Hands-Free Documentation

Karen Ward
Department of Computer Science
The University of Texas at El Paso
El Paso, TX 79968-0518
+1 915-747-6957
kward@cs.utep.edu

David G. Novick
Department of Computer Science
The University of Texas at El Paso
El Paso, TX 79968-0518
+1 915-747-6952
novick@cs.utep.edu

ABSTRACT
In this paper, we introduce an analysis of the requirements and design choices for hands-free documentation. Hands-busy tasks such as cooking or car repair may require substantial interruption of the task: moving the pan off the burner and wiping hands, or crawling out from underneath the car. We review the need for hands-free documentation and explore the role of task in the use of documentation. Our central analysis examines the roles and characteristics of input and output modalities of hands-free documentation. In particular, we review the use of speech as an input modality, and then visual means and speech as possible output modalities. Finally, we discuss the implications of our analysis for the design of hands-free documentation and suggest future work. The design implications include issues of navigating through the documentation, determining the user's task and task-step, establishing mutual understanding of the state of the task, and determining when to start conveying information to the user.

Categories and Subject Descriptors
H.5.2 [**Information Interfaces and Presentation**]: User Interfaces – *Training, help, and documentation, theory and methods*

General Terms
Documentation, Human Factors

Keywords
Hands-free, modality, speech, spoken-language interface

1. INTRODUCTION
Consulting documentation is, at best, a distraction to completing the task at hand. For some tasks and in some environments, however, it may be inconvenient or even dangerous to stop and grab a manual. When operating a vehicle, for example, both hands and eyes are occupied. Even when safety is not an issue, having to one's use hands to access documentation may interfere substantially with the task itself. Hands-busy tasks such as cooking or car repair may require substantial interruption of the task: moving the pan off the burner and wiping hands, or crawling out from underneath the car. In some cases, the interruption can even threaten the success of the task, as any cook who has been interrupted in the middle of making a Hollandaise sauce can attest.

In this paper we explore ways to support documentation access for tasks such as these. We address the general issue of how hands-free documentation differs from more traditional documentation and the specific issues involved in using alternative modalities to support hands-free documentation. Section 2 examines the nature and constraints of hands-free documentation. Section 3 focuses on the use of speech as an input modality. Sections 4 and 5 discuss the presentation of documentation, first using visual display and then through spoken display. Section 6 discusses the implications of our analysis for the design of hands-free documentation. Section 7 presents our conclusions and future work.

2. NATURE OF THE PROBLEM
Some research and development has been reported in creating documentation with hands-free techniques. A notable success story for hands-free interaction has been the use of speech recognition for capturing medical information (e.g., [17]). However, this involves using speech as input and does not address the issue of hands-free access to documentation. Where hands-free access to information has been studied, it has usually been in the context of helping disabled users (e.g., [10]).

Access to documentation differs from access to information more generally. When accessing information, the access itself is the immediate task. For example, a user might seek a particular piece of information—the date of the construction of the Panama Canal, for example—and when that information is located then the immediate task is satisfied. Documentation serves a different role, especially in hands-busy environments. Users are likely to be in the midst of a physical task, such as preparing a soufflé or repairing a jet engine, and their need for documentation will be closely tied to the structure of the ongoing task. Spoken-language interfaces will be important in many if not all of these applications.

2.1 The Role of Task
Task-oriented documentation is usually written sequentially and assumes that the user is at the beginning of a task. The user may not be at the beginning of the task when using that

SIGDOC'03, October 12–15, 2003, San Francisco, California, USA.

documentation, of course, but the combination of a sequential organization and the direct manipulation of the material allows the user to locate the correct step in the task quickly. In contrast, hands-free access to documentation will necessarily involve entering into the documentation, often in mid-task, without the benefit of direct manipulation.

Creating effective and usable hands-free access to documentation will pose multiple challenges:

- What modalities should be employed to request and deliver documentation?
- How can the user navigate through the documentation?
- How can the system determine the task, and the step within the task, for which the user needs information?
- How can the system and the user ground and establish the mutuality of their understanding of the state of the task?
- How can the system determine where in the system's task model to start conveying information to the user?

2.2 Design Constraints

Three factors constrain the design of a hands-free documentation delivery system: available input modalities, available output modalities, and the level of integration between the documentation delivery system and the task itself.

