Talk your Way round – A Speech Interface to a Virtual Museum

Lindsay Evett and Yeow Kee Tan

Department of Computing,

The Nottingham Trent University

Burton Street

Nottingham, NG1 4BU

UK

Tel: 0115 848 2168

Fax: 0115 848 6518

Email: lindsay.evett@ntu.ac.uk

Abstract

Purpose: To explore the development of a speech interface to a Virtual World and to consider its relevance for disabled users.

Method: The system was developed using mainly software that is available at minimal cost. How well the system functioned was assessed by measuring the number of times a group of users with a range of voices had to repeat commands in order for them to be successfully recognised. During an initial session, these users were asked to use the system with no instruction to see how easy this was.

Results: Most of the spoken commands had to be repeated less than twice on average for successful recognition. For a set of ‘teleportation’ commands this figure was higher (2.4), but it was clear why this was so and could easily be rectified. The system was easy to use without instruction. Comments on the system were generally positive.

Conclusions: While the system has some limitations, a Virtual World with a reasonably reliable speech interface has been developed almost entirely from software which is available at minimal cost. Improvements and further testing are considered. Such a system would clearly improve access to Virtual Reality technologies for those without the skills or physical ability to use a standard keyboard and mouse. It is an example of both Assistive Technology and Universal Design.

1. Introduction

There are many benefits associated with the use of a speech interface to a computer system. Such an interface would allow hands-free operation, telephone access and easy mobile use (e.g. dictating notes while walking around). A speech interface may be preferred, may be more compatible with certain tasks (e.g. 3D-navigation [1]) than standard keyboard and mouse [2], and is an important part of multi-modal systems. Multimodal operation is the natural mode of operation in many aspects of every day life. Speech would be a natural component of Ubiquitous Computing [3], where the aim is to make the technology invisible to the user, and so is clearly part of the future of computing systems. Importantly, speech is essential to enable systems to be used by many disabled people and so is important for Universal Design, where the aim is to design systems which are usable by all. Two good examples of this are the use of text-to-speech for the visually impaired and those with reading problems (clearly, spoken input would be desirable for its compatibility with such systems) and the use of spoken input for people who cannot effectively use a keyboard and mouse, whether this is because of lack of skill, control or strength. This is especially important for the motor-impaired, who do not have a simple motor relationship with the physical world to start with. Those without motor problems have to learn to use the keyboard and mouse and to map them onto operation – there is no simple, direct mapping – and this would be more difficult for the motor impaired. This would be especially true for navigation in a virtual world, which is the subject of this paper.

This paper describes the development of a speech interface to a virtual world. The virtual world in question is a representation of the Nottingham Castle museum. The Nottingham Castle Museum is situated on a hill, with steep walkways leading up to it and stairs to the main entrance. The museum itself is on several floors. Good facilities for disabled access exist in the museum. However, both for this and for other museums and attractions, a Virtual Museum, particularly with a speech interface, presents a number of additional benefits.

Firstly, the Virtual Museum allows the visitor to wander around and to determine places of interest. The visitor can then assess areas of interest in order to decide whether or not to actually go to them. This is useful for those with limited mobility. In other cases, it may be that this is the only way the visitor can view the museum. Virtual Reality (VR), according to the Virtual Reality Society [4], is a generic term associated with computer systems that create a real-time visual/audio/haptic experience. It should be noted that many instances of ‘virtual museums’ to be found on the World Wide Web consist simply of a sequence of linked photographs, so such systems are stretching the use of the term. At the other extreme, projects are concerned with creating a complete immersive, interactive, dynamic 3D experience (e.g., see [5, 6]). The present system gives a virtual representation of size, shape and layout, allowing the visitor to move around the museum and to judge for him/herself accessibility according to individual need. It is not immersive, but allows some interactive 3D experience. The fact that it is not immersive makes it more practically accessible.

