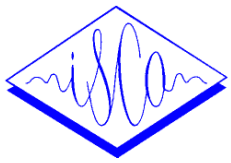

4th European Conference on
Speech Communication and Technology
EUROSPEECH '95
Madrid, Spain, September 18-21, 1995

HUMAN FACTORS OF A VOICE-CONTROLLED CAR STEREO

Reinhold Haeb–Umbach, Stephan Gamm

e-mail: {haeb,gamm}@pfa.philips.de
Philips GmbH Forschungslaboratorien, Weisshausstr. 2, D-52066 Aachen, Germany

ABSTRACT

Today speech recognition with small vocabulary can be realized so cost effectively that the technology can penetrate into consumer electronics. This paper presents some user interface guidelines that are adapted to or specific to voice-control, given the current state-of-the-art recognition technology. The design of a voice-controlled car stereo shows how said guidelines are turned into practice.

1. INTRODUCTION

Voice control indicates the ability to control a machine by means of spoken commands. However, it is by no means straightforward and obvious how to incorporate voice control successfully in an everyday consumer product. Just replacing button presses by speech input does not seem to deliver any discernible benefit. Speech has to be included from the outset of the development rather than simply adding it to an existing system [1], in order to exploit fully the unique properties of voice control, which are

a) *Hands-free operation:* Speech input allows hands- and eyes-free operation which is very important in hands- or eyes-busy situations, e.g. while driving a car.

b) *Remote control:* Speech input can be used for remote control, e.g. via telephone, in order to control a system which is out of manual reach.

c) *'Direct access':* Voice control often avoids the translation of a function into a code. Consider name dialling as an example: in order to place a telephone call the name of the person to be called is just spoken instead of translating the name into a code (the telephone number) and keying in that code.

The challenge of the user interface design is to exploit these strengths while at the same time cope with its shortcomings: The machine recognizer has a limited vocabulary, often a more or less rigid dialogue structure, and misrecognitions are possible. A user interface design must further take into consideration how the performance is affected by factors such as vocabulary size, dialogue structure, adherance to prompts etc.

The design of a voice control interface always has to take into account the capabilities of the speech recognition technology. New or improved recognition algorithms result in new freedom for the user interface design.

The consumer application domain poses very stringent restrictions on hardware costs. With today's technology recognition vocabularies of far more than 100 words seem to be unrealistical. A restricted recognition vocabulary, of course, has immediate consequences on the user interface and dialogue design. It is thus the goal of the design process to find the proper operating point between recognition accuracy, implementation costs and flexibility of the dialogue.

The outline of the paper is as follows. In the next section we will give a short overview of the state-of-the-art in recognition technology as much as it affects user interface design. Section 3 contains a summary of guidelines on how to incorporate voice control in a user interface. In section 4 a voice-controlled car stereo is presented where the principles of the previous section have been turned into practice. Then the paper is finished up in section 5 with some conclusions.

2. SPEECH RECOGNITION TECHNOLOGY

The technological capabilities of automatic speech recognition have a great impact on user interface design. In early systems the user had to speak a fixed vocabulary in a broken fashion to a computer that repeated the last statement, asked for confirmation, then requested another instruction. With the progress in speech recognition more and more constraints do no longer apply, allowing for a design of a more user-oriented rather than machine-oriented dialogue. In the following we give examples of how algorithmic advances lead to increased freedom for the user interface designer.

• *Continuous-speech recognition.*

Current state-of-the-art systems allow for continuous input, rendering the mandatory pauses between the words of an isolated speech recognizer superfluous. This results not only in a more natural way of speaking but also in a higher throughput in spoken words per second.

• *Speaker-independent recognition.*

Obliging a user to train a system before using it, might indeed often be asking to much, in partic-

10.21437/Eurospeech.1995-326

ular for everyday consumer products. A speaker-independent recognizer frees the user from having to train the system. Note, however, that the speaker-independent vocabulary is fixed and cannot be altered by the user. Therefore, often in practice a mix of speaker-independent and speaker-dependent vocabulary is employed: Command words and other common vocabulary (e.g. digits) are 'factory-trained', and the user can add speaker-dependent words for personal settings, e.g. the name repertory for name dialling. Further, it is often a good safety measure to allow a user to overwrite a speaker-independent template by a speaker-dependent one, either to improve the recognition accuracy for this particular word or to give the user the freedom to replace a speaker-independent word by a word he feels more comfortable with. Table 1 summarizes properties of the two recognition modes.

