Columbia University Department of Computer Science, Technical Report CUCS-016-03, June 2003.

Using Prosodic Features of Speech and Audio Localization in Graphical User Interfaces

Alex Olwal *Steven Feiner*

Computer Graphics and User Interfaces Laboratory

Department of Computer Science

Columbia University
New York, NY 10027
{aolwal,feiner}@cs.columbia.edu

ABSTRACT

We describe several approaches for using prosodic features of speech and audio localization to control interactive applications. This information can be used for parameter control, as well as for disambiguating speech recognition. We discuss how characteristics of the spoken sentences can be exploited in the user interface; for example, by considering the speed with which the sentence was spoken and the presence of extraneous utterances. We also show how coarse audio localization can be used for low-fidelity gesture tracking, by inferring the speaker's head position.

CR CATEGORIES AND SUBJECT DESCRIPTORS:

H.5.2 (User Interfaces): Graphical user interfaces, natural language, voice I/O; I.2.7 (Natural Language Processing).

GENERAL TERMS:

Design, experimentation, human factors.

KEYWORDS:

Speech, gesture, interaction.

1 INTRODUCTION

We describe a set of nonverbal metrics of speech for use as additional parameters in speech-based interaction. This information allows an application to react explicitly or implicitly to the characteristics of the user's speech. Igarashi and Hughes [2] discuss how duration, pitch and tonguing of nonverbal voice can be used for interactive application control, while Tsukahara and Ward [5] explore the use of prosodic features for appropriate emotional computer response in human-computer dialogues.

We extend this work by showing how prosodic features of *verbal voice*, such as speech rate, duration and volume can be used to control graphical user interfaces, and how audio localization can expand user expressiveness, as well as help resolve ambiguities in speech recognition. We introduce speech-based cursor control and 3D manipulation that use these metrics in Sections 2 and 3, describe our nonverbal metrics in Section 4, and present our conclusions and future work in Section 5.

2 SPEECH-BASED CURSOR CONTROL

Speech-based cursor control can make it possible for individuals who are physically disabled, or temporarily unable to use a keyboard or mouse, to interact with a traditional 2D graphical user interface. Problems with speech-based cursor control include precision and user strain. Karimullah and Sears [3] tried to address the lack of precision with a predictive cursor to compensate for speech-recognition delay (the time from when the command was spoken to its execution), but they concluded that it didn't provide the expected benefits. They found that cursor speed, target size, and speech recognition delays and errors were the factors that were most critical in speech-based cursor control. We address the problem by providing control of cursor speed through speech rate (i.e., how fast the user issues a spoken command).

We are experimenting with nonverbal features in a prototype system in which the cursor speed and direction is controlled by speech commands, speech rate, and the user's position extrapolated from their speech. In one approach, speech commands provide the direction (i.e., right, left, up, and down) and speech rate is used to control the cursor speed. Mapping speech rate to cursor speed is easy to understand and allows the user to execute slow, high-precision cursor movements by issuing commands at a slower pace, and to move the cursor quickly through fast speech (e.g., "Moooooove leeeeeeeft!" vs. "Move–left!") The cursor's speed can be changed while it is moving by reissuing the command at a different pace.

In a second approach, the user provides directional information by leaning to the left or right, and we use simple audio localization (see Section 4.4) to determine the side to which the user is leaning. If this second approach is used exclusively (we permit the simultaneous use of both approaches in our application), it could make it possible to use a smaller grammar and thus potentially improve speech recognition.

3 SPEECH-BASED OBJECT MANIPULATION

We are also experimenting with the use of nonverbal speech features for object manipulation, such as rotation. Consider, for example, rotation around the axis perpendicular to the screen. A head gesture (identified through audio tracking) to either shoulder corresponds to rotation around that axis, and we thus found it appropriate to map the operation to speech and head gesture, as shown in Figure 1. We find this more intuitive than specifying such an operation entirely with speech. The speech command, speech rate and the user's position are used to control the operation (rotation), rotational speed and direction of rotation.

