Voice Input For Graphics And Text Creation: A Case Study

Corine Bickley, Hong Z. Tan, and David Horowitz
Center for Rehabilitation Technology
Department of Rehabilitation Medicine
Tufts University School of Medicine, Boston, MA

ABSTRACT

In this report, we describe how voice control is used by a quadriplegic architectural designer to create graphics such as house plans, and reports including spec sheets, customer quotes, and master specs. Graphics systems typically use mouse control to create, size, and place geometric shapes (lines, arcs, etc.) and even generic walls, windows, and doors. Most of these operations can also be accomplished using text commands. We integrated a speech recognizer (BUG by Command Corp.) with a standard graphics program (AutoCAD) to enable the designer to create house plans by voice. A few AutoCAD operations (such as trimming, zooming, and object-selecting) must be specified by pointing. These functions are accomplished by integrating a head-pointer and joystick (FreeWheel by Pointer Systems) with the system. In order to allow the designer to create standard reports which he needs to use in his business, we created a knowledge base for the Kurzweil Voice Report system (see [7]).

BACKGROUND

Alternative input strategies (other than keyboard and mouse control) for computer-aided design packages such as AutoCAD have been in use for a number of years [1,2]. Burnett and his colleagues, as well as our own research team [3], have used combinations of voice recognition, head pointing, and joystick control as input methods for AutoCAD. These kinds of systems offer the capability of competitive employment for individuals who otherwise might be unemployed or underemployed.

The client (JEM) in this part of our project is an architectural designer who has retained good control of his voice and head movement, and some limited control of his arms and hands for short intervals of time, despite a C-5 quad injury in 1988. Prior to his injury, JEM owned and operated for 13 years a small business in the area of design and modification of custom houses, with a substantial established client base. Our goal throughout this project has been to provide the technology and support that will enable JEM to re-establish his business in a competitive way.

In order for JEM to do his work, he must create and manipulate both graphical entities, such as engineering drawings, and also text, such as reports and contracts. Prior to 1988, JEM drew the house plans by hand, and prepared reports such as specification sheets and contracts by writing or typing. As both of these sorts of paperwork utilize standardized formats and techniques, each was well-suited to being created by a computer program. Handling graphical objects can be accomplished by computer-aided design packages, solid modelers, and parametric design systems. Creating formatted text, in the form of reports and contracts, can be handled by voice-driven report formatters. In an earlier paper (see [7]), we discussed how we applied knowledge engineering to the Kurzweil Voice Report system to enhance JEM's productivity of text creation.

The strategic integration of these systems along with a user interface tailored to the capabilities of a specific individual can now be built with existing computer-based user input and engineering technologies. The policy within our project is to use existing technologies so that our clients can make use of the customer support and maintenance opportunities offered by the technology vendor, and thus move beyond reliance on our research team in the years to come.

Our general approach has been to utilize a combination of technologies, each one of which addresses one aspect of the goal of operating JEM's business. In the area of creating house plans, we use AutoCAD for the manipulation of graphical images, and speech recognition (the BUG system) as well as a joystick FreeWheel) and a head pointer to control the AutoCAD program. In addition to the basic AutoCAD facility, we incorporated several related pieces of software such as macro packages (e.g., AutoArchitect) for accessing precoded AutoCAD routines for kitchen appliances, bathroom fixtures, and other architectural symbols.

We are also using a combination of approaches in the task of text creation. We use speech recognition and knowledge engineering for the creation of reports with a specific format (using the KVR system mentioned above) and plan to incorporate a Kurzweil Voice system along with a text editor (WordPerfect) for generation of free text. JEM needs both sorts of facilities to accomplish various tasks such as writing customer contracts and bills as well as writing business letters and memos, and perhaps for creating and maintaining records of business activities.

SYSTEM DESIGN

Much of our design of the portion of the user interface involving voice was influenced by the restriction of a small vocabulary size: in the case of BUG, we are limited to approximately 200 words in any one lexicon. In order to be able to spell out infrequently used commands as well as labels and file names, we set aside 36 words in the lexicon to be used for letters and digits. We found it useful (in order to improve recognition accuracy) to use communication codes for some of the more acoustically similar names of letters, such as "T" and "P". We tried a combination of code names and letter names, using only the code names for the most easily confused letters like some of the E-set ("C", "E", "G", "P", and "T"). As it is easy to retrain a word and its associated prompt, this technique has worked well. During the AutoCAD training phase of the project, we were able to evolve and refine our choice of words over time.

