Foogue: Eyes-Free Interaction for Smartphones

Christina Dicke
Human Interface Technology Laboratory New Zealand
University of Canterbury
+64 3 364 2987 3070
christina.dicke@hitlabnz.org

Katrin Wolf
Human Interface Technology Laboratory New Zealand
University of Canterbury
+64 3 364 2987 3070
katrin.wolf@canterbury.ac.nz

Yaroslav Tal
T-Systems Switzerland
Industriestr. 21
CH-3052 Bern
+41 78 94660 94
ytal@gmx.net

ABSTRACT
Graphical user interfaces for mobile devices have several drawbacks in mobile situations. In this paper, we present Foogue, an eyes-free interface that utilizes spatial audio and gesture input. Foogue does not require visual attention and hence does not divert visual attention from the task at hand. Foogue has two modes, which are designed to fit the usage patterns of mobile users. For user input we designed a gesture language build of a limited number of simple but also easy to differentiate gesture elements.

Categories and Subject Descriptors
H.5.2 [**User Interfaces**]: *Haptic I/O*

General Terms
Design

Keywords
Auditory interface, Spatial audio, 3D, Gesture interaction, Mobile.

1. INTRODUCTION
Smartphones are not merely telephones anymore but offer a range of functionality that is comparable to mobile computers. So far most smartphone interfaces have not deviated far from the WIMP paradigm as graphical user interfaces (GUIs) are still predominant. The user can switch between 'views' or 'screens', icons and hierarchically structured menus are widely used. The mouse has been replaced by a stylus, touch or multitouch interaction.

Although GUIs are highly efficient in desktop computing and have a long history of research and optimization, in many countries using smartphones while driving is banned. This is due to three drawbacks visual interfaces have in mobile situations. Firstly, as a result of the limited screen size only little information can be displayed. Secondly, to retrieve the information the user has to hold the device up close and focus on the screen. As most mobile situations require visual attention, like driving a car or navigating through an urban environment, the consequences of distractions caused by looking at the screen (and not focusing on the task at hand) can be severe. Thirdly, most feedback is presented visually. Entering letters, selecting icons or scrolling require the user to continuously look at the screen while interacting with the device. This either forces the user to disrupt the primary task (e.g. stop walking / driving) and hence to turn the mobile situation into a stationary one or it diverts visual attention away from the primary task and results in the same conflict described above.

Using headsets for phone calls partly solves the haptic distraction caused by holding the phone to the ear. Miniaturizing icons or offering several selectable 'screens' makes more efficient use of the limited screen size. Adding additional buttons or using regions on the screen to access often used functions reduces the 'eyes on screen' time for these few functions. However, neither of these solutions accomplishes to overcome the distraction caused by pursuing two or more competing visual tasks at the same time.

We propose Foogue, a 3D audio interface that supports menu navigation, item selection, and 'window' management via haptic interaction. While audio has been widely used for alarms, notification, and feedback, Foogue offers a spectrum of functionality that is comparable to common visual interfaces. By employing audio and haptic interaction, Foogue avoids sensory conflicts with visual tasks. Foogue also enables vision-impaired users to fully access mobiles phones and it is transferable to other mobile or stationary devices.

2. WALKTHROUGH
We propose a design solution that is adapted to the challenges of operating a small device in a mobile situation. We presume that smartphone usage patterns deviate from desktop or laptop usage patterns in so far as the mobile user spends more time listening to files (e.g. music) and less time navigating and interacting with the device. Therefore, Foogue supports two modes: Menu mode and Listening mode (see fig. 1).

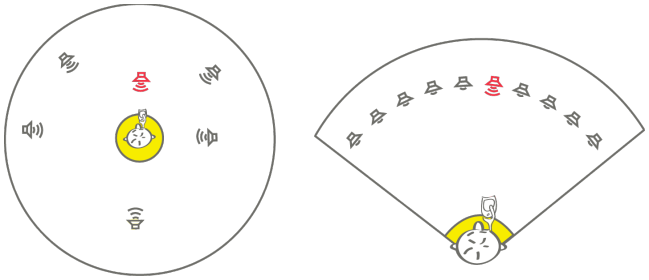

Figure 1: A user using gestures to interact in Foogue's different modes. Left: Listening mode with multiple active players at various positions and one selected player. Right: Menu mode browsing through a folder containing ten items.

