A Mechanism for Human – Robot Interaction through Informal Voice Commands

Sidney D'Mello
Institute for Intelligent Systems
University of Memphis
Memphis, TN 38152, USA
sdmello@memphis.edu

Lee McCauley
Department of Computer Science
University of Memphis
Memphis, TN 38152, USA
mccauley@memphis.edu

James Markham
Department of Computer Science
University of Memphis
Memphis, TN 38152, USA
jmarkham@memphis.edu

Abstract **- Interaction between humans and robots must, necessarily, be in a manner that is natural to humans, such as informal speech. While verbal command and control systems are fairly common, the human must know the exact phrases to issue in order for their robotic partner to respond appropriately. These systems define a few grammar rules that are used to match against incoming utterances. Here, we present a method of using these same grammar rules to expand the capabilities of command and control engines to include semantically similar utterances. Preliminary results from an experimental simulation will be presented along with a detailed methodology of a recently completed study aimed at collecting human speech for a more rigorous analysis.**

Index Terms – Speech recognition, Natural Language Understanding, Latent Semantic Analysis, Command and Control, Robots

I. Introduction

Truly useful interaction between humans and machines, especially robotic, will be through untrained verbal command and control. Several studies have been conducted and a variety of approaches have been proposed in an attempt to determine a plausible model for practical human-robotic speech interaction. Some of the sought after goals in this area include command and control, contextual understanding, and determination of the intent embedded within the vocal communication occurring between a person and a robot. Kismet, an interactive robotic face developed at MIT, focuses on detecting a user's intent and mood based on the acoustic speech patterns generated during the interaction session [4]. However, it ignores contextual understanding and interpretation. Mel, a robotic agent created at Mitsubishi Electric Research Labs, uses directed conversation to elicit a small set of possible responses, to limit the problems encountered when unpredictable responses or queries are given to it while interacting with people [20]. Robotic agents developed at Saitama University in Japan, use a combination of limited natural language processing, an object definition knowledge base, and sentence parsing and tagging to determine subject and verb requests in speech and visual communication between the agent and the human [15, 21]. Interactive Systems Labs at the University of Karlsruhe, Germany have focused on understanding recognition of command utterances directed towards a robotic agent when there is a human-human-robotic conversation in progress by applying methods of n-gram language models in conjunction with context-free grammars [10]. Carnegie Mellon researchers have used partially observable Markov decision processes to aid in command and control and intent determination with a robotic agent to aid the elderly in everyday actions [16]. Unfortunately, in most of these systems, the understanding is restricted to a handful of commands with a few variations.

Verbal command-and-control is available today, but only through a finite (and generally small) set of command formats. Speech recognition engines typically have two separate modes of operation. The first, called a dictation grammar, relies on complex statistical models of a language, while the second, called a rule grammar, requires simplified regular expressions that can be matched against incoming utterances. Dictation grammars are typically less accurate than rule grammars, but allow for free-form recognition. Rule grammars, on the other hand, are highly accurate given that the user utters a phrase that precisely matches one of the regular expressions in the grammar.

Described below is a method for supplementing rule grammars with Latent Semantic Analysis (LSA) [8, 13] in order to remove the need for rigorous phraseology. Ultimately, this enhanced recognition is only useful if the recognized text can be understood by the robot. "Understanding" here is used to mean that the robot can match an utterance to an existing grammar rule and extract the appropriate information.

We believe that one of the most natural and appropriate uses of this technology will be in the area

of human-robotic interaction. Our planned application involves the use of a humanoid robot, such as the Sony QRIO, that would guide FedEx Institute of Technology (FIT) visitors to the destination of their choice within the building. Previous work on this voice interface was conducted within the domain of natural voice interaction with Microsoft Outlook [14].

II. BACKGROUND

The responsibility of the natural language system being described here is to provide an analysis of the human's utterance in order to determine what action or actions need to be taken by the robot to satisfy the user's request. When a system is employed in a limited domain and has only a limited number of choices of what to say or do next, it need only classify the user's utterance according to what it needs to know in order to make that decision. For instance, if a simple navigation robot has just asked, "What room would you like to visit?" then it might classify the user's response according to labs, offices and other locations within its domain. The main natural language understanding (NLU) technology utilized here is based on classification.