Table 1: Comparison of speaker-independent (SI) and speaker-dependent (SD) recognition.

	SI	SD
Training by user	no	yes
Language dependence	yes	no
Complexity	rel. high	rel. low
Recognition accuracy	rel. low	rel. high

• *Keyword spotting*
Keyword spotting is a technique to artificially increase the vocabulary size of a recognition system as perceived by the user [2]. Commands can be embedded in carrier phrases and the recognition system extracts the commands whereas non-keywords are detected as garbage and thus discarded. Such a technique frees the user from adhering to only keyword input.

• *Word error rate*
The recognition accuracy of automatic speech recognizers has increased constantly over the last years. To give an impressive example, Table 2 presents recognition results on the adult speakers' portion of the Texas Instruments Conncected Digits Recognition Task [3]. The table shows that the string error rate has decreased by a factor of ten over the last ten years[1]. Today's systems achieve a string error rate below 0.9% (corresponding to a digit error rate below 0.3%, since the average string length is 3 digits). To give an example of how such a performance improvement pays off in terms of user interface design, good recognition accuracy allows for digit string input rather than isolated digit input with explicit confirmation after each digit.

[1]The authors are aware that the table is incomplete and would like to apologize to all those not mentioned.

Table 2: Development of string error rate (SER) on TI Digits.

Authors	Site	Publ. Date	SER [%]
Rabiner, Wilpon	AT&T	1987	7.8
Rabiner, Wilpon	AT&T	1988	3.0
Doddington	TI	1989	1.5
Wilpon et al.	AT&T	1991	1.4
Gauvain, Lee	AT&T	1992	0.9
Ney et al.	Philips	1993	0.84
Cardin et al.	CRIM	1993	0.84
Chou et al.	AT&T	1994	0.72

• *Robustness.*
Robustness means the ability of a system to maintain its good performance even under changing environmental conditions. Here, new algorithms have led to considerable improvement, although this topic is still a much addressed research issue. A practical consequence of improved robustness is that now desktop or handheld microphones can be used in situations where head-mounted microphones had to be used before.

The above examples showed that algorithmic advances have led to more freedom for the user interface design. Whereas the speech recognition researcher has a fairly simple measure of performance, the word error rate, the picture is much more complicated for the user interface designer: he has to optimize user satisfaction. He has to find the right balance between recognition accuracy and flexibility for the user, given an upper limit on the algorithmic complexity dictated by the implementation costs.

3. USER INTERFACE DESIGN GUIDELINES FOR VOICE CONTROL

A number of usability principles has found widespread acceptance [4]. In the following we mention those guidelines which assume a special or additional interpretation for voice control interfaces. Further, results of some specific guidelines for voice control interfaces [1], pertinent the our applications, will also be reviewed.

• *Give the user the choice of input modality.*
Systems that use several input modalities, such as voice input and keyboard input, should accept them alternatively whenever an input is demanded from the user [5]. The user should not have to opt once and for all for one input modality. Adherance to this guideline is essential if different input modalities should complement each other in such a way that one modality compensates for the shortcomings of the other [6].