Copyright is held by the author/owner(s).
MobileHCI 2010 September 7 - 10, 2010, Lisboa, Portugal.
ACM 978-1-60558-835-3.

Menu mode is designed to grant quick access to files, Listening Mode to make listening to files as comfortable as possible. In Menu mode the file structure is presented in a 120 degree arc in front of the user. Spatialized sound objects represent folders and items. The user can scan the content of the current folder by moving the phone like a torch along the sound sources. By doing so the item currently pointed at will be read to the user, e.g. 'music' or 'contacts'. By performing the 'open' gesture on a container the user descents into the hierarchy. Applying the 'open' gesture on a file will pass it on to a 'player'. Players resemble windows in graphical user interfaces. Depending on the type of file, the player is either a text-to-speech engine, a music player, a phone call pipe, or an acoustic notifier for new events, etc. Once a player is initiated it is displayed in Listening mode, which can be entered by performing the 'switch mode' gesture. Players are spatialized and initially positioned in front of the user. Each player can be selected by pointing the phone at the player. It can be repositioned on a 360 degree circle around the user with a 'drag and drop' gesture. If multiple players are active, the user can either focus on a player by pulling it closer or by pushing other players away so they are played from the distance and accordingly reduced in volume. This way Foogue supports multitasking but also offers an analogy to the 'minimize' and 'maximize' or 'foreground' and 'background' options in visual interfaces. If the user wants to display multiple files from different directories, he/she can use a variation of the 'clipboard' we call the 'buffer'. In Menu mode items or whole directories can be placed in the buffer. Performing the 'open' gesture on the buffer will send the content to the appropriate player. By dragging the buffer onto a folder in the menu the user can copy or move files.

Foogue allows users to interact with their smartphone in mobile situations without competing for visual attention. It is optimized for mobile usage patterns and although it is designed to be self-contained and fully functional, it can be complemented with visual output. Foogue is a high-level interface and hence does not require particular hardware but works on state-of-the-art-smartphones.

3. RELATED WORK

The Nomadic Radio by Sawhney & Schmandt [1] is a mobile, shoulder worn speaker and microphone system to manage voice and text-based messages, including voicemail, email, calendar entries, news, traffic, and weather updates. The user interacts with the system by either voice commands or tactile input. Nomadic Radio uses spatial audio, with audio cues being positioned in a circle around the user's head according to their time of arrival.

Dicke et al. [2] developed a spatial sound interface for navigating between multiple sound streams, which are positioned on a circle around the user's head. The user can either use gestures with the mobile phone or press keys to interact with the system. Panning gestures rotate the ring and allow source selection. Individual streams can either be focused by pulling them close or 'minimized' by pushing them away.

Marentakis & Brewster [3] found that mobile interaction with spatial audio displays is feasible. Participants were able to walk at 73 percent of their normal walking speed and mobility degraded their performance only by 20 percent.

Rico & Brewster [4] point out the important role of designing sociably acceptable gestures for interaction with an interface and provide gesture design recommendations.

4. INTERFACE DESIGN

4.1 Modes

Foogue has two modes to support both the predominantly active phase when interacting with the menu and the predominantly passive phase of listening to selected items. The user switches between modes by performing the 'switch mode' gesture. Changing between modes is confirmed by a feedback sound. If the user switches from Listening mode to Menu mode, all players are paused and the menu is displayed according to its last state.

4.1.1 Menu Mode

In this mode the user navigates, selects or manipulates items from a hierarchical menu. It is comparable to using the 'Explorer' in Microsoft Windows or the 'Finder' in OS X. As depicted in fig. 2, in Menu mode all items are spatialized and arranged in a 120 degree arc in front of the user.

