Pressure Player: Combined Pressure and Audio Interaction

David McGookin, Stephen Brewster
Department of Computing Science
University of Glasgow, Glasgow G12 8QQ
{mcgookdk, stephen}@dcs.gla.ac.uk

ABSTRACT
Interacting with touch screen devices demands that users look at the screen for even trivial operations. We propose a music player that allows users to preview following tracks by applying pressure on the player screen. We couple this with dynamic modification of the music being played. We present two alternative audio designs, and outline our future evaluation plans.

INTRODUCTION
There has been a recent trend with many mobile device manufacturers to move away from physical controls and buttons, and towards touch screen interaction. Devices such as Apple Inc's iPhone and iPod touch, have shown many of the advances that touch screen interaction affords, such as a dynamically reconfigurable interface allowing different functions of the device to use an optimised interface. For example, the iPod touch uses "disappearing" controls for the movie player function, allowing the entire display to be used to show the movie being played, with the controls overlaid only when the user wishes to interact. There are however drawbacks when using such interaction. Notably, there is no tactile relief on controls. The device must be removed from the user's pocket or bag and looked at for all operations, including those that could previously be done "in pocket" using tactile feedback of the controls. Other researchers have proposed augmenting screens with vibrotactile feedback to provide a tactile sensation when operating virtual buttons [2]. However in this paper, we propose that pressure input could be used to provide a limited subset of operations on touch screen music players. These could be coupled with other haptic technologies, such as vibrotactile feedback, to create a dynamic and useful "eyes-free" interaction.

RELATED WORK
Previous work on pressure based interaction has provided only visual feedback to the user about the level of pressure being applied, and the meaning of that pressure in the interface. Ramos *et al.* [3] carried out various studies involving a Wacom pressure sensitive graphics tablet, where participants selected a target with a cursor controlled by varying amounts of pressure applied via the tablet pen. Ramos *et al.* identified several important findings from their work, including that feedback on the pressure applied, and its meaning in the interface, should be continuous rather than discrete, and that a dwell-based selection mechanism (where the user maintains the pressure applied in a particular range for a duration of time) was the most accurate, but not the fastest technique. Chechanowicz *et al.* [1] found that the number of effective pressure levels depended on the force sensing technology used.

Figure 1. A Screenshot of the Pressure Player Interface.

PRESSURE PLAYER
Whilst work on pressure looks promising, there is a lack of work on both mobile devices, and providing non-visual feedback. We intend to investigate how audio can be used as feedback for pressure based interaction, with a particular view on the situations where current touch screen music players fail (see the introduction). Our test bed application, Pressure Player, is a simple touch screen mp3 player (see Figure 1) that runs on a mobile device. The player allows users to listen to a playlist of songs. Information about the songs is displayed, and users can skip forward or back through tracks, as well as play and pause the music. The player is implemented on a Nokia N800 internet tablet. The N800 is equipped with a touch screen capable of detecting around 500 useful pressure levels. Pressure Player uses these pressure levels to allow the user to easily preview following tracks by applying a force to the touch screen. In our initial implementation we allow the user to preview and/or select the following two tracks in the playlist using pressure.

Mapping Pressure to Sound
Our aim, in providing a sonification of forthcoming tracks, is to provide some feedback to the user of how much more pressure needs to be applied to "pop through" to the next track. As noted by Ramos *et al.* [3], dynamic feedback is more useful than discrete feedback. Providing feedback in such a way should help the user to vary the pressure applied to stay on a particular track. To map feedback to pressure levels we use a probability density function (pdf). This is a function which can be used to calculate the distribution of values for a random event when the mean is know. For each of the three audio sources that the user can browse (the currently playing (or selected) track as well as the following two), we center that source on a particular pressure value. Using that pressure value as the mean, we calculate a proba-

bility density value for each track at each possible pressure value. This acts as a crude measure of probability that the user is trying to preview that track (see Figure 2). The closer the input pressure is to this value, the better the preview of that track will be, and the ability of the user to select it as the playing track. E.g. if the current touch screen pressure applied is around 101, there is greater probability that the user is trying to preview track 2 rather than tracks 1 or 3 (see Figure 2). We use this to drive the audio feedback. We discuss more on the alternative audio designs in the following sections. In order to select a particular track as the currently playing one (e.g. track 2), the user must perform the selection operation within a smaller pressure "bin", again centred on the target pressure for that track. Although Ramos *et al.* [3] found that selection by dwelling in the pressure bin was the most accurate, it is unlikely to be an appropriate mechanism here, as the user may dwell in the pressure bin simply to preview an audio track. Instead we use the quick release metric of selection based on the rate at which pressure falls. I.e. the rate at which the user removes his or her finger from the touch screen.