• *Be consistent.*
Consistency asserts that mechanisms should be used in the same way whenever they occur. A particular system action should always be achievable by one particular user action such that a user is not required to learn which command for the same intended action is required at what stage of the machine [4]. If speech input is employed as another input modality, consistency also means that the result of a command should be the same irrespective of the way it has been invoked, whether by a speech command or by a button.
• *Provide appropriate feedback.*
The system should always keep the user informed about what is going on by providing him with correct feedback within reasonable time. Without feedback the user cannot learn from mistakes [4]. This issue is, however, particularly delicate for voice control interfaces. It is very awkward and tiring if the recognizer asks for confirmation each time a word has been recognized. If the outcome of a misrecognition is not fatal and can easily be corrected, it is more appropriate to execute the recognized command rather than asking for confirmation. (Implicit) feedback is then given by the reaction of the machine.
• *Take into account the user's expectations.*
Take into account the possibility that the user's expectations of the system will affect his interpretation of any dialogue with it. The dialogue should be designed to minimize confusions arising from these expectations [4]. For a speech interface the user's expectations might easily exceed the machine's capabilities. Therefore it is important to detect usage problems. One way to do so is to react upon recognition of non-keyword speech: if a non-valid utterance is detected, help menues may be offered automatically.
• *Do not overload the voice input channel .*
Each input modality has its own strength. The input of a location can ideally be done by a pointing device, a straightforward selection is ideally done by speech. But speech is definitely not suited for fine-tunings, i.e. adjustments of a value within a continuous range. Commands like 'up' and 'down' have only limited utility; they have to be used iteratively in order to accomplish an acceptable degree of precision in manipulating an object [1]. Since speech input is not useful for all functions, the right balance between the different input modalitites has to be found.

4. DESIGN OF A VOICE-CONTROLLED CAR STEREO

This section presents a voice-controlled car stereo where the design principles of the last section have been applied.
The hands-free operation has been mentioned as one of the unique properties of speech input. This property has been the motivation for bringing speech recognition into the car environment. The car stereo can be controlled while the hands remain on the steering wheel and the eyes are busy with the traffic.
Figure 1 shows the simulation of a car stereo. In addition to the ordinary control panel there exists an extra 3-button control element, which is shown on the left-hand side of the figure. This control element is assumed to be positioned on or close to the steering wheel, similar to the windscreen wipers or indicators. The microphone is a free-talk microphone mounted preferable on the car ceiling. Using voice commands and the 3-button control element most of the functions of the car stereo, at least the common functions, can be called.
The speech recognizer employs speaker-independent command words and speaker-dependent, i.e. user-defined, words. The speaker-dependent part is used for user-defined names of radio stations. The station names are trained as follows: the radio station to be programmed is tuned in and the user is asked to speak the name of the station, which need of course not be the 'official' name, a couple of times (typically 2 - 4 times). Afterwards this station can be tuned in by speaking its given name. This scenario is very much like the ordinary programming of presets.
A typical usage scenario is as follows. The car stereo is turned on by speaking "radio" or "turn the radio on" while pressing and holding down the recording button in the middle of the 3-button control element (the button with the label 'REC', see Fig. 1). The keyword "radio" is recognized and surrounding phrases, like in the second command, are discarded by the keyword spotting feature. The resulting action is that the car stereo is turned on and starts to play. Now stations can be tuned by speaking their names, such as "BBC" or "change to BBC now". The command word "volume" will program the up and down keys with the volume function, i.e. pushing the up-key will increase the volume, pushing the down-key will decrease the volume. But many other functions can be realized with these keys. To mention just one more, the command "scan station" will program the up/down keys to assume the tuning function: pushing the up-key looks for radio stations by increasing the frequency, pushing the down-key by decreasing the frequency. We have thus introduced the concept of speech-programmable softkeys.
The simulation includes a CD-player and changer. Continuous-speech recognition is employed to enable easy access to tracks on a CD with commands such as: "CD four, track five".
In the following it shall be shown how the presented guidelines have been turned into practice in this application.
• *Give the user the choice of input modality.*
The voice control has been added to the ordinary key control. For the most common functions the user has the choice between voice and key control. For tuning

