Interactive Speech Skimming via Time-Stretched Audio Replay

Wolfgang Hürst
Albert-Ludwigs-Universität Freiburg, Germany

Tobias Lauer
Albert-Ludwigs-Universität Freiburg, Germany

INTRODUCTION

Time stretching, sometimes also referred to as time scaling, is a term describing techniques for replaying speech signals faster (i.e., time compressed) or slower (i.e., time expanded) while preserving their characteristics, such as pitch and timbre. One example for such an approach is the SOLA (synchronous overlap and add) algorithm (Roucus & Wilgus, 1985), which is often used to avoid cartoon-character-like voices during faster replay. Many studies have been carried out in the past in order to evaluate the applicability and the usefulness of time stretching for different tasks in which users are dealing with recorded speech signals. One of the most obvious applications of time compression is speech skimming, which describes the actions involved in quickly going through a speech document in order to identify the overall topic or to locate some specific information. Since people can listen faster than they talk, time-compressed audio, within reasonable limits, can also make sense for normal listening, especially in view of He and Gupta (2001), who suggest that the future bottleneck for consuming multimedia contents will not be network bandwidth but people's limited time. In their study, they found that an upper bound for sustainable speedup during continuous listening is at about 1.6 to 1.7 times the normal speed. This is consistent with other studies such as Galbraith, Ausman, Liu, and Kirby (2003) or Harrigan (2000), indicating preferred speedup ratios between 1.3 and 1.8. Amir, Ponceleon, Blanchard, Petkovic, Srinivasan, and Cohen (2000) found that, depending on the text and speaker, the best speed for comprehension can also be slower than normal, especially for unknown or difficult contents.

BACKGROUND

While all the studies discussed in the previous section have shown the usefulness of time stretching, the question remains how this functionality is best presented to the user. Probably the most extensive and important study of time stretching in relation to user interfaces is the work done by Barry Arons in the early and mid 1990s. Based on detailed user studies, he introduced the SpeechSkimmer interface (Arons, 1994, 1997), which was designed in order to make speech skimming as easy as scanning printed text. To achieve this, the system incorporates time-stretching as well as content-compression techniques. Its interface allows the modification of speech replay in two dimensions. By moving a mark vertically, users can slow down replay (by moving the mark down) or make it faster (by moving the mark upward), thus enabling time-expanded or time-compressed replay. In the horizontal dimension, content-compression techniques are applied. With content compression, parts of the speech signal whose contents have been identified as less relevant or unimportant are removed in order to speed up replay. Importance is usually estimated based on automatic pause detection or the analysis of the emphasis used by the speaker. With SpeechSkimmer, users can choose between several discrete browsing levels, each of which removes more parts of the speech signal that have been identified as less relevant than the remaining ones. Both dimensions can be combined, thus enabling time as well as content compression during replay at the same time. In addition, SpeechSkimmer offers a modified type of backward playing in which small chunks of the signal are replayed in reverse order. It also offers some other

features, such as bookmark-based navigation or jumps to some outstanding positions within the speech signal. The possibility to jump back a few seconds and switch back to normal replay has proven to be especially useful for search tasks. Parts of these techniques and interface design approaches have been successfully used in other systems (e.g., Schmandt, Kim, Lee, Vallejo, & Ackerman, 2002; Stifelman, Arons, & Schmandt, 2001).

Current media players have started integrating time stretching into their set of features as well. Here, faster and slower replay is usually provided in the interface by either offering some buttons that can be used to set replay speed to a fixed, discrete value, or by offering a slider-like widget to continuously modify replay speed in a specific range. It should be noted that if the content-compression part is removed from the SpeechSkimmer interface, the one-dimensional modification of replay speed by moving the corresponding mark vertically basically represents the same concept as the slider-like widget to continuously change replay speed in common media players (although a different orientation and visualization has been chosen).

Figure 1a illustrates an example of a slider-like interface, subsequently called a speed controller, which can be used to adapt speech replay to any value between 0.5 and 3.0 times the normal replay rate. Using such a slider to select a specific replay speed is very intuitive and useful if one wants to continuously listen to speech with a fixed time-compressed or time-expanded replay rate. However, this interface design might have limitations in more interactive scenarios such as information seeking, a task that is characterized by frequent speed changes together with other types of interaction such as skipping irrelevant parts or navigating back and forth. For example, one disadvantage of the usual speed controllers concerns the linear scale. The study by Amir et al. (2000) suggests that humans' perception of time-stretched audio is proportional to the logarithm of the speedup factor rather than linear in the factor itself. So, an increase from, say, 1.6 to 1.8 times the normal speed is perceived as more dramatic than changing the ratio from 1.2 to 1.4. Thus, the information provided by a linear slider scale may be irrelevant or even counterproductive. In any case, explicitly selecting a specific speedup factor does not seem to be the most intuitive procedure for information seeking.

INTERACTIVE SPEECH SKIMMING WITH THE ELASTIC AUDIO SLIDER

In addition to a speed controller, common media players generally include an audio-progress bar that indicates the current position during replay (see Figure 1b). By dragging the thumb on such a bar, users can directly access any random part within the file. However, audio replay is usually paused or continued normally while the bar's thumb is dragged. The reason why there is no immediate audio feedback is that the movements of the thumb performed by the users are usually too fast (if the thumb is moved quickly over larger distances), too slow (if the thumb is moved slowly or movement is paused), or too jerky (if the scrolling direction is changed quickly, if the user abruptly speeds up or jerks to a stop, etc.). Therefore, it is critical and sometimes impossible to achieve a comprehensible audio feedback, even if time-stretching techniques were applied to the signal or small snippets are replayed instead of single samples. On the other hand, such a slider-like inter-

Figure 1. An audio player interface with speed controller and audio-progress bar