The speech interface allows hands-free access to the system. Hands-free access is desirable for many reasons (e.g., see [7]), but especially so for users without the strength, control or skills to tackle a keyboard and/or a mouse. The use of a speech interface means that such visitors can explore the museum. Visitors can navigate around the Virtual Museum by use of spoken commands, both within a room and between rooms. Feedback about navigation is given orally. Visitors can move to particular items in the museum and then request details about that item. These details are given both visually and orally. A Virtual Museum allows remote access, easy browsing, and autonomy, interaction and presence. The use of speech creates an interface which is more compatible with the navigation metaphor than the standard keyboard and mouse. The speech interface addresses the main problems for computer access for the disabled – that is, the use of keyboard and mouse for input and control, and the use of the monitor and screen for output and display [8].

People tend to have high expectations of speech recognition systems [9], derived to a large extent from portrayals of them in the media. Ideal systems, which are multi-user, large vocabulary, continuous speech recognition systems, are in fact still a long way off (for details of processes involved in the machine recognition of speech, see [10], [11]). Recent advances mean that more realistic systems, which are some combination of single user, limited vocabulary, non-continuous (i.e., single words and short phrases) speech, are becoming practical. Consequently, some applications are becoming a commercial reality. For example, the Dragon NaturallySpeaking [12] system, which enables spoken input, is now available at a reasonable price and is becoming ever more widely used. This system is large vocabulary, but requires training by an individual user and only accepts single word input. The system trains itself continuously while it is being used and requires a rather laborious initial training phase in order to achieve acceptable performance. Some users find speech input using this system preferable to traditional input. The error rate is also acceptable, once the system has been trained, especially to users who cannot use keyboard and mouse. For example, the first author knows of a colleague who suffers from arthritis in his hands, the severity of which can vary from day to day. Since the use of computers is an everyday part of his job, having the use of the Dragon system means that his life is much less disrupted (he was actually using an earlier version of this system, DragonDictate).

It is important that when systems such as this do make errors, those errors are dealt with in a way that is logical to the human user. The Dragon system gives a list of candidates when it fails to recognise a word, and bases its list on its analysis, so that the alternatives at least bear some acceptable similarity to the mis-recognised word. Other types of errors are concerned with the users' model of how spoken commands relate to the actions of the system. In the context of a speech interface to a virtual world, [1] considers semantic errors caused when the user either makes errors of spatial reasoning, or where the control metaphor of the system does not match the user's model of the system. In the system studied, a speech enabled VRML browser, the author identifies ways of supporting the interaction metaphor to prevent such errors. Methods of trapping speech recognition errors to prevent them having damaging effects on operation are also explored [1].

The present system is envisaged as being available to anyone who wishes to use it. Consequently it must be multi-user and require no training. Because of this, the system must be limited in other ways in order to obtain sufficient speech recognition performance. Since there are a limited number of actions necessary – movements around the world and requests for information – the system is naturally limited vocabulary. There is no need for continuous speech recognition since again it is natural for the system to be operated by single words or short phrases.

[13] has identified a number of issues which need to be addressed when designing speech interfaces. Because, as already pointed out, current speech technology limits the range of allowable utterances in order to achieve acceptable performance, an important concern is how to let the user know the constraints on utterances. People are more tolerant of errors in graphical interfaces, and of visual feedback on those errors, than they are of errors in spoken interfaces with spoken feedback. Visual feedback is natural and easy in the present application. The limited nature of the speech input required for the present application avoids the pitfalls encountered where more complex speech is involved [13].

2. The Nottingham Castle Speech Recognition Virtual Museum (SRVM)

A software quality study was carried out in order to develop a System Requirements Specification for the system (cf., [14]). The development of a draft user manual enabled clarification of proposed functions and identification of problem areas within them. A questionnaire study identified desired functions of the system. It established that potential users found the idea of the museum interesting and would like to try it; that they would like advanced navigation functions, to facilitate exploration and to overcome problems when lost; that they would like easy access to information about items; that they would prefer information to be presented as both text and speech, with the speech being in a soft, female, voice; that they would like the virtual world to represent a real world. A number of desired functions were identified [15].