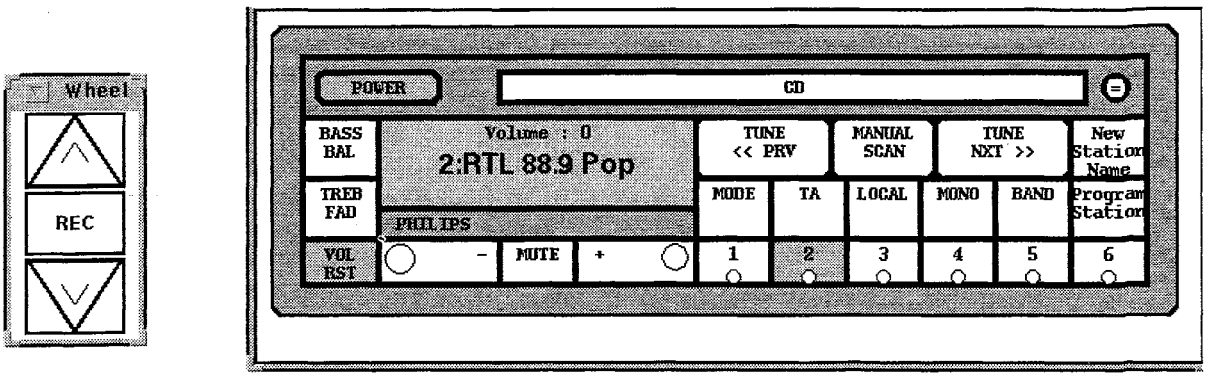

Figure 1: The voice-controlled car stereo

a radio station for example, he can either press a preset button or speak the station's name. During use he can switch between voice and key control whenever he wants. There is no need to stick to the input medium he has chosen once. The key control serves as a fallback mechanism in case of misrecognitions.

• *Be consistent.*

The operation is independent of the selected music source, be it radio or CD. Scanning for example is done in the same way for radio stations as it is done for CD tracks. Consistency is not only assured between the two functional parts of the car stereo but also between the two input modalities. When scanning radio stations for example, pressing the 'Next'-key has the same effect as speaking 'next', i.e. the feedback and the possible further functions are identical.

• *Provide appropriate feedback.*

In this application the feedback is mostly implicit since misrecognitions do not have fatal effects and can easily be corrected. After tuning in another radio station for example, the user gets the implicit feedback by the changed sound and in addition an explicit feedback by the display of the new station name.

• *Take into account the user's expectations.*

The user might not want to or is unable to adhere strictly to the limited command word vocabulary. Therefore synonymes have been introduced to allow the activation of a function by different command words. Further, keyword spotting has been implemented to cope with out-of-vocabulary input.

• *Do not overload the voice input channel.*

Voice control is ideal for a straightforward selection, e.g. for the selection of a preset radio station. Voice control is not suited for all kinds of fine tuning, i.e. the adjustments of volume, bass, treble and fading are better done by keys. The trade-off between voice and key control also concerns the activation of the speech recognizer. In principle the recognizer could be activated by pressing a button or speaking a codeword. Note, however, that an unintentional activation by speech is very costly in terms of user dissatisfaction. In addition, vocal activation tends to be tiring, since each command word has to be preceded by an activation word. Therefore we opted for an activation by button: for controlling the car stereo the 'REC'-button in Fig. 1 has to be pressed and held down for the time a speech command is uttered.

5. SUMMARY

Design guidelines have been formulated for voice control interfaces. It has been shown how those assume certain interpretations for the current state-of-the-art speech recognition technology. The case study of a voice-controlled car stereo showed how the theory has been turned into practice.

6. REFERENCES

[1] D. Jones, K. Hapeshi, and C. Frankish. Design guidelines for speech recognition interfaces. *Applied Ergonomics*, 20(1):47–52, Jan. 1989.

[2] J.G. Wilpon, L.R. Rabiner, C.H. Lee, and E.R. Goldman. Automatic recognition of keywords in unconstrained speech using hidden Markov models. *IEEE Trans. Acoust. Speech and Signal Processing*, 38(11):1870–1878, Nov. 1990.

[3] R.G. Leonhard. A database for speaker-independent digit recognition. In *Proc. IEEE Int. Conf. on Acoustics, Speech, and Signal Processing*, pages 42.11.1–42.11.4, San Diego, CA, Mar. 1984.

[4] J. Nielsen. *Usability Engineering.* Academic Press, San Diego, 1993.

[5] L.J. Stifelman, B. Arons, C. Schmandt, and E.A. Hulteen. Voice notes: A speech interface for a handheld voice notetaker. In *Proc. INTERCHI'93*, pages 179–186, Amsterdam, 1993.

[6] T. Falck, S. Gamm, and A. Kerner. Multimodal dialogues make feature phones easier to use. In *Proc. Applications of Speech Technology*, pages 125–128, Lautrach, Germany, Sep. 1993.