An additional nine words are used for punctuation. We use a period (.) for DOS file names and for decimal numbers (the lexicon entry "point" was added as an alternative for specifying a period), and a comma (,) for entering coordinates. A negative sign (also allowed as the word "minus") and the single and double quotes (called "feet" and "inches") were added to the lexicon for specifying coordinates, and a backslash for filenames. The at sign (0) is also needed for the AutoCAD specification of offset dimensions. Four words used for editing and entry functions of backspace, control-C, enter, and return were included.

The AutoCAD commands were broken into several functional groups that are roughly similar to some of the pull-down menus (see Fig. 1). This way of grouping AutoCAD commands into BUG command sets improved recognition accuracy by decreasing the target words available to the recognizer for matching at any one time. In terms of recognition robustness, this scheme has worked very well.

The increase in accuracy comes at the cost of an inconvenience, though: the set name must be spoken in order to switch the recognizer into a mode in which it expects only the words in the selected command set. There are two drawbacks to this approach. The first is the need to speak additional words, i.e. the command set names. For instance, in order to accomplish the AutoCAD actions of zooming the view in an area and then drawing a line, the spoken commands shown in Fig. 2 need to be spoken. Notice that only some of the spoken words (shown in the left column of Fig. 2) generate keystrokes (shown in the right column). Words like "tools" and "spell" do not generate keystrokes; they instead switch the BUG recognizer from one command set to another.

The second disadvantage of using command sets is the reduction in the number of voice commands that can generate keystrokes. For example, five words of the 200 or so are used to activate commands sets; therefore, only 195 words can be used to generate keystrokes.

Examples of voice commands and the associated keystrokes are shown in Fig. 3; the right-most column shows the command sets that are disabled or deactivated when each command is recognized. As soon as one of the command set identifier words is spoken, the recognizer augments the set of possible targets to include the words in that command set. The other command sets are at the same time deactivated, and their templates are removed from the set against which a spoken word will be matched. This strategy of activation (enabling) and deactivation (disabling) is under the control of the BUG user during recognizer training. We have chosen an approach in which only one command set in addition to the set that is never off is enabled at one time.

DISCUSSION

The design of the command sets and command words was motivated by the need to create small enough command sets such that the members of each command set were acoustically dissimilar enough for the BUG recognizer to perform robustly. It would be possible to structure the lexicon so that all 200-odd voice templates generate keystrokes and are active or enabled at all times. The drawback of this approach is that the maximum possibility for misrecognition exists because each voice command will be compared against all 200-odd possibilities in the vocabulary.

Another approach would be to use a maximally deep tree structure for organizing command sets. For instance, it would be possible to have only two templates active at any time for comparison with a spoken word. In this case, recognition can be maximally robust, provided that the two words in each command set are selected to be acoustically dissimilar. However, such a structure would prove to be unwieldy in practice because it would be necessary to speak six command set selectors to be able to speak one AutoCAD or keystroke command. Not only would the process of entering (speaking) a keystroke command (the eventual goal) be time-consuming and inefficient, but also the cognitive load placed on the user would be unreasonable.

Obviously, some compromise between these extremes is indicated. We chose the kind of strategy shown in Fig.1. Some words are always available; the BUG recognizer always has ready for recognition the voice templates for words such as those shown in the leftmost column. Our goal was to select for the Never Off set commonly used words as well as important commands such as control-C (to allow the user to return to top level in AutoCAD from any lower level prompt) and of course the word triggers for the command sets. The grouping of words into specific command sets was modeled in part on the AutoCAD pull-down menus. These menus group functionally similar commands, such as commands that modify a drawing, or AutoCAD tools commands. The commands for letters were grouped into the Spell set, and the numbers and punctuation for specifying coordinates were placed in the Draw set. Not all AutoCAD commands are available directly by voice commands in our system, due to the 200-word limitation. Any command, however, can be entered by activating the Spell command set and speaking each letter.

Inexpensive commercially-available speech recognizers are typically speaker dependent and permit only a small vocabulary to be used. Because JEM is the sole user of his system, the speaker dependent constraint is not a problem. As Evans [6] explains, only a relatively small set of AutoCAD commands is actually used very frequently. Thus, the training that is necessary with a speaker-dependent system is not a burden. Although only a few words are being used, the choice of these words is arbitrary, and the vocabulary items can be selected and modified easily by the user. These features are compatible with the requirements of voice control of menu-based programs such as AutoCAD, or any computer program with a fixed, relatively small-sized vocabulary. JEM's specific needs for creating architectural drawings are well satisfied within these constraints. For accessing AutoCAD commands that are not trained in the lexicon, JEM can use a spelling mode to enter any string of characters (e.g., a less common AutoCAD command, or label string, or file name). The actual rules of architectural design remain in JEM's head, and thus the large vocabulary of architectural terms does not need to be recognized. Only the words which correspond to the results, of applying those rules to the creation of a specific design are needed: the commands for drawing lines and arcs, and for moving and erasing graphical objects, and other words related to graphical tasks.

