

Browsing the Web from a Speech-Based Interface¹

Josiah Poon

Basser Dept of Computer Science
University of Sydney, Australia
josiah@cs.usyd.edu.au

Chris Nunn

Polymorphic Solutions Pty Ltd
Milton, QLD, Australia
cnunn@polymorphicsolutions.com.au

Abstract: The design of information presentation on the web is predominately visual-oriented. This presentation approach requires most, if not all, of the user's attention and imposes considerable cognitive load on a user. This approach is not always practical, especially for the visually impaired persons. The focus of this project is to develop a prototype which supports web browsing using a speech-based interface, e.g. a phone, and to measure its effectiveness. The command input and the delivery of web contents are entirely in voice. Audio icons are built into the prototype so that users can have better understanding of the original structure/intent of a web page. Navigation and control commands are available to enhance the web browsing experience. The effectiveness of this prototype is evaluated in a user study involving both normally sighted and visually impaired people. Useful lessons learnt from this experiment helps drive further research to a better speech-based interface.

Keywords: speech-based interface, phone browser, visually impaired, normally sighted

1 Introduction

World Wide Web (WWW) is rapidly emerging as the universal information source for our society. The WWW is generally accessible using a web-browsing package from a networked computer. The design of information on the web is visually oriented. The reliance on visual presentation places high cognitive demands on a user to operate such a system. The interaction may sometimes require the full attention of a user.

In addition, there are population demographics await resolutions. Take for example, the visually impaired who are limited in their use of standard PC-based web browsing systems. The other group is the aged people who do not generally have good eyesight and steady hands to read from the screen or to coordinate a keyboard/mouse. People also have difficulties to use existing browser technology if they have arthritis. Apart from these physical and/or health constraints, there are also socio-economic factors that constrain people from the possession of computers, e.g. financial ability. Surfing the web with computers is not an option for everyone.

The everyday telephone is an almost universally known interface for communication between people using voice. In recent years, with the introduction of the touch-tone system, phones have also become popular as a gateway to telephone-based navigation systems, for example phone banking. The telephone is simple to use, requires little attention from the user to operate, and has no associated visual output. This makes it an ideal communication tool for environments where the user is engaged in activities that require considerable concentration. It is also an ideal tool for the visually impaired.

Now we can imagine situations where it might be more useful to access information on the WWW via a telephone. Take for example information such as stock prices, weather information, international news stories, traffic details; the list is endless. A user may wish to stay informed about changes to such information while partaking in activities (for example, driving an automobile) that do not allow for the luxury of sitting in front of a computer and concentrating solely on its operation.

There have been a number of prototypes, research-based and commercially products incorporating non-visual interfaces. Their primary

¹ The research was conducted in the School of CSEE, U. of Queensland, Australia.

objective is to give visually impaired users an efficient and successful web browsing experience. However, to date, these commercial products as well as much of the research undertaken in this area, focus on the output from the system to the user. It is often *assumed* that the user has access to a standard input device (for example, a typical PC keyboard) for the purpose of issuing commands to the system.

The aim of this paper is to present a web browsing prototype. Section 2 looks at how current research relates to this project. The architecture and the functionality supported by the prototype, TeleBrowse, are introduced in Section 3. Details of the experiment are mentioned in Section 4 which present a user study to evaluate the effectiveness of the prototype as a web-browsing tool. Special emphasis is made to the different response from the normal sighted group and the visually impaired group. Discussion in Section 5 looks at the implication of the design of a speech-based interface regarding to the feedback from the study. The future work can be found in Section 6 of the paper.

2 Related Research

HTML is designed to be a mark-up language. Many of the structures in a document, such as hyperlinks, headings, tables and lists, are represented explicitly in the HTML file for the document by 'tags'. It is the task of a web-browsing program to interpret the tags, to format the content and to present the information to the user *visually*. There are several possibilities to re-represent the content through the audio channels.

One possible approach is to purposely design an audio document for the relevant web page. It may involve the author making an explicit recording of the document or parts of the document. Though this seems like the best strategy to ensure the author's intent is accurately rendered, it means that authors must create two documents for everything they write, which is obviously impractical.

