The Intelligent Voice-Interactive Interface

Christopher Schmandt
Eric A. Hulteen

Architecture Machine Group
Massachusetts Institute of Technology

"Put That There" is a voice and gesture interactive system implemented at the Architecture Machine Group at MIT. It allows a user to build and modify a graphical database on a large format video display. The goal of the research is a simple, conversational interface to sophisticated computer interaction. Natural language and gestures are used, while speech output allows the system to query the user on ambiguous input.

This project starts from the assumption that speech recognition hardware will never be 100% accurate, and explores other techniques to increase the usefulness (i.e., the "effective accuracy") of such a system. These include: redundant input channels, syntactic and semantic analysis, and context-sensitive interpretation. In addition, we argue that recognition errors will be more tolerable if they are evident sooner through feedback and easily corrected by voice.

INTRODUCTION

The input to "Put That There" is done by talking and pointing; this results in a system that is both highly interactive and requires little training. Both talking and pointing are ways of communicating which are second nature to people but which computers, up until this time, have had difficulty understanding. The system responds either graphically, by drawing on the screen, or vocally, using either a speech synthesizer or a digital audio playback system. The particular application in which the demonstration is embedded is the manipulation of shipping on a map of the Caribbean. The user can create, modify the shape or color, move, copy, name, annotate, and delete the ships by pointing and talking, as well as question the system about where particular ships and map features are located. The system has some memory so that ships can be restored to previous descriptions or relocated to former positions. It is possible to reconfigure the input/output structure by telling the system to stop listening, to stop talking and instead write its messages on the screen, or to use the digitized voice instead of the synthesized one. The details of the system's capabilities have been described elsewhere (1), in this paper we detail the algorithms which make the system's operation possible.

To operate "Put That There" the user sits in a chair facing a thirteen foot diagonal rear projected video screen in the "media room". This multi-media facility was built as an environment for experiments in interaction using multiple input/output channels. The user wears a headset microphone for the speech recognizer and the gesture recognizer is attached to a watchband worn at the wrist.

The speech recognizer, a Nippon Electric Company DP-100 (2), is capable of recognizing up to five words, from a 120 word vocabulary, of connected speech. Connected speech recognition was a necessary precondition of this research; any attempt to create a natural interface would fail if the user had to speak each work discretely. The gesture recognition is accomplished using a three dimensional digitizer built by Polhemus Navigation Sciences (3), which returns the position and attitude of a sensor with respect to a radiator forty times a second. A "gesture" consists of extension of the arm in the direction of the screen. The system computes from the sensor information the vector defined by the arm position and draws a cursor on the screen.

SPEECH RECOGNITION ENHANCEMENTS

"Put That There" is primarily a voice-interactive system, with gesture and visual feedback as ancillary input and output channels. The essence of the interaction is maintaining an intelligent conversation with the computer, where the intelligence is directed toward understanding both the user's speech and the task he/she wishes to accomplish.

The work reported herein was supported by the Cybernetics Technology Division of the Defense Advanced Research Projects Agency, under Contract No. MDA-903-77-C-0037.

Figure 1: In running "Put That There", the user points at the large screen using a position sensing device strapped to his wrist. He speaks to the system through the head-mounted microphone.

In exploring the application of speech recognition hardware to interactive systems, one quickly realizes that currently available speech recognizers, despite manufacturers' claims, are far from 100% accurate. This is particularly evident with connected speech recognizers, which are nonetheless more attractive for many applications, as they allow more natural spoken input. Although recognition technology is improving (with more of the processing happening at the chip level), we expect that the hardware will never come close to the human ear's ability to understand the complexities of variable pitch, intonation, and coarticulation. In fact our work starts from this assumption, exploring methods of overcoming shortcomings in speech input technology to take advantage of speech as a natural and intuitive human communication channel.

As a stand alone device, the speech recognizer is a context-free pattern matcher; it compares frequency spectra amplitude array samples of audio input with previously trained reference patterns. Upon completing a recognition, it communicates with the host computer ASCII codes for the recognized words. At this point we begin to apply context-sensitive speech processing. Although this analysis can be roughly divided into syntactic and semantic phases, the system will in fact branch back and forth between these phases, punctuated with spoken queries to the users, until it can execute a complete command.