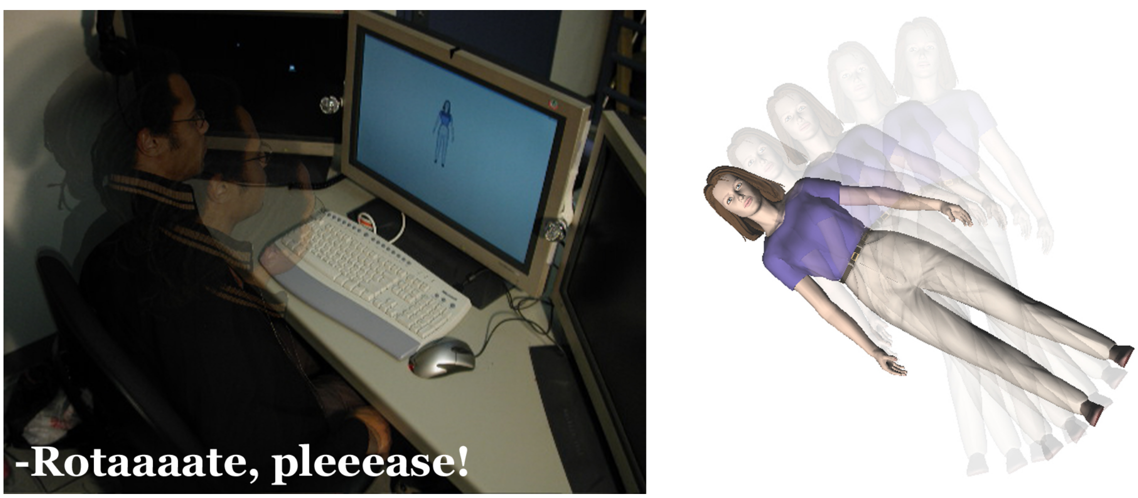

Figure 1. A desktop user manipulating a 3D model with speech commands, simple head gesture (tracked with audio) and speech rate. Rotation (shown at right) occurs after speech recognition.

4 NONVERBAL METRICS

Our prototypes use a set of new interaction techniques for controlling interactive applications, based on nonverbal features in the user's speech. In contrast to previous work [2], we consider the speech characteristics of the verbal sentence.

4.1 Speech rate

We approximate the speech rate as the number of spoken syllables per second. This metric is independent of what is said, and indicates how fast the user spoke. We use the speech rate to differentiate sentences that are spoken at varying speed (e.g., "Zoom in!" vs. "Zoooooom iiiiiiiiin!")

4.2 Duration

We define the duration of a sentence as the time it takes the speaker to speak it. In contrast to speech rate, this metric depends on what is said, since it is not normalized. Thus, we consider duration only when sentences with similar meaning are compared. We use duration to assign different meanings to different sentence formulations (e.g., "Put that there!" vs. "Move that object over there, please!")

4.3 Volume

We currently calculate two volume metrics, the average and maximum volume level from the start to the end of the sentence. Volume distinguishes sentences that are spoken with different loudness (e.g., "Zoom in!" vs. "ZOOM IN!")

4.4 Position

Without the need for a separate head tracker, the user's head position can be approximated by the originating direction of the speech. We use coarse audio localization to distinguish sentences that are spoken from different directions. Position data can be used to make assumptions about user gestures, and have the application react accordingly. For example, in car racing games, players instinctually lean to the left when they want to turn left quickly. The application could in this case let the car turn more when it detects that the user is leaning to either side. The disadvantage is that audio (verbal or nonverbal) is always required in this scenario unless other tracking mechanisms are employed. We can also use audio localization to improve recognition rate by taking into account redundant information from speech and gesture. Our speech recognizer (and most other available recognizers) provides an ordered *n*-best list over recognized speech. If such a list contains "Move left" and "Move in," and the user was leaning to the left during the speech, it might be reasonable to pick "Move left" over "Move in." Audio localization cannot only help reduce errors in speech recognition, but can also reduce cognitive load (and thus, potentially, user errors), by combining speech and gesture. For example, the user could speak "Move" and lean to the left, to perform a "Move left" action. Audio localization can also help the user specify operations that are hard to explain in words, through the use of simple head gestures.

