic interpretation of the multimodal language. Based on a comparison of user input and system processing, STAMP then generates automatic summaries of the multimodal system's recognition and mutual disambiguation rates averaged over subjects, conditions, or a whole corpus. Details of the multimodal data logger tool, its output, and the dependent measures and analyses that it supports have been described elsewhere [3].

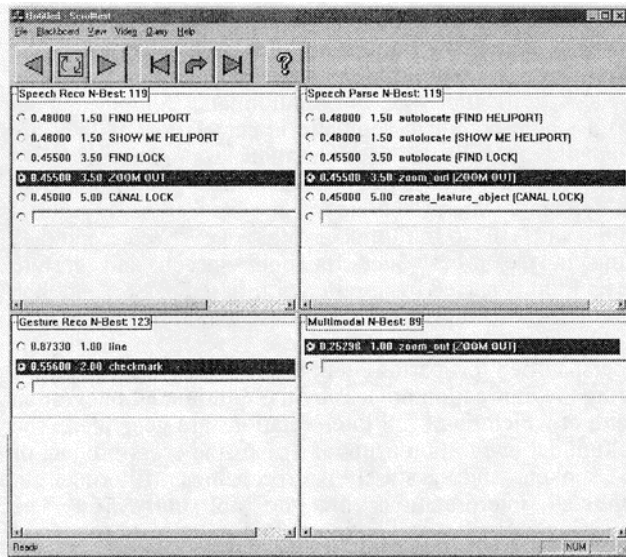


Figure 3: STAMP's display of system processing for a multimodal command, summarized as a set of n-best lists

Dependent Measures

Users' multimodal commands were scored for the measures outlined below except when: (1) a human performance error occurred (e.g., user gestured off screen), (2) a technical problem occurred (e.g., ink skipped), or (3) the command was extraneous or repeated too many times.

Mutual disambiguation

The rate of mutual disambiguation per subject (MD_j) was calculated as the percentage of all their scorable integrated commands (N_j) in which the rank of the correct lexical choice on the multimodal n-best list (R_i^{MM}) was lower than the average rank of the correct lexical choice on the speech and gesture n-best lists (R_i^s and R_i^g), minus the number of commands in which the rank of the correct choice was higher on the multimodal n-best list than its average rank on the speech and gesture n-best lists, or:

$$MD_j = \frac{1}{N_j} \sum_{i=1}^{N_j} \text{Sign} \left(\frac{R_i^s + R_i^g}{2} - R_i^{MM} \right)$$

MD was calculated both at the signal processing level (i.e., based on rankings in the speech and gesture signal n-best

lists), and at the parse level after natural language processing (i.e., based on the spoken and gestural parse n-best lists). Scorable commands included all those that the system integrated successfully, and that contained the correct lexical information somewhere in the speech, gesture and multimodal n-best lists.

Multimodal Pull-ups of Speech & Gesture

During MD, either the correct lexical choice for speech, for gesture, or for both were retrieved from a worse-ranked position than first choice on their respective n-best lists. When MD was present, the ratio of all such *architectural pull-ups* that involved speech versus gesture being retrieved and moved up in rank by the system also was calculated.

Speech Recognition

The total percentage of multimodal commands for which the speech input was correct was computed for each subject, and then averaged for each condition. Speech recognition was correct whenever the correct lexical choice was ranked first on the speech signal n-best list. Errors were scored at the utterance level, and any departure in verbatim lexical content was considered an error and therefore an incorrect utterance. This strict percent correct speech recognition rate was computed for all first attempts at a command, and also for all commands up to a maximum three tries apiece.

Gesture Recognition

The total percentage of multimodal commands for which the gesture input was correct was computed for each subject, and then averaged for each condition. Gesture recognition was correct whenever the correct gesture choice was ranked first on the gesture signal n-best list. Since the gesture set only contained individual gestures (i.e., not compound ones), this percent correct gesture recognition rate effectively was the inverse of a gesture word error rate. This rate was computed for both first attempts at each command, and up to three tries apiece.