The system was developed using software that is available at minimal cost. This means that such a system is potentially cheap to both develop and produce, making wide availability of such a system a realistic possibility. Additional costs would be incurred for any commercial enterprise. The system was developed based on Virtual Reality Modelling Language (VRML) using Microsoft Visual Basic through Microsoft Worldview for Developer. This allowed the VRML component to be combined with the speech component using Visual Basic. The speech component was developed using Microsoft's Speech Application Programming Interface (SAPI). Worldview for Developer and SAPI are components of Worldview Active X (not yet fully available and requires licensing). Microsoft Agent was used to integrate animated characters (such as Robbie the Robot and Merlin the Wizard) into the application. These characters were used to provide spoken narrative, act as tour guides, instructors and provide descriptions. A database contained information about the artefacts in the museum, and was consulted by the system when the user requested details about a certain artefact. The database contained the artefacts' location, picture, category, and information which the agent could read out. Platinum’s VR Creator – Learning Edition was used to create the virtual museum. This only allowed a world with a limited number of things in it. This meant the museum had to be divided into individual rooms containing a very limited number of things. Because of this the Virtual Museum was fairly empty, and transition through the museum could not be as smooth as would be desired. However, it is sufficient for prototype development to enable testing of the basic ideas. For full technical details, see [15].

The system contains the following features: an overview map, which indicates the users position within the museum; an artefacts database; a “What I can say” list of commands; navigation buttons; teleportation, allowing the user to move from one room to another instantly; synthetic narration; an interactive tour guide; an interactive help system. Functions were either validated by, or emerged from, the System Requirements Specification exercise outlined above.

Care was taken to make the interface clear and easy to use. For example, the following were used: large buttons which were compatible with their function (e.g., navigation buttons had appropriate arrows on them); buttons instead of text where possible; a map to inform the user of their current position in order to aid navigation; the use of simple language. When a movement command was made, the amount of movement was set to that considered acceptable to the user, based on viewpoints. Viewpoints were placed into the rooms, so that they were in reasonable positions with respect to the furniture, cabinets and cupboards. When the user wishes to move in any direction (forward, back, left, right) the program simply moves to the nearest appropriate veiwpoint. This means that movements are always to sensible places with respect to the room and its contents, and are not too far, to avoid disorientation. Attempts were made to use appropriate error handling, and appropriate use of feedback and confirmations (cf. guidelines for the design of effective speech interfaces, [16]).

3. Testing the system

10 subjects tested the ease of use of the system and recognition reliability (commands are listed in Table 1). These were computing students on the final year of their degree course. While not familiar with the use of speech recognition technology, the students were familiar with computer applications in general, and, hopefully, at ease with computers. Clearly, further testing is desirable with subjects less familiar with computer applications, and with subjects who have some kind of disability, particularly those with disabilities for whom this application is appropriate.

[Insert table 1 about here]

Each subject used the same system under the same conditions. They sat wearing a headset in front of the computer in a quiet room. An experimenter sat slightly behind them and was available to help if necessary and to take notes. The subjects represented a range of voices and pronunciations. Each subject was led through the system trying all the commands in the same, listed, sequence. If the system failed to recognise a command, nothing happened, and in such cases subjects were instructed to repeat the command. If the system mis-recognised the command as something else, subjects were instructed to wait until the system had carried out the wrong command, and then try again. Clearly, the latter could prove unacceptable in a real system. An undo command would be useful in this respect. One possible method of dealing with errors is to pre-empt them by the system producing a list of possible interpretations of any command with the user being required to verify the desired one before any action is carried out, as in the Dragon system, where verification is spoken. Using speech to deal with recognition errors in some way is desirable in order to maintain the advantages of the speech interface. Requiring verification for all commands can become tedious and may not be necessary in the present application, since errors tend to be ones of non-recognition rather than mis-recognition because of the limited command set. An undo function could well be adequate. This point needs further investigation. Dealing with recognition errors in speech interfaces is a challenge [17, 18].

The recognition results are shown in Table 2, where the average number of tries to achieve correct recognition for the different types of command is given.