5 IMPLEMENTATION

Our prototypes are implemented in Java and communicate locally with IBM ViaVoice 10 (through the Java Speech API), which provides the necessary data for computing speech rate, duration and volume. For audio localization, we use inexpensive omnidirectional microphones instead of special-purpose hardware, such as array microphones. In our experimental setup, we use two microphones, one each on the left and right side of a flat-panel display. Each microphone is connected to a separate computer running the speech recognition software, where all metrics except position are computed. The Unit dataflow framework [4] is used to communicate by Ethernet the recognized speech, the computed metrics, and the position of the associated microphone, to an application server that provides audiovisual feedback. Speech position is computed in the application server as the difference between the two volumes, and if the highest volume level is significantly higher (i.e., above a set threshold) than the other volume, we assume that the user is closest to this microphone, and the recognition result is taken from this source. The associated position is used to indicate the user's position. (The user is assumed to be in a neutral, centered position if no significant difference is detected.) The discrete use of spatial audio is due to our experimental setup not being sufficiently robust for continuous localization. Despite its lack of sophistication, this approach allowed us to rapidly test this low-fidelity audio tracker and our associated interaction techniques.

6 CONCLUSIONS AND FUTURE WORK

We have introduced a set of new interaction techniques based on spatial audio and prosodic features in speech. We show that even very simple speech analysis can increase the interaction bandwidth, and that spatial audio can expand user expressiveness in speech-based applications.

One limitation of using speech features is that they are normally used to convey emotion, rather than for interaction control. Another limitation is that our simple volume metrics do not distinguish between changes in the user's proximity to a microphone and changes in the volume of the speech itself, which could be handled by more sophisticated analysis. It is also possible to use other, more or less intrusive, tracking technologies for gesture tracking, such as cameras or electromagnetic trackers. On the other hand, it might be desirable to avoid additional technology, if sufficient data can be provided through the (already available) speech input.

Since our experimental setup provides us with extremely rudimentary audio localization, we intend to investigate the use of array microphones for accurate 3D audio tracking and better speech recognition. We are also interested in expanding our set of metrics to include pitch and energy, which are also important features for distinguishing emotions [1]. Our experiments naturally expand into the consideration of these features for adjectives (e.g., "Faster–faster–faster! Slooooweeer!"). Finally, we plan to perform a user study of our speech-based cursor control to investigate its benefits, and further investigate speech-based interaction techniques for 2D and 3D manipulation.

ACKNOWLEDGEMENTS

This research was funded in part by Office of Naval Research Contracts N00014-99-1-0249 and N00014-99-1-0394, NSF Grants IIS-00-82961 and IIS-01-21239, and gifts from Intel, Microsoft Research, and Alias | Wavefront.

REFERENCES

[1] Hirschberg, J., Liscombe, J., and Venditti, J. Experiments in Emotional Speech. IEEE Workshop on Spontaneous Speech Recognition. 2003.

[2] Igarashi, T. and Hughes, J.F. Voice as Sound: Using Non-verbal Voice Input for Interactive Control. *Proc. UIST 2001*, 155–156. 2001.

[3] Karimullah, A.S. and Sears, A. Speech-based cursor control. *Proc. SIGCAPH 2002*, 178–185. 2002.

[4] Olwal, A. *Unit—A Modular Framework for Interaction Technique Design, Development and Implementation.* MS Thesis, Dept. of Num. Anal. and Comp. Sci., Royal Inst. of Tech., Stockholm, Sweden. 2002.

[5] Tsukahara, W. and Ward, N., Responding to Subtle, Fleeting Changes in the User's Internal State. *Proc. ACM CHI 2001*, 77–84, 2001.