Andreas Zimmermann, Niels Henze,
Xavier Righetti and Enrico Rukzio (Eds.)

Mobile Interaction with the Real World

Workshop in conjunction with MobileHCI 2009

BIS-Verlag der Carl von Ossietzky Universität Oldenburg

Oldenburg, 2009

Verlag / Druck / Vertrieb
BIS-Verlag
der Carl von Ossietzky Universität Oldenburg
Postfach 2541
26015 Oldenburg

E-Mail: bisverlag@uni-oldenburg.de
Internet: www.bis-verlag.de

ISBN 978-3-8142-2177-X

Preface

Welcome to the proceedings of the 4th workshop on Mobile Interaction with the Real World. This workshop focuses on new mobile and wearable input and output interfaces which allow simpler and straightforward interactions with mobile services and applications. An inherit problem of mobile HCI are the limited output and input capabilities of current mobile devices. This workshop continues a successful series of workshops that focus on new approaches to overcome these issues. Examples are the usage of external visual interfaces, gestural input techniques, innovative applications, and underlying frameworks for mobile interaction with the real world.

We received 22 submissions and each submission received 2-4 blind reviews from the program committee and the organizers. After intensive discussions the 10 contributions you find in these proceedings were selected which leads to an acceptance rate of 45%. We would like to thank the authors for their contributions and the reliable reviewers for their work. Furthermore we would like to thank the European Network of Excellence InterMedia for their support. We hope that these proceedings provide interesting insights to the reader and foster active discussions in the areas of mobile interactions with the real world.

Andreas Zimmermann, Niels Henze,
Xavier Righetti, and Enrico Rukzio

Table of Contents

Workshop on Mobile Interaction with the Real World

Andreas Zimmermann, Niels Henze
Xavier Righetti, and Enriko Rukzio

Abstract

The workshop on Mobile Interaction with the Real World (MIRW 2009) invited papers that focus on new mobile and wearable input and output interfaces which allow simpler and straightforward interactions with mobile services and applications. An inherit problem of current mobile devices are their limited output and input capabilities. This workshop continues a successful series of workshops (2006-2008) that focus on new approaches to overcome these issues. Examples are the usage of external visual interfaces (e.g. projector phones, public displays, interactive surfaces) and additional input capabilities (e.g. gestures, on-body interfaces, pointing) and innovative feedback mechanisms (e.g. tactile feedback). The workshop combines technical presentations with the presentation of prototypes and focused discussions to drive interaction between participants.

1 Introduction

Mobile devices are a pervasive part of our everyday lives. People use mobile phones, PDAs, and mobile media player almost everywhere. These devices are the first truly pervasive interaction devices that are currently used for a huge variety of services and applications. Stordahl et al. for example forecast that in the year 2010 over 90% of the population in Western Europe will use mobile phones [1].

However, mobile device's immanent size restriction leads to key limitations such as a small visual display and limited input capabilities. Furthermore, current mobile user interfaces often disengage from the environment. To overcome these limitations we saw increased interest in extending the

interaction boundaries of mobile device by developing novel input and output interfaces.

The mobile interaction with the real world workshop series provides a forum which concentrates on mobile and wearable interaction with real world objects. The work on mobile applications, concepts, and techniques enabling the user to interact with real world objects using mobile devices have shown promising results [2,3,4]. Examples for this are for instances the usage of RFID/NFC equipped mobile devices for interactions with smart objects such as advertisement posters or vending machines; the usage of mobile devices as a universal remote control or the usage of mobile devices for direct interactions (e.g. based on image recognition) with objects in a museum.

Following the successful series of workshops on "Mobile Interaction with the Real World" at MobileHCI 2006 to 2008, we continue this workshop as a forum that concentrates on mobile and wearable interactions with real world objects.

2 Research Topics

Topics of the workshop are application, frameworks, and user studies in the area of mobile interaction with the real world. Research themes include (but are not limited to):

- Extending the user interface beyond the mobile device
- Mobile interaction with real world objects and smart objects
- Wearable computing and wearable input devices
- Multimodal interaction techniques using mobile phones
- Augmented and mixed reality on mobile devices
- Interaction techniques using external displays, projector phones or floor displays
- Using mobile device's sensors for pervasive applications
- Novel interfaces for conveying spatial information
- Pervasive interaction metaphors

- Guidelines and standards for mobile interactions with the real world
- Interaction techniques using multiple mobile devices
- Support of knowledge processes and collaboration through mobile and wearable technologies

3 Goals

The main goal of the workshop is to develop an understanding of how mobile and wearable devices can be used to interact with the real world. We seek for new ideas, prototypes, and insights as basis to develop a deeper understanding of the field. We provide an open forum to share information, results, and ideas on current research in this area. This workshop encourages discussion about future topics concerning mobile interaction with the real world. Furthermore we aim to develop new ideas on how mobile devices can be exploited for new forms of interaction with the environment. We bring together researchers and practitioners who are concerned with design, development, and implementation of new applications and services using personal mobile and wearable devices as user interfaces. Furthermore, the workshop aims at conveying hands-on experience with current state-of-the-art technology through demonstration sessions.

4 Organizers

Andreas Zimmermann

Andreas works as a senior researcher in the department Information in Context at the Fraunhofer Institute for Applied Information Technology (FIT) in Sankt Augustin (Germany). He has a strong research background in context-aware computing and artificial intelligence, and his further research interests include areas such as nomadic systems and end-user control of ubiquitous computing environments. Within the scope of two European projects he currently manages, he is responsible for the user-centred design process and for the design of software architectures.

Niels Henze

Niels is working as a researcher and PhD student in the Media Informatics and Multimedia Systems Group of the University of Oldenburg (Germany). He is involved in the Intelligent User Interfaces group at the research institute OFFIS. He worked for some national and European research projects and is currently involved in the European project InterMedia. He is interested in interaction with media using mobile devices and advances in accessing digital information using real world entities. Among his other research interests are tactile interaction and accessibility.

Xavier Righetti

Xavier is a research assistant and PhD student in the Virtual Reality Lab (VRlab) at Ecole Polytechniques Fédérales de Lausanne (EPFL), Switzerland. He is currently involved in the European Network of Excellence Intermedia in which he focuses on the design and development of modular wearable components for user input / output, processing and storage. His vision is the usage of wearable devices on demand and their ad-hoc collaboration once worn by a user.

Enrico Rukzio

Enrico is working as an academic fellow and lecturer at the Computing Department at Lancaster University. Enrico's research interests are physical mobile interactions and applications as well as context-aware mobile services. Enrico believes that mobile devices which were so far mostly used for interactions between the user and the device itself will more and more be used for interactions with objects in the real world. Currently he works new interaction techniques for projector phones and mobile interactions with floor displays, interactive surfaces and public displays.

5 Program committee

- Susanne Boll, University of Oldenburg, Germany
- Daniel Thalmann, EPFL, Switzerland
- Gregor Broll, DOCOMO Euro-Labs, Germany

- Andreas Lorenz, Fraunhofer FIT, Germany
- Michael Rohs, Deutsche Telekom Laboratories, Germany
- Jonna Häkkilä, Nokia Research Center, Finland
- Johannes Schöning, University of Münster, Germany
- Dominique Guinard, ETH Zurich and SAP Research, Switzerland
- Martin Pielot, OFFIS Institute for Information Technology, Germany
- Christian Kray, Newcastle University, United Kingdom
- Derek Reilly, Georgia Institute of Technology, United States
- Benjamin Poppinga, OFFIS Institute for Information Technology, Germany

6 References

[1] Stordahl, K., Gjerde, I. G., and Venturin, R. 2005. Long-Term Forecasts for the Mobile Market in Western Europe. In Proceedings of 16th Regional European ITS Conference, pages 76-77.

[2] Rukzio, E., Paolucci, M., Finin, T., Wisner, P., and Payne, T. (Eds.). 2006. Proceedings of the workshop mobile interaction with the real world in conjunction with the 8th International Conference on Human Computer Interaction with Mobile Devices and Services

[3] Broll, G., De Luca, a., Rukzio, E., Noda, C., Wisner, P., Cheverst, K., and Schmidt-Belz, B. (Eds.). 2007. Proceedings of the joint workshop mobile interaction with the real world in conjunction with the 9th International Conference on Human Computer Interaction with Mobile Devices and Services. Technical Report.

[4] Henze, N., Broll, G., Rukzio, E., Rohs, M. (Eds.). 2008. Mobile Interaction with the Real World: Workshop in conjunction with Mobile HCI 2008. BIS-Vlg. ISBN 3814221346.

LittleProjectedPlanet: An Augmented Reality Game for Camera Projector Phones

Markus Löchtefeld
Institute for Geoinformatics, University of Münster
Weseler Str. 253, 48151 Münster, Germany

Johannes Schöning and Antonio Krüger
German Research Center for Artificial Intelligence (DFKI)
Stuhlsatzenhausweg 3, 66123 Saarbrücken, Germany

Michael Rohs
Deutsche Telekom Laboratories, TU Berlin
Ernst-Reuter-Platz 7, 10587 Berlin, Germany

Abstract

With the miniaturization of projection technology the integration of tiny projection units, normally referred to as pico projectors, into mobile devices is not longer fiction. Such integrated projectors in mobile devices could make mobile projection ubiquitous within the next few years. These phones soon will have the ability to project large-scale information onto any surfaces in the real world. By doing so the interaction space of the mobile device can be expanded to physical objects in the environment and this can support interaction concepts that are not even possible on modern desktop computers today. In this paper, we explore the possibilities of camera projector phones with a mobile adaption of the Playstation 3™(PS3) game LittleBigPlanet™. The camera projector unit is used to augment the hand drawings of a user with an overlay displaying physical interaction of virtual objects with the real world. Players can sketch a 2D world on a sheet of paper or use an existing physical configuration of objects and let the physics engine simulate physical procedures in this world to achieve game goals.

1 Introduction and Related Work

Mobile phones are used for a wide range of applications and services in today's everyday life, but still they have many limitations. Aside from the lack of memory and processor power the small display size is one of the major bottlenecks. Digital projectors are shrunken to the size of a mobile phone. The next step is to integrate them directly into the mobile device. Such phones could overcome the shortage of the small screen and even make it possible to present large and complex information like maps or web pages without zooming or panning as presented by Hang et al. [11]. Up to now several prototypes have been presented, and the first series-production device is already up for pre-order. Considering the possibility of a phone with integrated camera and projector available in just a few months, still less research has been conducted to investigate the potential of such a mobile unit (in the following we use the term mobile camera projection units as a synonym for a camera projector mobile phone). We propose a mobile game combining hand drawn sketches of a user in combination with objects following a physics engine to achieve game goals (see Figure 1).

Initial research on mobile projection interfaces was conducted by Raskar et al. [13] followed up by Beardsley et al. [5] and Cao et al. [7]. Blasko et al. [6] explored the interaction with a wrist-worn projection display by simulating the mobile projector with a steerable projector in a lab. First mobile setups were presented by Hang et al. [11], Tamaki et al. [17] or Schöning et al. [15]. From this development a rich design space for mobile games could emerge. Actual mobile games are characterized by simple graphics and miniaturized input modalities. That is why many mobile games are just played when the user wants to overcome a period of unused time. With a built in projector not only the graphical resolution of the games can be increased, also the possibilities to develop mobile augmented reality games will improve. To create visual overlays for augmented reality games, in the past often head mounted displays where used [8]. This retrenched not only the comfort of the user it also limited the mobility. As a consequence of the display being attached to a single player, games using a head mounted display can only be played in multiplayer scenarios when using a large amount of hardware. Another common technique for dynamic overlays is to use the screen of the mobile device like a magic lens [14] and so be struggle again with the small

size and resolution. Moreover such a magic lens display is not really enjoyable to use with more than one player at the same time.

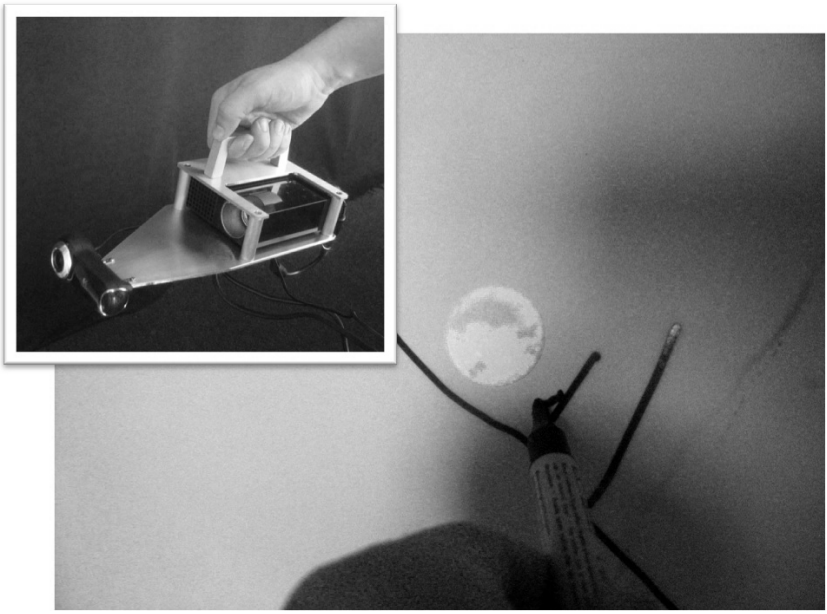


Figure 1: The LittleProjectedPlanet hardware prototype: (upper left corner). A user playing the game. He is sketching a marble run and projected tennis balls are bouncing on it (center).

Projecting a dynamic overlay directly onto a surface of the real world may enhance the playability even though it is hard to identify the overlay in bright light. Dao et al. have already shown first approaches for using a projected image in mobile gaming [9, 12] (but not in a mobile setting). In CoGame the players can steer a robot by connecting visual overlays with their mobile projectors, which contain parts of a path the robot should follow. With PlayAnywhere [18], Andrew Wilson demonstrated the possibilities of mobile camera projector units in mobile entertainment. It consisted of an easy to set up camera projector unit allowing the user to play games on any planar surface, which can be used as a projection area, by controlling the games with their hands. Enriching sketching in combination with physical simulation was presented by Davis et al. [4, 10]. The ASSIST system, was a

sketch understanding system that allows e.g. an engineer to sketch a mechanical system as she would on paper, and then allows her to interact with the design as a mechanical system, for example by seeing a simulation of her drawing.

In contrast to the related work in this paper we present a game called LittleProjectedPlanet that is designed for a mobile projector phones combing real world objects and projected ones utilizing a physics engine. We think that this kind of mobile projection camera unit can been utilized to improve the learning and collaboration in small groups of pupils (cause of the mobile setup of our prototype) in contrast to more teacher-centered teaching e.g. one interactive white board (as shown by Davis et al. [4, 10]).

2 Game Concept

The slogan of the popular Playstation 3 game LittleBigPlanet [2] by Media Molecule (some parts of the ASSIST sketch understanding system were used for the game) is "play with everything" and that can be taken literally. The player controls a little character that can run, jump and manipulate objects in several ways. A large diversity of pre-build objects is in the game to interact with, and each modification on such an item let them act in a manner physically similar to those they represent. The goal of each level is to bring the character from a starting point to the finish. Therefore it has to overcome several barriers by triggering physical actions. But the main fascination and potential of the game is the feasibility to customize and create levels. Creating new objects is done by starting with a number of basic shapes, such as circles, stars and squares, modify them and then place them in the level. Having done so, the user can decide on how these objects should be connected mechanically.

We took this designing approach as an entry point for a mobile augmented reality game using a mobile camera projector unit. It allows the user to design a 2D world in reality, which is then detected by a camera. Out of this detection a physical model is being calculated. Into this model the user can place several virtual objects representing items like tennis balls or bowling balls. These virtual objects then get projected into the real world by the mobile projector. When starting the physic engine, the application simulates the interaction of the virtual and the real world objects and projects the

results of the virtual objects onto the real world surface. Just like in LittleBigPlanet our application offers the user different ways of playing: One is like the level designer in LittleBigPlanet; the user can freely manipulate the 2D World within the projected area and place virtual objects in it. Similar to children building tracks for marbles in a sandpit, the player can specify a route and then let the virtual marbles running along it.

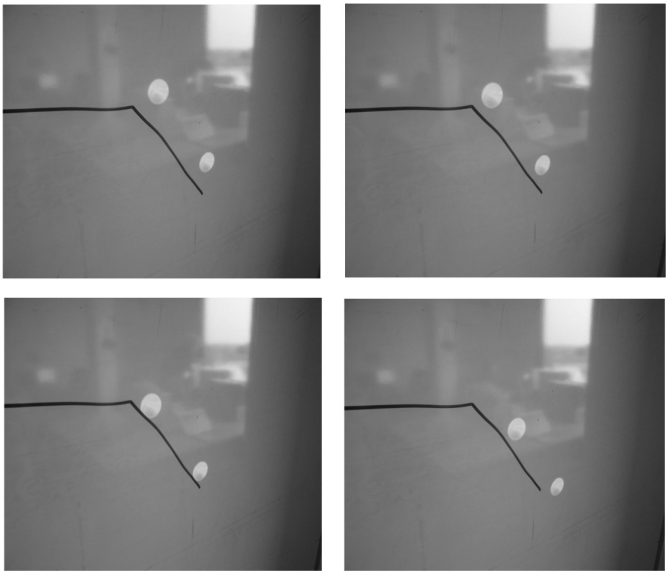


Figure 2: Projected tennis balls are bouncing on a run sketched by a user.

A different gaming mode is a level based modus, but instead of steering a character as in LittleBigPlanet, the user designs the world. As a goal the user has to steer a virtual object e.g. a tennis ball from its starting point to a given finish. The game concept uses a direct manipulation approach. Enabling the player to modify the world at runtime let the real world objects become the users' tangible interface. But not only the objects are used for the interface, by changing the orientation and position of the projector the user can also modify the physical procedures (e.g. gravity by turning the mobile camera projector unit).

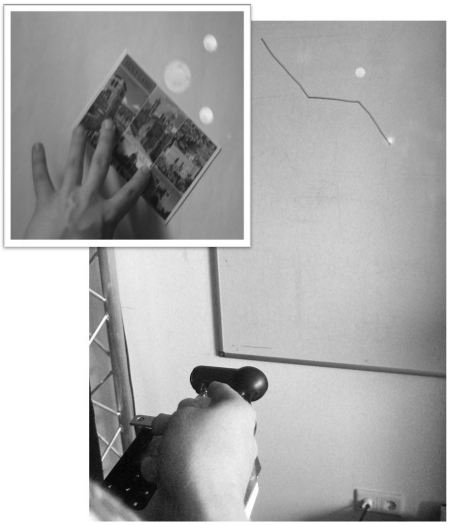


Figure 3: LittleProjectedPlanet game screenshot: A user playing the game with a postcard (upper left corner). User is sketching a marble run and projected tennis balls are bouncing on it (center).

3 Interaction Concepts

For designing a 2D world the players can use several methods. Basically they have to generate enough contrast that can be detected by using a standard edge recognition algorithm (utilizing the Sobel operator [16]). Sketching on a piece of paper or a white board for example can do this, but simply every corner or edge of a real world object could generate a useful representation in the physics engine. So there is no need for an extra sketching device or other for example IR based input methods. Just requiring the camera projector unit itself the game is playable nearly anywhere with nearly everything and it is easy to set up. Figure 3 show a user using a standard whiteboard as well as a user "playing with a postcard". An important problem to allow a smooth and seamless interaction for the user is that the \gravity in the projection" is aligned with the real worlds gravity. For that a Nintendo Wii is attached under the camera-projection unit (as can be seen in Figure 4 (left)). Also

gravity can be utilized in the game to control some action. A user can take control of the gravity by changing the orientation of the projector. Doing this the user can let virtual objects "fly" through the levels.

4 Implementation

Due to the unavailability of sophisticated projector phones (with an optimal alignment of camera and in-build projector) we used for our prototype a Dell M109S, a mobile projector with a maximum resolution of 800x600 and a weight of 360g, in combination with a Logitech QuickCam 9000 Pro. All together our prototype weighs around 500g and is therefore okay to handle (e.g. compared to the prototype used in [15] our prototype is "just 240g" heavier, but the projector has 50 lumen instead of just 10 and also has a larger resolution).

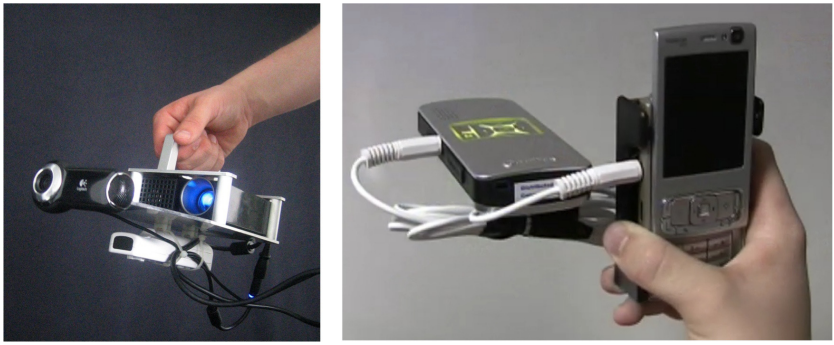


Figure 4: Different hardware prototpyes. Our current prototype (left) compared to the prototype used in [15] (right).

Table 1 compares some key characteristics of both prototypes. We think our prototype presented in this paper provides a good trade-off between mobility and sophisticated projection quality. In contrast to the few mobile devices with built in projectors, our projector and camera are mounted in such a way that the camera field of view fits nearly the projected area. But because of the different focal lengths of camera and projector in this setup the camera image is always wider than the projected image. Therefore the prototype needs a calibration to clip the right parts of the camera image. For controlling the

application and to determine the orientation (to set the gravity) a Nintendo Wii remote is attached to the camera projector unit. Most actual Smart Phones are already equipped with an accelerometer or an electronically compass, so the functionality of the Wii remote can easily be covered using a mobile phone. The application is fully implemented in Java using the QuickTime API to obtain a camera image. As a physics engine Phys2D [1], an open source, Java based engine is used. The communication with the Wii remote is handled by WiiRemoteJ [3]. Connected to a standard laptop or PC the camera projector unit has a refresh rate of approximately 15fps when running the application. Only the area of the camera image containing the projected image is processed via an edge recognition algorithm. This area is about one forth of the whole camera image of 640x480. Every pixel of a detected edge gets a representation as a fixed block in the physics engine. That gives the user total freedom in designing the world. Such a physic world update is done every 300ms but it can be stopped by the users, for example for editing the sketch. Adapting the gravity of the physical model to the actual orientation of the camera projector unit is done through calculating the roll[1] of the Wii remote. Up to now there is no correction on the projected image. In first preliminary user test we found out that this is not affecting the user experience. Several methods for image correction are already available (e.g. from Raskar et al. [13]), but are not implemented in the current prototype. The video http://www.youtube.com/watch?v=eCF2Q0w6hkg shows the game concept and our running prototype in different situations.

Characterics	LittleProjectedPlanet	Map Torchlight[15]
Weight	580g	340g
Lumen	50	10
Camera Res.	3 Megapixel	5 Megapixel
Projector Res.	800x600	320x249
Wireless	No	Yes

Table 1: Characteristics of the LittleProjectedPlanet prototype compared to the Map Torchlight prototype.

[1] This denotes the angular deviation along the longest axis of the Wii remote.

5 Conclusion and Future Work

We have introduced a mobile adaption of LittleBigPlanet for mobile camera projector units. The LittleProjectedPlanet augmented reality game shows the abilities and flexibility that a camera projector unit provides for mobile gaming. Expanding the interaction space to physical objects creates interaction techniques that are not possible on modern desktop computers. We think that these kinds of applications are helpful entertainment scenarios and in classroom settings and an informative user study is planned to evaluate the prototype. In addition we think that future of mobile gaming is definitely be influenced by the launch of camera phones with build in projectors. Especially the creativity in designing a world embodied by the nearly endlessly possibilities will be interesting to see. Definitely the approach of total freedom in the design space has its disadvantages. Also the detection of projected virtual objects in some strange lightning situations is an issue to work on. However the edge detection without any parameterization of the objects still seems to be the most flexible technique for a user to design a level without any restrictions.

6 References

[1] Phys2D Java based open source physic engine, http://www.cokeandcode.com/phys2d/, 2006. (Online; accessed 15-April-2009).

[2] LittleBigPlanet , http://www.littlebigplanet.com/, 2008. (Online; accessed 15-April-2009).

[3] WiiRemoteJ, http://www.world-of-cha0s.hostrocket.com/WiiRemoteJ/, 2008. (Online; accessed 15-April-2009).

[4] C. Alvarado and R. Davis. Resolving ambiguities to create a natural sketch based interface. In Proceedings of IJCAI-2001, 2001.

[5] P. Beardsley, J. Van Baar, R. Raskar, and C. Forlines. Interaction using a handheld projector. IEEE Computer Graphics and Applications, 25(1):39-43, 2005.

[6] G. Blasko, F. Coriand, and S. Feiner. Exploring interaction with a simulated wrist-worn projection display. In Ninth IEEE International Symposium on Wearable Computers, 2005. Proceedings, pages 2-9, 2005.

[7] X. Cao, C. Forlines, and R. Balakrishnan. Multi-user interaction using handheld projectors. In Proceedings of the 20th annual ACM symposium on User interface software and technology, pages 43-52. ACM New York, NY, USA, 2007.

[8] A. D. Cheok, K. H. Goh, W. Liu, F. Farbiz, S. W. Fong, S. L. Teo, Y. Li, and X. Yang. Human pacman: a mobile, wide-area entertainment system based on physical, social, and ubiquitous computing. Personal Ubiquitous Comput., 8(2):71-81, 2004.

[9] V. N. Dao, K. Hosoi, and M. Sugimoto. A semi-automatic realtime calibration technique for a handheld projector. In VRST '07: Proceedings of the 2007 ACM symposium on Virtual reality software and technology, pages 43{46, New York, NY, USA, 2007. ACM.

[10] R. Davis. Sketch understanding in design: Overview of work at the MIT AI lab. In Sketch Understanding, Papers from the 2002 AAAI Spring Symposium, pages 24-31, Stanford, California, March 25-27 2002. AAAI Press.

[11] A. Hang, E. Rukzio, and A. Greaves. Projector phone: a study of using mobile phones with integrated projector for interaction with maps. In MobileHCI '08: Proceedings of the 10th international conference on Human computer interaction with mobile devices and services, pages 207-216, New York, NY, USA, 2008. ACM.