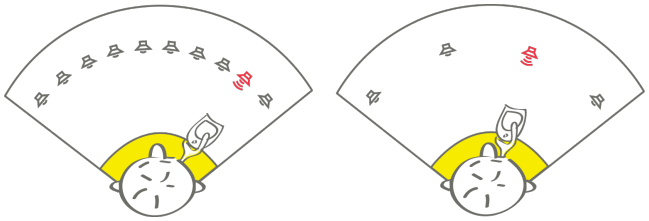

Figure 2: A user browsing through a list of items and selecting one item in Menu mode.

Items are displayed in sequence and ordered alphabetically, but can be alternatively ordered by file type or size. Items' positions are fixed to utilize spatial memory and allow users to 'jump' to a specific item without the need to scan through all displayed items. When pointing the device at an item, the item identifies itself by speaking its (file-)name. Foogue supports single and multiple item selection as well as selecting a range of items. For single file selection the user points at the item and performs the 'open' gesture. For multiple file selection from either one or multiple folders the user points at items and moves them individually to the buffer (as shown in fig. 3). Whole folders can also be moved to the buffer.

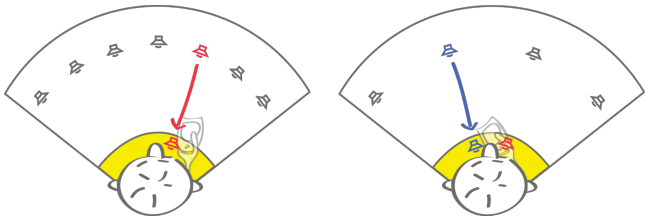

Figure 3: A user selecting items from different directories and moving them to the buffer.

For selecting a range of files the user points at the start item, performs the 'select range' gesture and moves the device to the end item. If the 'open' gesture is performed either on a single file or the buffer, the appropriate player is instantiated in Listening mode and the file/s is/are played when this mode is entered.

4.1.2 Listening Mode

In this mode the users mainly listen to what they have previously selected. Nevertheless, they can interact with players and rearrange their positions. Unlike in Menu mode, players can be positioned anywhere on a 360 degree circle around the user (see fig. 4).

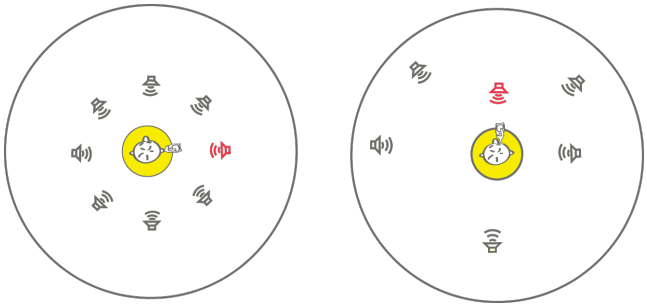

Figure 4: A user selecting and repositioning players in Listening mode.

Players can be clustered - for example all (sporadic) notifiers on the left, continuous playback (like music or podcasts) on the right side. Players can not only be rearranged in terms of direction but also in terms of distance. Foogue uses distance as a metaphor for minimizing/maximizing or, in other words, to focus/defocus attention. If the user wants to stay aware of a player, like one displaying notifications, it can be moved farther away. This way distraction from that player is minimized while the user is still able to maintain an awareness of the player being present. If the user wants to focus entirely on one player, as during a phone call, the player can be pulled into the 'Focus Zone' (see fig. 5).