The syntactic analysis compares the recognized parts of speech to model command sequences. Some commands require objects, those objects may need to be of a particular class (e.g. a type of ship), etc. In addition, some commands may translate into two models depending on syntax; "make that blue" means change it to blue, while "make a green freighter there" means create one. While matching input words against the command templates, the syntactic phase also builds up a data structure reflecting the current state of knowledge about the recognized speech.

At any stage of analysis, the software can question the user, through either the speech synthesizer or digital audio system. Each of these queries fills in missing or ambiguous elements of the command template, and any step of analysis is re-entrant by virtue of the stored command state structure. In particular, the syntactic analyzer will continue to ask questions and return to itself until it has adequate input to branch to routines handling unique commands.

At this point, semantic analysis begins (or resumes if more information has been requested). In this phase, the system tries to gather enough information to complete the action required by a command. The obvious first step is the meaning of the recognized speech; i.e., has the user asked to move something that is moveable (a ship) or something

that isn't (an island) or do we not yet know what the user wants to move? In the latter case, we again have the option of asking the user "what object" but first attempt intelligent interaction by looking at three sources of information to fill in the missing semantic data.

SPEECH ADJUNCTS

The first of these is gesture; specifically, was the user pointing at anything while speaking? Key elements of the gesture recognition are whether the user was reaching out in a pointing gesture, indicating intent, and approximate simultaneity of pointing and speaking. For example, if the user says "move that", while reaching out and pointing at some particular object, it is sufficient to recognize the word "move", as the gesture is what really specifies the object being referenced.

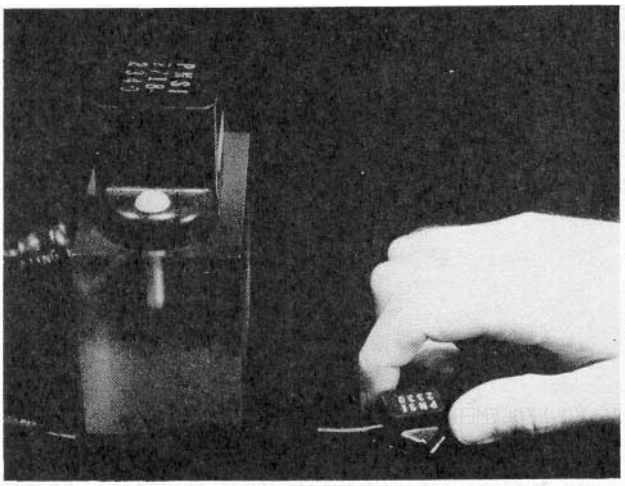

Figure 2: Gesture Recognition Hardware. Close up of the radiator and sensor pair of the six-degree of freedom digitizer.

The second source of added information is knowledge of the current state of the database. This is particularly powerful when combined with several assumptions of the user's intent, as described below. Cross-referencing a command with the database of existing shipping prevents attempting meaningless operations e.g. branching to a subroutine to move a "green freighter" when one does not exist. This process can also point out ambiguities which the user may not realize; if, for example, the user requests an operation on some type of ship when there are several instances of that type already in existence, the system will ask "which one".

A final source of intelligence exhibited by the system is the making of assumptions about the user's intent. This must usually be done in conjunction with information about the present state of the graphical database. If, for example, the speech recognizer hears "blue oil tanker" and one does not presently exist, the system assumes that the user wants to create one. On the other hand, if one or more blue oil tankers are already in use, it is time to ask "what command?".

At any stage of this analysis, questions may be asked of the user, resulting in an intelligent conversation between the operator and the machine. A person will be much more tolerant of imperfect speech recognition if the system's questions indicate of some degree of comprehension rather than if it simply says "please repeat that". The machine can continue to ask questions, incorporate the response into the syntactic model, and perhaps ask yet more questions, until a unique command is understood, at which point control passes to a particular subroutine programmed to complete that task with parameters passed from the analytic phase.

REDUNDANT INPUT CHANNELS

A key component of this experiment is the presence of redundant channels of input. The lack of accuracy in the speech input channel, to whatever extent it is present, requires that there be other ways for the computer to find out what the user wants done. In "Put That There" the second channel is gesture recognition. The ability of the computer to find out where the user is pointing allows it to continue to function when errors or omissions occur in the speech input.

