Natural Throw and Tilt Interaction between Mobile Phones and Distant Displays

Raimund Dachselt
University of Magdeburg
Faculty of Computer Science
User Interface & Software
Engineering Group
39106 Magdeburg, Germany
dachselt@acm.org

Robert Buchholz
University of Magdeburg
Faculty of Computer Science
Multimedia and Security Group
39106 Magdeburg, Germany
robert.buchholz@iti.cs.uni-magdeburg.de

CHI 2009, April 4–9, 2009, Boston, Massachusetts, USA.
ACM 978-1-60558-247-4/09/04.

Abstract
To provide intuitive ways of interacting with media data, this research work addresses the seamless combination of sensor-enabled phones with large displays. An intuitive basic set of tilt gestures is introduced for a stepwise or continuous interaction with both mobile applications and distant user interfaces by utilizing the handheld as a remote control. In addition, we introduce throwing gestures to transfer media documents and even running interfaces to a large display. To improve usability, data and interfaces can be thrown from a mobile phone to a distant screen and also fetched back to achieve mobility. We demonstrate the feasibility of the interaction methods with several advanced application prototypes facilitating a natural flow of interaction.

Keywords
Gestures, Interaction Design, Remote Interaction, Accelerometer, Music Browser, User Experience

ACM Classification Keywords
H.5.2 [Information interfaces and presentation]: User Interfaces – Input devices and strategies; Interaction Styles

Introduction

The availability of mobile phones and the variety of mobile applications has increased dramatically over the last decade. Nowadays, most mobile phones are able to store and manage all kinds of personal information and media. Despite many advances in the field, all of these devices are inherently limited in terms of their screen size and amount of information they can display. When combined, two advances could overcome this limitation. The first is the increasing integration of powerful sensor packs into mobile phones. Their potential for creating interaction techniques has been recognized in research for some time. The second is the increasing usage of large-screen displays for home entertainment and in public spaces, but still using traditional input devices.

As part of our research on the seamless interaction between multiple displays of various sizes we developed *Throw and Tilt* and introduced it in [2]. This approach combines sensor-enabled, gesture-based mobile phone interaction with large displays. One way is to use intuitive throwing gestures to transfer personal media data such as music, pictures, and map locations to a large screen. An even more advanced way is to seamlessly transfer the whole user interface (UI) between various devices. With it, the mobile phone is no longer treated as a small computer with a very limited display. It can extend its role to that of an input and control device for a distant display by using a simple and generic set of discrete and continuous tilt gestures to ensure natural interaction without a steep learning curve. This will be demonstrated with several sample applications.

Related Work

One of the first researchers to describe the usage of tilt as an input method for small screen devices was Rekimoto [8]. Early devices had no built-in accelerometers or other sensors. Therefore, researchers used the built-in camera of mobile phones for movement detection (e.g. [9, 1]) or manually added sensor packs to palmtop computers and mobile phones, e.g. for tilt-based browsing of photographs [5]. Cho et al. conclude that browsing with a single tilt gesture is indeed feasible and appealing to users. A typical application of using accelerometer sensors is the training and recognition of gestures on mobile phones, e.g. [6], as well as on the Nintendo Wiimote. In contrast, we focus on very few simple gestures that do not require machine learning for training and require almost no learning curve.

Amongst others, Cheverst et al. evaluated the interaction of mobile phones with situated displays [4]. However, their public displays act as whiteboards and are places to retrieve or drop information and not to extend a user interface to. Pering et al. also mentioned the idea of using a mobile phone to control other devices and even transfer media to these [7], but their main focus is on simplifying the process of connecting to these devices, not interacting with them. Similarly, with Toss-It, Yatani et al. developed pure information transfer techniques between PDAs [10].

Tilt Interaction with Mobile and Distant UIs

In the domain of personal media management and home entertainment, many tools rely on a simple set of interactions. One example is a mobile music player with hierarchical lists of music items which can be navigated using the up and down directions for scrolling and right and left directions for working with sub menus. These simple interactions can also be found as part of higher-level interaction tasks such as scrolling, zooming, panning, rate control, or menu selection. We identify the

Figure 1. Tilting along the x axis: *down, neutral, up* and along the y axis: *left, right.*

following basic set of interactions to be supported by a mobile phone: *up, down, left,* and *right* movements of either a cursor position, some widget highlight (such as in menus), or a direct mapping to navigate a movable application space (e.g. maps). These four movements can either be *discrete (stepwise)* or *continuous (fluent)*. For the latter we distinguish between a linear and non-linear mapping of movements to interface changes.