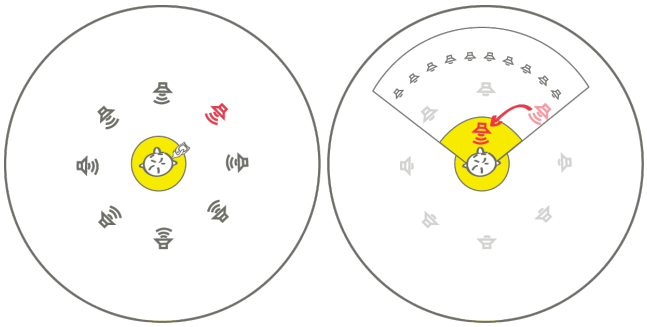

Figure 5: A user moving one player into the Focus Zone (yellow). Once the player is in the Focus Zone the right-click-menu becomes accessible.

This has three effects: All other players are paused, the stream from the player is played in stereo (if available), and the context or 'right-click' menu is displayed on a 120 degree arc just like in Menu mode. As items are silent unless they are pointed at, this keeps the context menu available without causing distraction.

4.2 Interface metaphors

Lakoff & Johnson [4] describe how thinking in ontological metaphors enables humans to refer to otherwise abstract concepts. Thinking of data in terms of an *object* that can be moved, copied, or named is a simple example of an entity metaphor. Foogue uses three ontological metaphors: Containers, entities, and substances. An example of a container is the folder, which contains either other folders or files. In Menu mode files are thought of as entities. Users can navigate through a structure of folders and select, copy or move files. But in Listening mode, when a file is 'played', it changes it nature and becomes a 'substance' ('water'). It is routed through a player, it can be 'diverted' (moved), it's 'flow' can be 'disrupted' (paused) and so forth. We find it important to reflect the different purposes of both modes in the metaphors we apply. Menu mode supports a perception of data in terms of a solid structure of containers and objects. But the actual playback in Listening mode refers more to the temporal, fluid nature of sound. We also use so called 'orientational' metaphors, which refer to spatial orientation such as front-back or up-down. In Foogue we apply the close-far metaphor or central-peripheral metaphor. In Listening mode users can push players away. This implies a 'downgrading' in terms of the amount of attention paid to them. By pulling a player closer into the 'Focus Zone' it gets full attention and accordingly all other players are paused.

4.3 User input

We developed a gesture language build of a limited number of simple but also easy to differentiate gesture elements. This language is a combination of 3D interaction techniques (point, torch, tilt, move, rotate, drag & drop) and 2D gestures on a keypad or touch screen (see fig. 6). We developed the gesture language based on user-designed gestures created in task-driven explorative experiments. We put special emphasis on designing gestures that are intuitive and easily discoverable, but also took into account the limited movement range of mobile users.

Figure 6: A user performing the 'select' gesture by pointing at a source and moving the device upwards.

4.4 Technical Feasibility

Utilizing inbuilt gyroscopes, digital compasses or cameras (visual feature tracking) the device's position in space can be determined. Smartphones like the iPhone already support 3D sound libraries like OpenAL. The interface described in this paper is partly realized on a Nokia N900 using OpenAL. An example of Menu mode is currently prototypically implemented on a PC using an Intersense InertiaCube3 for user input and OpenAL for creating the 3D sound space.

5. REFERENCES

[1] Sawhney, N. and Schmandt, C. 2000. Nomadic Radio: Speech and Audio Interaction for Contextual Messaging in Nomadic Environments. ACM Transactions on Computer-Human Interaction, 7, 3, 353-383

[2] Dicke, C. Deo, S., Billinghurst, M. and Lehikoinen, J. 2008. Experiments in Mobile Spatial Audio-Conferencing: Key-based and Gesture-based interaction", Mobile HCI, 91-100.

[3] Marentakis, G.N., Brewster, S.A. 2006. Effects of Feedback, Mobility and Index of Difficulty on Deictic Spatial Audio Target Acquisition in the Horizontal Plane. CHI, Montréal, Québec, Canada.

[4] Rico, J. and Brewster, S.A. 2010. Usable Gestures for Mobile Interfaces: Evaluating Social Acceptability. CHI, Atlanta, GA, USA.

[5] Lakoff, G. and Johnson, M. 1980. Metaphors we live by. Chicago: University of Chicago Press.