In addition to providing the computer with alternative input channels, the presence of redundant inputs has benefits for the user, over and above the increased performance of the system. There are many situations in the operation of a complex system where the operator is already using one or several of the available input channels for other purposes but still wants to command the system. For example, in an airplane cockpit situation a pilot may be flying manually engaging hands and eyes yet want to change the frequency of the radio; voice would be the natural channel over which to make that change. At another moment the plane may be flying on autopilot and the pilot talking to the copilot. In that situation it would be best to change the radio frequency by hand rather than interrupt the conversation.

The point is that there is no single channel of input (whether it be voice, gesture, eye tracking, or touch) to which any command should be limited; all command functions should be activated by any input channel all of the time. In many cases the user may use more than one of the channels at the same time, such as when he points and says "move that". Complex interactive interfaces should be designed with a systemic approach. Speech recog-

nition should not be used to replace the pushing of particular buttons but rather all functions be controllable by all modes of input.

In the "Put That There" system as it exists today there are only two input channels - voice and gesture. The Architecture Machine Group has purchased an eyetracker which we hope to integrate into the system, in the near future, as a third input channel. One could imagine the situation where the user said "move that" and instead of pointing at something merely looked at it. This would allow a style of interaction that is even more "human", analogous to the situation every night at the dinner table when one says "please pass me that" while looking at the bowl of vegetables.

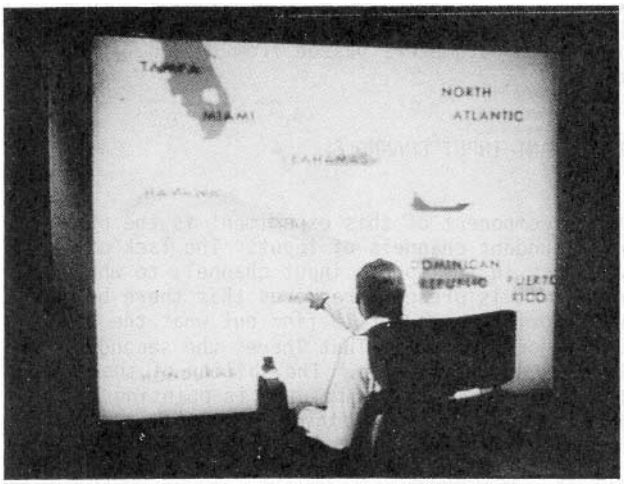

Figure 3: From a chair ten feet away, the user interacts with the large display.

FEEDBACK

Feedback is a very important aspect of interactive systems. When using the typical computer interface, a keyboard, feedback comes in the form of each key being echoed after it is hit; the feedback is unambiguous and immediate. When using speech recognition equipment for input such feedback does not exist (nor would it be appropriate). This lack of immediate feedback requires some degree of trust, on the part of the user, that the system understands what was said and what it should do. It also requires that the feedback, when it does arrive, be specific enough to assure the user of the system's comprehension.

Feedback occurs in both verbal and graphical forms in "Put That There". The system's verbal responses are quite detailed, asking precise questions. If, for example, the system cannot decide which of two objects are referred to it asks "which one". On the other hand, given a command (such as "move") without any object it asks "what object". The words "which" and "what" are specific indications of the system's knowledge.

Graphical feedback also aids the user. When the system determines which object is being referred to, it indicates this by outlining the object. This serves to reassure the user that the system has understood, giving encouragement. Before answering the system's questions it is helpful for the user to know what has already been comprehended. If the graphical feedback indicates the wrong ship the user can quickly abort the command sequence.

CONCLUSION

The focus of the work embodied in "Put That There" is on the person-machine interface. It is a demonstration of a style of interaction between a sophisticated computer based system and a human being. It is a realized argument for providing multiple channels of input/output in order to allow people to communicate with machines in a manner that is natural to them, as well as a testbed for ongoing research into the implementation of new channels of communication.

ACKNOWLEDGEMENT

The insight of Dr. Richard Bolt underlay the conceptual framework and interactive philosophy of the research described in this paper. His efforts are duly appreciated.

REFERENCES

1. Bolt, R.A., Put-That-There: Voice and Gesture at the Graphics Interface. Computer Graphics, Proceedings of ACM SIGGRAPH '80, Vol. 14. No. 3, 1980, 262, 270.

2. Kato, Yasuo, Words into action: A commercial system. IEEE Spectrum, June 1980, p 29.

3. Rabb, F.H., Blood, E.B., Steiner, T.O., & Jones, H.R. Magnetic position and orientation tracking system. IEEE Transactions on Aerospace and Electronic Systems. AES-15, No. 5, September 1979, 709-718.