Multimodal Recognition

The total percentage of multimodal commands that were correct was computed for each subject, and then averaged for each condition. A multimodal command was correct whenever the correct lexical choice was ranked first on the final multimodal n-best list. Errors were scored at the verbatim utterance rather than word level, so any error within a multimodal utterance was considered an incorrect command. For example, if "here" was recognized as "and here," then the command was not scored as correct even though an appropriate system response might have occurred. This percent correct multimodal recognition rate was computed for first attempts at a given command, as well as up to three attempts apiece.

Comparative Spoken Language Processing

To compare the performance of traditional spoken language processing with that occurring within a multimodal architectural framework, an estimate was made of the percent correct recognition rate for the spoken language processing component as a stand-alone (i.e., speech signal & natural language processing modules), as opposed to the percent correct recognition rate for comparable spoken language processing within the multimodal architectural framework (i.e., speech signal, natural language, and unification processing modules with architectural constraints). The latter

estimate was based on the same calculation as the multimodal recognition rate, after removing all commands known to have failed exclusively due to gesture recognition.

RESULTS

Mutual Disambiguation

One out of eight commands processed by the multimodal system produced the correct response because of mutual disambiguation that occurred between the input signals. More specifically, an average of 7.4% of multimodal utterances contained signal-level MD for native male speakers, and 9.6% for native females. These percentages increased to 14.8% for non-native male speakers, and 15.1% for non-native females. Analysis of variance confirmed that these MD levels were significantly different as a function of native speech status, $F = 8.04$ ($df = 1, 12$), $p < .015$. However, no significant difference was present as a function of gender, $F < 1$, or the interaction between native speech and gender, $F < 1$. A planned independent t-test confirmed that signal-level MD was significantly elevated for non-native speakers compared with native ones (i.e., 15.0% of utterances versus 8.5%, respectively), $t = 3.01$ ($df = 14$), $p < .005$, one-tailed. In short, signal MD values were 76% higher for non-native than native speakers.

This pattern of results was replicated with analyses based on parse-level MD values.⁵ On average, 25.2% of multimodal utterances contained parse-level MD for native male speakers and 25.8% for native females, increasing to 30.4% for non-native male speakers and 33.0% for non-native females. Analysis of variance also confirmed that these MD levels were significantly different for non-native than native speakers, $F = 5.24$ ($df = 1, 12$), $p < .045$. However, no significant difference was evident as a function of gender, $F < 1$, nor the interaction between native speech and gender, $F < 1$. A planned independent t-test again confirmed that parse-level MD was significantly elevated for non-native speakers compared with native ones (31.7% of utterances versus 25.5%, respectively), $t = 2.42$ ($df = 14$), $p < .015$, one-tailed.

Table 1. Relation between spoken command length, speech recognition errors, and cases of mutual disambiguation (MD) in which speech was pulled-up.

	% TOTAL COMMANDS IN CORPUS	% SPEECH RECOGNITION ERRORS	% MD WITH SPEECH PULL-UPS
1-SYLLABLE	40%	58.2%	84.6%
2-7 SYLLABLES	60%	41.8%	15.4%

Table 1 shows the relation between the length in syllables of spoken commands in the multimodal corpus, the percent of speech recognition errors accounted for, and the percent of multimodal commands in which the system pulled up the speech signal during MD. Table 1 basically reveals that although single-syllable words represented just 40% of all commands, they nonetheless accounted for 58.2% of speech

⁵ Since the same gestured or spoken lexical item could have different meanings in different multimodal command contexts (e.g., circle to create an area, or to select), this naturally generated ambiguity increased the baseline values for parse-level MD above those of signal-level MD.

recognition errors, which was significantly greater than chance according to Wilcoxon signed-ranks test, $z = 2.79$ ($df = 16$), $p < .003$, one-tailed. In addition, single-syllable words accounted for 84.6% of cases in which the speech signal was pulled up during MD, which again was significantly greater than chance according to Wilcoxon signed-ranks test, $z = 3.54$ ($df = 16$), $p < .001$, one-tailed.