[12] K. Hosoi, V. N. Dao, A. Mori, and M. Sugimoto. Cogame: manipulation using a handheld projector. In SIGGRAPH '07: ACM SIGGRAPH 2007 emerging technologies, page 2, New York, NY, USA, 2007. ACM.

[13] R. Raskar, J. van Baar, P. Beardsley, T. Willwacher, S. Rao, and C. Forlines. ilamps: geometrically aware and self-configuring projectors. In SIGGRAPH '06: ACM SIGGRAPH 2006 Courses, page 7, New York, NY, USA, 2006. ACM.

[14] O. Rath, J. Schöning, M. Rohs, and A. Krüger. Sight quest: A mobile game for paper maps. In Intertain, editor, Intertain 2008: Adjunct Proceedings of the 2nd International Conference on INtelligent TEchnologies for interactive enterTAINment, January 2008.

[15] J. Schöning, M. Rohs, S. Kratz, M. Löchtefeld, and A. Krüger. Map torchlight: a mobile augmented reality camera projector unit. In Proceedings of the 27th international conference extended abstracts on Human factors in computing systems, pages 3841-3846. ACM New York, NY, USA, 2009.

[16] I. Sobel and G. Feldman. A 3x3 isotropic gradient operator for image processing. Presentation for Stanford Artificial Project, 1968.

[17] E. Tamaki, T. Miyaki, and J. Rekimoto. Brainy hand: an ear-worn hand gesture interaction device. In Proceedings of the 27th international conference extended abstracts on Human factors in computing systems, pages 4255-4260. ACM New York, NY, USA, 2009.

[18] A. D. Wilson. Playanywhere: a compact interactive tabletop projection-vision system. In UIST '05: Proceedings of the 18th annual ACM symposium on User interface software and technology, pages 83-92, New York, NY, USA, 2005. ACM.

View & Share: Exploring Co-Present Viewing and Sharing of Pictures using Personal Projection

Andrew Greaves and Enrico Rukzio
Computing Department, Lancaster University
InfoLab21, South Drive, Lancaster, UK

Abstract

Co-present viewing and sharing of images on mobile devices is a popular but very cumbersome activity. Firstly, is it difficult to show a picture to a group of friends due to the small mobile phone screen and secondly it is difficult to share media, e.g. when considering Bluetooth usage; technical limitations and repetitive user interactions. This paper introduces the View & Share system allowing mobile phone users to spontaneously form a group. A member of the group has a personal projector (e.g. projector phone) which is used to view pictures collaboratively. View & Share supports sharing with a single user, multiple users or all users, allows members to borrow the projected display and supports a private viewing mode. The paper reports the View & Share system and an explorative user study with 12 participants showing the advantages of our system and user feedback.

Keywords: View & Share, projector phone, mobile interaction, personal projection, co-present.

1 Introduction

The purpose of viewing and sharing media is to communicate the experience with others. The co-present viewing and sharing of media provides communication of this experience between several people and often results in a collaborative task. Frohlich and several others suggest that sharing photos in this manner, face to face, is the most common and enjoyable [1]. The resulting collaboration between co-located people results in photo-talk. Here

photos are used as triggers to facilitate storytelling, reminiscing and to raise discussion within the group. Although this is highly desirable, it is very cumbersome and problematic to achieve with a mobile device. Typically the experience is conveyed to everyone by either gathering around a single mobile device or passing the device around the group. Although this satisfies the requirement of sharing, one could argue that this experience is not exploited to its full potential. The experience is of a distributed nature and not consumed by all simultaneously.

Kun et al. presented a prototype application that supports the sharing of photos between multiple devices [2]. In this situation the devices were synchronized to support co-present sharing between users. Although this is a great step in alleviating the need to pass the device around, the sharing semantics used here and also described in similar work require that content is shared with everyone in the group. The small screen issue is also still present. Similarly, Clawson et al. presented Mobiphos, a mobile photo sharing application that allows a co-located group of users to capture and simultaneously share photos with all in real time [3].

Alternative solutions solving the inherent small screen issue are to use large screens in the environment. Unfortunately such displays are neither readily available nor accessible, and certainly destroy the degree of portability that the mobile phone provides. Tabletops are another solution which support viewing of pictures by multiple people and provision easy user interaction. However, like public displays they lack availability, constrain the user to a certain environment are costly and lack portability.

In this paper we describe View and Share [4] and report the findings of an explorative user study whereby 12 participants evaluated View and Share in supporting the co-present viewing and sharing of media. By combing a mobile phone with a personal projection device the small screen issues are solved. This combination provides great opportunities for the mobile phone to be the dominant device in supporting mobile co-located viewing and sharing of media when compared to alternative approaches.

2 View and Share

The typical scenario which is addressed by the View & Share mobile application is supporting a group of friends who meet each other and want to view and share their pictures. View & Share supports quick and effective co-present viewing and sharing of media stored on the user's mobile devices. The combination of personal projectors and mobile phones allows several users to view pictures in a large format as opposed to the small limited mobile display.

The View & Share application encapsulates the sharing process and provides simple communication, co-ordination and media requests between single and multiple users. For this reason once a user is connected and thus a member of the group, media requests are handled automatically, transparent to the user and only require a single user interaction rather than multiple steps as is typically necessary. View & Share introduces two sharing interactions; presenter oriented and viewer oriented.

Within the group, users belong to one of two roles, either a presenter or a viewer. The presenter represents the user with the projector phone or a phone coupled with a personal projector who wishes to project their media for others to view and share. The presenter is the dominant role within the group. This principle is founded upon the presenter being the owner of the personal projection device and thus should always have full control. The viewers are the remaining users of the group. Their main role is to view the presenter's media and be recipients of the shared media.

Figure 1 illustrates the presenter oriented sharing technique. Here the presenter browses through their pictures which are then displayed to the group via the projection. The presenter can send a particular picture to all viewers through selecting the send to all function on the mobile device. The picture is then copied, displayed and stored on the viewer's mobile phones.

Figure 2 illustrates the viewer oriented sharing technique. Here the role of sharing is shifted and undertaken by the recipient of the media, the viewer. The viewer submits a request for the currently projected image. When received by the presenter the image is automatically sent to the viewer. Because the sharing originates from each viewer, sharing with a single viewer or multiple viewers is both possible and easily achieved.

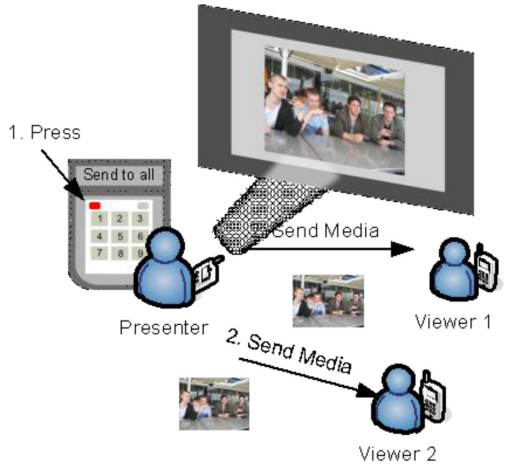


Figure 1. Presenter oriented sharing.

The presenter can switch to private mode in which the projection is not used. This mode can be used when viewing and sharing private pictures in a public setting. Using this mode, the presenter browses through their photos using their device and opts to share private photo(s). The photo is then displayed on all the viewer's devices. Here sharing is achieved by each user through viewing their private mobile screen but without users having to gather around a single screen as is currently required.

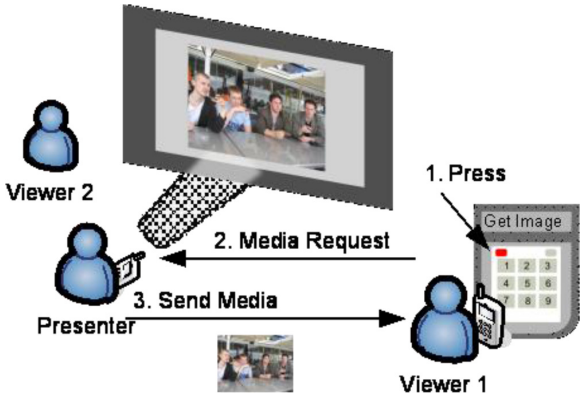


Figure 2. Viewer oriented sharing.

Viewers within the group can also temporarily borrow the projected display to allow them to view and share their media with others using the presenter's projector.

2.1 Implementation

Java ME (CLDC 1.1 / MIDP 2.0) was used to implement the View & Share application and was tested using three Nokia N95s. A battery powered mobile pocket projector (Samsung SP-P310ME) was used as the personal projector. We envision using projector phones in the future once they provide the necessary APIs and independent display support.

Figure 3 illustrates the system architecture. The resulting setup is similar to our previous work and allows independent control of the mobile screen and projection, this is not possible with the currently available projector phone [5]. The laptop is only used as a server allowing projection of content, which differs from the content displayed on the mobile phone screen. The server is not seen or used by View & Share users.

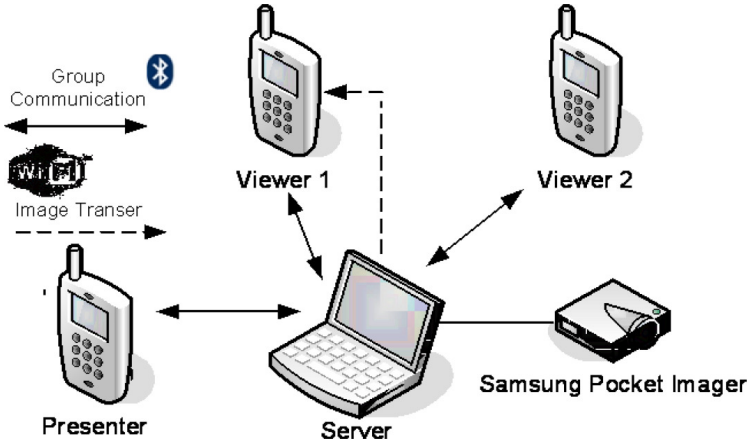


Figure 3. View & Share Architecture

Our approach allows the presenter to browse images on the mobile phone as a grid of thumbnails and at the same time allow others to view the current image but enlarged on the projection. Temporarily disabling the projection is also possible for the case of private viewing, here the projection is obscured.

Formation of the group and communication within (co-ordination and media requests) is achieved using Bluetooth and uses a lightweight message protocol. Members of the group must first connect to the presenter. This is achieved by performing a Bluetooth Device and Service Discovery operation. Message requests describe the origin of the message, the operating mode (presenter or viewer), the message function and current privacy setting (public or private). The Java APIs for Bluetooth (JSR 82) were used to facilitate communication within the group and the File Connection API (JSR 75) was used to access media content on the mobile phone. Pictures are transparently shared between devices using WiFi.

Figure 4 illustrates the presenter oriented (Figure 4 left) and viewer oriented (Figure 4 middle) sharing interactions. The received image, as a result of either interaction, is displayed on the viewer's mobile screen (Figure 4 right) and automatically saved.

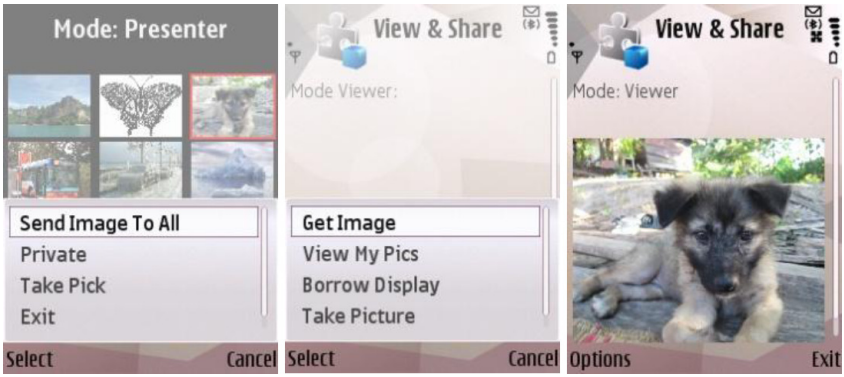


Figure 4. Sharing Interactions.

3 User Study

An explorative user study was conducted observing four groups of three friends using View & Share. The 12 participants were mainly students, all had prior computer experience to a high degree and owned a mobile phone, 11 of which had a camera. 9 of the participants were familiar with photo browsing software on mobile phones and 8 had prior knowledge of sharing

photos using Bluetooth. Participants were aged between 23 and 38 with a mean of 27 and consisted of 9 males and 3 females.

Members of each group were all friends and regularly participated together in social events and activities. Each member was asked to provide 30 to 50 photos to be used in the experiment. Backup photos were provided for members who failed to bring photos, this was applicable for two groups. For those who brought photo's it was mandated that at least some of the photo's included all members of the group.

By using groups of friends and familiar photo's we hoped to observe and capture realistic user behavior when using View & Share. We believed that the group experience would lead to collaborative "photo talk" as described by Frohlich [1]. Unlike previous research, we also wanted to observe users social behavior when browsing images in a group using a large projected mobile display, the impact this had on sharing (for example the effect the social setting has) and the benefits of sharing in this way. Usage behavior was also automatically recorded, this included the total number of sharing interactions and type (presenter or viewer oriented), the number of borrow requests, whether these were successful or not and the number of private viewing occurrences.

The experiment was conducted in two different social settings and locations. Group 1 was observed using the system in their own home during the evening. Here the social setting was more relaxed and members of the group indulged in snacks and a glass of wine during the experiment. The remaining groups performed the experiment in the Computing Department at Lancaster University during their working day.

Each group completed a short training phase whereby the functionality of View & Share was demonstrated and explained; participants were encouraged to ask questions. Following this the experiment began and participants were observed, for this purpose audio and video was recorded. The experiment was split into two halves. Firstly, participants were simply observed using View & Share for circa 15 minutes and were left to their own devices. The investigator stayed in the room to answer any questions or resolve any technical difficulties. The second half of the experiment was very similar, however members of the group were explicitly asked to complete certain tasks using View & Share with the intention of allowing every member to portray each of the two roles (viewer and presenter) and evaluate all functionality.

After completing both parts of the experiments each member of the group was asked to complete a short questionnaire containing selected questions from the IBM Computer Usability Satisfaction Questionnaire and the NASA Task Load Index. There were several questions regarding viewing and sharing habits and both their viewing and sharing preferences. Following this a group interview was conducted to elicit feedback, comments and suggestions.

4 Findings and Discussion

The total number of sharing interactions across all four groups was 129, 87 were viewer oriented and 42 presenter oriented. Between groups those who brought their own photos yielded the greater number of sharing interactions. Private viewing resulted in 25 occurrences and 22 cases of borrowing requests with 15 granted. The lowest mean score (1 = strongly disagree, 5 = strongly agree) for the IBM Computer Usability Satisfaction Questionnaire was 3.58 and the highest was 4.42. For the NASA Task Load Index the mean values were as follows: mental demand = 2.41, frustration level = 2, effort = 2.75 and performance = 4.25. For the latter the higher score closer to 5 represents the best case, for the first three the closer to 1 signifies the better the result.

Viewing media: All participants agreed that viewing photos using the projection when compared to the mobile phone was better. Participants commented that the big screen allowed simultaneous viewing experiences between all group members. One participant raised an issue that occurs when several people view a picture on a single mobile phone, which requires passing the device around. He commented that users have to typically wait whilst others are making comments, they are yet to see the picture and the excitement / suspense builds up as users wait while others make comments. However, when it is their turn to view they are often disappointed. A further participant commented, “The projection facilitates spontaneous interaction, everyone sees it without delay”. Benefits with regards to the increased size and resolution of the image were expressed but potential issues with ambient light and finding a suitable projection surface were also highlighted. Further comments included “Easier to view as a group, prefer the projection as it is a

natural response rather than a gradual one", "Fun times, fun to comment on together" and "Enables more interaction between the audience".

Sharing Media: Participants raised the following issues when using MMS and Bluetooth to send pictures to friends. MMS is expensive, but for some users MMS are free. One user commented that although she prefers our approach, MMS allows her to send to people far away. Several also commented that they have experienced problems when using Bluetooth, "it is error prone and can only send to a single recipient". All 12 participants agreed that they preferred using View & Share to support the sharing of media between friends when compared to the above methods. The following comments were made: "Very convenient to use", "never have used photo sharing software because it's a pain to setup" and "This method is more intuitive and direct, fulfilling the purpose very well".

Participants quickly grasped the concepts of the two sharing interactions, "The methods here are more straight forward". Although the presenter oriented interaction accounted for the lowest number of sharing interactions several commented, "Sending to all in the group is handy". This was also evident when viewing media privately. Further comments included "Methods used here are much more rapid in response and receiving and can upload numerous images at once in a very quick succession" and "Allows participants to get the images they wanted".

One participant in group 2 (the least number of sharing interactions) commented that he wouldn't really share images and was more likely to share videos in pubs, "YouTube videos are pretty funny". This participant also commented that he used Facebook to share images with others.

In both cases when viewing and sharing photos, several referred to Facebook as a means to achieve this and made suggestions to the presenter to add the photos to Facebook. In one instance the response to viewing a picture was, "Amazing, I think I might get that image, are you gonna put them on Facebook" and in another instance and by the same participant "That's cool, that's a well good photo, get that on Facebook".

It is unclear why the participant mentioned this, perhaps it was to bridge the gap between images on a mobile phone and those on the computer, which are uploaded to Facebook. However it is possible to upload pictures to both Facebook and Flickr via the mobile phone. From my personal experience I

continually see albums on Facebook with the title “phone pics” and typically these photos captured represent 'in the moment' experiences.

Communicating Experience: By using the projection to view images a high degree of sharing through viewing was achieved by all users simultaneously. Participants engaged in what Frohlich and many others describe as photo-talk; the reminiscing and storytelling of past events as a result of seeing a photo [1]. Participants enjoyed viewing media together, one user commented “Nicer with a group of people, provides spontaneous interaction, prefer to view in a group as people can discuss the photo and make comments”, another commented “It's kind of like a movie”. For the groups where all members provided photos a greater amount of, and more active communication occurred. In several occurrences discussion of a certain picture led to further discussion of another picture with participants actively engaged in describing and making comments about photos. Some of the members in group 1 had photos which were several years old. This resulted in reminiscing and comments which specifically mocked or embarrassed the individual(s) in the photo, some comments were even abusive but with no intent to offend. Group 1 appeared to be the most active group and were continually laughing, joking and having fun browsing the images. We believe the reason behind this was the social setting in which the experiment took place. Here it was about 8pm, participants were in their own home, appeared more relaxed and participated whist enjoying snacks and a glass of wine. This is representative of an idealistic setting at home. Here they were more actively involved when compared to the other groups who participated at work.

Borrowing: The majority of participants saw benefits in the ability to physically borrow the projected display. Comments included “That's cool” and “I LOVE to borrow”. One participant, who happened to be the eldest, made the following interesting comment that no one else raised, “Projector allows retention of personal property, no invasion of personal space or risk of personal data been seen”. This applies both to borrowing the projection and also to viewing media via the projection. There were however, some reservations with the process of borrowing the projected display. These included acts which could potentially occur when borrowing and what happens to the media content once the display is relinquished. Users debated whether the submission of a request to first borrow the display was necessary. The idea of being able to push content to the display automatically

“why would you ask, we are all friends”, or having a reserved area of the projection for each viewer whilst simultaneously supporting the viewing of multiple media sources, seemed appealing. Here the following scenario could then be satisfied, “I've got something to show you, look at this”. Further research into social protocols would be necessary to find a true answer to this question in this context of use. Two further issues arose concerning the validity of the borrowed content on the presenter's phone and situations in which the borrower projects unsuitable or embarrassing photos using the presenter's mobile phone. In the latter case participants did not assume that just because they were friends they wouldn't project inappropriate content, it may seem appealing to do so.

Privacy: When viewing media privately two cases emerged; Firstly participants would change to private mode, locate the image on the mobile phone and then use the presenter oriented technique to send the photo to the mobile device of each user. Here the photo remains private to passersby's. In the second case the presenter entered private mode, located the image and then made the display public allowing everyone to view the large image. This allowed the presenter to find the correct image without publicly browsing through their images, which they would not wish to do. The reason for this (also mentioned by several participants) is the typical flat file approach and a lack of a storage hierarchy for media content on mobile devices.

5 Conclusion

This paper evaluated View & Share in supporting the co-present viewing and sharing of pictures. All 12 participants preferred our approach to view and share media co-presently. The large projected mobile display facilitated simultaneous group viewing of pictures by all group members, which is currently not possible using a single mobile device. Furthermore, face-to-face in the moment experiences occurred and were shared amongst the group through viewing, resulting in active discussion and further enhanced the viewing experience. Support for sharing with a single member, multiple members or all members is easily achieved requiring only a single user interaction. Observations led us to believe that the social setting and relevance of the projected content specific to the users within the group, impacts users viewing and sharing behavior. In a more relaxed setting for

example the users home, an increased amount of photo-talk occurred and comments were of various natures: embarrassing, mocking, descriptive and in general participants seemed to have more fun in physically sharing the experience with others.

Future work shall explore supporting other forms of media, how we can better provision for multiple users, supporting further interactions and continuing to enhance the viewing experience.

Acknowledgement

This work is supported by the NoE InterMedia, funded by the European Commission (NoE 038419).

6 References

[1] Frohlich, D., Kuchinsky, A., Pering, C., Don, A., and Ariss, S. Requirements for photoware. CSCW '02. ACM. New Orleans, Louisiana, USA

[2] Ah Kun, L. M. and Marsden, G. Co-Present Photo Sharing on Mobile Devices. Mobile HCI 2007. ACM. Singapore.

[3] Clawson, J., Voida, A., Patel, N. and Lyons K. Mobiphos: A Collocated-Synchronous Mobile Photo Sharing Application. Mobile HCI 2008. ACM. Amsterdam, Netherlands.

[4] Greaves, A. and Rukzio, E. View & Share: A Collaborative Media Viewing and Sharing Framework for Projector Phones. In: MIRW 2008. Amsterdam, Netherlands.

[5] Gadget Graver. World's First Video Projector Mobile Phone Epoq EGP-PP01. http://www.gadgetcraver.com/ videoprojectormobilephone-p-198.htm

Supporting Hand Gesture Manipulation of Projected Content with Mobile Phones

Matthias Baldauf and Peter Fröhlich
Telecommunications Research Center Vienna
Donau-City-Strasse 1, Vienna, Austria

Abstract

The detection of a user's hand gestures enables a natural interaction with digital content. Recently, wearable gesture detection systems have been presented which use a camera to visually detect the gestures and tiny projectors to augment nearby surfaces and real-world objects with digital information. Still, current approaches rely on laptop computers restricting the systems' mobility and usability. In this paper, we present a framework for spotting hand gestures that is based on a mobile phone, its built-in camera and an attached mobile projector as medium for visual feedback. Other existing mobile applications can simply connect to our framework and thus, become gesture-aware. The proposed framework will allow us to easily and fast create gesture-enabled research prototypes shifting the user's attention from the device to the content.

1 Introduction

Mobile phones are the most widespread ubiquitous devices. Due to their inherent context-awareness [2], they are increasingly used to interact with the user's current surroundings and nearby real-world objects. Sensors integrated in today's mobile phones such as GPS receivers, compasses, accelerometers, NFC modules and cameras not only allow to view digital information about such objects but also to manipulate it. Previous research has proven the feasibility and usability of several interaction techniques. Examples include short-range techniques such as touching an NFC-enabled object with a

mobile phone [16] and medium- and wide-range techniques such as pointing the device at buildings [19].

More recently, demonstrations of MIT's project "6th sense" [11], a mobile gestural-controlled system, have attracted considerable interest. The portable combination of a common webcam, a laptop computer and a tiny projector (see Figure 1) allows the augmentation of arbitrary surfaces and objects by projected information while triggering actions through natural hand gestures. The visual detection of a user's gestures causes the involved computer to vanish into the background. Still, the system relies on the laptop computer which has to be carried in a backpack when used on the move.

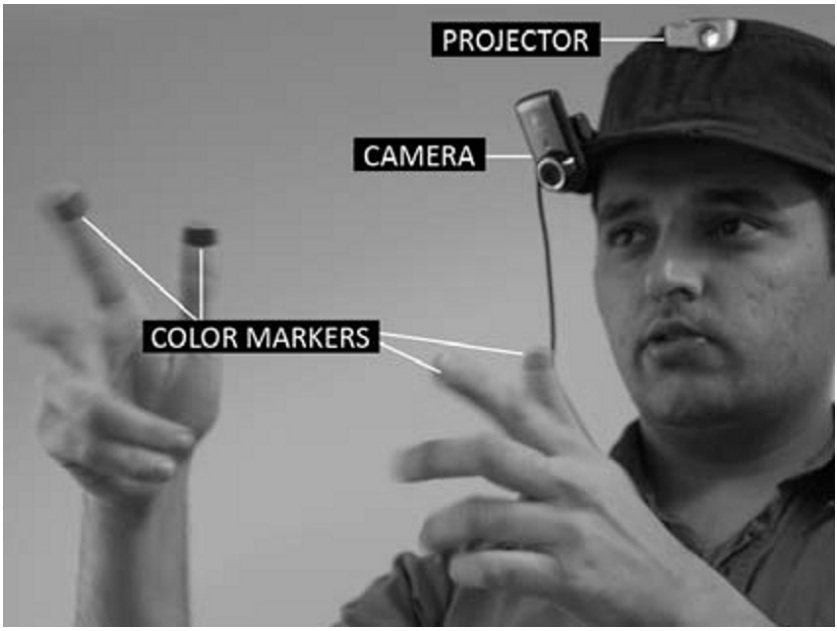


Figure 1: Setup of MIT's project "6th sense" [11]

Inspired by this work and with emerging projector phones in mind, we develop a framework supporting hand gesture manipulation of projected content through a mobile phone. Our aim is to make the mobile phone a wearable, truly unnoticeable mediator between the real and the virtual world changing the human interaction style from a device-centric over to a content

centric one. Existing mobile applications can simply connect to the proposed framework and thus, made gesture-aware. In this paper, we present our current work on this highly mobile and autonomous gesture detection framework. After giving an overview of related work in Section 2, we describe our hardware setup in Section 3. In Section 4 we describe the chosen event and gesture model. Section 5 explains the proposed system's architecture and provides implementation details. We conclude with an outlook in Section 6.

2 Related Work

In the past few years, gesture-based interfaces made their way to market-ready or even mainstream products. Examples include Microsoft's Surface [10], a table with a touch-sensitive top responding to hand gestures and real-world objects, and Apple's iPhone [1], a multi-touch-enabled Smartphone. Besides such devices that have to be physically touched for interaction, impressive touchless appliances emerge. E.g. Ubiq'window [8] enables gestural interaction with a screen behind glass through optical motion detection. g-speak [14] uses special sensor gloves for detecting spatial hand gestures. However, these applications rely on expensive custom hardware and are not mobile.