A similar approach is the development of a mark-up language for use with voice browsing applications. This is the long-term solution offered by the W3C group. All the web documents are expected to be marked up according to a VoiceXML specification (W3C, 2000), and that browsing products need then only read and interpret these voice-specific tags to produce an audio version of the document. However this requires not only a global acceptance of the specification, but that all authors then use this specification when designing

their HTML documents. Otherwise, only certain web pages will be 'viewable' using compliant voice browsing applications.

A third approach is to render the audio information by working directly from the visual representation. Many applications using this approach has the primary goal of improving access to the WWW for people with a visual disability. Whatever the origin or purpose, most applications can be categorised into one of the following: screen readers, speech-enabled browsers, or voice browsers.

Screen readers provide generic support and are not application-specific. The central issue is that the *visual* rendering of the underlying application determines how the document should be represented in audio. In this way, audio becomes a secondary interface modality for presentation of the information. Also, screen readers are sequential in nature. The applications do not understand what they are reading, not in terms of the actual comprehension of words, but in terms of the structures of a document. Thus, it is unable to skip to a certain paragraph when you ask a screen reader to move to the next paragraph. Another drawback of most screen reading products is that they continue to assume keyboards be used to interface with the system. Despite their drawbacks, screen readers are very popular with the visually impaired. The two most popular products on the market are JAWS (<http://www.hj.com/JAWS/JAWS.html>), and Window-Eyes (<http://www.gwmicro.com>).

Speech-enabled browsers are essentially screen readers customised for working with HTML pages. Products in this category are relatively new compared to screen reading technology. As such, there are research projects (Zajicek and Powell, 1997; Asakawa and Itoh, 1998) as well as commercial products, e.g. Marco Polo by Sonicon (<http://www.webpresence.com/sonicon/marcopolo/>). A speech-enabled browser typically works using a standard keyboard interface for input, and a combination of visual and audio for output. The primary difference between a speech-enabled browser and a screen reader is that the former one understands and can interpret the structure of the document it is reading. This allows a user to move between structures (via a suitable interface) such as paragraphs and links, and most importantly, it allows a user to skip information that is of no importance to them.

Research based browsers such as BrookesTalk (Zajicek and Powell, 1997) is experimenting with

the use of auditory icons to facilitate the representation of HTML structures through audio, while other browsers such as Emacspeak (Raman, 1996) uses speech cues to represent structures. However, the drawback with using all these browsers is their input interface, still relying on a standard keyboard. While this may be suitable in a PC environment, it is not at all useful when mobility is an issue, that is, when using a phone-based interface.

Voice browsers represent the class of products which use a totally voice driven interface; the prototype developed for this project is a member of this class. Voice recognition is used for user input, and speech synthesis and audio is used for output to the user. Commercial products such as SpeechHTML by Vocalis (<http://www.speechtml.com/>) and Conversa Web by Speech Technology (<http://www.speechtechnology.com/>) have been developed and deployed quickly to try and corner the rapidly expanding voice browser market, and as such, their interface and functionality are lacking.

A drawback of the currently voice browsers is that they do not support the ability to browse *any* web page. Rather, only pages formatted to strict compliance guidelines, and within a controlled domain can be viewed. This means that authors who want their pages viewed using voice-browsing technology often have to pay a subscription fee to get their pages published within this domain.

Another related research is the investigation of the general principles involved in the creation of a speech-based interface. The AHA system is a typical example. It is based on the principle that HTML files explicitly contain both the textual and structural content of a document, and that these two types of content are both essential for understanding the document (Frankie, 1997). AHA provides a framework for discussing audio marking techniques and how they can relate to different HTML structures to provide an intuitive audio interface to HTML. There are a few principles drawn from the experiments. The first one is to use different speaking voices to mark different HTML structures. This can be termed as Structure Differentiation. The other principle is related to recognisability, i.e. it is valuable for the selection of sounds to be used in an audio HTML interface. For example, to facilitate the target audiences to map a sound onto a marked document structure. Also, when we use a sound with an obvious metaphor (for example, the sound of a camera shutter to mark an image), the user can apply their recognition of the sound to interpret what is

being marked. A further principle is the avoidance of distraction. The provision of too much structural information can cause the user to be distracted from the textual content of the document.