The idea is to support this simple set of interactions by recognizing corresponding tilt gestures with an off-the-shelf sensor-enhanced mobile phone (compare Figure 1). Thus, we want to support everyday users by providing a natural and simple way of interaction. The force values measured by the built-in triaxial accelerometer can be interpreted as a vector pointing to the center of the earth. For many applications it is sufficient to directly interpret the x and y sensor values as measures for the rotation along the x and y axes.

Figure 1 depicts the supported tilting movements for both axes. While holding the mobile phone in a resting position on her palm, the user can easily perform tilting gestures in all directions along the x and y axes. For performing *discrete interactions*, the phone is simply moved back to the neutral resting position after each interaction step. For this type of interaction a simple threshold is used to recognize a single interaction in either direction. For the *continuous interaction* mode a neutral zone should be defined. This compensates for trembling of the user's hand and accidental movements. Compare Figure 2 for a schematic depiction of how rotational values of one axis are divided into zones which transform the values differently. Each zone can either have a linear mapping of the rotational values to the user interface or a non-linear, usually exponential function amplifying the movement. Beside the neutral zone, typically two additional zones are used. One allows for fine control in the near field, the other for accelerated control for distant parts, e.g. for scrolling documents. A good transfer function should ignore very small sensor values, accurately reflect medium ones and amplify big ones. We designed our implementation to allow for a flexible assignment of the zones and transfer functions depending on the application.

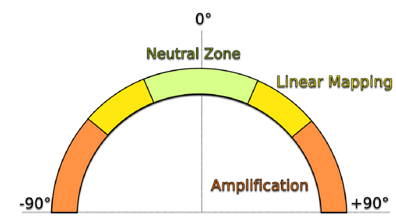

Figure 2. Different mapping zones are used for each axis.

Like previous research suggested (e.g. [8, 5]), the tilt gestures can be used to control applications on the mobile phone itself. We propose to also use the mobile phone as a remote control operating a distant UI with this set of basic tilt gestures. Due to their simplicity, attention does not need to be split, and users can concentrate on the large display while tilting the phone.

Application Scenario 1: A Zoomable Music Browser
Mambo (Mobile fAcet-based Music BrOwser) is a Zoomable UI for facet browsing of large personal music collections being available on mobile and other devices. It supports visual browsing and faceted filtering of songs, albums, or artists according to different hierarchical metadata facets [3]. Thus, scrolling long lists of items is entirely eliminated. The discrete tap-and-center navigation mode allows a stepwise navigation through

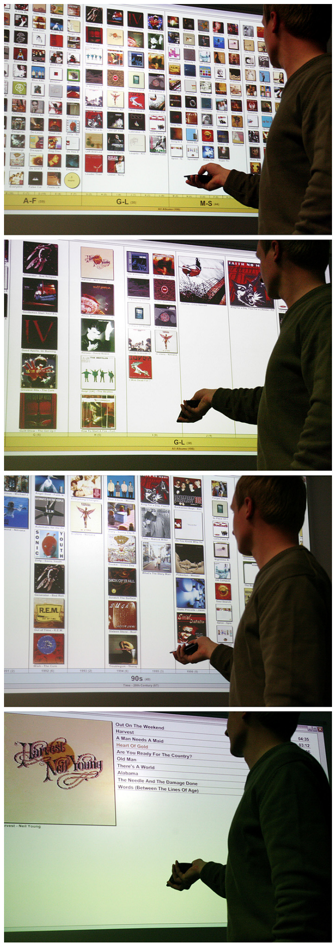

Figure 3. Using discrete tilt gestures to operate Mambo on a large screen. Top to bottom: all albums arranged by name; zoom to albums in the range of G-L; albums arranged by time; selected album.

and within levels of a facet hierarchy. It can be operated by mouse clicks on individual cells, by cursor or other directional keys and also by *discrete directional tilt gestures* of the mobile phone. With it, left and right gestures are mapped to stepwise panning, up and down to zooming in or out. Figure 3 shows the remote operation of the Mambo UI using a handheld phone with tilt gestures.