Research in the field of mobile computing also investigated the usage of acceleration sensors to detect manual gestures [9]. One of the favored use cases is the interaction with connected real-world objects such as distant displays through a gesture-aware mobile phone [4]. Another possibility to detect a phone gesture is to analyze the built-in camera's video stream [7]. With the enhancements in mobile hardware, more complex computer vision algorithms can be realized on smart phones leading to handheld augmented reality applications [17][15].

As a visual feedback medium, mobile projectors are increasingly applied. Examples include work augmenting real-world items such as maps with overlaid digital data [18] and studies evaluating the usability of such extended displays [6].

Recently, the aforementioned project "6^{th} sense" [11] combined visual gesture detection methods performed on a laptop computer with projector

feedback. The resulting wearable device built from off-the-shelf components visually augments objects the user is interacting with. An example for a similar but custom hand gesture interaction device is "Brainy Hand" [5], a small gadget that comprises an earphone, a color camera, and a mini-projector and is attached to one ear.

3 Hardware Setup

For our purely mobile-based setup, we attached a tiny projector to a Smartphone simulating upcoming mainstream projector phones. Figure 2 depicts the hardware components of our current setup.

We use a Nokia N95 mobile phone, a Smartphone running Symbian OS with the S60 platform. Due to its multitasking capability and its built-in camera, we are able to execute both the gesture tracking engine and the gesture-enabled application on one device. Lots of mobile research prototypes - applications we want to gesture-enable - are implemented using the Java 2 Micro Edition which is featured by Symbian OS. Alternative Smartphone operating systems are not suitable for our approach: e.g. the iPhone lacks multitasking support and is only scarcely used in research projects, and so far, none of the available Android-powered phones provides a video output.

In order to augment nearby surfaces or objects we apply the pocket projector PK101 from Optoma. This LED-based projector with similar dimensions as the N95 is perfectly suited for mobile usage and is connected to the phone through a short video cable.

Such an assembled gadget can be worn like pendant, i.e. both devices are arranged along a lanyard worn around the neck. In contrast to the setup in the "6h sense" project [11], the lanyard contains the complete equipment, there is no additional backpack needed. For alternative perspectives, only the camera phone can be attached to the lanyard and the projector is integrated in a hat or the user’s clothing.

Figure 2: A N95 mobile phone and a PK101 projector assembled to a wearable gadget

Thus, the presented equipment is highly mobile and easy to use: No backpack for a laptop computer is necessary, no annoying cables are involved. The equipment consists only of off-the-shelf components and is completely autonomous.

4 Events and Gestures

In order to ease the detection of gestures and identify single fingers, we attach colored markers to the user's fingers. During the detection process, we distinguish between low-level events and gestures as a combination of such events. Our current prototype is capable to recognize the following three low-level events.

– *Marker detected.* This event occurs when a marker is detected in a video frame but the same marker was not present in the previous frame, i.e. this marker just appeared. The event's parameters include the color of the detected marker as well as the position in pixels where the marker has appeared.

– *Marker moved.* When a marker was present in the previous frame and is detected at another location in the current frame, this event is triggered. Again, the event's parameters contain the spotted marker's color and its current location.

– *Marker lost.* A formerly detected marker can not be recognized in the current video frame, i.e. the marker disappeared. This event only owns one parameter, namely the color of the lost marker.

Based on these three fundamental events we define several gestures. Such gestures combine at least two low-level events and abstract from pixel-sensitive positions forming more meaningful high-level actions. Gestures can be either absolute or relative ones. Absolute gestures directly operate on the displayed information, e.g. by pointing at a shown photo to select it. Thus, some kind of calibration is necessary for absolute gestures in order to map camera-detected positions to display-coordinates. At the moment, our current prototype features relative gestures, i.e. gestures derived from the motion and geometric relation of the involved markers.

We named the implemented gestures according to their most likely uses.

– *Panning.* This gesture involves only one marker. The panning occurs when a formerly detected marker moves. Besides the type of marker, this gestural event informs about the relative movements on the horizontal and vertical axis.

– *Scaling.* This well-known gesture measures the distance between two formerly spotted markers. By moving the markers closer or farther away from each other, scaling or zooming actions might be triggered. The appropriate parameters include the types of the two involved markers as well as the relative distance between them with regard to the first measured distance.

– *Rotating.* Similarly to scaling, this gesture is based on the interaction of two spotted markers. Instead of the distance, the slope of the line defined by the markers is measured. The parameters again consist of the two involved markers and the change of the slope in degrees. This value is relative to the initially detected slope. Obviously, the scaling and rotating gestures can be combined by changing both the marker's distance and orientation at the same time.

So far, we implemented gestures making use of a maximum of two markers. The tracking engine can be extended to detect more markers and thus, support more complex gestures.

5 System Architecture

Figure 3 gives an overview of our approach's software architecture. It consists of two main components where one element is responsible for the actual tracking of gestures and the other one triggers the according actions as part of the application to be gesture-enabled.

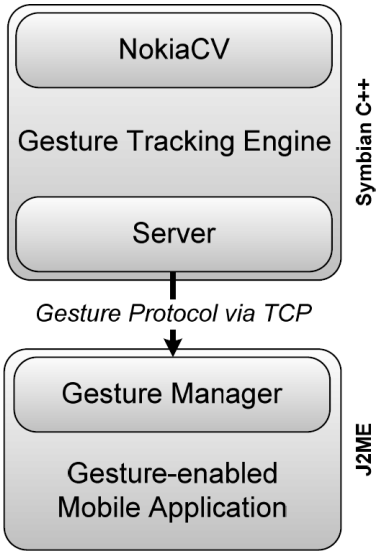


Figure 3: Architecture consisting of the Gesture Tracking Engine and the gesture-enabled application communicating via a local socket connection

In order to run computational intensive operations such as image recognition algorithms at interactive frame rates on resource-limited devices like mobile phones, they have to be implemented in native code. We based our gesture tracking engine on NokiaCV, a computer vision library written in Symbian C++. NokiaCV comes with source code and provides standard image operations and basic image recognition methods such as implementations of

corner or edge detection algorithms. We adapted some of the algorithms to analyze the video stream provided by the built-in camera. The presented gesture tracking engine uses a color-sensitive detection technique and therefore, is able to recognize and track differently colored markers in the viewfinder's image.

Once, a video frame has been completely analyzed, the tracking engine determines the occurred events described in Section 4. The according gestures are then derived by comparing the events to the ones detected in the former frame.

To notify another local application about spotted actions, the gesture tracking engine contains a small server component. As we only allow one client application to connect to this server, complexity of client management is reduced. For conveyance, the events and gestures are wrapped in a simple "gesture protocol", i.e. a short textual description of the events and gestures together with their particular parameters. An application might not only be interested in gestures but also in low-level events. E.g. an application might provide an acoustic signal to indicate a "marker detected" event - which usually marks the starting point of a gesture.

Obviously, any local application may connect to the gesture tracking engine, independent of the language it is written in. For a J2ME application to become gesture-aware, we provide the so-called "Gesture Manager". On initialization, this J2ME component connects to the gesture tracker engine and waits for incoming notifications to unwrap. Following the well-known observer pattern, a gesture listener has to be provided to the Gesture Manager. This gesture listener describes the operations to be triggered when a certain event or gesture is detected. In case no connection to the tracking engine could be established, the listener is ignored and the application's behavior is unmodified.

As an example, Figure 4 shows a mobile 3D urban exploration tool developed in our project 'WikiVienna' [3]. The application was made gesture-aware through the presented framework. We use the three supported gestures to move the point of view, to zoom in and out, and to change the viewing angle.

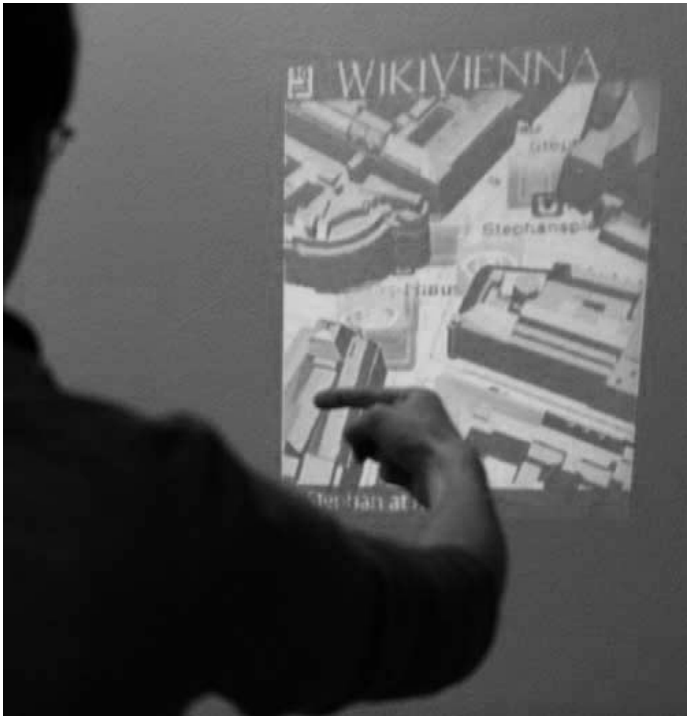


Figure 4: Controlling a mobile 3D urban model through a panning gesture

6 Conclusion and Outlook

In this paper we presented our ongoing work on a gesture detection framework for mobile phones. The framework aims at easily adding gestural interaction support to existing mobile application and, respectively, enables the rapid development of gesture-aware research prototypes. Our current framework prototype is deliberately designed for experimentation.

Future work will include the implementation of absolute gestures to directly operate on the projected content. Therefore, appropriate calibration and mapping techniques have to be developed.

A general problem when using a visual detection method and a projector as feedback medium, are the light conditions. Whereas the projected image can be recognized best in a rather dark setting, the visual detection works best in a well-illuminated ambience. Therefore, we will try to improve the robustness of our detection approach making it as illumination-invariant as possible in future work. The implementation of more sophisticated computer vision algorithms might even allow marker-less gestural interactions.

Acknowledgements

This work has been carried out within the projects WikiVienna and U0, which are financed in parts by Vienna's WWTF funding program, by the Austrian Government and by the City of Vienna within the competence center program COMET.

7 References

[1] Apple iPhone. http://www.apple.com/iphone/

[2] Baldauf, M., Dustdar, S., and Rosenberg, F. 2007. A survey on context-aware systems. Int. J. Ad Hoc Ubiquitous Comput. 2, 4 (Jun. 2007), 263-277.

[3] Baldauf, M., Fröhlich, P., and Musialski, P. 2008. WikiVienna: Community-Based City Reconstruction. IEEE Pervasive Computing Magazine, Vol. 7, No. 4.

[4] Dachselt, R., and Buchholz, R. 2009. Natural throw and tilt interaction between mobile phones and distant displays. In Proceedings of the 27th international Conference Extended Abstracts on Human Factors in Computing Systems (Boston, MA, USA, April 04 - 09, 2009). CHI EA '09. ACM, New York, NY, 3253-3258.

[5] Emi, T., Takashi, M., and Jun, R. 2009. Brainy hand: an ear-worn hand gesture interaction device. In Proceedings of the 27th international Conference Extended Abstracts on Human Factors in Computing Systems (Boston, MA, USA, April 04 - 09, 2009). CHI EA '09. ACM, New York, NY, 4255-4260.

[6] Greaves, A., Hang, A., and Rukzio, E. 2008. Picture browsing and map interaction using a projector phone. In Proceedings of the 10th international Conference on Human Computer interaction with Mobile Devices and Services (Amsterdam, The Netherlands, September 02 - 05, 2008). MobileHCI '08. ACM, New York, NY, 527-530.

[7] Hannuksela, J., Sangi, P., and Heikkilä, J. 2007. Vision-based motion estimation for interaction with mobile devices. Comput. Vis. Image Underst. 108, 1-2 (Oct. 2007), 188-195.

[8] LM3LABS Ubiq'window. http://www.ubiqwindow.jp

[9] Mäntyjärvi, J., Kela, J., Korpipää, P., and Kallio, S. 2004. Enabling fast and effortless customisation in accelerometer based gesture interaction. In Proceedings of the 3rd international Conference on Mobile and Ubiquitous Multimedia (College Park, Maryland, October 27 - 29, 2004). MUM '04, vol. 83. ACM, New York, NY, 25-31.

[10] Microsoft Surface. http://www.microsoft.com/surface/

[11] Mistry, P., Maes, P., and Chang, L. 2009. WUW - wear Ur world: a wearable gestural interface. In Proceedings of the 27th international Conference Extended Abstracts on Human Factors in Computing Systems (Boston, MA, USA, April 04 - 09, 2009). CHI EA '09. ACM, New York, NY, 4111-4116.

[12] Mobile Spatial Interaction Initiative. http://msi.ftw.at

[13] Nokia Computer Vision Library.
http://research.nokia.com/research/projects/nokiacv/

[14] Oblong g-speak. http://oblong.com/

[15] Rohs, M., Schöning, J., Krüger, A., and Hecht, B. 2007. Towards Real-time Markerless Tracking of Magic Lenses on Paper Maps. In Adjunct Proceedings of the Pervasive 2007.

[16] Rukzio, E., Broll, G., Leichtenstern, K., Schmidt, A. 2007. Mobile Interaction with the Real World: An Evaluation and Comparison of Physical Mobile Interaction Techniques. European Conference on Ambient Intelligence (AmI-07). Darmstadt, Germany.

[17] Schmalstieg, D., and Wagner, D. 2008. Mobile Phones as a Platform for Augmented Reality. Proceedings of the IEEE VR 2008 Workshop on

Software Engineering and Architectures for Realtime Interactive Systems (Reno, NV, USA), pp. 43-44, IEEE, Shaker Publishing, 2008-March

[18] Schöning, J., Rohs, M., Kratz, S., Löchtefeld, M., and Krüger, A. 2009. Map torchlight: a mobile augmented reality camera projector unit. In Proceedings of the 27th international Conference Extended Abstracts on Human Factors in Computing Systems (Boston, MA, USA, April 04 - 09, 2009). CHI EA '09. ACM, New York, NY, 3841-3846.

[19] Simon, R. and Fröhlich, P. 2007. A mobile application framework for the geospatial web. In Proceedings of the 16th international Conference on World Wide Web (Banff, Alberta, Canada, May 08 - 12, 2007). WWW '07. ACM, New York, NY, 381-390.

[20] Wang, J., Zhai, S., and Canny, J. 2006. Camera phone based motion sensing: interaction techniques, applications and performance study. In Proceedings of the 19th Annual ACM Symposium on User interface Software and Technology (Montreux, Switzerland, October 15 - 18, 2006). UIST '06. ACM, New York, NY, 101-110.

Magnification for Distance Pointing

Ferry Pramudianto and Andreas Zimmermann
Fraunhofer Institute for Applied Information Technology
Schloss Birlinghoven, 53754 St. Augustin, Germany

Enrico Rukzio
Computing Department, Lancaster University
InfoLab21, South Drive, Lancaster, UK

Abstract

We see currently the trend of having larger and larger displays in our living rooms and that more and more computer oriented applications which are usually controlled by a mouse are displayed by them. Examples for the latter are web browsing, picture browsing, chatting and email. This leads to two problems. Firstly, most applications and web pages are not designed for situations in which the user is sitting in a relative large distance from the display. The user is therefore often not able to read the text, to interact easily with buttons or to click on hyperlinks. Secondly, it is not appropriate to use the mouse as the input device as flat surfaces are typically not in reach when sitting on the sofa in the living room. This paper investigates firstly whether direct pointing would be a suitable interaction concept and secondly whether the usage of magnifiers helps the user when interacting from a distance. The paper reports a study comparing three different magnification techniques for direct pointing interaction with remote screens. The results provide clear evidence for the advantages of such interactions, especially when combined with linear and Fish Eye magnifications.

Keywords: Pointing, remote pointing, remote interaction.

1 Introduction

We see nowadays the trend of having larger and larger high resolution TV screens in our living rooms. Those devices are more and more used for PC applications such as web browsers, media players and file managers.

Many hardware vendors have offered so-called media Set-Top-Boxes such as Apple TV and Acer Set-top-Box, which enable people, not only to watch TV and DVDs, but also to browse the Internet and use computer applications on their TVs. Furthermore, many vendors also try to integrate conventional home entertainment devices such as TV and radio receiver into PCs and even provide them with software (e.g. Windows Media Center) that makes them similar to home entertainment systems.

The currently available remote controls are not suitable when considering interactions with web pages and media players that have many more and different user interface elements when compared with a standard TV user interface.

The main obstacle of adopting mouse and keyboard to a home entertainment system is that those input devices are designed to be used in a working environment which is deployed on a desk, where a keyboard and mouse can be used properly. In the contrary, people watch TV in their living room from a TV viewing distance, and most likely, while they sit on their sofa. Lorenz et al. study proved already that without a desktop like environment, using a mouse and keyboard in the setting is not suitable [1].

The second problem is that text and buttons of PC and web applications appear too small when considering that the user is sitting relatively far away from the TV in the living room. Users have problems to recognize the displayed content (e.g. Web page) and make mistakes because they can't see the labels or links clearly when interacting from a distance.

This work focuses firstly on pointing tasks in this context and assumes that remote pointing is a promising solution for this case since TV users are used to point at their TV with their remote controls anyway. Direct pointing has also certain advantages over indirect pointing techniques such as touch pad, track ball and mouse because it maps directly the hand or arm direction to the location on the screen.

The paper focuses secondly on the usage of magnifiers that magnify where to user points to in order to help her to read and interact with displayed information. Three different magnifiers were implemented and evaluated in a comparative study.

The paper is structured in the following way. Firstly we relate our work when compared with others. Following this, we discuss the three different magnifiers used within our study. The next section discusses our study design which is followed then by a report of the study results. Finally we analyze our findings and discuss future work.

2 Related Work

There has been a lot of research regarding pointing from a distance. Olsen and Nielsen introduced the idea of using a laser pointer for interactions with a remote display whereby a camera was used to track the position of the laser pointer [2]. Myers et al. conducted research that was aimed to inform the design of laser pointers used for distance interactions [3]. They analyzed the impact of the delay which is caused by the time the system needs to track the pointer, studied the jitter caused by hand unsteadiness and compared different laser pointers. The best laser pointer in their study was the heaviest one as this one had the least jitter due to unintended hand movements.

MacKenzie and Jusoh evaluated two input devices for remote pointing with a standard mouse as the baseline condition [4]. Their research shows clearly the advantages of the standard mouse when compared with direct pointing interactions.

Our research assumes that the usage of a standard mouse in a living room is an unrealistic assumption as there is often no flat surface close to a chair or sofa available.

This argument is also supported by Lorenz et al. who performed a user study about remote interactions with home media applications [1]. They used a similar environment setting as used in this work and compared interactions for internet browsing using a wireless mouse & keyboard, PDA stylus & virtual keyboard and PDA joystick & physical PDA keyboard. As the setting was on a chair without a desk, the user had to use the wireless keyboard and

the mouse without a flat surface. The results showed that mouse & keyboard were not suitable for this setting.

Research conducted by Freeman and Weissman proposed a television controlled by hands gestures [5]. Their system used image processing for detecting hand gestures. The disadvantage of his system is that the user interface needs to have large widgets to accommodate the hand size and is therefore not suitable for complex user interfaces or interactions with web browsers or media players.

Many direct pointing devices available and become more and more popular. Examples for this are the Nintendo Wii Controller (Wiimote), GyroPoint and RemotePoint. Pointing with a laser pointer has the problem that the pointer moves unstable due to unintended hand jitter. The Nintendo Wii overcomes this problem by using large UI elements. However, large widgets cannot be contained in a small space and therefore don't provide a solution for existing UIs.

3 Magnification for Remote Interaction

The research presented in this paper addresses two problems. The first one is the issue of not being able to use a conventional desktop mouse for pointing interaction with a remote screen. The second is the issue that web pages and PC application rendered on a remote screen are difficult to read and interact with. The reason for that latter is that there were designed for desktop use and not for interaction at a distance in a living room.

This work introduces magnifiers for direct remote interaction to overcome both problems. Three magnifiers were implemented and evaluated in a comparative study. The first and second one magnify the area around the mouse pointer with a linear (Figure 1a) and a hybrid fisheye transformation (Figure 1b) respectively. The third one magnifies the widget beneath the mouse pointer, as depicted in Figure 1c.

The linear magnifier gives very good detail information of the location the remote control points at. On the other hand it breaks the relationship between the magnified and the neighboring area through which the user loses orientation when moving the attention from the focus to the global context.

The advantage of the fisheye magnification is that it enhances the localized detail while preserving the continuity of transition to the global context. On the other hand, a continuous distortion over the image misleads the orientation of the focus. Thus, hybrid fisheye tries to combine the advantages of linear magnification and fisheye by transforming the surrounding area of the focus gradually and linearly magnifies the focus area. The advantage of this approach is that the users still get the continuity of the relationship between the magnified dimension and the non-magnified dimension while the focus area is not distorted.

The third approach magnifies the UI widgets such as buttons and toolbars with which the user interacts with. The advantage of this approach is that the magnified dimension does not move following the cursor so that the users have a more steady magnified area.

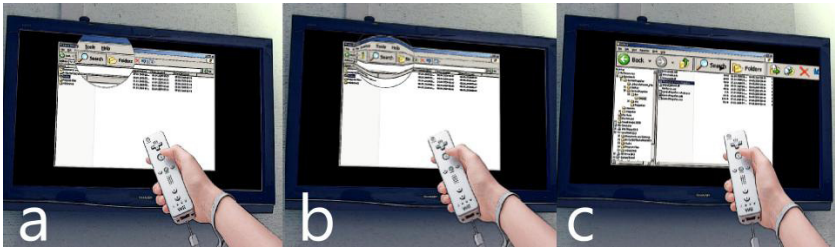


Figure 1. (a) linear magnification (b) hybrid fisheye magnification and (c) widget magnifier.

4 Implementation

The Nintendo Wii game controller (Wiimote) was used as the direct pointing device for remote interactions with a TV screen (42inch, 1024x768 pixel). The Wiimotelib library was used to determine the mouse pointer location on the screen. Using Wiimotelib, we could access the values of the IR camera of the Wiimote that was used to determine the mouse pointer location on the screen. The IR camera sends the location of the captured IR lights of the sensor bar attached to the TV. As a user points at the screen using Wiimote, the position of the mouse pointer is calculated based on the Wiimote's camera values.

To visualize the magnifier, the algorithm calculates a clipping area as big as the magnifier size on the surrounding area of the pointer. Then, it copies the pixels and interpolates these pixels based on the magnifier type. A more detail explanation of these interpolation algorithms can be found in [6].

To overcome hand's noise movement, we introduced a filter function. The filter function uses a dynamic averaging that works by measuring the pointer location's changes. If the changes are small, it averages a number of previous locations. If the changes are big then the user moves the pointer rapidly. Therefore it averages only a few previous locations. If the changes are very small the pointer should not be moved because this is normally unintended movement.

The widget magnifier works differently than the previous magnifiers (Figure 1c). It magnifies the widget that is currently under the mouse pointer by copying the whole area of the widget. Then it interpolates the copied area, uses it as a background of a transparent window, and then places this window on top of the original widget. When the user performs any mouse event, it is rerouted onto the original widget. When the pointer moves to another widget, the previous magnifier window is destroyed, and the whole process of visualizing a magnifier is repeated. The drawback of this method is if the user moves rapidly and crossing several widgets, then the all of these widgets would be magnified one by one.

5 Experiment

The purpose of evaluation was to test these following hypotheses:

- H1: The usage of magnifiers will reduce the error rate for selecting small targets in a home entertainment setting.
- H2: The usage of magnification will reduce the task completion time for practical tasks such as web browsing or interactions with media players.
- H3: Magnification leads to a higher selection time since targets seem to be moving faster in the magnified area. Therefore the user has to readjust her movement speed.

The experiment was conducted in a living room like setting where a sofa was placed 3.5 meter away from a 42inch TV which is mounted on the wall

(Figure 2). The study used a within-subject design with 12 right-handed participants whose ages ranged from 17 to 29 with a mean age of 25 years. The participants had different professional backgrounds.

Figure 2. Room used for experiment, shows used TV and sofa on which the participants sat.

The participants performed two different tasks. First a Fitts's law tapping task and then a practical task. In the tapping task, four widths of icon width (16, 32, 48, and 64 pixels) and three distances (100, 300, and 704 pixels) were used. The comparison of the sizes and distances is depicted in Figure 3. The tapping task followed ISO9241-9 standard for evaluating multidirectional tapping. ISO9241-9 describes the ergonomic requirements for office work with visual display terminals. Each participant hit 13 targets for every possible combination of magnifier (without magnifier, hybrid magnifier, linear magnifier and widget magnifier), width and distance. The sequence of interaction technique, width and distance was counterbalanced using a balanced Latin square algorithm in order to avoid any learning and exhaustion affects.

The practical task was conducted directly after the tapping task. The participants had to check their emails on Gmail and had to find particular news items on the BBC webpage. Furthermore they had to create a playlist and to play a particular song using the Windows Media Player. Both tasks were performed using the four different interaction techniques (without magnifier, hybrid magnifier, linear magnifier and widget magnifier) and completion times were recorded. Different tasks and counterbalanced sequences were used to avoid any training effects.

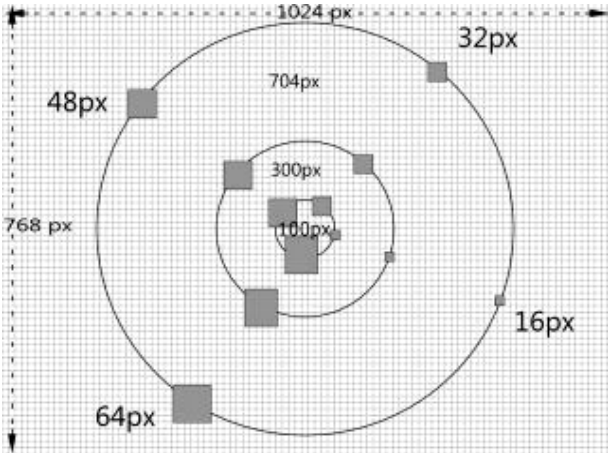


Figure 3. Comparison of targets and distances and the screen size for Fitts's law tapping task.

6 Results

The results of a 3-way ANOVA for the Fitts's law tapping task indicated that the selection times and error rates were significantly affected by magnifiers, widths, and distances.