During the initial stages of system design and implementation we found it necessary to choose alternative names for several AutoCAD commands in order to allow the BUG recognizer to perform adequately. This recognizer uses a simple strategy based on energy contours [5] to differentiate word targets.

The slow recognition rate of many currently available speech recognizers is also not a limitation in control of AutoCAD by voice. The important time savings results from the speed with which AutoCAD drawings can be modified and replicated, as compared to the hand creation of drawings. The architectural field has long recognized this fact and has incorporated the advantages of computer-generated drawings into standard practice, particularly in larger-sized operations. The focus of our work has been to make this technology available to a quadriplegic architectural designer - and in a way that uses voice instead of keyboard control. The goal is to demonstrate that voice input can be as fast as keyboard input to AutoCAD. The degree to which we are successful in achieving this goal will demonstrate the degree to which it is possible to make an individual who is neuromuscularly impaired able to be competitive in his field.

A side benefit of the implementation of the computer-based system for JEM is the standardization of the design process which is possible when systems such as AutoCAD along with various macro packages for fixtures, windows, doors, etc. are used in place of a process of hand drawing. In essence, using these macros is an initial step in the direction of rule-based design of house plans. Further work in this direction is in progress [4]: knowledge-based engineering using The ICAD System will facilitate the encoding of rules for architectural design into a computer program.

CONCLUSIONS

The voice technologies described in this paper have provided our client with an alternative form of input to a computer instead of a keyboard and mouse. His standard forms of reports and contracts are available to him once more, now in a voice-accessible format. Using the BUG recognizer and the FreeWheel joystick, JEM is now able to create house plans using voice and pointing. The BUG recognizer has served our task adequately well; the use of several command sets appears promising. Head pointing is a tiring activity for JEM after a few hours; he uses the joystick almost entirely for pointing.

JEM reopened his business during the fall of 1992, and expresses confidence in the voice technology and in his ability to carry on with his work.

FUTURE WORK

Further study is needed in the areas of long-term, daily use of a system which uses speech recognition. Issues to consider include determining guidelines for overall duration of use, both within one work session and throughout a work day, and for the recommended frequency of updating the voice templates.

ACKNOWLEDGEMENTS

This project was supported by the U.S. Department of Education Rehabilitation Services Administration Grant No. H128AO1011-92. We also appreciate the efforts of Christopher S. Jones and Janet Wang during the AutoCAD training phase of the project, and the loan of computer equipment by Prof. Harry West of the Mechanical Engineering Dept. of the Massachusetts Institute of Technology.

REFERENCES

1. Burnett, J.K. (in preparation) A networked, voice and head movement CAD workstation for quadriplegics.

2. Burnett, J.K., C.R. Klabunde, and C.W. Britell (1991) Voice and head pointer operated electronics computer assisted design workstations for individuals with severe upper extremity impairments. RESNA '91, Proceedings of the 14th Annual Conference. RESNA PRESS, Washington, D.C., pp. 48-49.

3. Bickley, C., C.S. Jones, J. Wang, H.Z. Tan, R. Ebert, and D.M. Horowitz (submitted) Case study of voice control of AutoCAD. Submitted to Proceedings of ECART 2.

4. Bickley, C. and S. Hunnicutt (submitted) A voice-accessed predictive knowledge-based engineering system. Submitted to Proceedings RESNA '93.

5. Command Corp., Inc. (1988) Bug voice command system, Bug DOS professional system user's guide.

6. Evans, D.E. (1990) Boosting CAD productivity in miles per hour, Cadence.

7. Tan, H.Z., R.R. Ebert, D.M. Horowitz, and C.A. Bickley (1992) A comprehensive speech recognition service delivery model illustrated with case studies. In Proceedings of CSUN '92, Technology and Persons with Disabilities, March 1992.

Figure Captions

Figure 1: Excerpt of chosen command set structure. (This listing does not contain every AutoCAD command which can be accessed by voice. Only a few representative commands are shown in each column.)

Figure 2: Segment of AutoCAD voice-input session.

Figure 3: Examples of voice commands and associated keystrokes. The command sets which are de-activated when each command is recognized are shown in the right-most column.