Users' MD rates did not change significantly as a function of presentation order between the first and second tasks, $t < 1$. That is, the MD rates appeared stable over the 1-hour test session, with no enhancement due to practice.

Multimodal Pull-ups of Speech & Gesture

The percentage of speech signal pull-ups during MD averaged just 3.7% of multimodal commands for native speakers, but increased to 11.2% for non-native speakers. An independent t-test confirmed that the percent of cases in which speech was pulled up was higher for non-native speakers than native ones, $t = 4.99$ ($df = 14$), $p < .001$, one-tailed. In contrast, the percentage of gesture signal pull-ups during MD averaged 7.1% of multimodal commands for native speakers and 5.2% for non-native ones. An independent t-test confirmed that the percent of cases in which gesture was pulled up was higher for native speakers than non-native ones, $t = 1.77$ ($df = 14$), $p < .05$, one-tailed. Overall, the average ratio of speech to total signal pull-ups was .35 for native speakers, but increased to .65 for non-native speakers. An independent t-test confirmed that the ratio of speech to total signal pull-ups also was significantly higher for non-native speakers, $t = 4.59$ ($df = 14$), $p < .001$, one-tailed.

Speech Recognition

Analysis of variance confirmed that the speech recognition rate was significantly different as a function of native speech status, $F = 8.65$ ($df = 1, 12$), $p < .015$. However, no significant difference was present due to gender, $F < 1$, or the interaction between native speech and gender, $F < 1$. As expected, the verbatim utterance-level speech recognition rate was 72.6%⁶ for native speakers, dropping to 63.1% for accented non-native ones—or a 9.5% degradation overall for non-native speakers. A planned independent t-test confirmed that this decrease in performance was a significant one, $t = 3.12$ ($df = 14$), $p < .004$, one-tailed.⁷

Gesture Recognition

Contrary to expectations, an analysis of variance also revealed that the gesture recognition rate was significantly different as a function of native speech status, $F = 4.90$ (df

⁶ The verbatim recognition rates reported in this study for speech, gesture, and multimodal recognition were adopted for making precise comparisons, but they are underestimates of the system's ability to respond correctly since close paraphrases were counted as errors (e.g., "zoom" and "zoom in"). The utterance-level rates reported here also result in lower estimates than a word-level rate, since multimodal commands averaged three words. As a result, the absolute recognition rates per se should not be interpreted literally as performance estimates.

⁷ Results reported for this and other system recognition rates were for first attempts at a given command, although all significant findings reported in this paper also were replicated with analyses based on users' first three command attempts.

= 1, 12), $p < .05$. However, no significant difference was present as a function of gender, $F < 1$, or the interaction between native speech status and gender, $F < 1$. The gesture recognition rate was 83.2% for native speakers, but increased to 86.5% for non-native ones— or 3.4% higher overall. This change represented a small but significant increase in performance for non-native speakers, $t = 2.32$ ($df = 14$), $p < .036$, two-tailed.

Multimodal Recognition

The multimodal recognition rate was predicted to remain lower for accented speakers than native ones, although the difference between groups was expected to be less divergent than their speech recognition rates. Instead, the verbatim utterance-level multimodal recognition rate was 77.2% for native speakers and 71.7% for non-native ones, a 5.5% departure that no longer represented a statistically reliable difference between groups, based on a planned independent t-test, $t = 1.31$ ($df = 14$), N.S., one-tailed. There also was no difference in multimodal recognition due to gender, $t < 1$.

Comparative Spoken Language Processing

As predicted, spoken language processing conducted within a multimodal architecture yielded significantly higher recognition rates for both user groups than spoken language processing as a stand-alone, paired $t = 14.48$ ($df = 15$), $p < .001$, one-tailed. The absolute change in the utterance-level recognition rate for speech processed within a multimodal architecture was +13.3%— which represented a 41.3% reduction in the total error rate for spoken language processing as a stand-alone. This advantage for speech processed within a multimodal architecture was equally evident in the native and non-native speaker groups.