As depicted in Figure 4, widget magnifier had the shortest selection time (M=0.88, SE=0.05), followed by without magnifier (M=1.04, SE=0.04), linear magnifier (M=1.21, SE=0.02), and hybrid magnifier (M=1.33, SE=0.05).

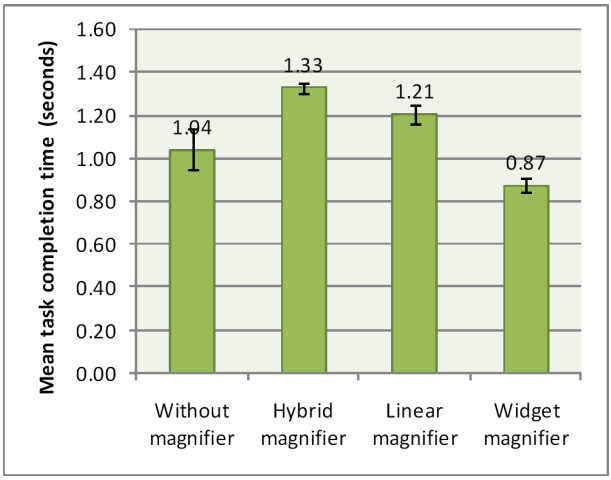


Figure 4. Mean selection times.

Figure 5 shows that hybrid, linear and widget magnifier had a significantly less error rate when compared with without magnifiers. The without magnifier interaction technique had especially very high error rates when considering small targets. Without magnification had on average a 123% higher error rate than the other interaction techniques when considering an icon size of 16 pixel. There was no significant difference of error rates among the different magnifiers.

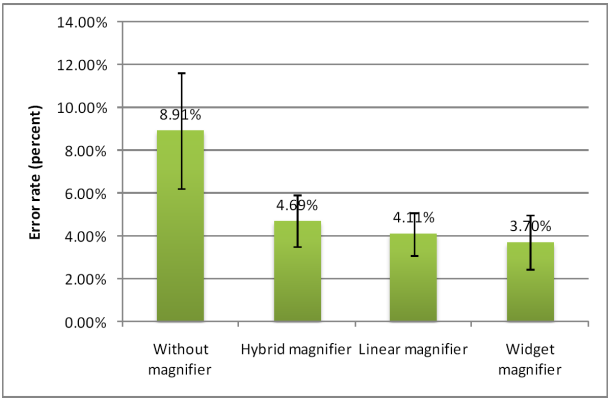


Figure 5. Mean error rates.

Figure 6 shows that in the internet browsing task, hybrid and linear magnifier were 73.09s (26%) and 71.40s (26%) faster than without magnifier. In contrast, widget magnifier was 17.80s (6%) slower. The result of the media player task indicated that hybrid and linear magnifier were 78.13s (26%) and 50.33s (26%) faster than without magnifier. In the contrary, widget magnifier was 35.42s (13%) slower.

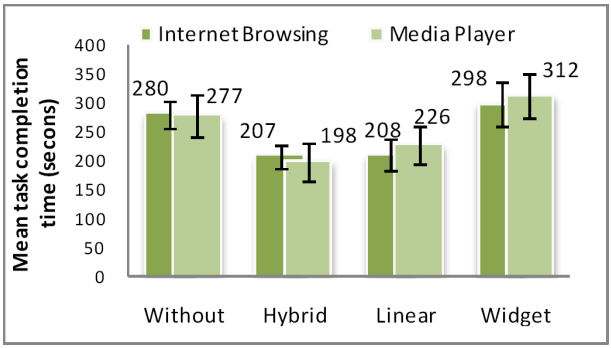


Figure 6. Mean task completion times for practical task.

The user acceptance was measured by questionnaires that were taken from IBM computer usability satisfaction questionnaires [7]. As showed in Figure 7 thought most participants that linear and hybrid fisheye magnifier are very enjoyable, quick, effective and satisfying interaction techniques. The widget magnifier on the other hand received constantly the worst ratings. Without magnifier received negative ratings regarding enjoyability, efficiency, quickness, effectiveness, and user satisfaction. On the other hand received this interaction technique rather positive ratings when considering intuitiveness, comfortableness and simplicity.

At the end of the experiment participants were asked to state their interaction technique order of preference. 50% of the participants saw linear magnifier and 42% saw widget magnifier as their first choice. 67% saw hybrid fisheye magnifier as their second choice. One can conclude from that, that the users would prefer the usage of magnifiers in general but have different preferences when it comes to the question which magnifier to use.

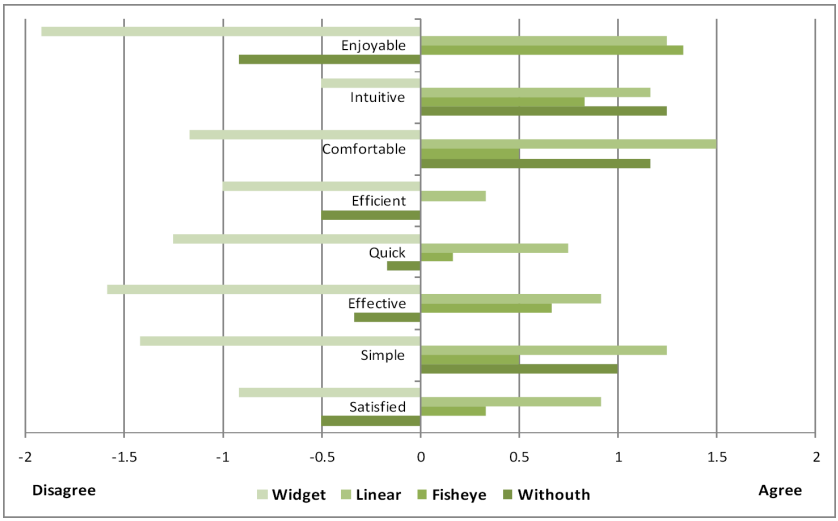


Figure 7. Feedback of participants.

7 Discussion

Magnifier had a very positive effect on error rate particularly for the selection of small targets. When the target size was 16 pixels, the error rate of without magnification was more than 20%. In contrast to that lead the usage of magnifiers to significantly lower error rates. This finding is consistent with H1.

In the practical task, the results showed that the completion time without magnifier was significantly higher when compared with hybrid fisheye and linear magnifier. The authors believe that when the participants performed the task without any magnifier, they had problems to see and point on the small hyperlinks and buttons. Moreover, the authors assume that this is due to the fact that text and buttons sizes of web browsers and media players are designed having a desktop usage scenario in mind. Because of that they are not suitable for the TV viewing distance. Therefore, magnifiers improve the usability in this context by making the user interface more visible to the user. This finding supports H2.

The result of the tapping task showed that hybrid fisheye had the highest task completion time, followed by linear magnifier. This might be caused by the magnified motion effect as the objects in a magnified dimension seem to be moving faster than the actual movement of the pointer. Hence, the users would have to hit moving targets, which is harder and more time consuming than hitting static objects. This argument is supported by the positive task completion time of the widget magnifier in the tapping task, which was better that any other magnification types. This finding is consistent with H3.

8 Conclusion

The presented research shows clear evidence for the advantages of using magnifiers for pointing interactions with remote screens. Those magnifiers help the user to read and interact with small buttons and hyperlinks displayed on remote TVs. This is proven through the significantly reduced error rates in the Fitts's law tapping task, the task completion times of the practical task, the user feedback and user preferences. As previously discussed provides each of the used magnifiers certain advantages and disadvantages which were also proven by the study results. The conclusion is to offer several magnifiers to the user and to let them decide which one to use.

Acknowledgement

This work is supported by the NoE InterMedia funded by the European Commission (NoE 038419).

9 References

[1] Lorenz, A., de Castro, C.F. and Rukzio, E. Prototype for Using Handheld Devices for Mobile Interaction with Displays in Home Environments. In Mobile HCI 2009.

[2] Olsen, D. R. and Nielsen, T. Laser pointer interaction. In CHI '01. ACM. Seattle, Washington, United States.

[3] Myers, B. A., Bhatnagar, R., Nichols, J., Peck, C. H., Kong, D., Miller, R., and Long, A. C. Interacting at a distance: measuring the performance of laser pointers and other devices. In CHI ’02. ACM. Minneapolis, Minnesota, USA.

[4] MacKenzie, S. and Jusoh, S. An Evaluation of Two Input Devices for Remote Pointing. In: Engineering for Human-Computer Interaction 2001. Springer. Toronto, Canada.

[5] Feeman, W. T. and Weissman, C. D. Television controlled by hand gestures. In International Workshop on Automatic Face- and Gesture-Recognition. 1995. Zurich, Switzerland.

[6] Keahey, T.A. and Robertson, E. L. Techniques for non-linear magnification transformations. In Symposium on Information Visualization (INFOVIS '96). San Francisco. USA.

[7] Lewis, J. R. IBM computer usability satisfaction questionnaires: psychometric evaluation and instructions for use. International Journal on Human-Computer Interaction, 7, 1 (Jan. 1995), 57-78.

Towards Interactive Museum: Mapping Cultural Contexts to Historical Objects

Ki-Woong Park, Sung Kyu Park, Jong-Woon Yoo, and Kyu Ho Park,
CORE Laboratory, Korea Advanced Institute of Science and Technology
373-1 Guseong-Dong, Daejeon, Korea

Abstract

In this paper, we present Interactive Museum Interface (IMI) which runs on a wearable computer. It allows people to efficiently and intuitively interact with historical objects in museums. Based on the IMI, the historical objects can be mapped to virtual icons containing cultural contexts as if making a shortcut icon in Desktop. A visitor can collect interesting contexts by pointing & selecting the virtual icons in the museum. In the IMI, the detection of users' pointing objects is a challenging issue because the virtual icons are widely dispersed in the physical space. In order to devise IMI pointing device, we performed simulations of G-sensor modules to extract system parameters like as the sensor threshold and the maximum density of virtual icons. Based on the simulation results, we designed and implemented IMI framework which includes a wearable platform and a ring-type 3D pointing device.

Keywords: Interactive Museum, Virtual Context Icon, Gesture-based Interface

1 Introduction

Recently, there has been an increasing interest in developing mobile interface for museum guides and sightseeing. Many museums still present their exhibits in a rather passive and non-engaging way. The visitor has to search through a booklet in order to find descriptions about the objects on display. However, looking for information in this way is a quite tedious procedure.

Moreover, the information found does not always meet the visitor's specific interests [1, 2]. Novel devices equipped with appropriate software can facilitate visually-impaired people in autonomous orientation and in exploring the surrounding environment. In this paper, we present an interactive museum interface (IMI) which runs on a wearable computer platform. It allows people to efficiently and intuitively interact with historical objects in museums. Over the interface, the historical objects can be mapped to virtual icons as if making a shortcut icon in Desktop. A visitor can collect interesting contexts by pointing to the virtual icons in the museum. Figure 1 shows the concept scenario of the interactive museum interface. The 3D museum space which includes various exhibitions [3] and those abundant spatial resources are mapped to a certain context. The visitor can select the exhibitions in the museum using a gesture-based 3D pointing device. And then, he can intuitively collect the context by just drag-and-drop into his/her wearable platform rather than searching through a booklet in order to find descriptions about the objects on display.

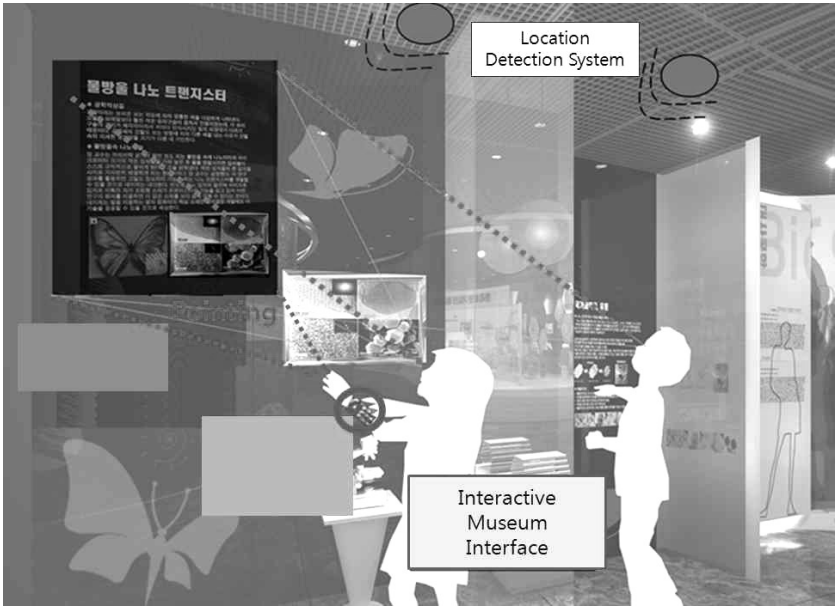


Figure 1. Concept Scenario of Interactive Museum Interface

In the IMI, the detection of users' pointing direction is a challenging issue because the virtual icons are widely dispersed in the physical space. In order to develop the 3D pointing device, we performed simulations of G-sensor modules to extract system parameters like as a sensor threshold and the maximum density of virtual icons for a certain space. Based on the simulation results, we designed IMI framework with a ring-type 3D pointing device. As a next step, we have implemented the prototype of IMI system which includes a wearable platform, and a ring-type 3D pointing device and deployed UWB-based location tracking system.

The outline of this paper is listed as follows: Section II introduces an application scenario of IMI and its components. The design flow of 3D-pointing device is described in section III. In section IV, we summarize our related works and conclusion is given in section VI.

2 Interactive Museum System

2.1 Application Scenario

In order to realize IMI system, we designed a prototype based on gestural input and mobile display output. Figure 2(a) shows our museum testbed which includes several physical objects. The location information of these objects is managed by the Virtual Map Manager (VMM) which contains the virtual map of the testbed as shown in Figure 2(b). As described in the previous section, the physical objects are mapped to virtual icons as if making a shortcut icon in Desktop.

When a visitor visits the museum, he/she can give a searching glance at around objects by scanning gesture (Figure 2-c and Figure 3-a). And the visitor can select interesting contexts by pointing & selecting the virtual icons in the testbed (Figure 3-b). Finally, the visitor can collect the contexts into his/her wearable platform by taking the drag-and-drop gesture (Figure 3-c). The gesture detection in our interface utilizes a three-axis accelerometer [4] and a three-axis magneto-resistive sensor [5]. The sensor produces signals that are interpreted as events by the gesture detection processing module of the wearable platform. The movements are detected by an accelerometer and, depending on the direction and speed of such movements, they are translated

into suitable actions/events (selection, scan or drag-and-drop) on the gesture interface.

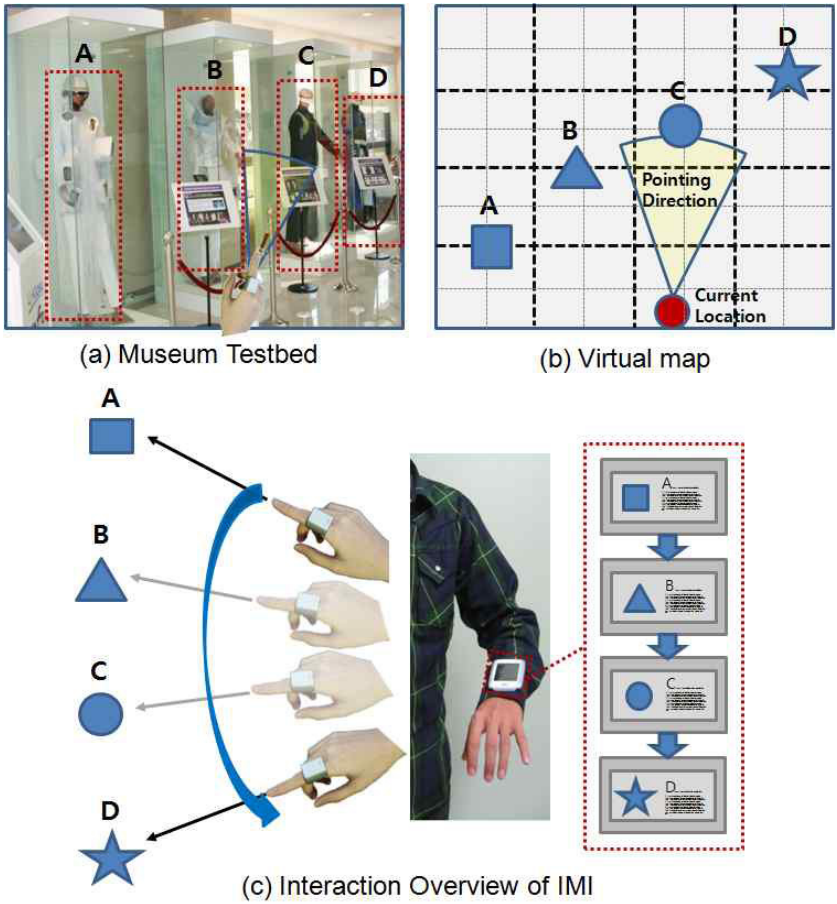


Figure 2. Virtual and Interaction overview of Interactive Museum Interface

2.2 System Components

The overall system of IMI is composed of four components: 3D Pointing Device, Location Tracking System, and Virtual Map Manager.

- *3D Pointing Device:* It is used to exploit spatial resources. A visitor can point to any objects in the physical space and input simple commands like point, select, scan operations. We have devised IMI pointing device. We explain it more detail in following section.
- *Location Tracking System:* It keeps track of the location of users and physical objects in 3D physical space. This system is essential because the absolute location information of users and physical objects are critical to find the target object that users point to with 3D pointing device. We have utilized an Ultra- Wideband (UWB)-based location tracking system whose typical accuracy is 6 inches (15cm).
- *Virtual Map Manager (VMM):* The role of a VMM is to manage virtual icons and the mapping information of the physical space. It contains virtual maps for a certain region like Figure 2-b. When a visitor points to certain object, the VMM automatically finds which virtual icon is mapped to that object.

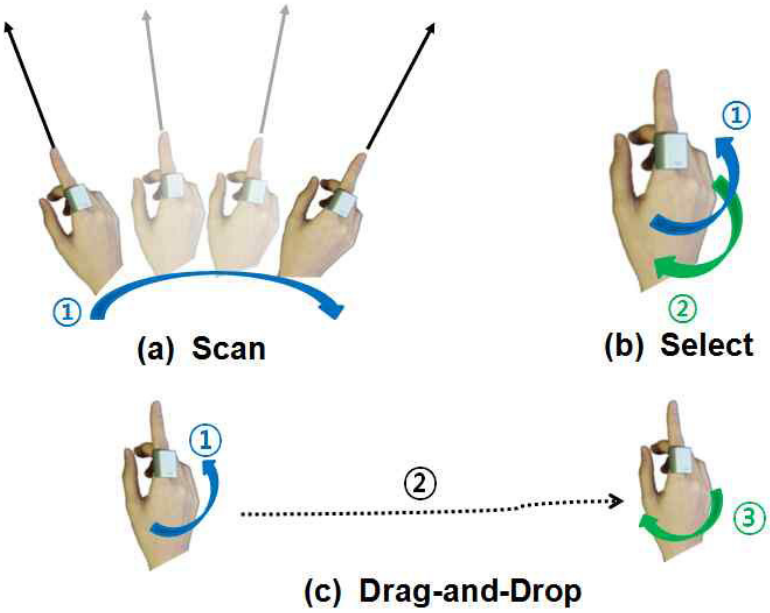


Figure 3. Basic Interactive Museum Interface

3 Target Selection Technique

3.1 Ray-based Target Selection Technique

Target selection that identifies which object is pointed by the visitor is the main issue of the IMI system. A sequence of the target selection operation to realize the scenario consists of three phases: First, a visitor can pick interesting contexts by taking selection gesture. Second, the visitors' location and pointing gesture are recognized by the location tracking system and gesture detection module. Finally, on recognizing the selection gesture, the virtual map server responds the information of the pointing object to the visitor's wearable platform.

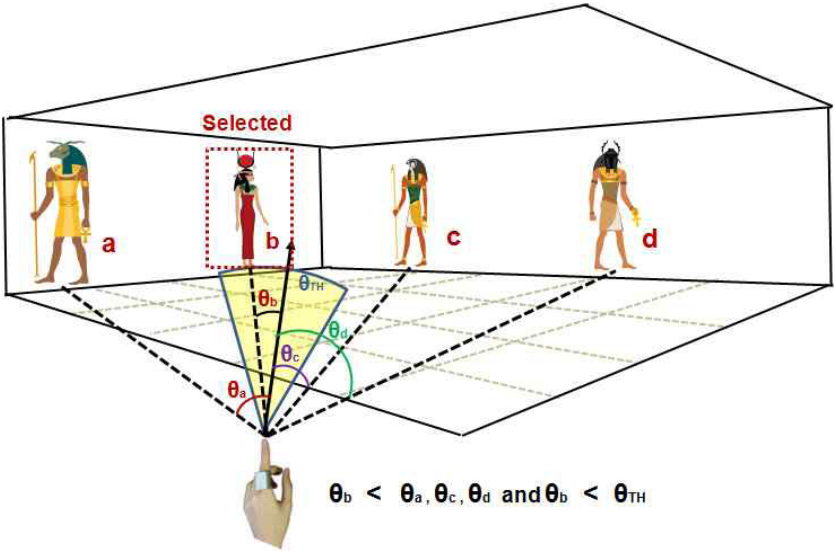


Figure 4. Ray-based target selection mechanism

To realize the target selection mechanism, we have taken a ray-based minimum angle selection approach which is described in if an angle between a ray and the selected icon is smaller than θ_{TH}. If a minimum angle is larger than θ_{TH}, no physical object is selected. The reason why θ_{TH} is needed is that we have to find an empty space or an empty physical object which is not

mapped a virtual icon. If not, we can't find an empty space when one virtual icon is mapped in a space.

3.2 System Parameters Extraction

In this study, the detection of users' pointing objects is a challenging issue because the virtual icons are widely dispersed in the physical space. Beside the pointing accuracy of the pointing device is affected by both human factors and device factors. The human factors are something like the hand trembling and miss alignment. And the device factors are the variation of a value from a sensor and the error of a location tracking system. Among them, device factors impress the pointing accuracy significantly comparing the human factors. Therefore, we mainly focus on the device factors when designing the pointing device.

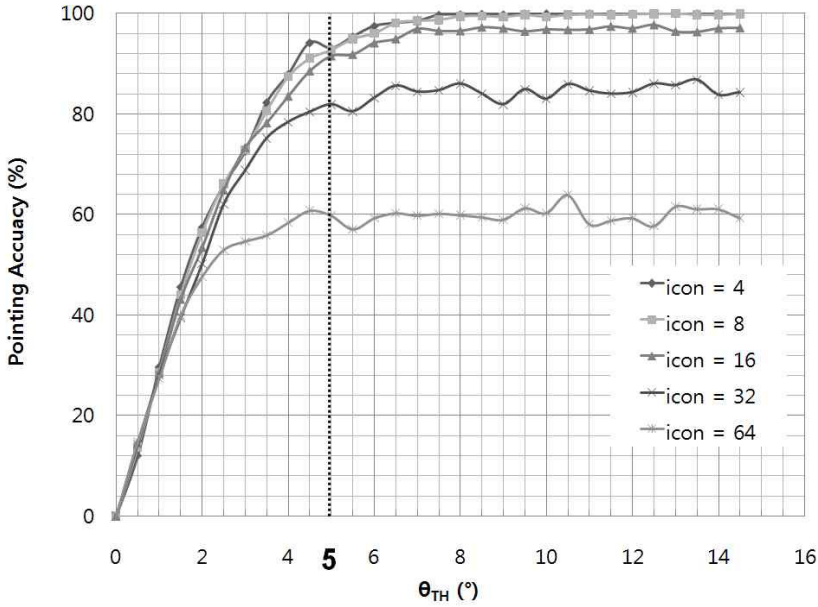


Figure 5. Pointing accuracy with the number of virtual icons and θ_{TH}

3.2.1 Simulation of Pointing Accuracy

We performed simulations to measure the expected pointing accuracy with varying the threshold angle (θ_{TH}), and the number of the virtual icons. In the simulations, we assumed that the average location error from the location tracking system [6] is 6 inches (15cm), and the one region space of a museum is 100m^2. Figure 5 shows how the pointing accuracy mutates as the number of virtual icons and θ_{TH} varies. As increasing the number of the icons, the pointing accuracy is decreasing due to the short distance between the virtual icons. θ_{TH} also affects pointing accuracy. If it increases, it is easy to point to a target icon because a selection range is extended. However, if θ_{TH} is too large, it is hard to find an empty space or an empty physical object which is unmapped to a virtual icon. Therefore, it is important to select an appropriate value of θ_{TH}. From the simulation, we have chosen an appropriate value of these parameters. An appropriate value of θ_{TH} is 5 and the appropriate number of virtual icons in 100m^2 room is 20. If applying these values of parameters to real system, we will obtain over 90% pointing accuracy.

3.2.2 Applying the parameters into the 3D-pointing device

Based on the results of the simulations, we applied the measured parameters into IMI 3D pointing device and a wearable platform called UFC [7]. As shown in Figure 6, it is a ring-type device for reducing an error of miss alignment. It has a three-axis accelerometer [4] and a three-axis magnetic sensor [5] for recognizing the defined gesture and the direction of the finger. It also has a ZigBee transceiver [8] for informing the recognition results to the UFC. Every time a visitor points to an object in the museum, the wearable platform displays the selected target object and context upon its screen. This feedback information helps the visitor find the correct target object. Similarly, a scanning gesture allows the user to investigate pointing object as described in Figure 2-c.