We assume that hands-free and possibly eyes-free access implies that the voice is the most likely—and usable—way for the user to communicate with the documentation system. Other possibilities are slim. It is true that a foot-operated mouse has been proposed for desk-workers [14] and a limited form of eye-tracking has been made to work for users unable to manipulate a traditional mouse, but these are awkward at best and would be problematic in an environment where the user needed to be able to move around. Thus, speech is the most suitable choice as an input modality.

For the output modality, that is, the display of the documentation, there are more possibilities. A visual display may be possible, either through a traditional screen located where the user can read it without interrupting the task, or—more likely—through a head-mounted display. If neither facility is available, documentation may have to be presented aurally.

The level of integration between the documentation system and the user task may vary widely. In the case of a documentation system of an non-automated physical task such as cooking or auto repair, the documentation will likely be stand-alone. This implies that the user will have to negotiate with the documentation system to establish precisely what task is in progress and at what point in the task the user requires help. At the other extreme, the user may be engaged in a task that has significant automated support, such as medical information entry. In this case, the documentation system may be able to infer what help is needed based on the state of the task.

Each of these options presents challenges that lead to different design considerations. In the remainder of this paper we give an overview of the technology available to support each option and discuss the design implications for designing usable hands-free documentation access.

3. SPEECH AS INPUT MODALITY

The use of speech to access documentation poses several challenges. Speech input is far more difficult to interpret than mouse or keyboard input. With mouse input, we know unambiguously which point on the screen was clicked. With keyboard input, variability in typing speed does not make it any more difficult to know which keys were pressed, nor does a noisy room interfere with understanding what was typed. With speech, however, ambiguity and variability can cause errors at all stages in processing the speech input. In this section we give an overview of several common sources of error in understanding speech input.

Speech is highly variable. At the acoustic level, the differences between the way two individuals say the same word may far exceed the differences between the way that one person says two different words. Sources of variation due to speaker characteristics include the vocal tract length and shape, dialect, and even the general health of the speaker. Consider, for example, the difference in sound between a 6-year-old girl from Boston saying the word "day" and a 45-year-old businessman from Canberra saying the same word. Humans, whose performance as speech recognizers far exceeds what we are capable of building, usually adapt to these variations within seconds upon hearing a few words from a new speaker. Computers do not do so either as well or as quickly.

A string of sounds often is phonetically ambiguous as well. Even humans may have difficulty determining which words were intended or even where words begin and end. Jurafsky and Martin [12] mention as an example the Jimi Hendrix lyric "'Scuse me while I kiss the sky," which is often misunderstood as "'Scuse me while I kiss this guy." The word error rate on speaker-independent, larger-vocabulary fluent-speech tasks is perhaps 20-40 percent [12], which implies that nearly every sentence-length utterance is likely to have at least one misrecognized word. Jurafsky and Martin give this example ([12], page 271).

> **Speaker said:** I um the phone is I left the portable phone upstairs last night so the battery ran out
>
> **System recognized:** I got it to the fullest I love to portable form of stores last night so the battery ran out

In a spoken language system, the speech recognizer output is typically handed to a language understanding component to convert the string of words, possibly including errors, into a semantic representation [13]. At this stage, some of the errors are corrected or dropped; the language understanding component may ignore words that do not appear to contribute to the meaning of a sentence. For example, even if the command "Switch to weather" were misrecognized as "Please weather," the correct action will still be taken (example from Yankelovich et al. [21]).

Noise poses additional problems for speech recognition. Background noise may confuse or drown out the speech signal. Although humans are remarkably proficient at isolating and following a single voice through noise, even when the background noise includes other voices, speech recognition accuracy declines sharply in even moderate noise. For example, a voice input system designed for use by medics achieved 90 percent recognition accuracy in a quiet environment but saw its accuracy drop to the 70-to-80 percent range under field conditions [7].

More troublesome than the noise itself, however, is the Lombard reflex: people speaking in noisy conditions reflexively modify their speech to make it more understandable. The modifications go beyond a simple increase in the volume: vowels are lengthened, consonants are shortened, and many of the signal characteristics that speech recognizers use are altered for nearly every phoneme [11]. Worse, the changes vary both by speaker [11] and by the amount and type of noise [18], making it particularly difficult for speech recognizers to compensate for the effect.