A. Classification Approaches to NLU

Notable among the various classification approaches are those that are based on word co-occurrence patterns such as LSA [8, 13] and HAL [5]. LSA is an attractive approach because it can be trained on untagged texts and is a simple extension to keyword based techniques.

In general, methods for analysing a large corpus of text, such as LSA, are described as generating "language models" rather than being applied to the specific task of speech recognition. It should be made clear that we are referring to corpus analysis based on large digital texts; this is quite different from the analysis of audio corpuses, as is quite common in building dictation grammars for automatic speech recognition systems. What is being proposed here will not alter the audio models used within a speech recognition engine. Instead, a large corpus of text is analysed in order to create a semantic representation. Our approach attempts to use these semantic representations to categorize incoming utterances as though it were one of the existing grammar rules.

There is previous precedent for the application of this type of corpus analysis to speech recognition, although it is generally assumed that the language models produced would replace or modify those in existing speech recognition engines. Two significant contributions would be [9, 19]. While the proposed research overlaps with this type of language modelling, it is more akin to information retrieval methods that do text classification. Aside from those already mentioned, some notable examples would be [1 ,2, 3, 11, 12].

B. Latent Semantic Analysis (LSA)

LSA is a statistical technique that measures the conceptual similarity of two text sources. LSA computes a geometric cosine (ranging from 0 to 1) that represents the conceptual similarity between the two text sources [7, 8]. To use LSA, one must first develop an LSA space, which acts as a lexicon. The simulations described below were conducted using a space generated from Grolier's Encyclopedia.

III. ALGORITHM

Understanding spoken speech really requires two very different capabilities. The first is the translation of the sound patterns into textual representations. This process is commonly referred to as speech recognition. This research does not delve into the area of audio to text conversion. We have chosen to use the commercially available Dragon Naturally Speaking™ software package as our speech recognition engine. Dragon Naturally Speaking™ can translate audio streams to text using two modes, dictation and grammar, discussed previously.

Our system uses both modes simultaneously. The speech recognition engine is set up to use a list of grammar rules as its primary matching scheme; however, if no rule from this list is matched, the engine will still provide text based on the dictation grammar. For this reason, all the grammar rules used as the basis for semantic classification will still function exactly as originally intended. Our system comes into play only when an existing rule is not matched. In these instances, LSA [5, 7, 8, 13] is used to extract and match grammar rules to the text provided by the dictation grammar.

Instantiated grammar rules are represented internally as vectors in LSA space. Here, we need to map an arbitrary utterance to a pre-existing rule that then states what knowledge needs to be updated and what actions need to be taken, if any. This is the same task as was performed by the rule-based approach, but, with the LSA approach, the phrases spoken do not necessarily have to match precisely, as long as their meanings, expressed as vectors in the LSA space, are similar enough.

Unfortunately, there is not a one-to-one correspondence between a grammar rule and a vector in the LSA space. A given utterance may match to a combination of rules in the grammar. For example, the utterance "please take me to the foyer" might match to a pair of rules such as the following:

1.<direction-request>= 'please take me to the <location>'

2.<location> = ‘foyer’ | ‘sun room’ |‘garage’