Application Scenario 2: Navigating 3D Digital Globes
Whereas Mambo utilizes the discrete tilt gestures to navigate the information space, we also implemented an example scenario for the *continuous tilt interaction* mode. Our first prototypes were implemented using the Google Earth COM API and Google Earth API. With them, one can remotely operate the virtual globe's interface, for example browsing the geo-referenced holiday photos on a large wall projection while sitting in an armchair. Panning of the map is simply performed by a *continuous tilt interaction* of the handheld device in the corresponding direction. To facilitate the various interaction modes provided with 3D map programs, mode switches are required. Pressing the central button on the phone, users can switch between *pan mode* and *zoom and tilt mode*. In the latter, zooming in is performed in a continuous way by moving down the phone, zooming out by tilting it in the opposite direction. Changing bird's eye view to street view, i.e. tilting the map, is achieved by tilting the phone to the left or right, which becomes more natural when rotating the phone about 90° to a position as shown in Figure 4.

Throwing Data and UIs to Wall Displays

In the domain of personal media management, items frequently need to be transferred between various devices, e.g. camera phones and media servers at home.

Figure 4: Remotely operating Google Earth with a touch-enabled mobile phone – tilting the surface and panning.

To provide an intuitive way of transferring personal data, we suggest simple *throw gestures* to connect a mobile device and a large display (i.e. stationary computer) and transfer data. One usage scenario is the transfer of newly taken pictures to a home media center: Performing a throw gesture with the phone (see Figure 5, center) within range of the computer transfers all data to the TV screen / HDTV projection over WiFi or Bluetooth. This connection is already established while entering the room. Afterwards, the user can browse or zoom the pictures on the large display by tilting the mobile phone, thereby experiencing a seamless interaction across devices. Visual explanations on how to use the gestures for interaction can be displayed on the phone itself as well as on the wall, e.g. as a miniature.

In the opposite direction, i.e. from a big screen to the mobile phone, we utilize a *fetch-back gesture* for transferring selected content by rapidly moving the phone towards the user's body. This gesture (similar to [10])

Figure 5. A photo taken by a mobile phone (top) is thrown to a distant display and appears there.

is somehow asymmetric to the throw gesture for two reasons: one is a missing metaphor for the opposite of throwing; the other is the necessary distinction between recognizing the tilting and fetch-back gestures.

Besides transferring your selected data (which in turn needs to be handled by different applications), we contribute the concept of *transferring a running user interface from a mobile phone to a large screen* (to improve usability) and back (to achieve mobility). This is also achieved by using throwing and fetch-back gestures. It provides a way of interacting in a truly seamless fashion across devices, assuming that an application exists for both the mobile and stationary device. This can be either an identical application implemented on both platforms or similar applications from the same domain which can exchange data, such as an image manager.

In addition to transferring data (e.g. documents), the state of an application running on the mobile phone needs to the transferred to a large screen and back. This also applies to interface configuration information, for examples open tool palettes. In case of a remotely controlled UI, control data needs to be passed between the phone and distant computer. If the basic tilt interactions are supported by the program, the user can immediately continue using the application on the large display. Since the interaction method, i.e. tilting the handheld, stays the same, throwing the interface can be conceptually seen as just switching displays. However, unlike VNC-enabled remote screens, the interface might look different on both devices.

Application Scenario 3: Image Browser

Similar to the description in the previous section, this application allows throwing images or other captured media from a personal mobile phone to a large, distant display, e.g. a home cinema projection (see Figure 5 and 6). For browsing photographs, discrete tilt gestures can be used on the mobile phone itself. After entering a room, data and the whole user interface state (e.g. currently selected pictures) can be thrown to the wall. On connecting, all necessary data is received and cached by the PC based on a MD5 hash for later use. While using only the phone, the image browser is limited to simple full screen or tiled picture browsing with tilt gestures. When transferring control to the large screen, smooth cover-flow browsing and a map-located photo display can be used. Moreover, additional information (e.g. EXIF data) can be displayed, and comfortable image selection is supported. Thus, the interface capabilities are extended. The ring menu view of pictures can be navigated smoothly by phone tilt gestures.