[Insert table 2 about here]

While the majority of commands required less than 2 tries in order to be recognised correctly, greater difficulties were experienced with the teleportation commands. Some effort was put into choosing commands that were distinct, in order to reduce potential confusions by the system. It is clear from the pattern of results that improvements can be made immediately. For example, the command ‘View’, an explore command, was recognised less well than average (1.9). This is because the sound for ‘V’ has a slow onset. A different command needs to be chosen, which remains transparent and simple as to function, but is more distinctive acoustically. The teleportation commands were problematical. They were chosen not for their distinctiveness but rather were determined by the method used for coding rooms and the location of objects within them, the consequences of which were not anticipated. These commands were all of the form ‘Go’ followed by the letter identifying the room. Since it was only that letter which determined the difference between the commands, and some of the letters were very similar acoustically (e.g., B, D and E), it is not surprising that difficulties were experienced by the system for these commands. Clearly it would be easy to make these commands such as to be recognised more reliably. This would require changing both the formats of the object and room identifying codes and the organisation of the database.

Overall, the speech recognition results are encouraging. The subjects represented a fairly wide range of voice and pronunciation, and the choice of commands can be greatly improved. Under these conditions initial performance is quite good.

In order to assess the suitability of the interface, the 10 subjects who tested the speech recognition tried out the system in an initial session. The users were invited to try the system without guidance or instruction for as long as they wanted to. If any problems occurred whilst using the system, these were noted and help was on hand to enable the user to carry on using the system.

Users were able to use the SRVM without instruction, and none needed clarification. The users found the information presented by the system easy to follow, and the narration presented by the tour guide helpful. Their only problem with this feature of the system was not being able to interrupt it. The users' main complaint about the system was the lack of objects in the rooms and the quality of the images of artefacts presented. While the images were generally presentable, the quality of the digital camera used did limit their clarity.

None of the users raised any criticisms of the actions carried out on recognition of the commands. It would be useful for a further study to vary the distances moved and the viewpoints in order to optimise these, although it is encouraging to note that they were apparently successful. No problems of disorientation or "getting lost" were reported; the map proved useful in this respect.

Generally, users thought it was harder to navigate using speech than using traditional methods. However, it should be remembered that these are users who are very familiar with using keyboard and mouse for controlling computer applications.

4. Discussion

Within its limitations, the system worked reasonably well. There are a number of straightforward ways in which the system could be improved. The set of commands could be changed to improve their discriminability – this was an obvious limitation in the case of naming the rooms with letters. The speech recognition could be immediately improved by using a better quality of microphone and by using more sophisticated recognition software. The quality of the virtual world could be immediately improved by using a better quality of camera and using complete versions of VR Creator and Worldview Active X. This would allow the Museum to be adequately populated with good quality visual images, and would allow smoother movement through the rooms. While these improvements would obviously make for a more expensive system, it is important to note how much has been achieved on a minimal budget. Even with higher quality equipment and software, the system would not be hugely expensive, as these items are becoming more easily available and less expensive all the time.

Further testing is necessary to establish improvements in more detail, such as how effective the feedback and general controls are. While the way the veiwpoints were laid out appeared to be satisfactory, this would need to be tested more thoroughly, especially with a more densely populated museum. Ease and speed of navigation should be assessed and any errors investigated. The design and use of the map should be investigated in order to optimise its effectiveness.

Testing with disabled users is also a necessity. For a practical application, it would be necessary to tailor the system much more to the requirements of particular user groups, and especially of those with different kinds of disabilities. The purpose of the present study was to develop a prototype in order to demonstrate the feasibility of the project. Clearly, further work is necessary to enhance the functionality and usability of the system.

In his discussion of the next generation of post-WIMP interfaces, [19] argues that the ideal interface is no interface: ‘The goal we wish to strive for with today’s user interfaces is to minimise the mechanics of manipulation and the cognitive distance between intent and the execution of that intent’. This would make it ever more possible for the user to attend to the task rather than the technology of the task. To achieve this goal, the use of gesture and speech recognition would play an important part. This would be especially the case for 3D tasks, where the mapping between 3D and 2D control devices is particularly strained. These arguments are consistent with the notion of Ubiquitous Computing, where the use of computers is enhanced by making them effectively invisible to the user [3]. Such developments are entirely compatible with the concept of Universal Design. Universal Design is ‘the concept of designing products that are usable by all people, including people with disabilities’ [16]. Universal design will improve the usability of an application as well as making it suitable for users with disabilities. The present system addresses the main problem areas for the disabled in computer access [8]: use of the keyboard and mouse for input and control, and use of the monitor and screen for output and display. The system is an example of both Assistive Technology and of Universal Design [16]. Assistive Technology is targeted at the needs of a particular, usually small, user group (e.g., sip and puff device) and can be expensive. Technology developed in this way may be of benefit to other groups. Universal design attempts to design systems that are usable by all, including those with disabilities. Universal design builds in compliance with disability legislation, and because it is intended to have a large market, is potentially easier to fund and therefore develop.