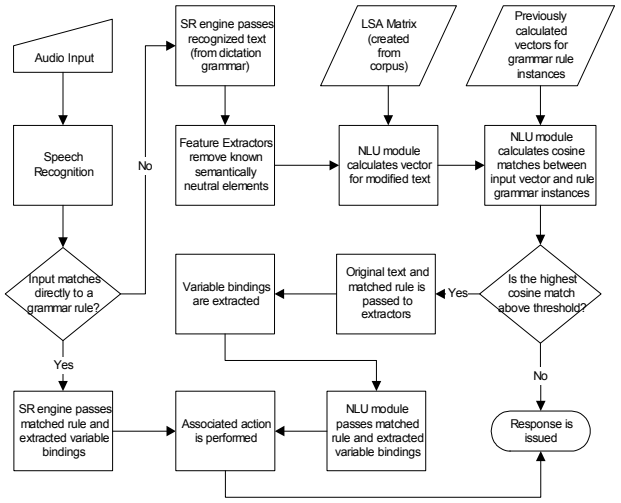

Figure 1. Algorithm flow for understanding spoken commands

Stated loosely, these two rules recognize the words "please take me to the" followed by any of the locations listed in (2). This very simple combination (not actually used in the research) recognizes three different sentences. LSA would be able to match the utterance against any of the instantiations of the rules, but there is no LSA vector that would match to (1) or (2) individually. Instead, a LSA vector is generated for each possible rule instantiation. Therefore, when an arbitrary utterance is received, its LSA vector is matched against the instantiations of all of the grammar rules currently in context. If a match is found then the system instantiates appropriate data extractors based on the types of variables contained within the matched rule. These extractors are then used to bind values to the necessary variables. These values would be things like a proper name, a date, etc., as well as domain specific variables such as locations. For the example given above, the <location> variable would be bound to “foyer.” Since the grammar rule prescribes which variables need to be bound and their type, regular expressions specific to those types can be used to extract the necessary information from the text. A more complete version of the algorithm is provided in Figure 1. The result is that the appropriate grammar rule is “fired” with the correct variables bound just as if the user had uttered the phrase exactly as prescribed in the grammar rule. In other words, in our example, the human could say, “I want to go to the foyer” and the robot would respond with the action sequence associated with, “Please take me to the foyer.”

IV. Experimental Simulation

These preliminary analyses focus on assessing the system’s ability to locate semantically appropriate rules even when a given rule phraseology does not match.

A. Task Domain

The task domain chosen for these tests was Microsoft™ Outlook's Calendar program. This selection was motivated by a number of factors. First, it is narrow enough to have a tractable number of possible action or query types that might be presented. Second, the domain is broad enough to allow for a large number of possible utterances, thereby giving the system a sufficient test of semantic possibilities. Additionally, it is a widely used program that can be easily tested in real-world situations and has an easily accessible API.

B. Experimental Setup

A set of fourteen initial grammar rules in the Outlook domain was expanded into a set of 14,792 instantiations of those rules, even after not elaborating items such as dates, times, etc. A statistically significant sample size to evaluate these instances was determined to be approximately 700 by a method specified by [17]. The confidence interval was set at 95% with a margin of error of 3.705%. These test instances were divided into three categories: *gibberish* instances, *feature* instances, and *semantic* instances.

1) Gibberish Instances: Gibberish is considered to be any out of context text or random noise. The gibberish test instances are divided into three categories based on the amount of gibberish present in the samples. The gibberish text in these simulations were randomly selected snippets of text selected from cnn.com [6]. In the first category, all test samples exclusively consisted of gibberish. The second set of gibberish instances address the situation where the speech recognition engine has accurately recognized a complete spoken utterance, but has introduced some garbage due to noise. Finally, the third category addresses situations where a part of the spoken utterance has been recognized along with some added gibberish.

2) Feature Instances: The feature instance test samples are a type of controlled semantic equivalence test in which the utterance replaces one or more features in the rule instantiation. These samples are also divided into three sub-categories based on the replacement feature selected. In the first category, the replacement is chosen from a small, but frequently used set of synonyms (taken from [18]) of the replaced feature. In the second category, the replacement is chosen from a larger set of less frequently used synonyms. For example, replacing the feature *foyer* with *entrance* and *lobby* would be considered as valid

instances for the first and second test categories respectively. Finally, the third category consists of test instances where a binding is replaced with a highly specific instance of itself. An example would be, "Please take me to the *vestibule*", with *vestibule* being a highly specific instance of a foyer.

3) Semantic Instances: The semantic equivalence test instances address situations where the spoken utterance means the same as a rule instantiation, but is specified with completely different wording. For example consider the instance, "I need directions to the foyer", which is semantically similar to an instance of the grammar rule *direction-request,* but is syntactically quite different. The test instances for this experiment were manually created by human volunteers. Each volunteer was given the text for several rule instantiations and was asked to rewrite each instance in a way that preserved its meaning while being as structurally dissimilar as possible.

C. *Scoring Procedure*

It should be reiterated that the system described here is a supplement to the standard rule grammar systems already in use for command and control. For this reason, all of the phrases that the command and control grammars are designed to recognize will still be recognized as usual. The results of the system will, therefore, be providing a level of increased recognition that would not have been made by the original rule grammar alone. For this reason, it does not make sense to test exact matches. All of the test instances being used for these experiments *would not have been matched by the grammar alone*. In essence, the performance of a speech recognizer endowed with just the corresponding rule grammar would be 0.