When a recognition error occurs, users often attempt to speak more distinctly on the assumption that doing so will improve the speech recognizer's performance. Unfortunately, this hyperarticulation makes matters worse [14]. Speech recognizers are trained on the relaxed pronunciations normally used in fluent speech, e.g., "fordy" for the number forty. When a user presents hyperarticulate speech, the speech recognizer is likely to misrecognize the input ("Did you say 'fourteen'?"). The unsuspecting user responds by trying to pronounce the words even more carefully, and the interaction rapidly degenerates into a spiral of errors [14].

3.1 Background

Current speech-input systems achieve their success by sharply limiting some aspect of either speaker variability or phonetic ambiguity. Dictation systems, such as IBM's ViaVoice [9], are capable of recognizing a very large number of words, although they do not attempt to interpret what those words mean. They require the user to provide several lengthy speech samples, so that the system can tune its recognition algorithms to the user's particular dialect and vocal tract characteristics. Dictation systems also may build up frequency models of the task vocabulary (e.g., business letters) and of the user's individual vocabulary. When more than one word matches a particular speech signal, the system then can select the word more commonly employed by this user in this context.

When there is no opportunity to tune a system to a particular user, as with the systems for accessing airline flight information by telephone, the vocabulary is limited instead. The system may use a series of carefully-worded prompts to request information from the user [4], e.g., "If you know your flight number, please say it now." In this way, the system attempts to limit the user's likely input to a small set of words at each point in the dialogue. If these words have been selected to sound as little alike as possible, the recognizer can achieve acceptable accuracy despite a wide variation in speaker characteristics. For example, a system designed to be used by both the Boston child and the Canberra businessman above would avoid a dialogue design that would require it to distinguish between the words "day" and "die" at any one point.

Whether taking the approach of customizing to the user or limiting the vocabulary, speech recognition errors certainly will occur. For this reason, systems generally provide immediate feedback to show the user what words were recognized and to allow the user to correct errors. Dictation systems display the words as they are dictated; that is their function. Task-oriented interfaces generally provide this feedback either by performing the requested action or by including the recognized information in the next prompt, e.g.,

System: What is your flight number?

User: Sixty five oh two.

System: Flight 6502 arriving in Phoenix at 12:28 p.m. today is on time.

Explicitly asking the user to confirm each input is slow and quickly becomes annoying.

System: What is your flight number?

User: Sixty five oh two.

System: Did you say 6502? Please say yes or no.

This technique generally is reserved for situations in which the system has misrecognized several user inputs and is trying to correct the errors to get the task back on track.

3.2 Design Considerations

Speech recognition accuracy will be central to the success of a hands-free documentation system. The following factors should be considered:

- Will the system be single-user, or must it accommodate a wide variety of walk-up users? If the system must accommodate a variety of users, then prefer a limited-vocabulary-prompt approach.
- For single-user systems, will the overhead of customizing the recognizer be worthwhile? The user may be reluctant to invest the time required to customize a recognizer solely for the purpose of accessing documentation. Unless the user has reason to use speech input regularly for other purposes, or unless the user expects to use the documentation system very heavily, prefer a limited-vocabulary-prompt approach.
- Dialogue design for spoken-language interfaces is at this time more of an art than a science. In general, though, attempt to design system prompts to ask specific questions and to suggest the words that the system is prepared to recognize.
- Must the system function in a noisy environment? If so, consider the use of noise-cancelling microphones.
- Is the noise fairly steady? If so, consider having the user customize the recognizer in the noisy environment so that recognizer will have a chance to learn the Lombard-reflex-affected pronunciations for this user and this noise level.
- Provide immediate feedback to the user as to what was recognized. This can be done by incorporating the recognized information in the next prompt or by taking the action requested by the user.
- If possible, explicitly train the user to avoid hyperarticulation when errors occur. Consider including a prompt during error recovery dialogues to remind the user to speak normally and in a relaxed manner.
- When repeated recognition errors occur, consider improving the chances of correct recognition by moving to a more guided interaction in which the vocabulary is

constrained as much as possible and the user is explicitly informed as to what vocabulary the system is prepared to recognize. If necessary, fall back to yes/no questions.

4. VISUAL DISPLAY AS OUTPUT MODALITY

For some tasks, it may be possible for the user to view the documentation visually. The nature of the task may make it reasonable to assume that an appropriately placed computer display will be available, for example. Another possibility is the use of head-mounted displays. Augmented reality applications are beginning to appear in certain manufacturing applications [5], including a successful application to assist in assembling cable harnesses for the Eurofighter [6]. Of particular interest for this paper is the successful use of head-mounted displays for reviewing assembly or engineering data while inspecting components and assembly processes [20]. A possible drawback: users report that current-generation head-mounted displays and wearable computers are not comfortable for lengthy use [6], [15].