Table 2. Overview of mutual disambiguation levels and recognition rate differentials for native and accented speakers.

	NATIVE SPEAKERS	ACCENTED SPEAKERS
MD LEVELS:		
Signal MD level	8.5%	15.0%*
Parse MD level	25.5%	31.7%*
Ratio of speech signal pull-ups	.35	.65*
RECOGNITION RATE DIFFERENTIAL: †		
Speech	+9.5%*	—
Gesture	—	+3.4%*
Multimodal	—	—

*Rates representing a significant elevation between groups.

†Recognition rate differentials show the percentage of advantage for a user group when a significant difference was present.

DISCUSSION

This research has demonstrated that multimodal systems can be designed that are capable of functioning in a more robust and stable manner than individual recognition technologies, which are inherently error-prone. In fact, a 41% reduction in the total error rate was revealed for spoken language processing within a multimodal architecture,

compared with spoken language processing as a stand-alone.

One by-product of designing a multimodal architecture clearly is the superior error handling that is possible due to mutual disambiguation of the system's input modes. In the multimodal system analyzed in this study, one in eight commands that were recognized correctly by the multimodal system succeeded because of mutual disambiguation, even though one or both component recognizers had failed to identify the user's intended meaning. This disambiguation occurred because architectural constraints imposed by semantic unification ruled out incompatible speech and gesture integrations, which effectively pruned recognition errors from the n -best lists of the component input modes. As a result, it frequently was possible to retrieve a correct lexical item ranked lower on an n -best list in a manner that basically yielded an architectural pull-up.

The example shown in Figure 1 illustrates double MD during an error-prone monosyllabic command by an accented speaker. In this case, the fourth-ranked speech choice "pan" was the only alternative that could integrate with the second-ranked arrow gesture, and none of the other speech alternatives integrated with the line on the gesture n -best list. During this integration process, each input mode provides a context for interpreting the other, thereby helping to disambiguate its meaning. In the design of future multimodal architectures, research should explore natural language and dialogue processing techniques other than unification which also may be effective in supporting or optimizing mutual disambiguation of errors.

The rate of mutual disambiguation also was higher for accented non-native speakers than native speakers of English— by a substantial 76%. Although speech recognition rates were much poorer for accented speakers, as would be predicted— their multimodal recognition rates actually did not differ significantly from those of native speakers. The factor mainly responsible for closing this gap in multimodal performance was their higher MD levels. A second factor appears to have been their slightly but significantly elevated gesture recognition rates. It is possible that accented speakers' self-awareness about the vulnerability of their speech recognition may have resulted in efforts to compensate via their gestural input.

There often may be asymmetries in a multimodal interface as to which mode is the more fragile in terms of reliability of recognition. When one mode is expected to be less reliable, the most strategic approach will be to select an alternate mode that can act as a complement and stabilizer in promoting overall mutual disambiguation. In this study, speech recognition was the more fragile mode for accented speakers, with two-thirds of all architectural pull-ups retrieving poorly ranked speech input. However, the reverse was true for native speakers, with two-thirds of pull-ups retrieving lower ranked gestures. Future research could be helpful in defining how different user groups and usage contexts may influence overall MD rates, or asymmetries in the reliability of modes that require compensation in a multimodal interface.

When a spoken language system must process a diverse array of accented speech patterns, as from speakers in the present study, one problem is that the recognizer's

substitution errors will be extremely heterogeneous. In contrast, for native English speakers the pattern of lexical confusions typically is relatively predictable, such that a speech vocabulary can be crafted for an application that minimizes highly-confusable errors. Unfortunately, when a realistic array of accented speech must be processed, the strategy of minimizing errors by tailoring vocabulary selection becomes infeasible. Given this more challenging real-world usage context, a multimodal architecture that supports mutual disambiguation may provide a more viable and flexible long-term alternative for reducing system errors. In general, the present results suggest that multimodal interfaces can be developed that support diverse user groups in tangible ways, and that function more reliably than unimodal recognition technologies during challenging real-world usage conditions.