For any given test instance, the output of our system could be categorized in three ways: (1) a *true positive* is when the phrase is exactly matched to the intended command, (2) a *false positive* occurs when the phrase is matched to a command that is unrelated to the intended command, (3) an *unclassified* phrase might *not be matched* to any command and would, therefore, not result in an action. In order to reduce the amount of frustration caused by false positives, we introduced a similarity threshold such that any match below the threshold is considered to be *unclassified*. As the threshold increases, the requirements for a match (both correct and incorrect) become more stringent and the likelihood that the phrase will not be matched increases. We introduce a *performance* metric such that a true positive is scored as a 1, a false positive -1, and an unclassified match is scored as 0.

This performance metric was applied to the 700 instances distributed among the gibberish (n = 300), feature (n = 300), and semantic (n = 100) categories.

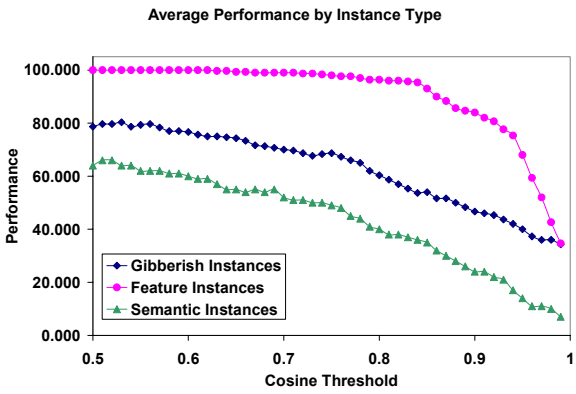

Figure 2. Performance scores over 0.5 – 1.0 cosine range

Since, the gibberish and feature categories have three sub-categories each, the performance scores were averaged over the three sub-categories. Finally, it should be noted that a slightly different performance metric was applied to the sub-category of instances that comprised of complete gibberish. All of these instances should not match to any command. For those instances any match was scored as -1, while an unclassified instance was scored as 1.

D. *Results and Discussion*

A typical cosine threshold when using LSA as an extension for keyword matching is 0.65. Since one of the major motivations of these simulations was to select an appropriate cosine value to accommodate all three categories of instances, we varied the threshold over the 0.5 – 1.0 range with uniform increments of 0.01. Figure 2 depicts performance of our system in handling the three types of test instances over this range.

The gibberish instances handle situations where the utterance was somewhat unrelated to the subjects addressed by the command and control grammar. Figure 2 indicates that the system reaches its maximum performance in handling gibberish instances with a score of 80.33 at cosine threshold 0.53. This is a positive indication of the system's ability to tolerate both partial and complete gibberish.

The feature instances consider replacements of one or more features of the original grammar rule with synonym words not included in the original rule. The ability to successfully handle these instances greatly broadens the scope of the natural language understanding module. Figure 2 shows that even at the most liberal cosine values, the performance is excellent.

Test cases for the semantic instances were made up of phrases that mean the same thing (based on human interpretation) as a provided grammar rule, but do not necessarily share any structural similarity. From figure 2, performance for the semantic instances is lower than gibberish or feature instances, but is quite acceptable.

The results presented here, although not performed on "live" speech, demonstrate a clear benefit over rule-based grammars. It should be noted that simple keyword matching techniques can be successfully applied to the gibberish and feature instances. However, one would not expect them to perform satisfactorily on semantic instances in which the keywords are replaced. Therefore, it is safe to assume that none of the semantic test instances would have been recognized by a keyword based approach. Any command correctly categorized by this new approach can only serve to increase the accuracy of the overall system. The trade-off of possible miscategorizations seems reasonable although testing under real-life conditions needs to be conducted.

V. EMPIRICAL DATA COLLECTION

The experimental simulations presented above serve as proof of concept evidence regarding our approach of natural language understanding. However, since they did not involve real speech, one would expect a performance degradation when the system is embedded into a real world application like a robot. We therefore introduce the procedure of a study designed to test the system under "real-world" conditions in order to gauge its full usefulness. In particular, we want to know what percentage of utterances are likely to match the original grammar rules as opposed to being semantic equivalents. We also would like to find out what percentage is going to be simple synonym replacements (feature instances) and how many will be complex restatements (semantic instances).

A. Study Methodology

1) Participants: The participants used in this study consisted of 28 undergraduates. They were selected from the department of psychology subject pool at the University of Memphis.