4.1 Background

Visual display of documentation is appealing. The documentation designer may be able to take advantage of existing documentation for the application in question, and can at least take advantage of existing graphical user interface (GUI) documentation guidelines. Also, the presence of visual feedback makes it easier to give immediate feedback on speech input; the recognized command can be displayed directly.

In converting an existing GUI-based documentation display to a speech-based system, it may be tempting to begin by simply implementing the GUI commands as speech commands. A relatively small vocabulary that is grounded in the interface itself may suffice: commands that correspond to labels on graphical items may seem a plausible approach. The vocabulary to be recognized then can be derived automatically from the page itself, on the fly if need be.

This approach has several shortcomings, however. The fact that the user must manipulate and navigate the documentation display by voice imposes some additional considerations [21]. In a GUI, for example, one discovers what commands are available by moving the mouse to a menu and pressing a button. How should that action be expressed verbally? It is likely that the user will have to learn some additional command conventions for manipulating the graphical elements of the display, e.g., “edit menu” to display the edit menu.

One of the advantages of direct manipulation over speech is the ability to point to something in an unambiguous fashion. For example, a screen layout with several identically-named links or buttons is not problematic if the context is established by the surrounding text. If the user must refer to them verbally, however, the reference may be ambiguous. In an early implementation of a speech-controlled web browser, House [8] allowed for users cycling through the available links on a page using commands such as “next.” This works, but clearly is awkward and slow compared to pointing. Yankelovich found that users tend use relational and positional terms to specify which is wanted [21], suggesting that references such as “the first one” or “the second ‘here’ button” may be expected.

Finally, one of the advantages of speech over direct manipulation is the ability to refer to things not immediately visible [2], and an implementation that merely speechifies the GUI interface may be failing to take advantage of that power. For example, requiring the user to verbally "pull down" the edit menu to get to the submenu is slow—and slower with speech than with a mouse, due to speech recognition delays. If the user is sufficiently familiar with the interface to know what function is wanted, this could be especially frustrating. This issue is discussed in greater depth in Section 5.

4.2 Design Considerations

When considering visual display of hands-free documentation, the following factors should be considered:

- Is it feasible to position a conventional display where the user an read it easily without interrupting the task?
- Is it feasible to use a head-mounted display?
- Does the task itself suggest the use of a head-mounted display? Note that users engaged in a hands-busy task would not be able to don a head-mounted display merely to access documentation, so this approach is most likely for those tasks that mandate the use of the technology otherwise.
- Do the characteristics of the application and of the user suggest that a head-mounted display and wearable computer will be tolerable? Factors to consider here include the physical demands of the task: would wearing this equipment interfere with the task?
- Is the user willing to wear the head-mounted display for long periods of time, or is it deemed annoying?
- Ensure that labels on referable graphical items, especially links, are distinct. Often this can be accomplished by enlarging the scope of the link to include enough context to make it unique.
- Include the use of relational references such as "next" and “last” in the speech recognizer grammar.
- Consider including hotkey-style shortcuts for frequently-used references or functions, especially if the users are likely to be using the documentation frequently or if the functionality is similar to other interfaces that the users would know well.

5. SPEECH AS OUTPUT MODALITY

When no visual display is possible, the documentation designer is left with speech as the medium for conveying the documentation to the user. Speech as an output modality poses many profound implications both for the presentation of information and for the user's communication with the system.

5.1 Background

When using speech as the sole output modality, we find ourselves relinquishing some of the advantages of speech as an input

modality. Speech is a serial, temporally-constrained medium, that is, information must be presented one item at a time and in order. The listening user cannot skip around on the page looking for the most relevant item, as can be done with a visual presentation. An item once presented is gone. These characteristics will make navigation within the documentation difficult. Furthermore, human short-term memory limits the amount of information that can be presented at one time. When presenting the user with command options, for example, the list of commands must be short enough to allow the user to remember them and select the most appropriate one.

The lack of visual information means that we also lose the opportunity to establish understanding in the visual channel. Presentation of graphical information is difficult at best, impossible in many cases. Also, the user may have difficulty determining even what state the system is in. With no visual feedback, for example, the user may not know whether a silence from the system means that the system is working or that the system didn't receive the last command [1], [12].