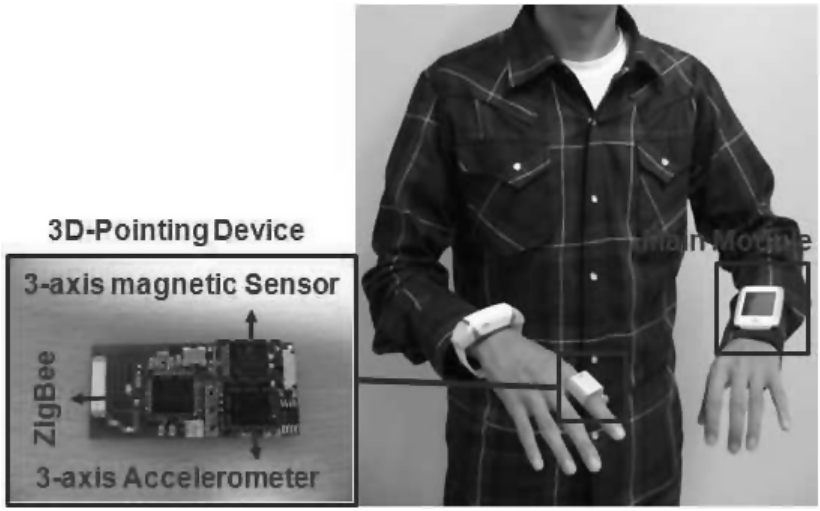


Figure 6. 3D-pointing device and wearable platform for IMI

4 Related Work

Recently, numerous researches have done in devising various technologies to assist in museum's exhibition such as Interactive Museum Guide [1], an audio-augmented museum guide, called LISTEN[13], RFID-based Museum Guide [2], Electronic Guidebook [9], PDA with Semantic Web [10], Museum Guide enhanced with tangibility called ec(h)o [14], and many more. As a service aspect, Herbert Bay proposed the prototype of an interactive museum guide. It runs on a tablet PC that features a touch-screen, a webcam and a Bluetooth receiver. This guide recognizes objects on display in museums based on images taken by the webcam on the tablet PC. In order to provide blind users with a museum guide services, Giuseppe Ghiani [2] made effort to investigate how tilt-based interaction, along with RFIDs for localization, can be exploited to support blind users in interacting with mobile guides. They presented a tilt-based interaction and RFIDs for accessible mobile guides.

As a mobile technology aspect, many research efforts have been performed to achieve high usability of a mobile device. Rukzio et al. [11] proposed a framework for the development of systems which takes physical mobile interactions into account. They mean any communication between the entities user, mobile device, and physical objects in the physical world whereby every entity can exist one or more times with it. They have used typical technologies supporting these interactions that are Radio Frequency Identification (RFID), visual marker recognition, Near Field Communication (NFC), or Bluetooth. They have made mobile interactions with various services, which were inadequate and inflexible in a mobile device for small screens, fiddly keys and joysticks as well as narrow and glutted menus, easier and more convenient. Valkkynen et al. [12] presented a user interaction paradigm for physical browsing and universal remote control. The paradigm is based on three simple actions for selecting objects: pointing, scanning, and touching for choosing tags with readers. Therefore, these paradigms should be supported for any tagging technology. They have provided an optimal support for natural interaction with physical objects. And they can control augmented physical objects with tags and get information from them. All of the previous works, however, used augmented physical objects with tags. Therefore, they can only interact with specified objects that have augmented tags. Compared to the previous work, we can use more abundant spatial resource because the physical objects do not need augmented tags for the interaction in our mechanism.

5 Conclusion

We have presented Interactive Museum Interface (IMI) which runs on a wearable computer. Our work aims to allow people to efficiently and intuitively interact with historical objects in museums. In order to devise IMI pointing device, we performed simulations of G-sensor modules to extract system parameters like as the sensor threshold and the maximum density of virtual icons. Based on the simulation results, we designed and deployed IMI framework which includes a wearable platform called UFC and a ring-type 3D pointing device.

6 References

[1] H.Bay, B.Fasel and L.Gool, "Interactive Museum Guide", UbiComp 2005, September 1114, 2005, Tokyo, Japan.

[2] G.Ghiani, B.Leporini, F.Paternò, and C.Santoro, "Exploiting RFIDs and Tilt-Based Interaction for Mobile Museum Guides Accessible to Vision-Impaired Users", ICCHP 2008, LNCS 5105, pp. 1070-1077, 2008.

[3] G. Broll, S. Siorpaes, E. Rukzio, M. Paolucci, J. Hamard, M. Wagner, A. Schmidt, "Supporting Mobile Service Usage through Physical Mobile Interaction", In proc. the 5th IEEE International Conference on Pervasive Computing and Communications (PerCom'2007)

[4] Freescale Semiconductor, 3-axes accelerometers, MMA7260Q, http://www.freescale.com/

[5] Honywell, Magneto-resistive sensor, HMC1053, http://www.ssec.honeywell.com/

[6] Ubisense, http://www.ubisense.net

[7] J.Lee, S.H Lim, J.W Yoo, K.W Park, H.J Choi, K.H Park, "A Ubiquitous Fashionable Computer with an i-Throw Device on a Location-Based Service Environment", In Proc. 21st IEEE Symposium on Pervasive Computing and Ad Hoc Communications, Vol. 2, pp. 59-65, May 2007.

[8] IEEE 802.15.4 Specification, "Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications for Low Rate Wireless Personal Area Networks (LRWPANs)", IEEE Specifications, October 2003.

[9] M. Fleck, M. Frid, T. Kindberg, R. Rajani, E. O'Brien-Strain, and M. Spasojevic (2002), "From Informing to Remembering: Deploying a Ubiquitous System in an Interactive Science Museum", [Online]. Available: http://www.hpl.hp.com/techreports/2002/hpl-2002-54.pdf

[10] C. Shih-Chun, H. Wen-Tai, F. L. Gandon, and N. M. Sadeh, N.M.(2005), "Semantic Web Technologies for Context-aware Museum Tour Guide Applications", 19th International Conference on Advanced Information Networking and Applications, 2005, Volume: 2, pp. 709- 714

[11] E. Rukzio, S. Wetzstein, and A. Schmidt, "A Framework for Mobile Interactions with the Physical World", Wireless Personal Multimedia Communication (WPMC'05), Aalborg, Denmark, 2005.

[12] P. Valkkynen, I. Korhonen, J. Plomp, T. Tuomisto, L. Cluitmans, H.Ailisto, and H. Seppa, "A user interaction paradigm for physical browsing and near-object control based on tags", In proc. Physical Interaction Workshop on Real World User Interfaces, in the Mobile HCI Conf. 2003, Udine, Italy, 2003.

[13] Zimmermann, A., Lorenz, A. "LISTEN: A User-adaptive Audio-augmented Museum Guide", Springer User Model User-Adap Inter (2008) 18:389-416 Ec(h)o

[14] Ron W., Marek H. "ec(h)o: Situated Play in a Tangible and Audio Museum Guide", ACM DIS 2006

Cocktail: Exploiting Bartenders' Gestures for Mobile Interaction

Jong-Woon Yoo, Woomin Hwang, Hyunchul Seok, Sung Kyu Park, Chulmin Kim, and Kyu Ho Park
CORE Laboratory, Korea Advanced Institute of Science and Technology
373-1 Guseong-Dong, Daejeon, Korea

Abstract

Recent mobile devices are capable of creating and storing a large amount of multimedia data, but sharing those data with others is still challenging. This paper presents Cocktail, a new gesture-based mobile interaction system that exploits bartenders' motions. Like a bartender who pours drinks to a glass, a user can pour (transfer) multimedia data to other device. The user can also mix music files and pictures into a music video by shaking the mobile device, as a bartender does to make a cocktail. We have implemented a prototype of Cocktail system with Ultra Mobil Personal Computers and a touch-screen monitor and demonstrate its usability.

Keywords: Mobile interaction, gesture interface

1 Introduction

Most advanced mobile devices are capable of creating new multimedia data. For example, mobile phones are equipped with high resolution cameras, allowing users to take pictures or make movies anytime, anywhere. However, sharing those data with others is still challenging. Transferring the taken pictures from one mobile phone to another one usually takes several steps of inconvenient user intervention, including manipulating buttons to execute the file exchange program, enabling radio interfaces (e.g., Bluetooth) followed by wait time for identifying the target's network address (e.g., Bluetooth MAC address), selecting pictures to be sent, and pressing 'transmit' button.

Therefore, it is essential to devise novel solutions for exchanging multimedia data in more user-friendly ways to enhance usability of mobile devices.

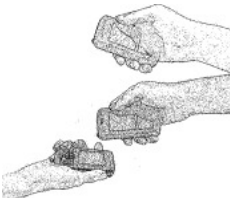
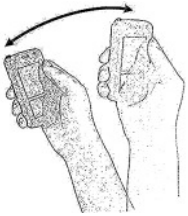
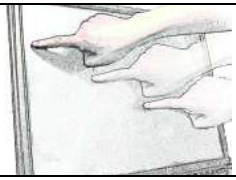
In a bar		In our system	
	Pouring drinks to a shaker or a glass		File transfer
	Mixing drinks to make a cocktail		Making a music video
	Hand over glasses or bills		File transfer to stationary devices and quick execution

Table 1: Comparison of the gestures used in a bar and in our system

This paper describes a gesture-based mobile interaction system, Cocktail, which is designed for intuitive data transfer and contents creation. Motivated from bartenders who mix drinks to make cocktails, our system uses their gestures for mobile interaction: a user can pour (transfer) data in her mobile phone to other device, like a bartender who pours drinks to a shaker. The user can mix her pictures and music files into a music video by shaking the mobile phone, as a bartender does to mix drinks into a new cocktail. In addition, our system includes a touch-screen-based table-like computer system called SmartTable which support 'pushing' interaction with stationary devices, such as a TV, or printer. In the same way that a bartender pushes glasses or bills to

customers on a table, users can push icons on SmartTable towards the stationary devices to transfer the data. Table 1 compares the gestures used in a cocktail bar and our system.

2 Cocktail System

2.1 Overview

Figure 1 shows the overall Cocktail system. It consists of mobile devices (e.g., mobile phones), a touch-screen called SmartTable, and several types of stationary devices (e.g., networked storage systems, displays, or printers). Cocktail provides gesture-based intuitive interaction among them.

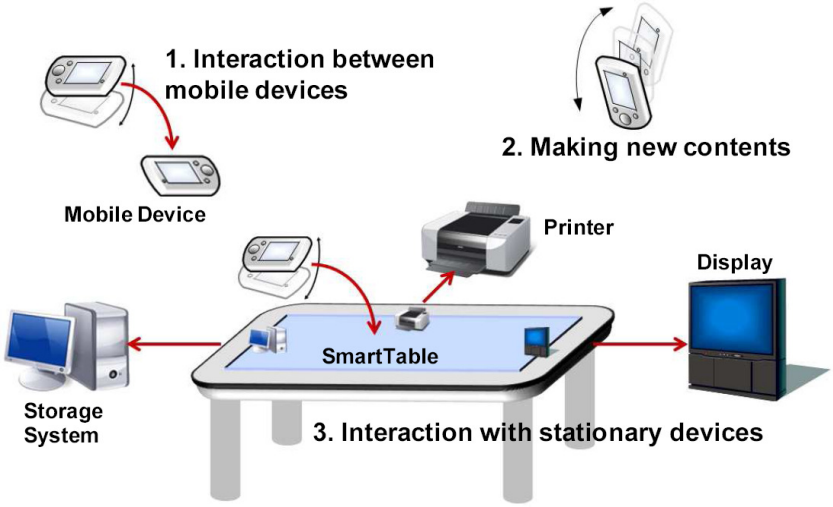


Figure 1: An example of Cocktail system.

Those devices shown in Figure 1 are mapped to the objects in a real cocktail bar: a mobile device is mapped to a bottle, multimedia data in the mobile device represents drink in the bottle, and SmartTable is mapped to a table in the bar. In this environment, mobile users become digital bartenders handling multimedia data in their mobile phones.

Our system provides three types of interactions: (1) data transfer from a mobile device to another device (including both mobile and stationary devices), (2) creating new contents, and (3) data transfer from SmartTable to stationary devices. All three interactions are based on gestures used by bartenders.

First, in order to transfer data in a mobile device to another device, we use the sprinkling gesture (which is similar to the pouring gesture). Like a bartender pours drink from a bottle to a glass, a mobile user can transfer data, such as pictures, in her mobile phone to another device, such as her friend's phone, by sprinkling her device above the target device. Then, our system automatically finds the target, establishes a connection, and begins the data transfer. Section 2.2 discusses the sprinkling interaction in detail.

Seconds, like a bartender who makes a new cocktail by mixing various drinks, a mobile user can create new contents by shaking her mobile phone. Then the contents (music or pictures) in her mobile phone are mixed into new contents (music video). This interaction is described in Section 2.3.

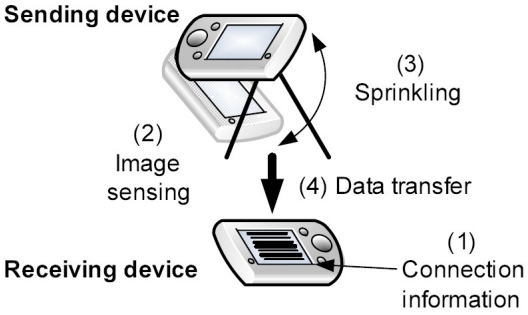
Finally, we have designed the SmartTable interaction to support intuitive data transfer to stationary devices. SmartTable is a touch-screen-based computer system. A user can transfer data in the mobile phone to SmartTable with the sprinkling gesture, i.e., by sprinkling the mobile phone above the touch-screen. Then, an icon that represents the received data appears on the touch-screen. In addition to this, SmartTable allows the user to transfer data from SmartTable to nearby stationary devices, such as a TV, printer, or computer, by pulling the icon to the direction of the target device. The details about the SmartTable interaction are given in Section 2.4.

2.2 Sprinkling Interaction

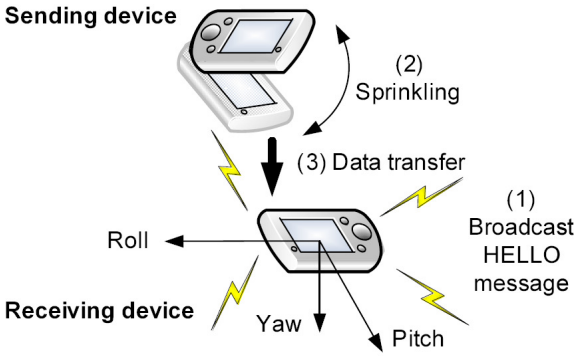
Figure 2 shows the concept of data transfer with the sprinkling gesture: a user transfers data from her device to other device by sprinkling her device upon the target device like sprinkling salt on fish. Note that the target device can be both mobile devices (like a friend's phone) and stationary devices (like SmartTable or a printer). To realize the sprinkling interaction, we need to answer to two fundamental questions: (1) *how to determine the target?* and (2) *how to recognize the sprinkling events?*

Several previous works have explored the idea of using gestures for file transfer. Toss-It [1] uses the 'tossing' gesture to transfer a file from the mobile

device to a remote target in the line-of-sight. iThrow [2] allows users not only to transfer data but also to control remote targets using gestures. These approaches, however, require real-time high-resolution location tracking systems to identify the position of each user and the target device.



(a) Target detection using markers.



(b) Target detection using signal strength.

Figure 2: Data transfer with the sprinkling gesture.

Adopting the sprinkling gesture alleviates the requirement of location tracking because this gesture forces the sending device to be placed above the target device. Therefore, we can use high-resolution cameras in recent mobile phones and well-studied image processing techniques [4] for target detection,

as shown in Figure 2(a). First, if the receiving device becomes ready, it displays its connection information as an encoded code (like a marker) on its screen. Then, the sending device extracts the information from the marker and establishes a connection. Finally, the user sprinkles her device to initiate data transfer.

Figure 2(b) shows a simpler approach using Received Signal Strength Index (RSSI) for target detection: the receiving device first broadcasts HELLO messages containing connection information when it becomes ready. The sending device can extract needed information from the HELLO message.

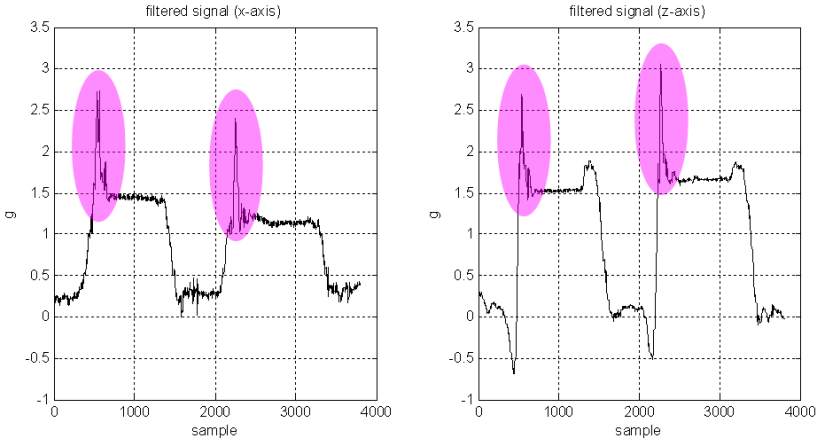


Figure 3: Extracted acceleration features of the sprinkling gesture.

Using RSSI, sending device can identify the target which is closest to itself. However, when the sending device is surrounded by many other devices, the distance may not be sufficient to identify the target among them. Therefore, in our system, the receiving device should stay (almost) horizontal to the ground, i.e., only devices which has zero roll and pitch are looked up when target identification is on-going. This reduces ambiguity of target identification, but not completely avoids it. Therefore, we pop up a list of detected targets when multiple devices are searched. In our implementation, we use the second approach because of its simplicity.

We use 3-axes accelerometers to answer the second question. We can extract the features of sprinkling gestures (as well as shaking gestures) as shown in

Figure 3. The original accelerometer outputs include some obstacles due to the unintentional noises and hand trembles. They can be reduced by passing the outputs through low pass filters. From the filtered signals, we can define the feature of the sprinkling gesture as several points that exceed 2-g in x- and z-axis.

2.3 Shaking Interaction

The shaking interaction is designed for contents creation. Figure 4 illustrates the concept of shaking interaction. First, user C receives a music file and a picture from users A and B via sprinkling interaction. Next, user C shakes her device up and down. Then, the received music file and picture are mixed into a music video.

Support of creating new contents by mixing existing data will increase the usability of mobile devices, but this is a hard work for a hand-held device, whose computing power is limited. In order to avoid the extensive computations inside the mobile device, we apply an o²oading technique: the computations are done in the other powerful machine, such as SmartTable. When a user takes pictures or records songs, the contents are transferred from her mobile device to the powerful machine. Then, if the user makes shaking action, the powerful machine starts to generate a new content rapidly, and sends it to the shaken device.

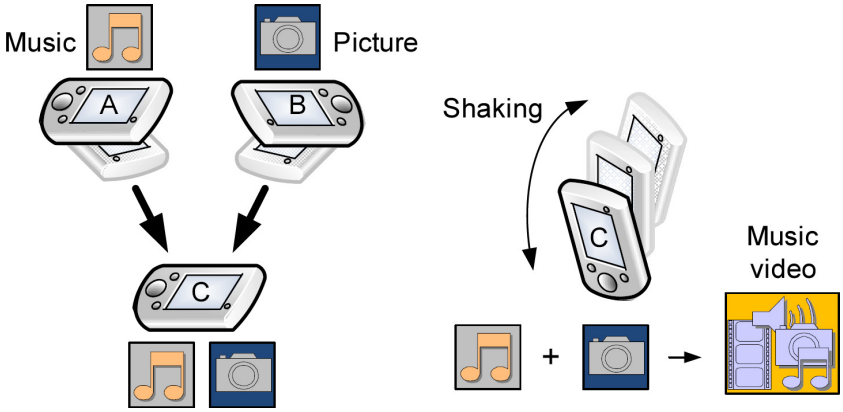


Figure 4: Creating a music video by mixing a picture and a music file.

2.4 SmartTable Interaction

This section describes SmartTable that provides touch-screen-based interaction with stationary devices, such as a printer or TV. To do so, SmartTable manages virtual icons of physical stationary devices which surround it. The virtual icons are located in an edge of a display from the center of a display toward stationary devices. SmartTable can manage a map which includes location information of virtual icons. Therefore, we can easily interact with stationary devices using these virtual icons.

Figure 5 shows an example of SmartTable interaction with a printer, a computer, and a TV. Each device is mapped to an edge of SmartTable as if a virtual icon. Therefore, a user can easily store, print, and display a picture in SmartTable by dragging it to a location of a virtual icon. For example, we map a printer to a location, (20, 60). If a user want to print a picture, he/she drags a picture to a location (20,60). If a user wants to register a new device to SmartTable, he/she has only to update a map. The main difference between SmartTable and MS Surface [3] is that SmartTable can interact with surrounding devices. MS Surface has many operations using a multi-touch-screen. However, it cannot interact with surrounding devices, but only in its own fixed display. In contrast to MS Surface, SmartTable is expanded into a physical space which includes various devices. Iconic Map [5] introduced a similar concept, but the main contribution of our work is providing the bartender-like natural interface for mobile interaction.

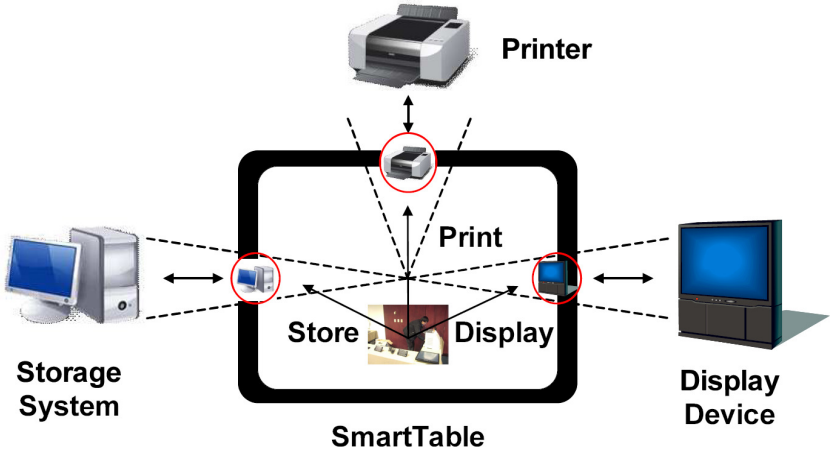


Figure 5: An example of SmartTable interaction.

2.5 Bottle-like Graphical User Interface

Our GUI is composed of pictures and Background. Pictures represent the contents such as images, songs, movies and etc. The operation of GUI is specialized for the available gestures - rubbing, pouring, shaking and sprinkling. The pictures can move as if it is in the real world since the physical model including gravity, friction and torque is used to implement the motion of pictures. Figure 6 illustrates the various motions according to the gestures. Figure 6(a) shows the straight movement by rubbing and pouring. If we rub a picture, the picture has velocity determined by rubbing speed. Then, the picture follows uniformly negative-accelerated motion which acceleration is made by friction force. In case of pouring, it has same motion but the acceleration is determined by gravity. Figure 6(b) describes bumping motion of a picture. Too much force on a picture can move it away from the screen. So, the Background captures the picture which wants to escape, and invert the velocity of opposite axis direction of the border line capturing the picture. The motion due to shaking is illustrated in Figure 6(c).

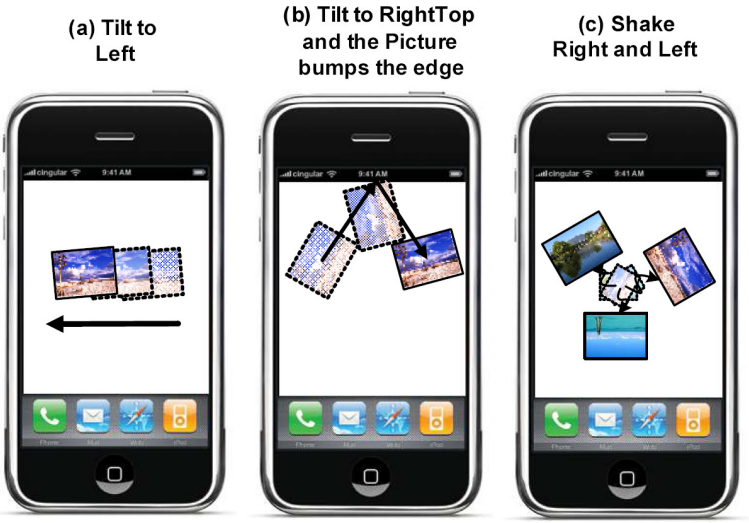


Figure 6: Bottle-like Graphical User Interface.

Actually, the shaking in here and the shaking in the above section are recognized as different gestures. When users do the shaking of this section, the pictures getting together are spread out. So, the users can see the part of

the pictures which is hidden by other pictures. Sprinkling functions to transfer the picture in the device to the other device. When sprinkling occurs, the pictures in the sender are eliminated from the screen one by one and the pictures are created in the receiver's screen one by one to indicate that the transfer is going well.

3 Demonstration

We have implemented the prototype of Cocktail (Figure 7) using two Ultra Mobile Personal Computers (UMPC) [6] as mobile devices, a touch-screen monitor for SmartTable, and a printer as a stationary device, to demonstrate the usability of the proposed system. We have also implemented a tiny sensor module that consists of a 3-axes accelerometer [7] for motion sensing and a CC2430 ZigBee transceiver [8] for RSSI measurement. It is attached to the back of each UMPC as shown in Figure 8. The sensor module is directly connected to the UMPC via a USB cable. RSSI-based target detection approach discussed in Section 2.2 is used. One can see our demonstration video on http://core.kaist.ac.kr/cocktail/.

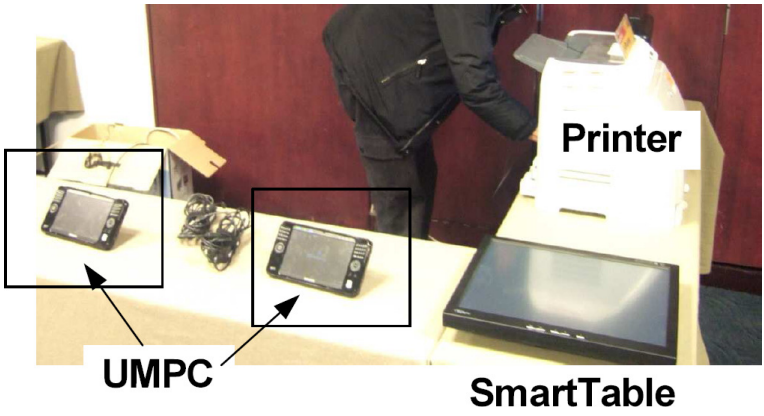


Figure 7: Cocktail system prototype.

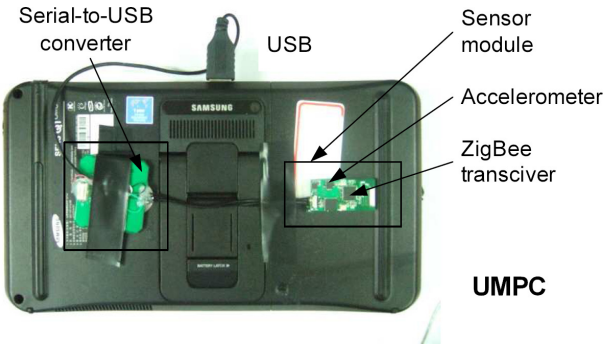


Figure 8: A UMPC and the attached sensor module.