3 Design of the System

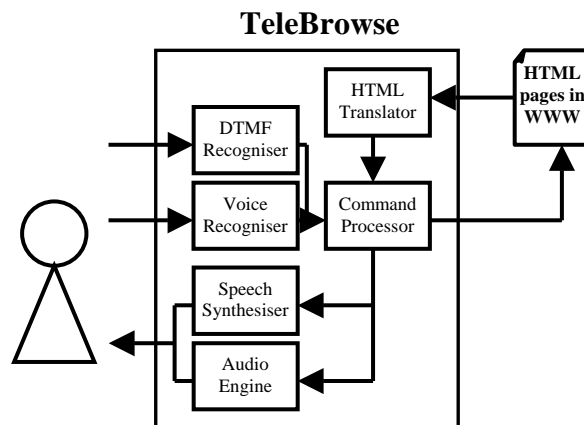
This section aims to give an overview of the design of a phone browser, TeleBrowse. The prototype is assumed to have a phone-based physical medium. The architecture is first introduced, which is followed by the functions provided by the prototype.

3.1 System Architecture

The system consists of three modules: voice driven interface, HTML translator and a command processor (Figure 1).

Figure 1: System Architecture of TeleBrowse

The voice driven interface of the prototype includes a voice recogniser, a DTMF recogniser, a speech synthesiser and an audio engine. The



physical medium for communication is intended to be a telephone, where the user can be remote to the system's location, but it can also be simulated with a microphone/speaker combination connected directly to a PC running the system. A voice recogniser and a DTMF (Dual Tone Multi-Frequency) recogniser are available for input to the system, while the speech/audio synthesis is suitable for output to the user. Hence, a command issued to the system can be in one of two forms: either spoken commands by the user, or DTMF tones punched in on a phone's keypad. DTMF assigns a specific frequency, or tone, to each key on a touch-tone telephone so that it can easily be identified by a microprocessor. The purpose of DTMF in the architecture is justified by the cost effectiveness to supplement the voice functions. Although voice recognition technology is

capable to handle most of the translation task, it is however an expensive one. This will be used for issuing commands that are not complex enough to warrant the translation to a voice driven format.

The **Voice Driven Interface** essentially accepts spoken words as input. The input signals are then compared against with a set of pre-defined commands. If there is an appropriate match, the corresponding command is output to the Command Processor. The engine also handles auxiliary functions (in conjunction with the speech synthesis engine) such as the confirmation of commands where appropriate, speed control and the URL dictation system.

The Text-to-Speech/audio engine is responsible for producing the only output from the system back to the user. The output is in the form of spoken (synthesised) text, or sounds for the auditory icons. The input for these two engines comes from the Command Processor. The input can either be a text stream consisting of the actual information to be read out as the content of the page (processed by the TTS engine), or an instruction to play a sound as an auditory icon to mark a document structure (processed by the audio engine).

The other major component is an **HTML Translator**. When a user requests an HTML document, the contents of the document must first be parsed and translated to a form which is suitable for use in the audio realm. This includes the removal of unwanted tags and information, and the re-tagging of structures for compliance and subsequent use with the audio output engine of the interface. The translator also summarises information about the document such as the title and positions of various structures for use with the document summary feature.

A **Command Processor** sits between the HTML translator and the interface. The command processor is responsible for acting on the voice / DTMF commands issued by a user. The Processor retrieves HTML documents from the WWW and feeds them to the HTML translation algorithm. It also controls the navigation between web pages, and the functionality associated with this navigation (bookmarks, history list, etc.). This component also processes all the other system and housekeeping commands associated with the program. A stream of marked text to the speech synthesis / audio engine is output. The stream consists of a combination of actual textual information, and tags to mark where audio cues should be played.