With the loss of visual display comes a loss of common vocabulary and of a common ground for referring to actions and objects of interest. When using a graphical user interface (GUI), a user can see exactly what options are available. With spoken input, however, the user may be uncertain as to what commands will be accepted [17]. Furthermore, the user may not know even what actions are possible [19]. Thus, when a user attempts to perform some action and fails, it may not be clear whether the failure occurred because the system cannot perform that function, because the user failed to use the correct term in requesting the action, or because the speech recognition system failed to correctly interpret the words. Having the system dictate a list of possible actions is likely to be painfully slow, and if the list is more than a few items long then the user will have difficulty remembering the options. Especially when the users are unfamiliar with the system or do not use it frequently, as may be expected with a documentation system, users may prefer a more system-directed style of interaction in which the system prompts the user for the information that it needs to locate the desired documentation [19].

Another effect of the loss of visual common ground is that the user vocabulary changes. Instead of being able to point to an item on the screen, the user must describe which item is intended. Yankelovich noted an increase in the use of relative terms such as "next Monday" over the use of absolute terms such as "Monday, the 10th." Even when users are very familiar with a graphical interface for a similar task, they may not remember or use the GUI terminology [21].

Yankelovich also found that other GUI conventions do not translate well into speech-only interfaces. For example, users often ignored confirmation requests, e.g., "Your message is being sent to Matt Marx. Okay?" or "Did you say to hang up?" [21]. They speculate that this may be because, in human speech, yes/no questions are rarely answered with a simple yes or no. In the case of confirmation messages such as these, people tend to respond with the next relevant action they wish to take [3].

The very need for a hands-free interface suggests that the user is likely to be interrupted during presentation of the documentation. With a visual display, it is relatively easy to attend to the distraction and then resume reading where one left off. With an audio presentation, however, the user may lose part of the presentation and may have difficulty backing up to the correct place to repeat it. Worse, the presentation itself may distract the user from attending to more important physical events, especially when users are managing tasks with high cognitive load. In reporting on the deployment of a speech-based medical information capture system, Holzman stressed the importance of making it easy for users to suspend and resume operation of the system [7]. He also noted that nonverbal audio feedback was preferred by very busy users (medics working on patients in the field) over lengthy verbal readbacks and error reports. Simple, distinct sounds such as pings to signal successfully-processed commands and clicks for failure were faster, less distracting and heard more easily in noise.

Finally, be aware that use of a speech-only interface results in an interaction that is far slower than that afforded by a visual display or even by human conversation. In addition to the delay inherent in the processing of the speech recognition and understanding systems, possibly with errors that must be corrected before the interaction can continue, the output of an appropriate response will be very slow by human standards. If it is possible to determine all possible system responses in advance, system responses can be constructed from human-voice recordings—but large recordings can require a noticable delay to load and begin playing. If some or all of the system outputs cannot be known at the time that the documentation system is designed, then responses may have to be generated on the fly using synthesized speech.

5.2 Design Considerations

Avoid the use of speech-only output if possible. Navigation will be substantially slower and more difficult than with a visual display, and information presentation will be far slower. Graphical information may be difficult or impossible to translate to an aural presentation.

If speech is the only output modality available, though, consider these factors:

- Provide feedback to the user that commands have been received, and do so quickly. A small number of distinct tones or clicks can provide fast, unobtrusive, robust signals as to the system's state.
- If users will not be consulting the documentation frequently enough to become expert users, prefer a more system-directed style of interaction in which the system prompts the user for each command.
- Provide feedback as to the user's current location in the document.
- Be aware of the possibility that the user may be distracted by other events. Include stop-continue-repeat commands to allow the user to quickly interrupt and then resume presentation of the documentation.
- Keep presentations as short as possible [21, 7]. A literal reading of written documentation is likely to be too wordy for effective verbal presentation. Elide words where the previous presentation has established context

[21]. Use hierarchical organizations to limit the number of options presented at any one time.