2) Electronic Materials: Each participant was required to use Dragon Naturally Speaking's™ user profile training facility in order to construct an individualized speaker profile for more accurate speech recognition.

Participants then interacted with a simple program that presented a text based stimulus and operated in two modes, *formulation* and *recitation*. In the formulation mode, participants were presented with a task description and were instructed to formulate and speak a verbal command. In the recitation mode, participants were presented with verbal commands and were required to speak them out. When they were ready to speak, participants utilized a push to talk facility that resulted in the program recording their speech for offline analyses.

3) Procedure: As participants came into the lab, they were presented with an informed consent. Then, they created a speaker profile tailored to their unique speaking patterns. Next, they interacted with the audio recording program that presented the stimuli.

In the *formulation* mode, participants were presented with 15 high level task descriptions and were required to mentally formulate and speak out voice commands to convey the meaning underlying the tasks. These verbal commands formulated by the participants were recorded. For example, a task description would be "Your advisor needs to meet with you to discuss your thesis sometime next week. Check your availability". An appropriate verbal command to achieve this task might be "Show my calendar for next week".

The *recitation* part of the study consisted of participants speaking out pre-formulated commands. The commands were similar to the test instances utilized for the baseline tests discussed above. The instances were a combination of grammar instances, feature instances, and semantic instances. The grammar instances consist of exact instantiations of grammar rules with absolutely no variations. Therefore, each grammar instance should, theoretically, match one grammar rule. However, due to environmental noise and restricted accuracy of all speech recognition engines, several spoken commands will not match a grammar rule. Once we calculate the rate of gibberish we can assess any performance improvement provided by our system, in addition to rule grammars. The feature and semantic instances were identical to those described above. Each participant recited 100 different instances that were a combination of grammar, feature, and semantic instances. Additionally, each instance was recorded by 4 different participants.

B. Data Analysis

Analysis of the recorded speech first involves transcribing the recorded audio. We are planning to delve into a rigorous analysis of the data that investigates several issues that are directly related to performance.

1) Classification Algorithms: Our current approach to classification is a nearest neighbour search utilizing a geometric cosine as the distance metric. Other approaches to be considered would be strict keyword matching, a variety of distance metrics, boosting, etc.

2) Multiple Corpora: Our system currently utilizes a corpus constructed from Grolier's Encyclopedia for its semantic matches. We plan to experiment with different corpora such as the TASA corpus which significantly differs in size as well as content.

3) Formulation Analysis: This analysis would focus on the speech gathered from the participants in

the formulation mode. Once the speech patterns related to the various tasks have been explored, they will be incorporated into the rule grammar. This will broaden its scope and enable it to account for a majority of the variations in which humans express verbal commands.

4) Recitation Analysis: This study considered three types of instances: grammar, feature, and semantic. We will compare the efficacy of our approach on the basis of its ability to handle these three categories.

5) Gibberish Content: Gibberish is considered to be any out of context words that were incorrectly recognized by the speech engine when it is operating in dictation mode. The proposed analysis here would be to quantify any performance degradation of our approach based on the amount of gibberish present in the instances.

6) Speech Recognizers: All audio transcriptions will be transcribed by a variety of speech recognition systems to ensure that our approach to natural language understanding remains robust independent of the speech recognizer. Two speech recognizers under consideration are Dragon Naturally Speaking™, and CMU Sphinx.

7) Speaker Profiles: An attractive feature of some speech recognition systems is speaker independence. These systems require no training. According to the study methodology, we have trained speakers exclusively on the Dragon Naturally Speaking™ system. This assessment will evaluate the ability of our approach to improve performance of untrained speech recognizers.

The data collection phase is complete. We are currently in the process of transcribing the recorded audio into text to begin the analyses described above.

VI. CONCLUSION

Once the analysis of the human subject data is complete, we intend on using this system in additional real-world scenarios. In the immediate future, this involves an intelligent kiosk system for the lobby of the FedEx Institute of Technology that will converse in natural language. A planned extension of this project will be a mobile robot that will guide visitors to their desired locations throughout the building. The speech domain for such a robot would be smaller than the Microsoft Outlook domain tested previously, but will provide specific real-world challenges. We believe that the technique described here is an important step towards freeing users from the constraints of the desktop and allowing natural interaction with an increasing number of robotic helpers.