3.2 Functionality

All communication from the user to the system is made by issuing voice commands or using DTMF tones. Such commands are arranged into objects known as menus. Depending upon the functionality requirement/availability, different menus are available at different points in the program's execution.

A grammar set is defined to recognise the speech commands. Some of these rules are for administrative control. To name a few administrative controls:

- `<exit|quit[program|application|TeleBrowse]>`
- `speak <faster|slower>`
- Where am I?
- What is my homepage?

The other rules are used to control navigation. It is anticipated that they are the most frequently used commands. The navigation is supported in various ways: within the same page (intra-page navigation), browsing a new web page (inter-page navigation), bookmarks, history list, document structure or to follow a hyperlink in the web page. The following grammars display the nature of these rules:

- Start browsing by `<location|bookmark|homepage>`
- Maintain bookmarks
- Start reading `[all|again]`
- Load `<location|bookmark|homepage>`
- Go to the history list
- Jump `<forwards|backwards x <structure>>`
- `<Next|Previous <structure>>`

A `<structure>` is one of paragraph, link, anchor, level 1/2/3 heading, list, page or table, and `x` represents a positive integer value. All three versions of this command represent one action - moving between structures within a document.

Another way to navigate to a specific target page is via dictation. Dictation is invoked whenever a 'browse by location' type command is requested, and it is responsible for fetching a URL address from the user. Users dictate to the system by saying words representing a single letter to improve recognition accuracy. A good example is the military code - 'alpha' for 'a', 'bravo' for 'b', 'charlie' for 'c', etc. The grammar recognises this military code, and also common animals, such as 'frog' for 'f'. Macros and shortcuts are also used to simplify the dictation process. The 'http://' at the beginning of every URL is automatically added, and the system recognises phrases like 'World Wide Web', 'company', 'aussie' among many more to

represent 'www.', '.com', and '.au' respectively. The dictation menu also allows for corrections to be made, a review of what has been dictated so far and an ability to restart the dictation session.

Output from the system is either synthesised text or sounds (as auditory icons). The synthesised text can represent either actual information being read from a web page, or feedback about the system's operation to the user. When a page is to be read out (post translation), the page is broken up one piece at a time and analysed. Two situations can occur: if the piece is a tag with an associated auditory icon, this icon is played out, or, if the piece is simply text, it is synthesised into voice. Typical application of auditory icons include the creaking opening door (creaking) to represent internal link (link to an anchor within the same web page), or a doorbell to represent an email address, or the clicking sound of a camera shutter to relate to an image.

When certain tags are encountered (end of paragraph, end of list, end of table row, etc.), speaking ceases and the user is returned to the menu they were last at. Alternatively, if a user wishes to interrupt the speaking prior to the next break point, the interrupt key '*' can be used.

4 Experiments and Results

The TeleBrowse prototype was developed under the Visual Basic 6 environment. Three ActiveX controls were used in conjunction with the VB project. The Microsoft Telephony Control (see <http://www.microsoft.com/speech>) has the voice recognition engine, speech synthesis engine, and audio output engine. The Microsoft Internet Transfer Control is able to retrieve HTML documents from the Web using the HTTP protocol, while the HTML Zap Control (Newcomb, 1997) provides a simple interface for analysing HTML documents.

The primary goal of the evaluation of the TeleBrowse prototype is to determine the usability and acceptance of the application as a voice-driven web-browsing tool. Measurement is done via the operational efficiency and the quality of the transformed data. The operational efficiency is represented by the physical interface, speech recognition & synthesis capabilities, document and page navigation, prompts and feedback from the prototype. The quality aspect is evaluated through the availability, integrity and usefulness of the data after the translation, and also the support of various structures in the original HTML page, e.g. headings, links, tables.

The experiment was carried out on two different groups of users who had similar characteristics, i.e. they were all above 18 years old, they had prior experience on using current web browsing products and had limited knowledge of the WWW and the related technology. The major difference between these two groups was subjects in the first group (G1) were normally sighted, while the second group (G2) were visually impaired, or to be precise, they suffered from complete blindness. In other words, subjects in G1 were familiar with typical visually oriented web browsers such as Microsoft Internet Explorer. In this experiment, there were five people in G1 and four subjects in G2.