- Allow the user to control the pacing of interaction, particularly the speed of the documentation presentation. A fast-forward presentation speed can allow the user to locate the desired documentation section more quickly.
- Warn the user about the length of upcoming aural presentations. Arons, for example, used a short, high-pitched tone to signal a brief presentation and a longer, low-pitched tone for a long one [1].
- Confirmation and error dialogues may not translate well into speech presentations. Users are likely to fail to respond to questions such as "Is that OK?" If such dialogues are needed, prompt for an explicit reply, e.g., "Please say yes or no."
- Synthesized speech is harder to understand than recorded human voices, and the additional time needed to generate the response may be unacceptable. If all possible system outputs cannot be determined in advance, however, synthesized speech will be required.

6. DISCUSSION

In this section we discuss the implications of our analysis for the design of hands-free documentation. In particular, we address the issues of navigating through the documentation, determining the user's task and task-step, establishing mutual understanding of the state of the task, and determining when to start conveying information to the user.

6.1 Navigating Documentation

The practicality of using speech to navigate documentation depends in large part on the available output modalities. Where the system's output is aural only, the issue arises of helping the user understand where it is possible to navigate. Traditional documentation offers multiple means of navigation. Among these are reading the table of contents, reading the index, and leafing through the body of the documentation. Of these approaches, only the table of contents appears to be practical for speech input and output, and the table of contents would have to be structured and presented hierarchically to avoid outputting long lists of items via speech.

Because visual output can convey information more quickly than spoken output, it is the modality of choice for navigation through the documentation. Even if the input modality is spoken language, the visual display provides powerful advantages. For the table of contents and index, the visual display provides both a fast means of output and a source of target vocabulary items for a speech recognizer. This would build on the technique proposed by House [8] for spoken-language access to Web pages. Browsing or scanning, too, is made easier with visual output. The actions of leafing through a manual could be approximated through voice commands such as "next page" or "go 12 pages." The system could also provide a diagrammatic representation of the documentation's contents that could also be navigated by voice, using both directional commands and keyword matching. (In such cases, it would be a poor idea to have content topics with names like "next" or "up.")

Speech output is the low-bandwidth sibling of visual output. But in some cases where the user's vision is occupied with the underlying task, it may be the only modality available. In such cases, how can the interface support the user's navigation through the documentation? Part of the solution is to make the documentation hierarchical, as navigating an ordered tree is more efficient than traversing a list. Another part of the solution is to produce output templates that can help the user to understand the context quickly. For example, when the user is navigating a tree structure such as a table of contents, the system could establish context at each node with a quick recap of the current path through the tree ("html 4.01, tables, table captions"). Finally, the system could use non-speech audio output to represent navigation state. For example, different sounds could represent intermediate and leaf nodes, or descriptive or prescriptive sections.

For any non-trivial system the approach of navigating by a traditional index would not be practical for spoken-language output. We expect that users scan an index looking for terms that they see as relevant to their task; if they already knew relevant terms they would have navigated directly to that section in the first place. Consequently, the brute-force approach of presenting each index item would be far too time-consuming to be practical, especially considering that tasks requiring hands-free documentation are often time-critical. This suggests that speech-only hands-free documentation ought to have a feature that replaces the index with an interactive keyword help system. A speech-only substitute for leafing through the documentation likely would be difficult to produce. Because the high-level of output of printed documentation is not possible with speech, spoken-language output for leafing or browsing likely would be equivalent to navigating the table of contents.

6.2 Determining the User's Task

A speech-input system is no worse off than other systems in terms of determining the user's task, and the step within the task, for which the user needs information. Indeed, if the underlying task is also being carried out through a speech-recognition system, the system may have a more articulated task model because usually it is constrained by task element in order to improve recognition. In the general case, though, this remains an exceedingly thorny problem. The system will need substantial task knowledge and sophisticated reasoning tools to deduce task and task-step associated with the user's interaction with the documentation. Authors may be able to provide partial solutions by organizing the descriptive elements of documentation along the lines of relevant tasks.

With respect to output modalities, visual output may be more useful than speech output in helping the system determine the task and task-step for which the user needs information. Visual output has a continuous presence, and the documentation system can thus know the user's current focus with reasonable certainty. In contrast, as noted in Section 5, aural output does not have continuous presence. As a result, the system may literally be unable to tell what page the user is on. With visual output the system can also use dynamic methods for reference, such as highlighting headers and sections of text. The system could then ask the user to respond verbally to questions such as "Is this what

you're working on now?" The design goal here is to fit the interaction as appropriately as possible to the asymmetry between the system's visual output and the user's spoken output.