A key piece to this endeavour will be the accurate understanding of natural human language. Described above is a method for the enhancement of speech recognition and natural language understanding through the coupling of highly accurate rule-based grammars with a powerful classification technique. The technique presented here, in conjunction with standard rule-based methods, should allow for accurate recognition and understanding of human-to-robot commands or requests without the constraints of strict phraseology.

VII. REFERENCES

[1] Baker, L.D. and McCallum, A.K., Distributional clustering of words for text classification. in *SIGIR'98*, (Melbourne, Australia, 1998), 96-103.

[2] Bellegarda, J.R., Butzberger, J.W., Chow, Y.L., Coccaro, N. and Naik, D., Automatic Discovery of Word Classes Through Latent Semantic Analysis. in *EUSIPCO-96 Signal Processing VIII, Theories and Applications*, (1996), Edizioni Lint Trieste.

[3] Bellegarda, J.R., Butzberger, J.W., Chow, Y.L., Coccaro, N. and Naik, D., A Novel Word Clustering Algorithm Based on Latent Semantic Analysis. In *Proceedings of the 1996 international conference on acoustics, speech, signal processing* (*ICASSP-96*) , 172-175.

[4] Breazeal, C and Aryananda, L 2002. Recognizing affective intent in robot directed speech. *Autonomous Robots*, 12:1, 83–104, 2002.

[5] Burgess, C., Livesay, K. and Lund, K. Explorations in Context Space: Words, Sentences, Discourse. *Discourse Processes*, *25*. 211-257.

[6] CNN online news, http://www.cnn.com, 2004.

[7] Deerwester, S., Dumais, S.T., Fumas, G.W., Landaur, T.K. and Harshman, R. Indexing by latent semantic analysis. *Journal of the American Society for Information Science*, *41*. 391-407, 1990.

[8] Dumais, S.T. Latent semantic indexing (LSI) and TREC-2. in Harman, D. ed. *National Institute of Standards and Technology Text Retrieval Conference*, NIST, 1994.

[9] Gotoh, Y. and Renals, S. Topic-based mixture language modeling. in *Natural Language Engineering 6*, 2000.

[10] Katzenmaier, M., Stiefelhagen, R., and Schultz, T., Identifying the Addressee in Human-Human-Robot Interactions based on Head Pose and Speech, In: International Conference on Multimodal Interfaces ICMI 2004, State College, PA, USA, October 2004

[11] Khudanpur, S. and Wu, J., A maximum entropy language model integrating n-grams and topic dependencies for conversational speech recognition. in *ICASSP-99*, (Phoenix, AZ, 1999), 553-556.

[12] Kuhn, R. and De Mori, R. A cashe-based natural language model for speech recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *12* (6). 570-583.

[13] Landaur, T.K. and Dumais, S.T. A solution to Plato's problem: The Latent Semantic Analysis theory of the acquisition, induction, and representation of knowledge. *Psychological Review*, *104*. 211-240.

[14] McCauley, L., D'Mello, S., & Daily, S. (2005). *Understanding Without Formality: augmenting speech recognition to understand informal verbal commands.* Paper presented at the ACM Southeast Conference, Kennesaw, GA.

[15] Mohd Hanafiah, Z., Yamazaki, C., Nakamura A., and Kuno, Y., Human-robot speech interface understanding inexplicit utterances using vision. CHI Extended Abstracts 2004: 1321-1324

[16] Montemerlo, M., Pineau, J., Roy, N., Thrun, S., and Varma, V., Experiences with a mobile robotic guide for the elderly *Proceedings of the International Conference on Artificial Intelligenc*e (AAAI 2002). Edmonton, July. 2002

[17] Rea, L. *Designing and Conducting Survey Research*. Jossey-Bass, 1997.

[18] Resource.com, http://thesaurus.reference.com, 2004.

[19] Siivola, V., Language modeling based on neural clustering of words. in *IDIAP-Com 02*, (Martigny, Switzerland, 2000).

[20] Sidner, C., Kidd, C., Lee C., and Lesh, N., Where to Look: A Study of Human-Robot Engagement. *In Proceedings of Intelligent User Interfaces 2004*. Madeira, Island of Funchal, Portugal

[21] Yoshizaki, M., Kuno, Y., and Nakamura, A, Human-Robot Interface Based on the Mutual Assistance between Speech and Vision. *Proc. Workshop on Perceptive User Interfaces*, CD-ROM, 2001.