Evaluation sessions were run on a one-to-one basis. There was also a general discussion involving the group at the end of the individual experiments. No interaction occurred between subjects from differing groups.

Each subject was briefed about the operation of the prototype, and the methods for interacting with it. Along with the briefing, each user was given a sheet on which was printed a vocabulary listing of what phrases the program would respond to, and at what points in the program's execution such phrases could be used. In the case of the subjects in G2, the sheet was printed using a Braille printing device. The sheet also contained a set of tasks the subjects had to complete using the program. Some of the tasks that subjects were required to complete included: starting the application, checking to see what the current homepage was set to, commencing reading of the web document once loaded, jumping to another location by dictating a URL address directly into the system etc. Accompanying each task was a description of the behaviour the system would demonstrate during the task's execution, and what phrases to use in interaction with the system to complete those tasks. All tasks were first completed using the software running in emulation mode on the laptop computer. The speech recognition engine was not trained to adapt to any specific person. After completing this and gaining a degree of experience in using the system, subjects were then given an opportunity to use the system as they chose over the phone, completing the effect of a phone-based web-browsing tool.

Subjects were also asked to view the same web pages that had just been 'viewed' using the prototype with a browser they would typically use. In the case of G1, this was either Microsoft Internet Explorer or Netscape Navigator. In the case of G2, this was again Microsoft Internet Explorer, but this

time with the edition of the JAWS screen reading program.

After using the prototype for a sufficient amount of time (in most cases this was a period of about twenty minutes to half an hour), each subject was asked to complete a questionnaire to record their experiences with the prototype. The questionnaire was arranged into two parts: the measurement of efficiency and integrity. Some of these questions asked the participants to give numerical scores in a scale of 1 (very poor) to 7 (very good). Some of the other questions were free format where the subjects could provide their own comments. The comparison of the relative frequencies per grade between the two groups is shown in Figure 2. This gave us an initial and very broad insight into how subjects from each group responded to using the prototype in the experiments. By graphing both groups' results on the same axes, we could also see a comparison of acceptance between the two groups.

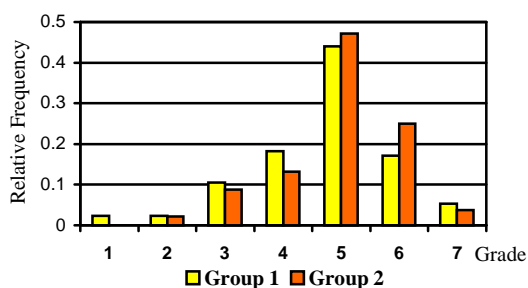


Figure 2: Histogram of Relative Frequencies per Grade

In addition to the rated response questions, The evaluation questionnaire contained a further thirteen questions of free-form response in nature. These free-form questions were designed to draw out any comments, problems, criticism, or general feedback from the test subjects. There were, on average, two such questions per section of the evaluation criteria.

Voice Recognition. It was one of the more poorly rated criteria. While both groups considered it a necessary technology for the idea of a phone browser, subjects suffered from its shortcomings, and it did result in a loss of efficiency for most users. Subjects from G2 responded more favourably than those from G1. Of particular concern was the dictation of URL addresses. This was noted as a shortcoming in the interface by every user from both groups. The idea of having to spell out URL addresses one letter at a time (and wait for confirmation of each letter) was not well received. The idea of using shortcuts like 'company' to spell

out '.com' was considered a strong improvement, thus this technique must be further explored.

Speech Synthesis. There was little or no problem with this sub-system. Subjects found the voice easy to understand and of suitable volume and pitch. The major contrast between the two groups was the usage of the speed control feature. Subjects from G1 saw no reason to adjust the speed of the synthesised voice. They were content with the default normally paced speaking voice. However, subjects from G2 tended to change the speed to a much higher rate before doing anything else.