Dynamic Audio Feedback Designs

Our initial implementation of Pressure Player uses a simple pressure to volume mapping. However we are considering how another technique, granular synthesis, might be applied.

Volume Mapping

Volume mapping is conceptually the simplest design. Here all audio tracks are played simultaneously, but the volume of each is mapped to that track's probability density function at the current pressure level. E.g. if the user applies a pressure of 101 (see Figure 2), track 1 will be played at about 5% of its volume, track 2 at 35% and track 3 at 0%. Therefore the track with the highest probability will be played with the highest volume. Volume is limited to the user selected global volume. I.e. if the user is playing music at 50% of the level the device is capable of, a probability of 1 from the probability density function will be mapped at 50% of the overall device volume level.

Granular Synthesis

An alternate option for audio feedback, granular synthesis [4], is more complex. Here small samples of each audio stream (grains) are selected from each audio track and randomly combined creating a discordant noise. As a user approaches the selection bin of a track (and the probability of that track increases (see Figure 2)) more grains are chosen from that track causing it to "emerge" from the audio mix. Metaphorically this is like tuning an old fashioned radio to a different station; as the pressure level changes, the audio breaks to static before "tuning" into the next track [1].

Each of these options offers advantages and disadvantages for interaction. The volume control system is the most obvious to understand. However even if the tracks used are normalised for volume, there will still be quiet and loud parts in different songs that may confuse the user as to the pressure level being applied. This may be less of an issue if the user is familiar with the track being played. The granular synthesis technique, whilst overcoming these problems has other potential issues, as the playing of white noise (as will occur when the probability of all of the audio streams is low) may be annoying to the user. To measure the extent of these problems we intend to carry out an experiment to compare and contrast each technique.

[1]For this to be successful the probability across all audio sources at each pressure level must add up to 1. This does not always happen so the remainder will come from another "track" of white (static) noise, which is not shown in Figure 2

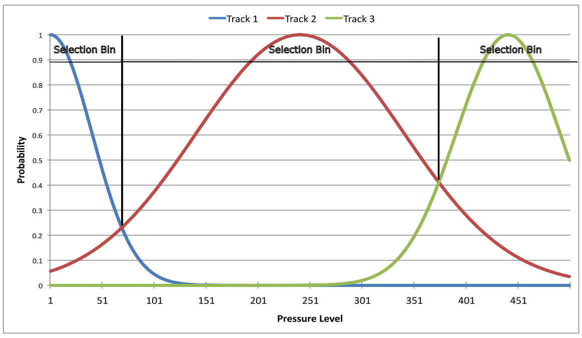

Figure 2. Probability density functions are calculated at each possible pressure value to determine the probability that the user is trying to preview a particular track.

CONCLUSIONS

Whilst touchscreen interaction provides many compelling benefits for mobile devices, it requires visual attention from the user for even simple operations, which may not be convenient or possible. In this paper we have discussed the possibility of using pressure based interaction to effect non-visual useful interaction in a music player. We are currently considering alternate audio mappings, and also designing an experiment to evaluate our different audio designs as well as the number of pressure levels that are possible. We believe the results of this will show that pressure based interaction can be a useful and compelling mobile haptic and audio interaction design.

ACKNOWLEDGMENTS

This work is supported by Nokia and EU framework 6 Open-Interface (IST-35182) and framework 7 HaptiMap (ICT-224675) projects.

REFERENCES

1. Cechanowicz, J., Irani, P., and Subramanian, S. Augmenting the mouse with pressure sensitive input. In *Proceedings of CHI 2007* (San Jose, California, USA, 2007), ACM, pp. 1385–1394.

2. Hoggan, E., Brewster, S., and Johnston, J. Investigating the effectiveness of tactile feedback for mobile touchscreens. In *Proceeding of CHI2008* (Florence, Italy, 2008), ACM, pp. 1573–1582.

3. Ramos, G., Boulos, M., and Balakrishnan, R. Pressure widgets. In *Proceedings of CHI 2004* (Vienna, Austria, 2004), ACM, pp. 487–494.

4. Williamson, J., Murray-Smith, R., and Hughes, S. Shoogle: excitatory multimodal interaction on mobile devices. In *Proceedings of CHI 2007* (San Jose, California, USA, 2007), ACM, pp. 121–124.