ACKNOWLEDGMENTS

Thanks to J. Clow and C. Slattery for assistance with data collection and analysis, and to J. Clow for implementing the STAMP multimodal analysis tool. Thanks also to D. McGee for adapting the Quickset interface for the fire and flood management scenarios, and to M. Johnston and J. Pittman for extending the speech and gesture vocabulary and grammar. Finally, thanks to P. Cohen for numerous discussions about multimodal architectures, and to our research volunteers for their enthusiasm and generous commitment of time.

REFERENCES

1. Bolt, R.A. Put that there: Voice and gesture at the graphics interface. *Computer Graphics*, 1980, 14 (3): 262-270.
2. Carpenter, R. The logic of typed feature structures. Cambridge, MA.: Cambridge University Press, 1992.
3. Clow, J. & Oviatt, S. L. STAMP: A suite of tools for analyzing multimodal system processing, *Proceedings of the International Conference on Spoken Language Processing*, in press.
4. Cohen, P., Dalrymple, M., Moran, D., Pereira, F. Synergistic use of direct manipulation and natural language, *CHI '89 Conference Proceedings*, ACM/ Addison Wesley: New York, NY, 1989, 227-234.
5. Cohen, P., Johnston, M., McGee, D., Oviatt, S., Pittman, J., Smith, I., Chen, L. and Clow, J. Quickset: Multimodal interaction for distributed applications. *Proceedings of the Fifth ACM International Multimedia Conference*, New York, NY: ACM Press, 1997, 31-40.
6. Johnston, M., Cohen, P.R., McGee, D., Oviatt, S.L., Pittman, J.A. & Smith, I. Unification-based multimodal integration. *Proceedings of the 35th Annual Meeting of the Association for Computational Linguistics*, San Francisco, CA.: Morgan Kaufmann, 1997, 281-288.
7. Koons, D.B., Sparrell, C.J. & Thorisson, K.R. Integrating simultaneous input from speech, gaze, and hand gestures. In *Intelligent Multimedia Interfaces*, M. Maybury, Ed. MIT Press: Menlo Park, CA, 1993, 257-276.
8. Neal, J.G. & Shapiro, S.C. Intelligent multi-media interface technology. In *Intelligent User Interfaces*, J. Sullivan & S. Tyler, Eds. ACM: New York, 1991, 11-43.
9. Oviatt, S.L. Ten myths of multimodal interaction, *Communications of the ACM*, in press.
10. Oviatt, S.L. Multimodal interactive maps: Designing for human performance, *Human-Computer Interaction*, 1997, 12 (1 & 2) 93-129.
11. Oviatt, S.L. Pen/voice: Complementary multimodal communication, *Proceedings of Speech Tech '92*, New York, NY.
12. Oviatt, S.L., Bernard, J. & Levow, G. Linguistic adaptations during spoken and multimodal error resolution, *Language and Speech*, in press.
13. Oviatt, S.L., Cohen, P. & Wang, M. Toward interface design for human language technology: Modality and structure as determinants of linguistic complexity, *Speech Communication*, 1994, 15 (3-4), 283-300.
14. Oviatt, S. L., DeAngeli, A. & Kuhn, K. Integration and synchronization of input modes during multimodal human-computer interaction, *Proceedings of the CHI '97 Conference*, New York, NY: ACM Press, 415-422.
15. Oviatt, S. L. & Kuhn, K. Referential features and linguistic indirection in multimodal language, *Proceedings of the International Conference on Spoken Language Processing*, in press.
16. Oviatt, S. L. & Olsen, E. Integration themes in multimodal human-computer interaction, *Proceedings of the International Conference on Spoken Language Processing*, (ed. by Shirai, Furui & Kakehi), Acoustical Society of Japan, 1994, vol. 2, 551-554.