Navigation: The overall ability of intra-page and inter-page navigation using the system was rated favourably by both groups. The use of auditory icons to mark HTML structures was viewed by the G2 subjects as being superior to any similar screen reader marking scheme. Subjects from G1 also appreciated the ability of the voice icon to quickly and to simply mark structures from a document, in a way that was natural and easy to remember. The idea of metaphorically matching the meaning of sounds with the structure they were representing was well liked and accepted. There was little problem with remembering the mapping of sound to structure, especially after using the system for an extended amount of time. A criticism with the auditory icons was that they appeared too frequently, and could be seen as breaking up the flow of text unnecessarily.

A comment made by many subjects was that the prototype offered similar and familiar functionality to that of browsers they have previously used. Thus, features such as bookmarks, the history or 'go' list and the ability to save a 'home' page were all well received. The ability to follow links contained in documents was well liked. Using different auditory icons for the different types of links allowed subjects to know in advance whether the link would be to a target within the same document, or an external link to another document. This too was well liked. Again, the problem with dictating URL addresses was brought up.

Online Help: This section of the criteria did not rate well, due to the lack of help associated with commands and prompts used in the system. It was thought by all users that more detailed help (context based) explaining the meaning and usage of commands, should be available at any point during the system's operation, as opposed to the simple listing of commands currently available.

Certainly the need to refer to other supporting documentation for more detailed information should

be avoided, as access to this information would not be available in environments where a phone browser might be used.

The tutorial available from the system's main menu was well accepted in terms of its content, but perhaps a similarly detailed tutorial should be available at every menu in the system, customised for the relevant set of commands.

Information & Structure: There was a marked difference between the two evaluation groups in these two aspects. G2 was more willing to accept the level of integrity of information presented to them by the prototype than G1. Comments were made from users in G1 concerning the lack of ability to quickly visualise an entire page at a "glance". They reported frustration when they were forced to listen to the content in sequential fashion.

Overall Impression: The subjects from both evaluation groups accepted the prototype as a viable method of browsing the web in the audio realm by phone. The efficiency of the product was quite highly regarded by most subjects. The system interface fared very well. The only major problem was the dictation of URL addresses to the system.

4 Discussion

Referring to the histogram in Figure 2, subjects from G2 were more prepared to accept the prototype than people from G1. People in G2 had a higher tendency to give higher grades to the prototype. This might be due to the reasons that the subjects from G2 had limited options in getting quality non-visual browser aides. This group of people generally knew the limitations and performance of existing speech-based tools. When they were introduced to this prototype and found that its performance was better than the tools they have seen, they would give better grades to the prototype. The new tool performed beyond their expectation and has extended the capabilities of the available browser aides. However, the subjects from G1 found themselves more restricted than their usual browsing experience. One of their frequently used information receiving channels came from the visual field, which was disabled in this experiment. They also did not have prior experience in using non-visual browser aides. This constrained them from understanding the current limitations and the state-of-art of these non-visual tools. The people in G1 could only compare the encountering of this prototype with the web surfing experience in visual domain. This had implication to the people from G1 in scoring of the

voice browser. We believe the lower scores from G1 were not due to the functionality of the prototype, but the decreasing channels in receiving information once they were so used to.

The different tolerance level to voice recognition could be due to the fact the visually impaired users already had exposure to recognition technology, their expectations would not have been so high. Normally sighted subjects (of whom only two had used voice recognition software to any extent) were more easily disappointed. Due to the same reason that the visually impaired subjects were very used to listening to synthesised voices, they had the foreknowledge that the efficiency of information delivery could be achieved by speeding up the rate at which text was presented to them. We hypothesise that normally sighted users will become more accustomed to the synthesised voice, and likewise, will increase speed to improve efficiency. In other words, when the experience and familiarity with a speech interface accumulate, the tolerance level or understanding of a system's limitation will increase. The availability of user-controllable features is necessary to compensate some of the deficiency of such a speech-based interface.

The use of sound icon/metaphor is both beneficial as well as distracting. The sound of a structure-denoting icon should be more smoothly integrated into a text flow, removing any breaks and pauses. Further experiments have to be carried out to balance off the delivery of author's intent and distraction.