Determining the user's task and step-task for which the user needs information, is relatively difficult with speech output. The system could still ask users to confirm its understanding of their task state, but the spoken representation of the information probably will require an effective means for truncating or eliding the descriptions. Of course, this shortening of descriptions carries with it an increased risk of misunderstanding. In practical terms, there will be a need for careful usability testing to ensure that the output of task descriptions is meaningful for users.

6.3 Establishing Mutual Understanding

With speech input, on-line documentation and the user can ground and establish the mutuality of their understanding of the state of the task with relative ease as long as things are going smoothly. If the system has a hypothesis about the state of the task, it can output a representation of the state and ask the user to confirm it the state is correct. If the system's hypothesis isn't correct, though, repairing the misunderstanding may be non-trivial. One approach might be to output a limited number of the system's task-state hypotheses and ask the user to choose from these. Of course, grounding isn't always accomplished perfectly by humans, either, so there are natural limits as to what can be achieved in this regard. If a visual output modality is available, then the system could present more detailed explanations of its hypotheses, possibly including diagrams representing different states of the domain task on which the user is working.

With visual output, the system and the user can similarly ground and establish the mutuality of their understanding of the state of the task. For example, the system could present a picture of a state of the object on which the user is working and ask a question such as "Does the item look like this, now?" As discussed with respect to speech input, there may be problems if the interaction gets far enough off-track that the system is unable generate a small set of plausible hypotheses about the state of the task. Imagine being asked by the system about 10,000 different possible states!

Similar issues affect the process in which system and user ground and establish the mutuality of their understanding through speech output. Rather than asking the user to confirm that the state of the task conforms to the state represented in an (unavailable) picture, the system may have to inquire about individual elements of the state. In some ways this may actually be more effective than asking the user to compare reality to a picture: the degree of focus inherent in the questions asked by the system likely will increase the salience of the key factors for determining the state of the task. To the extent the system's initial hypothesis is incorrect, though, the user may be subjected to an interrogation that is unacceptably lengthy.

6.4 Conveying Information

The issue of determining where in the system's task model to start conveying information to the user appears to be independent of speech as an input modality. A user can halt output by saying "stop" to the documentation system, whether the output is visual or aural. This assumes, though, that the speech system permits "barge-in." Should this feature not be available, outputting help at inappropriate times may render the system unusable. But in general, the same task model could be used regardless of the means by which the system provided information to the user. The real issue for when to convey information involves output modalities. In the next two sections, we address visual and speech outputs.

On the output side of the design, the issue of determining where in the system's task model to start conveying information to the user is easier with visual output than with speech output. The penalty for volunteering information—in terms of distraction or of using up attention—is much smaller for the visual modality than for the speech modality. Users can typically determine at a glance if visually proffered information is worth understanding; for speech output, they may have to listen for quite a while before being able to make a comparable judgment. The set of design choices for visual output is relatively large, too. Possibilities range from the obvious (e.g., popping up manual pages deemed relevant) to the subtle (e.g., activating an icon that indicates that the system has information to offer).

In contrast, the speech modality for output will require extra attention to economy in design. Authors should consider using non-speech signals to indicate the availability of different kinds of information. Moreover, the system should let the user determine when information is provided. Otherwise, the busy user may miss part of the speech output. In this case, a "barge-in" capability is essential.

7. CONCLUSION

In this paper, we introduced an analysis of the requirements and design choices for hands-free documentation. We reviewed the need for hands-free documentation and explored the role of task in the use of documentation. Our central analysis involved examination of the roles and characteristics of input and output modalities of hands-free documentation. In particular, we reviewed the use of speech as an input modality, and then visual means and speech as possible output modalities. Finally, we discussed the implications of our analysis for the design of hands-free documentation.

While the capabilities of the spoken and visual modalities available for hands-free documentation are well documented in the literature, the implications for design of hands-free documentation presented in this paper are largely conjectural. Our analysis has been based on our understanding of both the modalities and the functions of documentation, but there remains the enormous work of verifying that our suggestions will actually produce effective use of documentation, particularly in the special contexts that require free hands. We expect, as our suggestions are implemented, applied and tested, that significant refinements of our analysis will be made possible.

Finally, research should address the possibility of using more specialized technology such as gaze-tracking.

8. ACKNOWLEDGMENTS

This research was supported in part by an award from the National Science Foundation, EIA-0080940. The authors thank Javier A. Aldaz Salmon for helpful discussions.