5. References

1. Turunen M. Error handling in speech user interfaces in the context of virtual worlds, K.-J. Räihä (ed.), Advanced Course on Human-Computer Interaction, Proceedings of ACHCI'98, University of Tampere, Department of Computer Science, Report B-1998 **4:** 68-74
2. Connor B., Snibbe S., Robbins D., Zeleznik R., van Dam A. Three-dimensional widgets. In *Symposium proceedings on Interactive 3D graphics,* 1992: 182-188, ACM
3. Weiser M. Some computer science problems in ubiquitous computing. *Commun. ACM* 1993, **36**: 74-84
4. Virtual Reality Society glossary, http://www.vrs.org.UK/VR/reference/glossary.html, accessed 2.10.01
5. Sparacino F. City of news. SIGGRAPH 99: Emerging Technologies, exhibit, 1999
6. Sparacino F. Wearable cinema/wearable city: Wearables for 3D browsing, interactive cinema and museum tours. MIT Media Lab research project, 2001, http://www-white.media.mit.edu/~flavia/projects.html, accessed 3.10.01
7. Evans J. R., Tjoland W.A., Allred L.G. Achieving a hands-free computer interface using voice recognition and speech synthesis [for Windows-based ATE]. *IEEE Aerospace and Electronics Systems Magazin*e 2000, **15:** 14-16
8. Wilson L. Assistive technology for the disabled computer user, UNC-CH Institute for Academic Technology, 1996, http://www.unc.edu/cit/guides/irg-20.html, accessed 9.01

9. Dobroth K., Interview in Weinschenk S., Barker D T. Designing Effective Speech Interfaces, New York: Wiley, 2000, pp. 215-223

10. Jurafsky D, Martin J H. *Speech and Language Processing*, New Jersey: Prentice Hall, 2000

11. Holmes J N. *Speech Synthesis and Recognition*, Van Nostrand Reinhold (UK), 1988

12. Dragon NaturallySpeaking Standard, Dragon Systems, Bucks. UK, 1999

13. Yankelovich N., Interview in Weinschenk S., Barker D T. Designing Effective Speech Interfaces, New York: Wiley, 2000, pp. 314-324

14. Somerville I., Sawyer P. Requirements Engineering Practise: A Good Practise Guide. Chichester, John Wiley & Sons, 1997

15. Tan Y-K. Speech Recognition Virtual Museum, Unpublished Final Year Project, Department of Computing, The Nottingham Trent University, Nottingham, UK, 1999

16. Weinschenk S., Barker D T. Designing Effective Speech Interfaces, New York: Wiley, 2000

17. Karat J., Interview in Weinschenk S., Barker D T. Designing Effective Speech Interfaces, New York: Wiley, 2000, pp. 33-42

18. Oviatt S., Interview in Weinschenk S., Barker D T. Designing Effective Speech Interfaces, New York: Wiley, 2000, pp. 253-265

19. van Dam A. Post-Wimp user interfaces, *Commun. ACM* 1997, **40:** 63-67

Table 1: The set of commands

Type of Command	**Possible Commands**
Explore mode	Forward, Backward, Turn Left, Turn Right, Search, Teleport, Zoom Map, View, Quit
View Mode	Previous, Next, Say, Search, Exit, Previous Page, Next Page
Exit Query	Yes No
Teleport	Go *letter*

Table 2: Average number of tries to achieve correct recognition for the different types of spoken command

Type of command	**Average number of tries**
Explore commands	1.6
View commands	1.5
Teleportation commands	2.4
Yes/No (confirmations)	1.7
Exit	1.4