4 Conclusion and Future Work

This paper has presented Cocktail, a new mobile interaction system using bartenders' gestures, and demonstrated its usability using the real prototype. Although our prototype implementation includes additional sensor modules for motion and RSSI sensing, we expect that our system can be easily integrated into most advanced mobile phones without hardware extension because recent mobile phones are increasingly equipped with gesture and wireless communication (e.g., Bluetooth) modules. Our future works include the implementation of Cocktail on mobile phones, performance evaluation, and user studies.

5 References

[1] K. Yatani, K. Tamura, K. Hiroki, M. Sugimoto, and H. Hashizume, "Toss-It: Intuitive Information Transfer Techniques for Mobile Devices", ACM CHI'05, pp. 1881-1884, 2005.

[2] J. W. Yoo, Y. W. Jeong, Y. Song, J. Lee, S. H. Lim, K. W. Park, and K. H. Park, "iThrow: A Gesture-Based Wearable Input Device with Target Selection Algorithm", International Conference on Machine Learning and Cybernetics, vol. 4, pp. 2083-2088, 2007.

[3] S. Bathiche and A. Wilson, Microsoft Surface, http://www.microsoft.com/surface/.

[4] D. Scott, R. Sharp, A. Madhavepeddy, and E. Upton, "Using Visual Tags to Bypass Bluetooth Device Discovery", ACM Mobile Computing and Communication Review, vol. 9, no. 1, pp. 41-53, 2005.

[5] R. Gostner, E. Rukzio, and H. Gellersen, "Usage of Spatial Information for Selection of Co-located Devices", ACM International Conference on Human Computer Interaction with Mobile Devices and Services, pp. 427-430, 2008.

[6] Samsung, Q1 Ultra, Ultra Mobile Personal Computer, http://www.samsung.com/sec/.

[7] Freescale Semiconductor, MMA7260Q, 3-axes low-g accelerometer, http://www.freescale.com/.

[8] Texas Instruments, CC2430, IEEE 802.15.4/ZigBee transceiver, http://focus.ti.com/.

Shopping in the Real World: Interacting with a Context-Aware Shopping Trolley

Darren Black
Systematic A/S
Søren Frichs Vej 39, DK-8000 Århus

Nils Jakob Clemmensen
Nordjyske Medier
Langagervej 1, DK-9220 Aalborg East

Mikael B. Skov
Aalborg University
Selma Lagerlöfs Vej 300, DK-9220 Aalborg East

Abstract

Shopping in the real world is becoming an increasingly interactive experience as stores integrate various technologies to support shoppers. Based on an empirical study of supermarket shoppers, we designed a mobile context-aware system called the Context- Aware Shopping Trolley (CAST). The aim of the system is to support shopping in supermarkets through context-awareness and acquiring user attention. Thus, the interactive trolley guides and directs shoppers in the handling and finding of groceries. An empirical evaluation showed that shoppers using CAST adapted in different shopping behavior than traditional trolley shoppers by exhibiting a more uniform behavior in terms of product sequence collection and ease of finding products and thus, CAST supported the shopping experience.

1 Introduction

Shopping in the real world, i.e. grocery shopping in supermarkets is becoming an increasingly interactive experience. Concept stores like the Metro Groups Future Store have started using radiofrequency identification (RFID) tags to streamline supply chains as part of a checkout-free store concept [8]. Other stores have integrated self-checkout points to speed up the paying process, while others integrate barcode scanners where shoppers can get information about products. Finally, manufacturers like Siemens- Nixdorf and MediaCart produce shopping trolleys with interactive touch-based screens where shoppers can find information related to the shopping activity, e.g. the shopping list or information about selected products.

With high-speed wireless networks like 3G, people can now access a wide array of information, e.g. cooking recipes or product information, from the Internet on their mobile devices while shopping and thus, create their own unique shopping experiences. Hence, future shopping in the real world is likely to involve both handling of real world objects or smart objects (groceries) and at the same time mobile technologies for enhancing the experience.

Product placement and exposure is important for supermarket shopping and promoting products on a trolley screen raises concerns about e.g. what products should stores show on the display, when should these products be displayed, and where in the store should certain products be displayed? One of the potential promising solutions to such questions could be to integrate context-awareness into the product. Research within context-awareness has focused on the above questions related context where one should consider who, what, where, when, and why questions related the context of the system [6]. However, this is still poorly understood.

In this paper, we describe the design and evaluation of a prototype for enhancing the shopping experience in supermarkets by actively acquiring and maintaining the users' attention. The prototype is called CAST (Context-Aware Shopping Trolley), and it directs shoppers through a supermarket by outlining product placement. Elements of context-awareness are used in the solution as attempts to integrate the handling of products with the interaction with the mobile devices (CAST).

2 Related Work

Context-awareness in mobile devices provides users opportunities and ways for interacting with mobile computing devices [4]. One of the promising features of context-awareness is the support of the limited means for input and output of mobile devices [11]. Numerous research studies on context-awareness focus on design of context-aware technologies and illustrative examples are many [7]. Several mobile context-aware systems and prototypes have been proposed during the last years. Mobile tourist guides, e.g. [5, 9], are typical examples of mobile context-aware systems.

Schilit and Theimer explain context-awareness as the ability of an application to discover and react to changes in the environment [10]. Environment is a complex entity and defines circumstances or surroundings assigned to an application's context. Environment can be understood in terms of entities denoting people, places, or objects that are relevant when using the application [6]. Thus, a context-aware mobile system should be able to discover changes in these three entities, e.g. when an object is in close proximity to the system.

Few studies have investigated opportunities of context-awareness for the shopping environment. Bohnenberger et al. illustrates the implementation and testing of a decision-theoretic shopping mall guide [3]. The system works on a 'macro' level, instructing the user which shops to visit, and in which order, to make their shopping activity more effective. The system achieved this goal using an extremely simple interface, which directed the shopper towards shops using arrows. The system proved both effective and likable according to testing, reducing time spent shopping by a small but significant amount (11%). Issues highlighted included users feeling they lacked an overview and felt like they were being "led blindfolded" through the mall. The Shopping Assistant by AT&T Bell Laboratories is another example of a context-aware mobile system [4]. This device can guide shoppers through a store and provide details about products in the store. Further, the device can help shoppers locate products within the store. But its applicability is limited understood.

Figure 1: Illustration of CAST. The picture on the top illustrates the shopping list and the overview map of the store including the nearest product (red dot in the map). The centre picture shows a nearby product in the shopping list and its location, while the bottom picture illustrates the system reacting to a shopper taking a loaf of bread.

3 CAST - A Shopping Trolley

In the following, we outline our context-aware shopping trolley solution (CAST). First, we describe the motivation for CAST as explained through a field study. Secondly, we illustrate the overall idea behind CAST, and finally, we present the design of CAST.

3.1 Background and Motivation

We based our research at the eastern Aalborg branch of føtex (a Danish chain of medium-sized supermarkets). We conducted in-situ contextual interviews with seven shoppers while shopping for groceries in føtex. Five of the shoppers were provided with a pre-generated shopping list while two shoppers brought their own list. Their movement through the store was logged and we recorded their utterances. Both the logger and the test leader wore audio recording devices to facilitate recording observations. Following the shopping sessions semi-structured interviews were conducted to obtain further contextual and demographic data for analysis. In the following we elaborate on three findings from the study. More information can be found in [3].

Our first observation showed that shoppers often find it difficult to locate products. This was expressed in two ways. First, shoppers sometimes failed to notice products despite being in extremely close proximity to the product often caused by poor product recognition and sometimes due to a belief that the product in question was located elsewhere. Secondly, shoppers often had difficulties in recognizing the products visually.

Our second observation concerned complexity of the setting. The store føtex, like many other supermarkets, presents shoppers with an array of stimuli. Brightly colored signs, aromas from various products fill areas, music, audio adverts, and announcements playing over the public announcement system. Also, shelves are crowded with brightly colored products and packages. All of these sources compete for the shopper's attention.

Our third observation illustrated the movement through the store. The physical layout of the store encourages shoppers to follow a U-shaped route through the store. Shoppers expressed disdain for any need to 'go back' on the route. Also, our shoppers stated that a shopping list would always be ordered in some fashion. They were offered the opportunity to re-order the

list they were provided, but only one did so. Opinions as to how such a list should be ordered varied - some claimed that it would be by where they thought things were placed in the shop, while others would use a mental model of product groupings to order the list.

3.2 Design

Based upon the findings of the field study, we designed CAST. CAST provides contextually relevant information to the user while shopping. CAST provides the shopper with information on product location and appearance when contextually relevant, and it registers products put into the trolley. By reacting to user context, the need for direct interaction with the system is reduced.

Applying understandings and definitions of context [6, 10], we define the context of shopping in føtex as follows: (1) task - to collect the items on the shopping list, (2) location - the location of the shopper, as well as the locations of products and shelves and the spatial relations between all three, (3) objects - the physical properties and states of products and shelves, and (4) people - the other shoppers. We see social context as manifesting in shoppers' need to follow the route through the store. Since the vast majority of shoppers walk in one direction, backtracking becomes difficult. In addition to physical issues, the route appears to be considered a social norm.

CAST's graphical user interface consists of a 7" TFT touch screen in the 16:9 screen format divided into two sections; in its basic state the left side of the screen shows the user's shopping list, while the right side shows a map of føtex (see figure 1, top). The touch screen is mounted on a regular shopping trolley. CAST supports shoppers by sensing the user's context and, where necessary, acquiring the user's attention through a simple sound notification.

1) Task. The user's task is to collect the items on the shopping list. The inclusion of the shopping list in CAST gives a direct representation of the task. The dynamic ordering of the list presents the sub-parts of that task in the order which best suits the shopper's current context.

2) Location. CAST provides location information in several dimensions namely between the spatial relations between products and the trolley, trolleys and shelves, products and shelves, and products and products. The items on the user's shopping list are represented with icons corresponding to those displayed in the shopping list to aid recognition and the spatial

relationship to the users' current position. Finally, the map represents the user's position as a red spot. As the shopper moves with CAST the map updates; the red spot stays in the middle of the map's display area and the map moves such that the red spot correctly depicts the shopper's location. Furthermore, the shopping list reorders itself according to the proximity of products.

In addition to supporting the user's awareness of his/her location, the system uses its awareness of its location and the location of products to inform the user of nearby products which are on his/her shopping list. When the user nears a product (or products) the system alerts the user and displays a list of nearby products. Tapping an item on the list shows its location - its icon is then highlighted on the map. In this state, only products that are considered nearby (i.e. listed in the popup) are shown on the map; all other icons are temporarily removed to reduce complexity (see Figure 2). All of these representations are offered to support the user's interaction with their context such that they may locate products more easily.

Since the small touch screen has somewhat limited visibility in the supermarket context, and due to the noted issues context-aware systems can suffer, the shopper has a Bluetooth headset, which is used for alerts. As such, we are able to use the aural channel for alerting, ensuring that the systems alerting functionality is not compromised by its limited visual output.

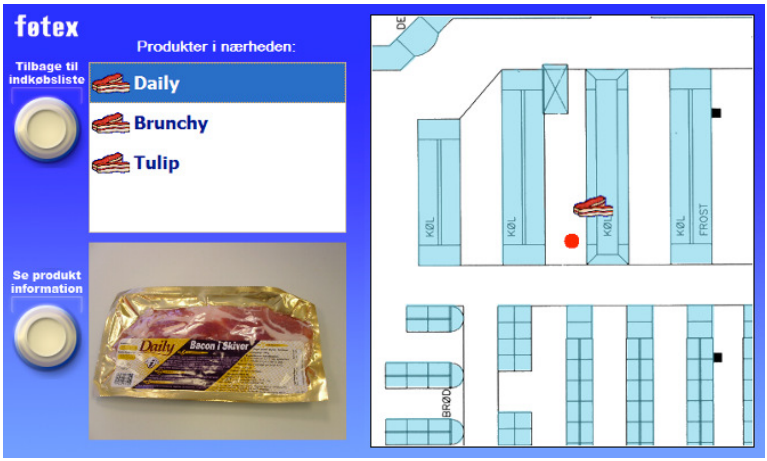


Figure 2: Screen shot of CAST. Nearby products - "Daily" bacon is highlighted on the map and its packaging displayed

3) Objects. Objects in the shopper's context in føtex are included in CAST in the form of the physical properties and states of products and shelves. CAST supports the shopper's interaction with objects in his/her context in multiple ways:

- Location of objects
- Visual appearance of objects
- Existence of relations between objects
- Descriptive information about objects

CAST's core functionality combines the first two dimensions by providing a photograph of products that are nearby. This photograph is included on the panel which appears when products are nearby, and is updated if/when the user selects a different nearby item from the list. CAST supports the shopper's knowledge of the existence of and relations between objects by showing similar and related products to those on the shopping list. The user can also select alternative products to the ones on his shopping list. CAST will then display a picture, and detailed information related to the product (see figure 1). By offering the map of føtex containing the physical layout of the store, the user is provided with contextual information related to the objects in the physical environment that would otherwise be unavailable.

4) People. CAST's design incorporates support of the supermarket's social context through its dynamically ordered shopping list. The shopping list component continues to display items which haven't been collected, but which have been 'passed' by the shopper, as the top item on the list. The system does so despite other products being physically closer to the shopper. In doing so, the system encourages the user to collect the 'missed' item before proceeding with the rest of their shopping and allows the user to avoid backtracking.

In order to achieve this functionality, the store was divided into four blocks, derived from the movement data obtained in our field study. These blocks are sequential, and create a 'loose' ordering for the products. CAST provides location-based updates to the shopping list and location-based notifications as long as the shopper collects the products in their current block before moving on to the next block in the sequence. For further information on CAST's implementation and design, see [1, 2].

4 Evaluation

We conducted a number of field trails to evaluate how shoppers could utilize and the context-aware information in CAST.

4.1 Method

18 participants took part in our evaluation; nine participated as users of CAST (M=35.11, SD=13.79, 1 female) while nine others participated using a traditional shopping trolley (M=27.44, SD=3.57, 3 females). Seven of these participants shopped at the east Aalborg branch of føtex at least once every month. Like the field study, a 23 item-shopping list was created, containing a blend of items which are considered daily goods, as well as items which are less frequently purchased. All participants were informed to envision a scenario where they had been sent shopping for all the members of their household. Thus, several items on the shopping list would be easily recognizable, while others would not be.

All nine CAST participants were introduced to the system and we gave them a brief introduction to the functionality of the system. Afterwards, all participants started the shopping activity. In a similar configuration to the field study, a test leader joined the participant to elicit feedback, and a logger recorded movement and product collection data. Each participant was equipped with an mp3 player to capture audio data from the shopping session. In our evaluation, context awareness and positioning was simulated from a Pocket PC using a Wizard of Oz approach. Following the shopping sessions, each participant was interviewed in the cafeteria and they were asked to complete a questionnaire including a workload assessment (NASA TLX test).

4.2 Findings

Our findings from the evaluation showed that CAST influenced the shopping activity and experience. CAST participants walked a significantly shorter route through the store than the traditional shopping trolley participants (app. 470 meters compared 620 meters for the traditional trolley participants). Furthermore, the participants using the traditional shopping trolley backtracked noticeably more than those using CAST. The only CAST users who backtracked were those who opted to collect items in a different order to

that recommended; these shoppers said they were aware of CAST's recommendations but felt there were closer items and opted to collect those instead. In terms of context-awareness, we can see that our implementation of location helped the participants navigate through the store. Furthermore, it was notable that shopping list participants visited aisles not visited by CAST participants when searching for products. Despite these differences, they were very similar in task completion time (app. 30 minutes on average in both conditions).

Our data indicated that the CAST participants found it easier to find products than the group using traditional tools. Traditional shopping trolley participants asked store staff for help 33 times in order to locate products, while only one CAST participant asked for help. The order in which the CAST users collected products was notably more uniform than the users of the traditional shopping tools. The CAST users picked up the same product at the same point in the product collection sequence 131 of 207 times, compared to 79 of 207 times for the trolley and paper list users.

Shoppers using CAST showed a tendency to devote an increasing amount of attention to the system as the session progressed. Initially the CAST users entered the shop floor and only glanced occasionally at the display. Following the first audio notification, and subsequent product collection, most users gradually reduced their sampling of the physical context until some became almost totally reliant on the system for guidance. Several CAST users reached points where, after blindly trying to move the red spot towards the icon on the map, they exhibited a moment of clarity.

Our data indicates that not only did CAST acquire the shoppers' attention at the key times intended by its design; it maintained possession of their attention to an unexpected degree. Also of note is what can be interpreted as attention division techniques on the part of the CAST users. While moving with the system, the shoppers could be observed repeatedly switching gaze between the system and the environment at short intervals. This is likely part of an orienting activity, where the user samples the environment and the map to construct a more complete understanding of his/her physical context.

5 Conclusion

Shopping in the real world is likely to become a more and more interactive experience where shoppers use interactive shopping trolleys and self-check out points. In this paper, we outlined the design of a prototype for enhancing the shopping experience in supermarkets and we called the system CAST (Context-Aware Shopping Trolley) as it directs shoppers through a supermarket by outlining product placement.

Our initial field trails showed that context-awareness provide an opportunity for enhancing and affecting the shopping experience. While using approximately the same amount of time for shopping, shoppers using CAST were more successfully in finding the listed products on the shopping list and they asked for help fewer times than traditional shopping trolley shoppers. Also, they seemed to adapt to a more similar sequence of collecting product in the store. Our research calls for further studies within interactive shopping trolleys. First, our findings seem to confirm that shoppers are open to engage with touch screen interfaces that provide information about the shopping activity. Further studies could investigate other types of information to be integrated into the trolley, e.g. interactive cooking recipes or product nutrition information.

Acknowledgements

We would like to thank the participating subjects in the two evaluations studies and several reviewers on previous versions of the paper.

6 References

[1] Black, D. and Clemmensen, N. J. (2006) Researching Context-Awareness for Supermarket Application - Appendices. Aalborg University

[2] Black, D. and Clemmensen, N. J. (2006) When More is Less: Designing for Attention in Mobile Context-Aware Computing. Aalborg University

[3] Bohnenberger, T., Jameson, A., Krüger, A., Butz, A. (2002) Location-Aware Shopping Assistance: Evaluation of a Decision-Theoretic Approach. Proc. Mobile HCI 2002. Springer Berlin, 18-20

[4] Chen, G. and Kotz, D. (2000) A Survey of Context-Aware Mobile Research, Technical Report TR2000-381, Department of Computer Science, Dartmouth College

[5] Cheverst, K., Davies, N., Mitchell, K., Friday, A., Efstratiou C. (2000) Developing a Context-aware Electronic Tourist Guide: Some Issues and Experiences. Proc. CHI 2000, ACM Press, 17-24.

[6] Dey, A. K. (2001) Understanding and Using Context. In Personal and Ubiquitous Computing, Vol. 5, Springer-Verlag, pp. 4-7

[7] Dey, A.K., Barkhuus, L. (2003) Is Context-Aware Computing Taking Control away from the User? Three Levels of Interactivity Examined. UbiComp 2003, Springer-Verlag, 149-156

[8] Metro Group Future Store Initiative (2009) www.futurestore.org Future Store Rheinberg. Viewed on Wednesday, June 07, 2006

[9] Pospischil, G., Umlauft, M., and Michlmayr, E. (2002) Designing Lol@, a Mobile Tourist Guide for UMTS. In Proceedings of the fourth Conference of Mobile Human- Computer Interaction, Springer-Verlag, LNCS, pp. 140-154

[10] Schilit B. N., Theimer M. M. (1994) Disseminating Active Map Information to Mobile Hosts. IEEE Network 8(5), IEEE, 22-32

[11] Skov, M. B, Høegh, R. T. (2006) Supporting Information Access In A Hospital Ward By A Context-Aware Mobile Electronic Patient Record. Personal and Ubiquitous Computing 10(4), Springer London, 205-214

What is That? Object Recognition from Natural Features on a Mobile Phone

Niels Henze
OFFIS - Institute for Information Technology
Escherweg 2, Oldenburg, Germany

Torben Schinke and Susanne Boll
University of Oldenburg
Escherweg 2, Oldenburg, Germany

Abstract

Connecting the physical and the digital world is an upcoming trend that enables numberless use-cases. The mobile phone as the most pervasive digital device is often used to establish such a connection. The phone enables users to retrieve, use, and share digital information and services connected to physical objects. Recognizing physical objects is thus a fundamental precondition for such applications. Object recognition is usually enabled using visual marker (e.g. QR Codes) or electronic marker (e.g. RFID). Marker based approaches are not feasible for a large range of objects such as sights, photos, and persons. Markerless approaches that use the image stream from the mobile phone's camera are commonly server-based which dramatically limits the interactiveness. Recent work on image processing shows that interactive object recognition on mobile phones is at hand. In this paper we present a markerless object recognition that processes multiple camera images per second on recent mobile phones. The algorithm combines a stripped down SIFT with a scalable vocabulary tree and a simple feature matching. Based on this algorithm we implemented a simple application which recognizes poster segments and conducted an initial user study to get an understanding of the implications that accompany markerless interaction.

Keywords: natural features, mobile phone, object recognition, markerless, camera

1 Introduction

Mobile devices with the mobile phone on its forefront are a ubiquitous part of our daily life. Not only because of their limited in and output capabilities there is an increasing interest in extending the interaction between the user and her phone to an interaction between the user, the phone and real world objects. Typical application are mobile tour guides which enables the user to point at sights [17, 1] to get further information, access related services for advertisements [10], or go shopping in physical stores [16]. To implement such an interaction with a real world object it is necessary that the mobile phone senses objects in its surrounding in some way.

Approaches to sense real world objects are usually based on visual markers (e.g. QR-Codes or other 2D barcodes) or digital markers (e.g. RFID tags). For certain types of objects, such as sights, buildings, and living beings marker based approaches are often not sensible or considerably restrict the interaction radius. Markerless approaches, for instance based on natural features, can overcome some of these limitations. However, they suffer from high demands on the available processing power. Thus, markerless object recognition is usually performed on a remote server (see e.g. [10]) or preformed on the mobile device with a delay of up to several seconds. Either way this delay clearly restricts the interaction.

The work by Wagner et al. [14] showed that estimating the 3D pose of a 2D object from natural features with a high frame rate is feasible on recent mobile phones. In this paper we build up on this approach and describe an algorithm, which combines it with a vocabulary tree to recognize a number of objects. Based on this algorithm we implemented a prototype (see Figure 1) to conduct an evaluation which offers first insight into the implications of real-time markerless object recognition that provides direct feedback to the user.

In Section 2, we present work related to object recognition, in particular, on a mobile phone. The developed algorithm is described in Section 3. We present a first user test in Section 4 and close this paper with a conclusion and outlook to future work in Section 5.

Figure 1: User interacting with an interactive poster.

2 Related Work

Fitzmaurice was one of the first who predicted the use of mobile devices for pointing based interactions [3]. He described for instance an application with which the user could point onto certain locations on a map in order to get additional information [3]. In recent years systems developed for mobile phones emerged which provide information related to physical objects. A common and commercially successful way to implement such systems is to mark objects with visual markers [11] or electronic markers [15]. However, not all types of objects are suitable for equipment with markers. Sights and buildings are simply too large or out of range to be reasonably equipped with either type of markers. It is questionable if objects whose visual appeal is important, can in general be sensibly equipped with visual markers (see e.g. [4]). In addition, markers restrict the interaction radius in a specific way.

Another approach uses images from a camera for the recognition of digital images to find the corresponding digital information using content based image analysis. Lowe presented the influential Scale Invariant Feature Transform approach (SIFT) [7] which allow the recognition of arbitrary physical objects. SIFT is invariant against rotation by design and robust against scale, light changes, partial occlusion, and perspective changes. A survey of local feature descriptors in general can be found in [8].

In recent years the size of the database whose content can successfully recognized increased dramatically. While Lowe reported to successfully recognize around 50 objects Nister et al. presented the recognition using a vocabulary tree [9] which enables to find an image out of two million images in two seconds on a high-end computer. Schindler et al. refined the vocabulary tree [13] and improved its performance by a factor of more than ten.

Recognition of a large number of objects is feasible if enough processing power and memory is available. However, even computing the widely used SIFT descriptor alone, without any matching or storage issues, overcharges mobile phones today, which is the reason why remote server processing is so popular. Liu et al., for example, use the camera of a mobile phone to select documents displayed on computer screens [5]. Erol and Hull use mobile phones' cameras to enable the user to select presentation slides by taking images of the slides [2]. Zhou et al. developed a system to acquire information related to sights by taking a photo [18]. However, recently Wagner et al. [14] adopted the SIFT and FERNS algorithms for mobile devices and estimate the pose of a single image with high frame rates.

3 Object Recognition Algorithm

Widely used object recognition approaches such as SIFT are too expensive in terms of processing power. SIFT consist of three steps: keypoint detection, feature description, and feature matching. Wagner et al. describe a simplified SIFT algorithm to estimate the 3D pose of a 2D object [14]. Their approach is capable to process camera frames with a size of 320x240 pixels at a rate up to 20Hz. However, only results from processing a single image are reported and it was not analyzed how the algorithm performs with an increasing number of objects. In the following we describe the extension of the

approach developed by Wagner et al. using a scalable vocabulary tree [9]. Our recognition pipeline is outlined in Figure 2.

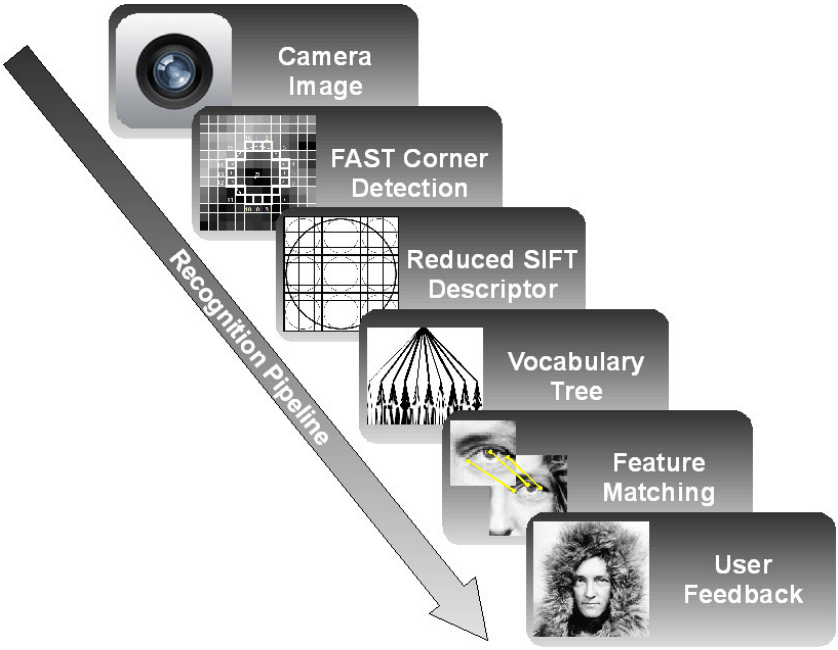


Figure 2: Overview of the recognition pipeline.