Implementation

Currently, we have implemented all three application scenarios. The research prototypes serve as a test bed for evaluating and refining the gesture interactions. We are using a Nokia N95 8 GB Symbian S60 phone and Python for rapid prototyping. In addition, the iPhone with its SDK is used, allowing us to also utilize touch events to reduce mode switches. Both phones are equipped with a triaxial accelerometer. The uncalibrated sensor delivers data which is subject to some jitter. This can be compensated by smoothing on the phone at the cost of reducing the temporal resolution or accuracy. A simple or composite transfer function to map the sensor data to meaningful interaction (e.g. zoom) values is executed on a PC after the sensor data has been transmitted. To recognize a throwing gesture, we use a threshold of the Euclidean length of the raw force vector delivered by the sensor. Currently, we are considering the usage of a more sophisticated gesture recognition framework to better distinguish explicit and non-explicit gestures. For the communication, we decided to use WiFi, the low-latency UDP for transmitting sensor data and the reliable TCP for files.

Conclusion and Future Work

In this work we investigated a basic repertoire of intuitive gesture interactions on accelerometer-enabled mobile phones. The continuous and stepwise tilt gestures can be mapped to a variety of elementary interactions typical for media-centered applications. The simple but powerful gestures can be used to control various applications both on mobile screens and large displays. In addition, we proposed simple throw gestures to connect a mobile and distant display by transferring data and even running interfaces back and forth, thus facilitating a natural flow of interaction. With these gestures, mobile devices and large displays are combined in a seamless manner. We also implemented several prototypes, which need to be supplemented and carefully evaluated as future work. We will investigate the transfer of interfaces more closely, which is a non-trivial software engineering and usability challenge, and examine additional application scenarios combining private mobile and public displays. This also includes issues of multi-user collaboration, proximity, and directionality of devices.

Figure 6. Image Browser on mobile phone and large display. From top: UI on the phone; thrown to wall; tilt browsing; selecting images for deletion; tilt browsing with reduced number.

Acknowledgements

We thank Jürgen Scheible with his MobiToss video-tossing application for the original inspiration and Niklas Röber, Mathias Frisch, Lars Uebernickel, and Jens Henneberg for their great support. Our work was funded by the "Stifterverband für die Deutsche Wissenschaft" from funds of the Claussen-Simon-Endowment.

References

[1] Ballagas, R., Rohs, M., Sheridan, J. G. Sweep and point and shoot: phonecam-based interactions for large public displays. In *Ext. Abstracts CHI 2005*, ACM Press (2005), 1200-1203.

[2] Dachselt, R., Buchholz, R. Throw and Tilt – Seamless Interaction across Devices Using Mobile Phone Gestures. In *Proc. MEIS 2008*, Lecture Notes in Informatics, Vol. P-133 (2008), 272-278.

[3] Dachselt, R., Frisch, M. Mambo: A Facet-based Zoomable Music Browser. In *Proc. MUM 2007*, ACM Press (2007), 110-117.

[4] Cheverst, K., Dix, A., Fitton, D.. Kray, C., Rouncefield, M., Sheridan, J., Saslis-Lagoudakis, G. Exploring Mobile Phone Interaction with Situated Displays. In *Proc. Workshop PERMID* at Pervasive 2005.

[5] Cho, S., Choi, C., Sung, Y., Lee, K., Kim, Y., Murray-Smith, R. Dynamics of tilt-based browsing on mobile devices. In *Ext. Abstracts CHI 2007*, ACM Press (2007), 1947-1952.

[6] Kela, J., Korpipää, P., Mäntyjärvi, J., Kallio, S., Savino, G., Jozzo, L., Marca, D. Accelerometer-based gesture control for a design environment. *Personal Ubiquitous Computing*, 10, 5 (Jul. 2006), 285-299.

[7] Pering, T., Anokwa, Y., and Want, R. Gesture connect: facilitating tangible interaction with a flick of the wrist. In *Proc. TEI 2007*. ACM Press (2007), 259-262.

[8] Rekimoto, J. Tilting operations for small screen interfaces. In *Proc. UIST 1996*. ACM Press, 167-168.

[9] Wang, J., Zhai, S., and Canny, J. Camera phone based motion sensing: interaction techniques, applications and performance study. In *Proc. UIST 2006*, ACM Press (2006), 101-110.

[10] Yatani, K., Tamura, K., Hiroki, K., Sugimoto, M., and Hashizume, H. Toss-it: intuitive information transfer techniques for mobile devices. In *Ext. Abstracts CHI 2005*, ACM Press (2005), 1881-1884.