9. REFERENCES

[1] Arons, B. (1991). Hyperspeech: Navigating in speech-only hypermedia. In Hypertext '91 Proceedings, December, 1991, p. 133-146.

[2] Cohen, P. R. and Oviatt, S. L. (1993). The role of voice in human-machine communication, in Human-computer interaction by voice, D. B Roe and J. Wilpon (eds.), Chapter 3, National Academy of Sciences Press, Washington, DC, 34-75

[3] Clark, H. H. and Schaefer, E. F. (1989) "Contributing to Discourse," Cognitive Science, 13, 259-294.

[4] Cole, R. A., Novick, D. G., Vermeulen, P. J. E., Sutton, S., Fanty, M., Wessels, L. F. A., de Villiers, J. H., Schalkwyk, J., Hansen, B., and Burnett, D. (1997). "Experiments with a spoken dialogue system for taking the U.S. Census," Speech Communications, 23(3), 243-260.

[5] Doil, F., Schreiber, W., Alt, T., and Patron, C. (2003). Augmented reality for manufacturing planning, Proceedings of the workshop on Virtual environments 2003, Zurich, Switzerland, 71-76.

[6] Hammerschmidt, C. (2003). 'Augmented reality' speeding assembly and service tasks. EETimes, July 7, 2003. http://eetonline.com/story/OEG20030707S0066. Viewed July, 2003.

[7] Holzman, T. G. (2001). Speech-audio interface for medical information management in field environments. International Journal of Speech Technology 4(3-4), July-Oct. 2001, 209-226.

[8] House, D. (1995). Spoken language access to multimedia (SMAM): A multimodal interface to the World-Wide Web, Master's thesis, Oregon Graduate Institute of Science & Technology.

[9] IBM (2001). ViaVoice Pro User's Guide, Release 9.

[10] James, F. (2003). Voice over workplace (VoWP): Hands-free access to SAP software. SAP Design Guild, http://www.sapdesignguild.org/editions/edition4/vowp.asp, viewed 2003.

[11] Junqua, J.-C. (1993). The Lombard reflex and its rols on human listeners and automatic speech recognizers. In Journal of the Acoustical Society of America, Vol. 93, No. 1, pp 510-524.

[12] Jurafsky, D., and Martin, J. H. (2000). Speech and Language Processing. Prentice Hall, New Jersey.

[13] McTear, M. F. (2002). Spoken dialogue technology: Enabling the conversational user interface. In ACM Computing Surveys, Vol. 34, No. 1, March 2002, 90-169.

[14] Oviatt, S. L., MacEachern, M. and Levow, G. (1998). Predicting hyperarticulate speech during human-computer error resolution, Speech Communication, 24(2), 1-23.

[15] Patokallio, J., and Ward, N. (2001) A wearable cross-language communication aid. In Fifth International Symposium on Wearable Computers (ISWC'01), October 08-09, 2001, Zurich, Switzerland, 176-184.

[16] Pearson, G., and Weiser, M. (1988). Exploratory evaluation of a planar foot-operated cursor-positioning device. In Proceedings of the SIGCHI conference on human factors in computing systems, Washington, DC, 13-18.

[17] Teel, M.-M., Sokolowski, R., Rosenthal, D. and Belge, M. (1998). Voice-enabled structured medical reporting. In CHI 98, Los Angeles, CA, 595-602.

[18] Wakao, A., Takeda, K., and Itakura, F. (1996). Variability of Lombard Effects Under Different Noise Conditions. In Proceedings of the International Conference on Spoken Language Processing (ICSLP) '96, Philadelphia, PA, Vol. 4, 2009-2012.

[19] [Walker, M. A., Fromer, J., Di Fabbrizio, G., Mestel, C. and Hindle, D. (1998). What can I Say? Evaluating a spoken language interface to email. In CHI '98 Proceedings, Conference on Human Factors in Computing Systems, Los Angeles, CA, April 18-23, 1998, 582-598.

[20] Wohlgemuth, W., and Triebfürst, G. (2000). ARVIKA: Augmented reality for development, production and service. In DARE 2000, 151-152.

[21] Yankelovich, N., Levow, G.-A., and Marx, M. (1995). Designing SpeechActs: Issues in speech user interfaces. In CHI '95 Proceedings, Conference on Human Factors in Computing Systems, Denver, CO, May 7-11, 1995, 369-376.