3.1 Keypoint detection and feature description

For the keypoint detection we adopt the simplified SIFT algorithm [14]. In the pre-processing phase images are successively downscaled with a factor of p2 to achieve scale invariance. For each scale level the image is smoothed with a 5x5 Gaussian kernel. Afterwards a FAST corner detector [12] detects keypoint candidates. During the pre-processing we detect up to 300 features per scale step. During the online phase the image is not downscaled detecting keypoints from a single image plane.

The keypoint candidates are feed into the creation of feature descriptors. Image patches around the origin of the candidate with a size of 15x15 pixels are used to derive the descriptor. First the patch's main orientation is derived

from the pixels' of the patch. The pixel's gradients are weighted by a Gaussian function and quantized to a natural number between 0 and 36. The result is inserted into an orientation histogram. For each peak in this histogram one feature is created.

The feature's according descriptor is again derived from the 15x15 pixel image patch. The patch is subdivided in 3x3 sub-regions. For each pixel the orientation is weighted by their distance to the patch centre as well as to the subregion centre and quantized to a natural number between 0 and 4. The 3x3 sub regions and the 4 quantization steps form a descriptor with 3x3x4=36 entries. The descriptor is further normalized and each of its values is cropped to 80% of the descriptor's overall length.

3.2 Feature matching

Wagner et al. employs a "Spill Forest" (a combination of a number of Spill Trees [6]) to match features extracted from the camera image with features from all scale steps of the reference image. Since our aim is to recognize a number of images we employ a different approach. Vocabulary trees [9] are able to reduce the problem to find the matching object by multiple magnitudes. Nister and Stewenius trained a vocabulary tree which reduced the problem to find an image out of two million to a problem to find an image out of a hundred candidates. Unfortunately the vocabulary tree described by Nister and Stewenius has a size of hundreds of megabytes and must be stored in RAM for performance reasons. We downsize the tree by reducing its level to five instead of six and a branching factor of eight instead of ten. In addition, our descriptor has only 36 entries instead of 128. Through this our empty tree needs only two megabyte. We trained our vocabulary tree with 10000 images, mainly high quality photos.

Reference images are inserted into the vocabulary tree by extracting the features from each of the image's scale steps. Each scale step is then treated individually and inserted into the tree. By treating the scale steps individually we obtain not only object candidates from the vocabulary tree but scale step candidates. During the online-phase three scale step candidates are retrieved from the vocabulary tree.

Since we do not aim at fine grained pose estimation no sophisticated feature matching is necessary. Thus, we rely on simple brute-force matching to compare the 100 features from the camera image with the 300 features from

each of the three scale-step candidates using the sum of squared difference. To further reject potential outliers we compute a difference of orientations histogram for each candidate's matches. If this histogram shows a consistent rotation and the respective candidate's number of matches is above a certain threshold in two consecutive camera images the according image is considered as a match.

3.3 Performance

The algorithm described above was implemented for Windows Mobile 6 devices using C. We tested the speed using an ASUS P535 Smartphone equipped with an Intel XScale PXA270 processor running at 520 MHz and 64MB built-in RAM (26MB RAM available for applications). The respective durations are averaged over a short test sequence with a resolution of 320x240 pixels. Initial smoothing of the camera image takes 10ms. Afterwards keypoints candidates are detected using 13ms and the descriptors are computed in 27ms. Finally the features are matched against the vocabulary tree containing 343 scale levels corresponding to 57 images in 6ms. The three best candidates are matches with brute-force in 44ms. The overall time to process an image from the camera accordingly takes 100ms.

4 User Study

In order to get a first impression of the implications that accompany markerless interaction with physical objects we conducted an early user study. Our aim was to observe how the participants interact if no visual marker highlight interactive areas.

4.1 Developed prototype

In order to conduct the user study we developed a simple prototype shown in Figure 3. The prototype displays the camera image in full screen. The camera image is constantly delivered into the recognition algorithm described in Section 3. If an image is recognized a small thumbnail of the recognized image overlays the camera image. The user can get details about the object by clicking the thumbnail with her finger.

Figure 3: ASUS Smartphone running the used prototype.

4.2 Method

Six male colleagues from the lab participated in the study. All were between 25 and 35 years old. The evaluation consisted of three tasks described in the following. The sequence of the second and the third task was randomized. We asked the participants to fill a NASA TLX questionnaire after the second and third task.

In the first task the 45x55 cm large poster shown in Figure 4 was used. The poster sketches a street setting and contains seven interactive regions. If a participant selects one of these regions the phone displays advises about how to behave in the respective traffic situation. The poster lay flat on a table. The participants were asked to find all interactive areas without knowing its number. It was up to the respective participant to decide when to end this task.

In the second and third task two very similar posters hanging on a wall were used. Each poster contained 24 clearly identifiable interactive regions. Figure 4 shows a cut-out of the poster. For some of these regions the thumbnail that was displayed, when a region was recognized, contained a question mark. If the participant clicked the thumbnail the phone displayed either a happy green or a sad red emoticon but each poster contained only one happy

emoticon. So the participants' task was to find the happy one. One aim of these two tasks was to get an understanding of the participant's behaviour if the recognition fails. Therefore, three regions were deactivated in the third task.

Figure 4: The poster used in the first task of the evaluation (left) and an extract of the poster used in the second task.

4.3 Results and Discussion

In the first task the participants found between three and six interactive regions ($\emptyset = 4.66$ $\sigma = 1.21$). No participant was able to find the region located in the upper right of the poster. All but one participant started by systematically scanning the poster in zigzag. After scanning the whole poster once some started to scan specific regions of the poster. All but one participant permanently aligned the phone with the orientation of the poster. Three participants held the phone in an almost constant height. Two participants mentioned that additional hints to surrounding interactive regions would be helpful and one participant said that it is difficult to remember the parts of the poster that were already scanned.

All participants managed to complete the second and third task. However, probably due to the small number of participants the NASA TLX showed no significant difference between the two tasks. Some participants rushed through these tasks and two did not even notice the three deactivated regions.

The longest time a participant tried to select one of these regions was around 20s. All but two participants permanently aligned the phone with the orientation of the poster. One participant rotated the phone by 90^{O} and one participant did not show a consistent behaviour. All but one participant focused most of the time on one region after the other so that the respective region approximately filled the phone's screen.

Because of the used methodology and the selected participant the study can obviously not be generalised. However, the results indicate that users intentionally align the phone with the object. This is consistent with the observation we made in earlier work. It could imply that the recognition pipeline can be simplified by removing orientation invariance in tasks such as ours. Unsurprisingly the participants had problems to find all interactive regions if these are not clearly distinguishable. When marking interactive objects is not feasible additional hints displayed by the phone could ease finding nearby objects.

5 Conclusions and Future Work

In this paper we described an algorithm which enables to recognize hundreds of objects on a mobile phone. We employ a FAST corner detection, stripped down SIFT descriptors, and a vocabulary tree combined with brute-force matching. The implementation is able to process about ten images per second on an Asus P535 Smartphone. The algorithm is used to implement a basic prototype to evaluate the implications of markerless image recognition on a mobile phone.

The implemented algorithm is far from being optimized and all stages can probably be improved in terms of speed and accuracy. In particular, the brute-force matching can be replaced by more sophisticated techniques. Furthermore, more detailed user studies are necessary to get a deeper understanding of the implications that accompany markerless object recognition. This is especially true if going beyond 2D objects by enabling interaction with 3D objects.

Acknowledgements

This paper is supported by the European Community within the InterMedia project (project No. 038419).

6 References

[1] N. Davies, K. Cheverst, A. Dix, and A. Hesse. Understanding the role of image recognition in mobile tour guides. Proc. of MobileHCI, 2005.

[2] B. Erol and J. J. Hull. Linking presentation documents using image analysis. Proc. of Signals, Systems, and Computers, 2003.

[3] G. W. Fitzmaurice. Situated information spaces and spatially aware palmtop computers. Communications of the ACM, 36(7), 1993.

[4] N. Henze and S. Boll. Snap and share your photobooks. Proc. of ACM Multimedia, 2008.

[5] Q. Liu, P. McEvoy, and C.-J. Lai. Mobile camera supported document redirection. Proc. of ACM Multimedia, 2006.

[6] T. Liu, A. Moore, A. Gray, and K. Yang. An investigation of practical approximate nearest neighbor algorithms. Advances in neural information processing systems, 2004.

[7] D. Lowe. Distinctive image features from scale-invariant keypoints. International Journal of Computer Vision, 60(2), 2004.

[8] K. Mikolajczyk and C. Schmid. A performance evaluation of local descriptors. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2005.

[9] D. Nister and H. Stewenius. Scalable Recognition with a Vocabulary Tree. Proc. of Computer Vision and Pattern Recognition, 2006.

[10] M. Pielot, N. Henze, C. Nickel, C. Menke, S. Samadi, and S. Boll. Evaluation of Camera Phone Based Interaction to Access Information Related to Posters. Proc. of Mobile Interaction with the Real World, 2008.

[11] M. Rohs and B. Gfeller. Using camera-equipped mobile phones for interacting with real-world objects. Proc. of Pervasive Computing, 2004.

[12] E. Rosten and T. Drummond. Machine learning for high-speed corner detection. Proc. of European Conference on Computer Vision, 1, 2006.

[13] G. Schindler, M. Brown, and R. Szeliski. City-scale location recognition. Proc. of Computer Vision and Pattern Recognition, 2007.

[14] D. Wagner, G. Reitmayr, A. Mulloni, T. Drummond, and D. Schmalstieg. Pose tracking from natural features on mobile phones. Proc. of ISMAR, 2008.

[15] R. Want, K. P. Fishkin, A. Gujar, and B. L. Harrison. Bridging physical and virtual worlds with electronic tags. Proc. of CHI, 1999.

[16] Y. Xu, M. Spasojevic, J. Gao, and M. Jacob. Designing a vision-based mobile interface for in-store shopping. Proc. of NordiCHI, 2008.

[17] T. Zeidler, B. Brombach, E. Bruns, O. Bimber, and P. Föckler. Phoneguide: museum guidance supported by on-device object recognition on mobile phones. Proc. of Mobile and ubiquitous multimedia, 2005.

[18] Y. Zhou, X. Fan, X. Xie, Y. Gong, and W.-Y. Ma. Inquiring of the Sights from the Web via Camera Mobiles. Proc. of Multimedia and Expo, 2006.

Separation of User Interfaces from Services of Ambient Computing Environments: A Conceptual Framework

Andreas Lorenz
Fraunhofer Institute for Applied Information Technology
Schloss Birlinghoven, 53754 St. Augustin, Germany

Abstract

The use of mobile and handheld devices is a desirable option for implementation of user interaction with remote services from a distance. This paper describes the design of a general solution to enable mobile devices to have control on services at remote hosts. The applied approach enhances the idea of separating the user interface from the application logic, leading to the definition of virtual or logical input devices physically separated from the controlled services.

1 Introduction

The design and implementation of interaction in ambient computing environments cannot rely on traditional input devices like mouse and keyboard. For remote interaction from a distance, [10] revealed a dramatical increase of the error rate using wireless mouse and keyboard compared to a handheld device. Whether a control device is suitable for an intended interaction depends on the capabilities, personal preferences, situation and task of the user. If the physical shape of the equipment causes complaints or errors in operation, then the interaction could be improved either by revised design of the input hardware or by freedom to switch to another input device more aligned to the task and the personal attributes of the user. For example, [2] developed six different devices and interaction techniques to operate a gaming application from the distance. The key lesson learned was that users

are interested in selecting input devices according to their own preferences and performance of that device [8].

The opportunity to use mobile devices is a desirable option to enhance interaction with remote services, in particular if the user is experienced in its operation. These shifts in usage of computer technology go hand in hand with re-thinking of user interface technology. Disconnecting the input from the remote service host requires fundamental research in system models and architectures [12]: "*Lots of good research into input techniques will never be deployed until better system models are created to unify these techniques for application developers.*" The research described in [9] elaborates the fundamental characteristics of a distributed interactive system and derives the technical components for transmission of user input from an input device to remote services.

The main objective of this work is to elaborate a generic solution enabling mobile devices to express user input to remote services. This paper introduces a framework using virtual input devices to specify the input of the user interface without constraints regarding metaphor, shape, location, or modality. The main requirements to the design of the framework are abstraction, architectural design, and being independent from hard- and software. The specification of the framework identifies the components, defines the relationships between the components and illustrates the data flow within an intended system. The approach enables developers to create interfaces that depend on the meaning of the input rather than on the concrete device.

2 The Separated User Interface

The key concept used in this work is separation of the user interface on the mobile device from the internal logic of the service of ambient computing environments. Developers of interactive systems create a logical separation between application and user interface, enabling higher specialization in development and flexibility of use [5]: "*Separation lets specialists develop the user interface and the application independently, promotes interface consistency across applications, and allows application functions to be added or combined in new ways.*" The separation makes user interface development more efficient because the design, building, and evaluation of the user

interface are separated from the code of the application. It implies that the user interface must have sufficient access to application internals in order to keep the user aware of the application semantics (the application objects, operations and effect of the interaction).

2.1 Architectures

Moran proposed an important model for the specification of user interfaces [11], decomposing the user interface into three components: Physical interface, communication, and conceptual component. In this view, only the conceptual component needs direct contact to the functional aspects of the system. On a semantic level, the concepts represent the system's functional capabilities and provide operations to the user for manipulating the system's state. This work has been influential to the development of architectures for separable user interfaces.

The first proposal of a user interface software architecture was probably developed by Edmonds [1]. The proposed architecture built directly upon Moran's concepts. The I/O processors transform physical input actions from the user into corresponding internal representations, and vice versa transform internal representation of processing results into physical output action(s) displayed to the user. The dynamic processor determines the action(s) that the computer system should take. The background tasks are the set of possible functions that may be performed by the background application.

A widely used architecture further elaborates the separation of the user interface from functional code. The Seeheim-model [3] consists of three components: Presentation, Dialog Control and Application Interface Model. The presentation covers all issues for controlling the visual appearance and physical device for the actual interface. The application interface model, also referred to as semantic interface, defines the interface to the functions of the application. It is a representation of the application from the viewpoint of the user interface. In between, the dialog control defines the structure of the dialog between the user and the application. It serves as a mediator between the presentation component and the application itself. It receives an input stream from the presentation component and the output stream from functional calls of application, defines the interaction and routes the information to the appropriate destinations.

More oriented to the realization of large or complex system, the Model-View-Controller (MVC [6]) uses a modular approach for separating the visual appearance and the user input components from other objects in Smalltalk-80. The model represents the data, functions and behavior of the system. The view (visually) presents the model to the user, and the controller updates the model on behalf of the user. Each model can be combined with several view/controller pairs, whereas each view/controller pair is bound to only one model. The approach uses the observer-pattern to notify and update all dependents when one object changes state.

3 Definition of the Framework

This work further separates the user interface from the application logic. In particular, the concept separates the user input elements, i.e. the controller of the MVC-approach, from the physical device and location of the ambient service host. This enables to move the controller to any computing device able to connect to the model and the view. Like for the MVC-approach, several controllers can connect to one service, potentially using different interaction metaphors for receiving input from the user.

The visual appearance and physical device of the presentation of the service to the user, the display of its internal state, and notification on updates of the model, i.e. the views of the MVC-approach, are designedly left untouched. The views remain on the service host or move with the controller to the mobile device. Like the MVC-approach supports to have several views connected with one model, it is also possible to have several views, duplicated views, or combined views on the service host and mobile device.

The separation of the input elements from the service requires to have a mechanism for accessing the application functions. The approach used in this work adheres to the event-based approach and observer pattern that achieved dominant position in terms of actual usage. From the point of view of the service logic, input is then received in form of events from any controller residing on a mobile device. The hardware and realization of the interface on the mobile device play a minor role; the source of the input remains abstract and is therefore labeled as "virtual" input device in this document.

For recurring development of similar software applications, the development and use of frameworks is a well-known technique to not only reuse code fragments but to elaborate the fundamentals of a potential solution [4]: "*Frameworks shoulder the central responsibilities in an application but provide ways to customize the framework for specific needs.*" Figure 1 illustrates the framework. In the center of the image, the virtual input device spreads across both hardware components, covers the input events and introduces virtual event delivery mechanisms. User input elements implement the specification of the virtual input device, and services hosting on the controlled device are observers to implementations of virtual input devices. A mechanism in between distributes the input as internal events of the virtual input device. Because the presentation of the service (i.e. the views) is not addressed in this work, a virtual input device does not specify feedback to the user on behalf of the controlled service.

Virtual Input Device: A virtual input device is a source of meaningful user input without constraints for physical shape of the device, type of user interface or interaction style.

The hot spots of the framework, where programmers add their own code to add the functionality specific to their own project, are illustrated as small rectangles at the left and right border of the virtual input device.

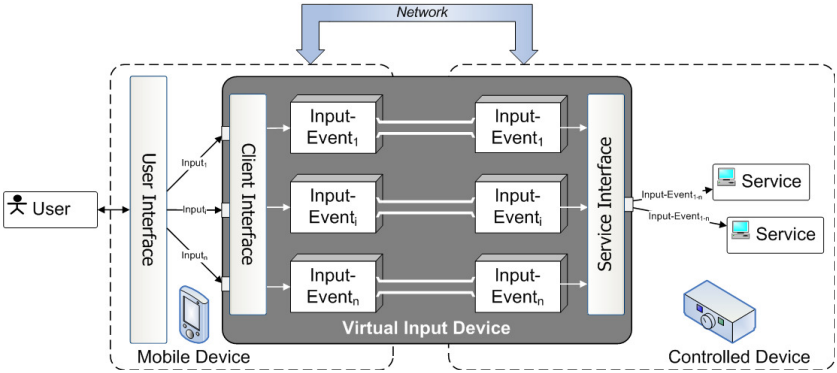


Figure 1: The Framework

3.1 Components

The virtual input device covers three main components: The definition of the input event, a client interface and a service interface. An additional component resides on the mobile device to realize the user interface. It implements the specification of the virtual input device. The user engages the provided input methods to express his demands. The recognition of the demand of the user from the interaction with the user interface is transferred into software events that are delivered to the controlled service. The physical device hosting the client component acts as the input device for the remote service.

3.1.1 User Interface

A realization of a way to express input from the user to the service with the mobile device. The user interface is the technical source of the input. It is supposed to adhere to common usability guidelines, for example the system's interpretation of activating a specific feature must be predictable by the user.

Examples: A graphical user interface on a mobile phone, a voice recognition tool, a gesture recognition engine.

3.1.2 Input Event

The event delivered from a client to a server. The input event determines the data exchanged between the distributed components. It denotes the type of the input and encloses event-specific data. Each input event has a name, an event source (usually the mobile device), and a list of parameters.

Examples: Button-pressed events, key-typed events, mouse- move events. The events include additional information like the character assigned with a pressed key or the position where a mouse click occurred.

3.1.3 Client Interface

The client interface is the software implementing the role of the information provider in the client/server approach. It is the application running on the mobile device, receiving input from the user interface and delivering the input in a feasible manner to the event-consumer.

The provision of methods is determined by the specification of input events of the virtual input device. As hot spots at the client side of the virtual input

device, the client interface provides a single method for each defined input event. This work distinguishes four types of client implementations, depending on their location and behavior: Local client, remote client, mobile client, and proxy client. This document exclusively addresses the remote and mobile client:

Remote Client.

The client implementation resides at another host than the server part. It uses the network channel for distributing events from the client implementation to the server side.

Example: An application mapping finger movements on an interactive table to movements of a remote mouse pointer.

Mobile Client.

A special case of the remote client. The location of the remote client is not relevant, and might not be known by the server part.

Example: An application mapping pen movements on a touch-sensitive screen of a mobile phone to movements of a remote mouse pointer.

3.1.4 Service Interface

The service interface is the software implementing the role of an information receiver in the client/server approach. It is the application running on the controlled device distributing the information to the local interactive services. As hot spot at the server side of the virtual input device, the service interface provides a subscribe-inform mechanism to services of ambient computing environments.

This work distinguishes four types of server instances, depending on their behavior and availability.

Single Server.

Only one object is active on the server host. Usually, it is one single service receiving specific input from single or multiple clients. There is only one single processing line on the server. Dedicated services allocate unique communication ports. The server exits when the processor exits.

Example: A single movie player application waiting for specific control commands.

Shared Server.

Multiple active objects share the same server instance. The services receive input of different types from multiple clients. The shared server instance distributes the events to registered processors. Different input devices access fixed communication ports. The server instance branches into different processing threads inside one single process. The server exits when the last processor exits.

Example: A multi-media application switching between different media players.

Persistent Server.

A special case of a shared server. The server is started by an entity other than the service provider (operating system, web-server, etc.). Multiple active objects share the server. Usually, the server is intended to be always on, no matter of available input devices or input processors. The server exits only on explicit shutdown.

Example: A standard application server.

System Queue Server.

A special case of a shared or a persistent server. Incoming events are integrated into the event queue of the operating system. Services do not directly receive events from the server instance, but indirectly from the local operating system. This approach enables for high integration with standard graphical user interface technology. It enables event delivery to any service already available on the operating system without changes of the service. The creation of events is restricted to meaningful default events available in existing user interface specifications.

Example: A server instance that integrates external mouse- events into the event queue of the local operating system.

3.2 Control Flow

Figure 2 exposes those details that are required to be implemented to apply the framework.

The specification of the virtual input device determines the methods of the client interface (hot spots). An adapter implementing the client interface logically provides access to the services in the environment. The user interface invokes the corresponding methods in order to submit the input events on request. From the point of view of the user interface, the adapter consumes the event locally. It returns to be ready to receive the next input from the user.

The consumption of the input event by the adapter implementing the client interface internally triggers the distribution of input events to the corresponding event notification part at the server side. The adapter handles the connection management with the remote service, encodes the event into the network representation, and transmits the message over the network.

The service interface maintains a list of event consumers. An adapter implementing the service interface retrieves the data from the network. It decodes the enveloped input event from its network representation, and notifies any service enrolled in the list.

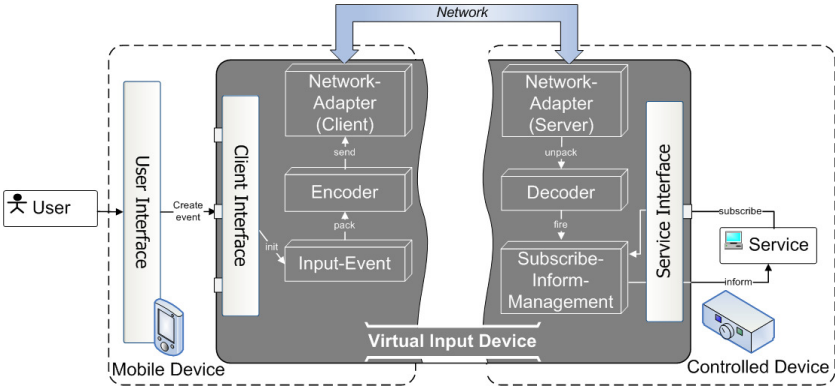


Figure 2: The details covered by the framework

4 Review of the Framework

The framework was evaluated in a technical review with four reviewers. The objective of the technical review is to diagnose whether the framework is adequately designed and documented to provide a basis for connecting user interfaces on remote input devices with services of ambient computing environments. The reviewers were employees from the department who were not involved in the development of the framework at all. Two reviewers are professional software developers, one is a psychologist and business information technologist, and one is a student in computer science. Three reviewers are male, one female.

For formal assessing the reviewers' statements and comments, a questionnaire of 19 items was used, each with a scale from "1" (strongly agree) to "7" (strongly disagree). The questionnaire was adopted from the "Computer System Usability Questionnaire"(CSUQ) of [7]. In average, all rates from the reviewers lie below the median of "4 ".

4.1 Positive Results

The reviewers were satisfied with the framework, its components and information flow. Three of four reviewers agreed, that the framework covers all required components. All reviewers agreed that the framework is free of overhead components.

The reviewers accepted the general purpose and overall design of the framework. The reviewers were able to match their solutions onto the architecture proposed by the framework. One of the reviewers elaborated a similar architecture only using different names for the components. Another reviewer would have needed a feedback channel towards the input device. However, the other reviewers explicitly agreed, that such a feedback channel is not necessarily part of the framework. It would be difficult to implement, especially if the input device offers no capabilities for rendering the feedback. In conclusion, the input device remains treated according to its name as pure information source.

4.2 Negative Results

The lowest level of satisfaction was associated with clarity of documentation, and finding information when required.

Regarding the usage of a framework, the reviewers provided inconsistent and sometimes contradictory statements. On one hand, they emphasized that a framework defines the architecture of a solution rather than provides software components; whereas other reviewers stated contradictory they missed libraries and usage like eclipse plug-ins.

5 Summary and Future Work

This paper described the design and specification of a framework for open human-computer interaction with services of ambient computing environments. The applied approach supports the physical separation of user input components from the service logic. The defined virtual input device allows for the specification of event exchange without constraints regarding shape, location and modality of the implementing user interface. The specified components cover the complexity of network management, connection establishment, and information exchange between client and server. The evaluation of the framework in a technical review agreed on the design of the framework. The main critics of reviewers addressed more careful documentation of the framework and its application.

Future work will elaborate to use standard technology of distributed systems for instantiation of the framework's components. Using standard technology from distributed system development will enable for implementation of special purposes, and to auto-generate source code from the specification of virtual input devices. This will deliver reference implementations to specialize the framework for a wide range of platforms and programming languages.

Acknowledgements

This research was supported by the European Commission within the InterMedia NoE (project No. 038419).

6 References

[1] Edmonds, E. The man-computer interface: A note on concepts and design. International Journal of Man-Machine Studies 16, 3 (1982), 231-236.

[2] Eisenhauer, M., Lorenz, A., and Zimmermann, A. Interaction-kiosk for open human-computer interaction with pervasive services. In Adjunct Proc. of Pervasive2008 (2008), Österreichische Computer Gesellschaft, pp. 134-137.