Although a combination of bookmarks, the history list, and the 'home' page alleviate the need to continually dictate addresses, it is certainly still necessary to dictate to the system in many cases. Also, it was suggested that a better interface be designed to handle management of bookmarks. It can be seen that if the list of bookmarks was to grow to a large number, it would become tedious and difficult to find a specific one.

In terms of the integrity of information, G2 was more willing to accept the level presented to them. The visually impaired subjects could only compare the prototype to other non-visual browsers or screen reading products they had used. The system compared favourably to these, with the information and structures associated with a page being presented in a similar or better form. Members from G1 are used to viewing pages in the visual realm, hence, any shortcomings in the translation from visual to audio form are more obvious to them. Both groups felt that support for structures was generally

good, with the exception of tables. It was suggested a more advanced method of navigating around table structures be introduced. Another suggestion by many users was the need to be able to use fill-out forms, and also support for frames.

Subjects in G1 complained the lack of ability to quickly visualise an entire page at a "glance". They reported frustration when they were forced to listen to the content in sequential fashion. However, this problem can only be solved through adaptation by the user. Subjects from G2, on the other hand, have always been forced to listen to all GUIs sequentially through voice where the conceptual image of a page is formed as it is read out. The different responses highlighted the different visualisation process adopted by the two groups, this is worthy to further investigate to see what are the common mechanisms underneath these two seemingly different visualisation processes, and how a better approach using voice can be built.

The dictation is a major problem. At this stage, it seems to relate to URL addresses alone. The problem is far fetching than this. If a system is to support form-filling (where the information can be ill-related search terms, or name and address of a person), the dictation facility may be used in these different places. In the command recognition, there is a grammar to govern what information can be entered. In a text understanding tool, the context of the document helps resolve the ambiguity of a word. The troubles with the recognition of URL addresses or search terms are that they neither have syntax nor a context. This is an important issue to be addressed in order to deliver a user-friendly speech-based interface.

5. Conclusions

The TeleBrowse prototype developed for this project proved its success in the evaluation as a telephone based web-browsing tool. Problems were highlighted in the user study. Part of these problems are related to the features currently available in the prototype, for example, the navigation of a table, the interruption of an output using voice rather than "*" key on the phone pad, the missing function of filling in forms on a web page etc. At the same time, there are some deeper issues which require more research

include the dictation function (how and where contextual information can be acquired to improve the accuracy of a dictation), visualisation process (what kind of model is made available which provides a feeling of information coming in parallel). Testing of the prototype in real life environments rather than in a standard laboratory condition is also an important direction of future work, e.g. walking in the street, driving a car. This may uncover further issues that relate to situated interaction. Research into domain-specific speech interaction model may also improve the accuracy and effectiveness of the system. It is the intention of this project to create an environment where web surfing with voice is possible and surfing experience is a pleasant one. The knowledge accumulated in the production of the first generation prototype, TeleBrowse, has given us a lot of insights and lay the groundwork that helps us move on to develop the second generation prototype.

References

- Asakawa, C. and Itoh, T. (1998). User Interface of a Home Page Reader. Proc. of the Third International ACM Conference on Assistive Technologies, pp.149-156, Marina del Rey, CA USA.
- Frankie, J. (1997). AHA: Audio HTML Access. In M. R. Genesereth and A. Patterson (ed.), The Sixth International World Wide Web Conference, pp. 129-139, Santa Clara, California.
- Newcomb, M. (1997). HtmlZap ATL ActiveX Control. See <http://www.miken.com/htmlzap/>.
- Raman, T. V. (1996). Emacspeak - Direct Speech Access. ASSETS '96 : The Second Annual ACM Conference on Assistive Technologies, pp. 32-36, New York, ACM SIGCAPH.
- W3C (2000). Voice eXtensible Markup Language (VoiceXML) version 1.0. See <http://www.w3.org/TR/2000/NOTE-voicexml-20000505>.
- Zajicek, M. and Powell, C. (1997). Accessing the World Wide Web by Telephone. See <http://www.brookes.ac.uk/schools/cms/research/speech/publications/43hft97.htm>, Oxford Brookes University.