[3] Green, M. Report on dialogue specification tools. In User Interface Management Systems, G. E. Pfaff, Ed. Springer-Verlag New York, Inc., 1985, pp. 9-20.

[4] Hong, J. I., and Landay, J. A. An infrastructure approach to context-aware computing. Human-Computer Interaction 16, 2-4 (2001), 287-303.

[5] Hurley, W. D., and Sibert, J. L. Modeling user interface-application interactions. IEEE Software 06, 1 (1989), 71-77.

[6] Krasner, G. E., and Pope, S. T. A cookbook for using the model-view controller user interface paradigm in smalltalk-80. J. Object Oriented Program. 1, 3 (1988), 26-49.

[7] Lewis, J. IBM computer usability satisfaction questionnaires: psychometric evaluation and instructions for use. International Journal of Human-Computer Interaction 7, 1 (1995), 57-78.

[8] Lorenz, A. Lessons learned from open selection of input devices for a gaming application. In Adjunct Proc. MobileHCI2009 (2009). to appear.

[9] Lorenz, A., Eisenhauer, M., and Zimmermann, A. Elaborating a framework for open human computer interaction with ambient services. In Proc. PERMID2008 (2008), pp. 171-174.

[10] Lorenz, A., Fernandez de Castro, C., and Rukzio, E. Using handheld devices for mobile interaction with displays in home environments. In Proc. MobileHCI2009 (2009). to appear.

[11] Moran, T. P. A framework for studying human-computer interaction. In Methodology of Interaction, R. Guedij, P. ten Hagen, F. Hopgood, H. Tucker, and D. Duce, Eds. North-Holland, 1980, pp. 293-301.

[12] Olsen, D. R. Evaluating user interface systems research. In Proc. UIST'07 (2007), ACM Press, pp. 251-258.

Amazon-on-Earth: Wedding Web Services with the Real World

Amnon Dekel, Barak Schiller, and Niv Noach
The Selim and Rachel Benin School of Computer Science and Engineering,
The Hebrew University of Jerusalem
Jerusalem, Israel

Abstract

We describe the Amazon-on-Earth project which enables users to look for objects of interest, to navigate and find those objects in a physical space, and to pick them up, purchase them and walk out with those objects. We implemented a working prototype system in one of the libraries on our campus and ran an initial user study to see if there was any advantage to using the system relative to the existing library information services. Results show that users found information faster with the system, but because the subjects knew the library well, no advantage was found in the time it took them to navigate to the physical location of the book.

Keywords: Mobile, Interaction, Real World, Navigation, Tagging, Services

1 Introduction

For a number of years now efforts have been made in research and product circles to enable digital information services within the context of real world scenarios. The growing sector of powerful Smart Phones with their multiple embedded sensors and networking systems have made them a prime focus in consumer based location based services. These have spawned a number of commercial systems that can point out relevant physical services close to a person's current location [13]. Such systems are mostly used while driving or walking within a shopping center [7]. While our system uses location as an important dimension, we focus on using indoor navigation to enable users to

find objects in the physical world and to interact with them. Thus, our "objects" do not transmit their existence to the world (there are simply too many of them to make this feasible)- but once they are found by a user using the navigation maps provided, the system enables a number of relevant functions to be enacted in relation to the object.

2 Related Work

2.1 Mobile Interaction with the Real World

Smith Et. Al. [11] presented a prototype for mobile retail and product annotation services. Their system enabled the user to scan the object's barcode and received relevant information about that object which was found on existing web services such as Amazon. Their system used a special purpose barcode scanner to decode the object's ID for further querying (since then 1D and 2D visual tag decoding software have become available for most Smart phone systems). But their system did not help users find an object within a physical space, nor to conduct a transaction to buy the object if the user wished to. As the venue for this paper suggests, many additional research projects have focused on this space in the last few years [1, 2, 5, 6, 8, 9, 10, 12, 14].

2.2 Indoor Navigation

Nokia recently [7] announce a public trial of their Locate Sensor system in the Kampi shopping center in Helsinki. The system enables mobiles phones to track and present the location of special tags on the phone's screen. In this case the use was mostly for advertising- enabling a person to look for a specific store in the shopping center and receive promotional coupons relevant to their location.

Guinard, Streng & Gellerson's RELATE system [3] presents a system that identifies and shows the location of services relative to the mobile client device. The relative location of a service was shown by placing signifiers of the services as visual widgets on the sides of the screen in the direction that the service exists relative to the mobile device. Although such a method has a lot of promise in that it presents a way of discovering, navigating to and

using services without needing any user end installation processes, we feel that it is less suitable for our scenario since we deal with larger numbers (tens to hundreds) of objects that need to be navigated to and interacted with.

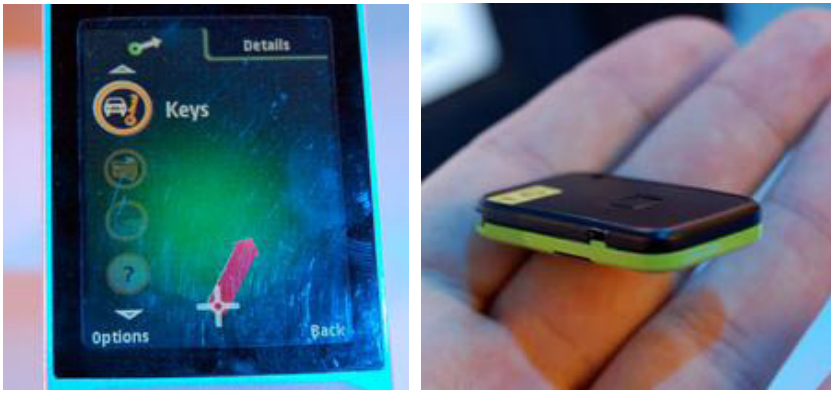


Figure 1: Nokia Locate Sensor system

Our project explores a method of enabling map based navigation in a physical space, but also offers pre and post object services. We looked for a simple and low cost way to enable a person to find the location of an object of interest. To us it did not matter what the technique is as long as it is acceptably accurate and robust enough.

3 Amazon-on-Earth

Our project focuses on enabling a person to do the following:

- Search for information about an object they are interested in
- Physically find that object in physical space
- Receive recommendations about it and relevant alternative objects
- Pick-up-n-Go: Pick the object up, purchase the object, and carry it out of a store.

3.1 System Test Location:

Our system has been implemented in one of the libraries in our campus. The reason for this was proximity and ease of access, and should not be taken to mean that we are focusing on libraries. The opposite is true- a library to us is a representation of a physical retail store. Such a store has stock (the books), a physical space to view the stock and handle it (the book cases and desks), and a check out counter where people can buy (lend) the books. To us, such a system is conceptually parallel to retail stores, while allowing us to explore and test flows and methods without the obvious difficulties involved in using a real store location. Thus, anytime we use the word "book" in the paper, it can be replaced with the phrase "physical object" which can be one of many: a music CD, a toaster, a refrigerator, etc. When the word "library" is used, it can be replaced with the word "store". Lastly, when the phrase "check out" is used, it can be replaced with "buy" or "purchase".

3.2 Scenario:

The AoE system enables our user with the following scenario:

Finding Information about a book

Our user is looking for a specific book they need for their work. They go to a web site and run a search for the book (using key words, authors or ISBN number). They receive an information page about the book and can browse the information which has been gathered from the Google Books and Amazon web sites using their public application programming interfaces (API's).

Adding a book to their personal list

If they are interested in the book, they can enter it into their book list after signing in to the system. They can now go to the "store" to the view book, and check it out.

Navigating to the book in the library

Once they arrive at the library, they launch the AoE mobile application and select the book they are interested in. This brings up a navigation map which shows them the path they need to take in order to reach the book. If they navigate properly, they will reach the book case that holds the book they are

looking for. If they get lost, they can walk to one of a number of public and centrally located navigation tags and snap (photograph) it. This will give them a new map with an updated path to reaching the book case.

If they find that the book is not there, they can get information about additional books that can be relevant for them. The other books can be snapped using their bar codes and available information about them can be viewed.

Taking the book with them

Lastly, if the user wants to take the book with them, they can select the Check Out option under the book screen and receive feedback that the book has in fact been checked out and that they can take it with them.

3.3 Technical Description:

Figure 2 presents the main Amazon-on-Earth (AoE) back end system modules:

External DBs

- External comments DB: uses Google Book information and Amazon's ratings and opinions
- External books DB: Use the library's web site
- Mapping system: Returns results as text (may include links to maps stored on the web)

Book Comments Database Manager (BCDM)

An item may have several servers where users' comments are stored. The BCDM layer supplies a convenient abstraction for multiplexing comments from and to several DBs.

Internal comments DB

Since not all DBs are writeable, and there might be none which are so, an Internal DB might be used to store users' comments.

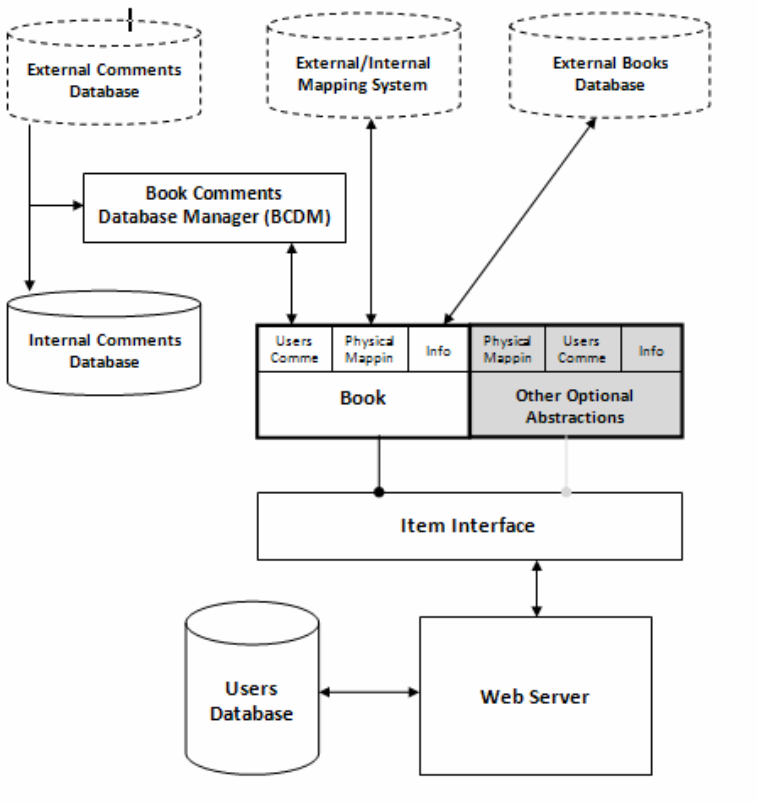


Figure 2: AoE System Architecture

Users DB

Stores data regarding users who are eligible to access the system (i.e. User names, passwords, and custom data: "My Books", Preferences).

Item interface

This is an interface for generalized access to information about an item. The information is separated into 3 sub-categories:

– *Info:* General information for identifying and describing the specific item.

- *Position:* Positioning (global and/or local, absolute and relative to a given point)
- *Comments:* Getting/adding comments capabilities (might use the BCDM interface if many sources for comments are available).

Mobile Client Application

The mobile part of the service is enabled via a native Android application that we developed and implemented on a G1 phone. The application is responsible for:

- Preserving internal state of the user's requests
- Initiating requests to the server(s) based upon requests.
- Parsing data returned from server(s) as response to requests.
- Displaying the incoming data.
- Interacting with Physical World Objects via the PWC (see below)

The Physical World Connectivity (PWC)

This module is responsible for acquiring and analyzing information from the world in the following methods:

- 1D/2D barcodes
- Keyboard input
- Optional: RFID and Voice Recognition

Output of this module is unified for all acquisition methods. At this point in time only 1D/2D barcodes and Keyboard input are supported.

Figures 3 – 5 present parts of the scenario in action.

Implementation

We implemented the system using a mixture of web based and native mobile technologies. The server modules were written in Python, and hosted within an apache HTTP server. We wrote a mapping application in .NET which created an xml map file for the library, above which the navigation path was drawn at run time with Python. The Databases were in SQL Lite.

The check out part of the scenario was only simulated, but we are working to implement a solution. All other parts of the scenario worked.

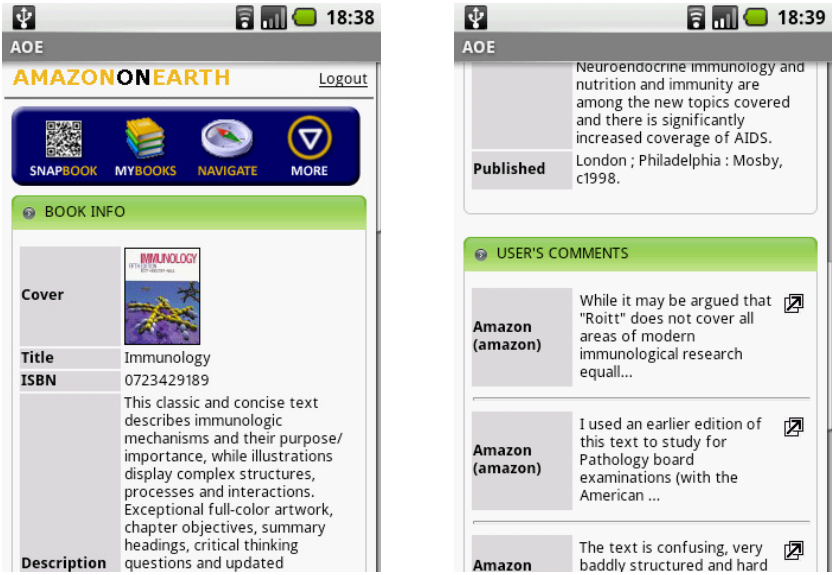


Figure 3: Book information and User Comments view

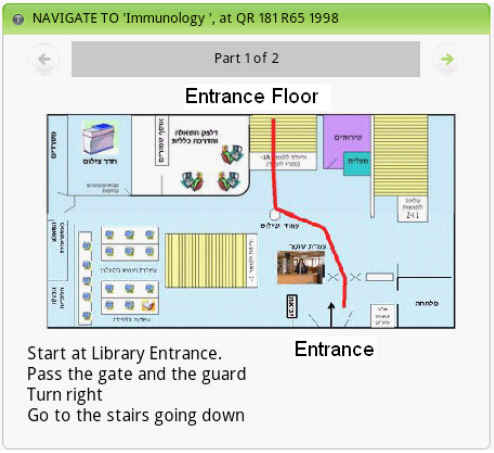


Figure 4: Navigation Map View to the Book

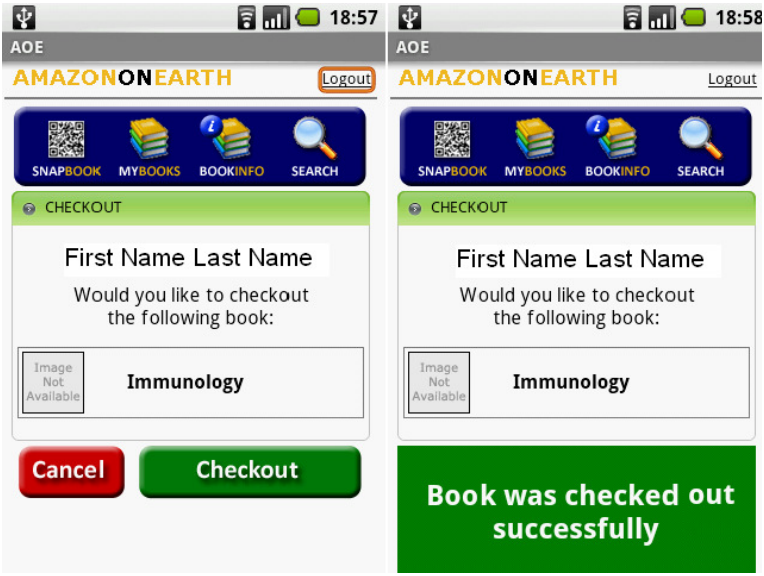


Figure 5: Book Check Out Interface

4 Initial User Study

Once we had most of the scenario working we ran an initial user study to understand if the move from paper prototype to working system had added any unforeseen issues. We ran the test with 10 subjects, none of which had used an android phone before, but all of which were students who frequently use the library. For the initial test we had them do two counter balanced tasks: a. Look for information about a book and find its location information from either the library catalog system or our system on the phone, and b. to physically navigate and find the book in the book cases.

4.1 Results

Finding the information about the book showed a clear advantage for the AoE system: it took users on average 34 seconds less to find the information about the book with our system (01:57 for AoE versus 02:31 for the search

without our system). We did not find any advantage in using the system in navigating to and finding the book on the shelf (AoE 04:26 vs. 04:31 without our system).

4.2 Discussion

The results seem to show that finding information about a book and it's location in the library is easier with our system. This is not surprising since the number of steps needed to achieve this with our application is somewhat smaller: enter search information about a book into a form, receive results and then click on the Navigate button. Compared with going to the old computer catalog systems, searching for the book, and finding its corresponding code and writing it down on a piece of paper, our system enabled the subjects to do this more than 30 seconds faster.

On the other hand we were surprised to see that no advantage was exhibited by our system in helping people physically navigate to the book. After a short analysis it quickly became evident that the subjects knew the physical layout of the library intimately and also knew the physical section of the library that houses the books that were used in the test. In such a case it is not surprising that our system did not exhibit any advantage.

5 Future Work

The implementation of AoE in the library is only beginning. First of all we will rerun the test described above to check if our system does in fact help people navigate to a physical location within a space when they do not know the place vey well. We will also be implementing a number of additional parts to the scenario and testing them:

- Implementing a physical check out system that is interfaced with the library check out system
- Testing system effects when a book is not in its place or missing
- Implementing electromagnetic tag systems for navigation and check out
- Exploring new interfaces for information and navigation, including Augmented Reality.

– Exploring new scenarios such as Physical E-Commerce with reverse auctions.

6 References

[1] Dekel, D. 2006. Finding the Path from Here to There: Some Questions about Physical-Mobile Design Processes. MIRW 2006. Workshop at MobileHCI 2006, Espoo, Finland.

[2] Geven, A., Strassl, P., Ferro, B., Tscheligi, M. and Schwab, H. Experiencing Real-World Interaction - Results from a NFC User Experience Field Trial. 9thInternational Conference on Human Computer Interaction with Mobile Devices and Services, MobileHCI 2007, Singapore, September 9 – 12, 2007.

[3] Guinard, D., Streng, S., Gellersen, H. (2007) Extending Mobile Devices with Spatially Arranged Gateways to Pervasive Services. In: International Workshop on Pervasive Mobile Interaction Devices (PERMID 2007) at the International Conference on Pervasive Computing, PERVASIVE 2007

[4] International Organization for Standardization: Information Technology – Automatic Identification and Data Capture Techniques Bar Code Symbology – QR Code. ISO/IEC 18004, 2000.

[5] Kindberg, T., Barton, J., Morgan, J., Becker, J., Bedner, I., Caswell, D., Debaty, P., Gopal, G., Frid, M., Krishnan, V., Morris, H., Pering, C., Schettino, J., Serra, B. and Spasojevic, M. 2002. People, Places, Things: Web Presence for the Real World. In: MONET Vol. 7, No. 5, pp. 365-376 (Oct. 2002).

[6] Kindberg, T. 2002. Implementing physical hyperlinks using ubiquitous identifier resolution. In: Proceedings of WWW 2002: pp. 191-199.

[7] Nokia Locate Sensor debuts at CES. [http://conversations.nokia.com/2009/01/12/nokia-locate-sensor-debuts-at-ces/]

[8] O'Neill, E., Thompson, P., Garzonis, S. and Warr, A. 2007. Reach Out and Touch: Using NFC and 2D Barcodes for Service Discovery and Interaction with Mobile Devices. In: Proceedings of Pervasive 2007: 19-36.

[9] Rukzio, E., Broll, G., Leichtenstern, K. and Schmidt, A. 2007. Mobile Interaction with the Real World: An Evaluation and Comparison of Physical Mobile Interaction Techniques. AmI-07: European Conference on Ambient Intelligence, Darmstadt, Germany, 7-10 November 2007.

[10] Semacode. http://semacode.org/

[11] Smith, M., Davenport, D., Hwa, H., Combs Turner, T. Object AURAs: A Mobile Retail and Product Annotation System. ACM-EC'04, May 17–20, 2004, New York, New York, USA

[12] Siorpaes, G. Broll, M. Paolucci, E. Rukzio,J. Hamard, M. Wagner, and A. Schmidt. Mobile Interaction with the Internet of Things. In Adjunct Proc. of Pervasive 2006 Late Breaking Results, 2006.

[13] UrbanSpoon. [http://www.youtube.com/watch?v=LQwUZe5Ms08]

[14] Välkkynen, P. and Tuomisto, T. 2005. Physical Browsing Research. PERMID 2005: 35-38

A Strategically Designed Persuasive Tool For An iPhone

Prithu Sah
Software and User Interface Design, National Institute of Design
Gandhinagar, India

Oliver Emmler
LifeSensor Product House, InterComponentWare AG
Walldorf, Germany

Abstract

WHO projects that by the year 2015, approximately 2.3 billion adults will be overweight and more than 700 million will be obese. This article is about designing a concept of a second generation persuasive tool for an iPhone and how it can help users in fighting obesity. There are numerous applications in the market which claim to aid users in fighting weight issues. What makes our concept different from others is the emphasis on usability at every stage of design process which is fundamental to success. We started off with research on iPhone user profiles, demographics and health, moved on to user interviews, requirement analysis, interaction models, use objects, information architecture, visual design, and ended up with Hi Fidelity clickable mock ups of the application. The application is intentionally designed to change a person's attitude or behaviour in a predetermined way. The final result is a robust and a user friendly persuasive tool with the age group of the target users being 18-40 years. The application leads the user through a step by step sequence of actions with relevant, customized interventions, providing the right kind of motivation and thereby providing a better user experience in turn making the process more engaging and enjoyable. The usability evaluation tests ensure that any potential issues are highlighted and fixed before the product is launched. The article also addresses the impact of usability on the final design and how it affects and is the key to the success of the application.

Gestural Control of Pervasive Systems using a Wireless Sensor Body Area Network

Oleksii Mandrychenko, Peter Barrie, and Andreas Komninos
Glasgow Caledonian University
70 Cowcaddens Road, Glasgow G4 0BA, UK

Abstract

This paper describes the prototype implementation of a pervasive, wearable gestural input and control system based on a full body-motion-capture system using low-power wireless sensors. Body motion is used to implement a whole body gesture-driven interface to afford control over ambient computing devices.

1 W-BAN BODY GESTURE CAPTURE

Our system is comprised of sensor "nodes" that can be attached to key locations on a user's body, monitoring the movement of major body parts, detailed technically in [2]. An internal processing system provides us with an updatable skeleton model of the use, which is a method also used by other researchers, e.g. [3]. The posture of the skeleton is calculated in real-time through forward kinematics. Kinematics simplifies computations by decomposing any geometric calculations into rotation and translation transforms. Orientation is obtained by combining (or fusing) these information sources into a rotation matrix – an algebraic format that can be directly applied to find the posture of the user. The result is a simple skeletal model defined as a coarse representation of the user. In general terms, gesture recognition consists of several stages, like feature extraction, pre-processing, analyzing and decision-making. Our experimental method consists of using linear angles between any two links in the skeletal model as a dataset that is fed into the gesture recognition algorithms described below. Analyzing

sequences of linear angles and performing the gesture recognition itself was implemented with the help of AMELIA general pattern recognition library [6], which we used as a basis to implement our own customized Hidden Markov Model. Our system allows users to record their own gestures for predefined actions that control the behaviour of ambient computing devices. As such, different actors may use diverse gestures, which can combine multiple body parts moving in different ways, for the same action. Typically, to record one gesture an actor repeats it for 3-4 times, as in [1] [5]. Once a few "recordings" of a gesture have been made, the system is then trained on the captured motion data set in order to be able to recognize the gestures. After training, the user can perform gestures in different sequences as well as performing actions that are not gestures. Our system recognizes gestures with the probability of 80-90% (determined experimentally). Examples of our gesture recognition systems are available to view online in video form[1].

At this point in time, our system has two limitations: Firstly, saving of the recorded gestures training data is not yet implemented (due to development-time constraints) but we consider it as a simple goal. Secondly, our current recognition model does not allow a gesture to stop in the actor's relaxed position. For example, if a user stands still and tries to record a gesture, finishing it at the relaxed posture, the recognition system will not determine when the gesture ends. However, this limitation will be removed in the near future.

2 CONCLUSIONS & FURTHER WORK

Our system is comparable to existing commercial offerings (e.g. XSens, EoBodyHF). These systems use sets of wired sensor packs, connected to a wireless hub, which transmit aggregated data wirelessly using Bluetooth or 802.15.4 respectively. Our system's advantage is that all sensors are wirelessly connected to a coordinator/transmitter node, which allows for improved wearability and flexibility in the configuration of the system, for full or partial body motion capture. We are particularly interested in its potential in mixed reality situations for gaming. We also wish to investigate issues in human-human interaction through embodied agents, controlled

[1] http://www.mucom.mobi/Projects/BodyArea

through the motion capture system. We are looking into the control of VR agents, as well as robotic agents for which the metaphor of "transferring one's soul" will be used to investigate response and interaction with other humans. Finally, we are interested in pursuing applications in tangible interfaces and semi-virtual artifacts, as well as gesture-based whole-body interaction with large situated displays. We hope to be able to create new types of human-computer interfaces for manipulating program windows, arranging or opening files using ad-hoc large projected or semi-transparent situated displays.

3 REFERENCES

[1] Philip Smit, Peter Barrie, Andreas Komninos. Mirrored Motion: Pervasive Body Motion Capture using Wireless Sensors. Whole Body Interaction Workshop, ACM CHI2009, Boston MA.

[2] Crossan, A., Williamson, J., Brewster, S., Murray-Smith, R., 2008. Wrist rotation for interaction in mobile contexts. In Proceedings of the 10th international conference on Human computer interaction with mobile devices and services. Amsterdam, The Netherlands: ACM, pp. 435-438.

[3] S. Rajko, G. Qian, T. Ingalls and J. James, Real-time Gesture Recognition with Minimal Training Requirements and Online Learning, IEEE Conference on Computer Vision and Pattern Recognition, 2007.

[4] Bodenheimer, B., Rose, C., Pella, J., Rosenthal, S. 1997. The process of motion capture: Dealing with the data. Computer Animation and Simulation. pp. 3-18

[5] AMELIA: A generic library for pattern recognition and generation: http://ame4.hc.asu.edu/amelia/ (link valid 5/09)