Finteraction

Finger Interaction with Mobile Phone

Mahsa Jenabi, Harald Reiterer
Human-Computer Interacton Group University of Konstanz D-78464, Konstanz, Germany
{mahsa.jenabi, harald.reiterer}@uni-konstanz.de

ABSTRACT
Touch interaction with mobile phones enable users to have a more natural interaction with the device, since touch is a natural way of direct accessing an object of interest. But one disadvantage is occlusion; i.e., the user loses a high percentage of information presented on the small screen of the mobile phone during the interaction using his ”big finger”. On the other hand, interaction enhanced with accelerometers can react according to the device movements, e.g., while the user is tilting the mobile phone, the UI will be rotated. A drawback is that the user needs to move the whole device while interacting and consequently loses his eye contact on the phone's screen (screen-absence problem).

Finteraction (Finger Interaction) is a new interaction concept that solves the occlusion and screen-absence problems. User interact with a large public display using his mobile phone. Moving the index finger in front of the camera at the backside of the mobile phone, the user can interact with the large public display even on the move.

Categories and Subject Descriptors
H5.2 [**Information Interfaces and Presentation**]: Interaction styles, input devices and strategies

General Terms
Design, Human Factors

Keywords
Input and interaction techniques, mobile interaction, interaction design, handheld device, gesture recognition

1. MOTIVATION
Mobile phones are small, technology-rich devices people carry around almost all the time. They store their contact information, as well as their appointments there, which makes mobile phones Personal Information Management (PIM) systems. Furthermore mobile phones can save user-specific configurations and preferences. Using these context information, the interaction designers can realize better fitting interaction possibilities for individuals. As an example the system gets the information from the mobile phone's calendar that the user has a meeting now. Consequently the mobile phone is switched to the silence mode and the answering machine is switched on. In addition the interaction meanwhile should calmly take place, without giving any audio feedback to the user.

These features make the mobile phone a good candidate as an input device for Large Public Displays (LPD). The number of LPDs is increasing in public places, such as airports, shopping malls, and bus stations, but mouse and keyboard are not any more proper input devices for such displays in ubiquitous computing environments, because there is no desktop metaphor available. Touch interaction is also not an option, since for LPDs the large size will make it difficult for the user to reach the display's upper corners. Using mobile phones in combination with LPDs can also improve the interaction in comparison with the mobile phone alone, since the small screen of the phone is not appropriate for high resolution data representation.

Existing research projects define new ways of interaction with mobile phones using one or more sensors, such as: camera, accelerometer, touch, microphone, speaker, light sensors, GPS, Bluetooth, WiFi, etc. Although these projects have improved the interaction with mobile phone, there are still a few problems, which are discussed in the following:

In a scenario that a user is presenting slides on an LPD using his mobile phone as an input device, and having eye contact with the participants in the meeting, or when the user is working with some images on the LPD and want to rotate them, the interaction technique should enable him to interact *eye-free*; i.e., without any need to keep eye focus on the mobile phone. We call the interaction techniques that need eye focus, ***eye-engaged*** interaction. Furthermore during a presentation the user should not need to turn to the display each time he wants to move forward to the next slide. He prefers to keep eye contact with the listeners. We call the interaction techniques that need the user to face the display ***focus-turning*** interactions.

In another scenario user navigates a geographical map on an LPD. More details about each region he selects is shown on the mobile phone's screen, e.g., the number of population, and languages people speak in this region. In this scenario during the interaction both GUIs of the LPD and mobile phone should be kept visible to the user. We call the interaction techniques that can not obey this rule, ***screen-absence*** interactions.

Another problem may occur when a user touches the display of his mobile phone for selecting some menus. He would not be able to see the pop up messages displayed on the screen, since his "big fingers" are covering a large percentage of the small screen of the mobile phone. This problem is called an ***occlusion*** problem, which is also mentioned in [10].

In order to have a clearer understanding about the above problems, we mention related work in the following section.

2. RELATED WORK

In order to give a similar feeling of playing golf in real life to users, Keir et al.[6] have introduced an interaction method that reacts against swinging a mobile phone in the air for playing a golf game on a mobile phone. This accelerometer-based interaction has the screen-absence problem, since moving the mobile phone in the air will prevent the user from keeping his eyes on the mobile phone's screen; i.e., the user can not see the visual feedback of the system and can not react to it.

Another new concept in HCI is the Organic Interface, which is about using flexible electronics, such as: organic batteries, circuits, sensors, and organic light emitting diode display [8]. Gummi[9] is a prototype based on this concept. There user interacts with the device by bending it. Bending the device would make the screen unstable and therefore can cause the screen-absence problem.

Some brands of mobile phones have integrated a touch display to their products for enhanced interaction. The Apple iPhone has even a multi-touch display. By touching the display, user can select, zoom, and type. Unfortunately the small screen size of the mobile phone may cause the occlusion problem during interaction.

Wigdor et al. [10] have introduced Lucid Touch: a transparent display that enables user to interact with the system by using touch (fingers) at the backside of the display. This technique has solved the occlusion problem, since the interaction takes place at the backside of the display and introduces a direct, absolute interaction with the mobile device itself. However it was not developed to support interaction in combination with LPDs, which would need an indirect interaction.

Ballagas et al.[2] have introduced the sweep technique that uses a mobile phone to control the cursor on an LPD. They use an optical flow algorithm that takes sequential pictures with the camera of mobile phone, while the user is moving the phone in the air. The difference between the pictures enables them to compute the relative motion and map it to the cursor's relative movement on the display. With this technique the user can concentrate on his intended task with the LPD and does not need to look at the mobile phone during the interaction (eye-free interaction). With long-term usage, sweep can be a relatively high fatigue interaction because of the arm movements[1]. Another disadvantage is the screen-absence problem while moving the mobile phone in the air.

Riisgaard et al.[5] have also experimented camera-based interaction with a mobile phone. Additionally a marker in front of the camera, drawn on a paper or a large display, is used for the relative positioning. The camera of the mobile phone recognizes the marker (circle) on the paper and computes the relative movement of the device. Using this method they can perform interactions such as zooming or panning. This technique is eye-engaged, since the user needs to concentrate on the circle marker he has drawn to make sure that the marker is on the camera view and this can take his concentration away from the actual interaction task with a specific application. E.g. when the user wants to work with images to move them or zoom them, he needs to look at the image itself and not a circle marker, which is drawn in front of him. Furthermore this method is not designed for mobile interaction; i.e., the users cannot walk around in a room and continue their interaction with the system, e.g. while presenting slides in a meeting. They may try to do it by carrying a paper with the circle marker around. But in this case the user needs to take care all the time that the marker stays inside the camera's view although both are on the move, which adds extra perceptual and motor load (according to [3]) to the user.

Maunder et al.[7] define SnapAndGrab, which is an interaction technique using mobile phone as an input device for interacting with an LPD. The user takes a picture from the media package of his interest, displayed on the LPD, and sends the picture to the SnapAndGrab display. The corresponding media object will be sent back to the user using the Bluetooth connection. This interaction technique can be used for selecting one item from a limited number of items. Interactions, such as zooming or scrolling or pointing is not considered in this method.

Although the previous techniques are promising experiments using mobile devices, but they have a few problems, namely: eye-engaged, screen-absence, non-mobility, and occlusion problems. Our interaction concept aims to overcome these problems. In the following section we introduce our concept.

3. NOVEL INTERACTION TECHNIQUE

In this section we introduce a new interaction concept for LPDs using a camera-based mobile phone. So the user controls applications on the LPD, using his mobile phone. Information is represented to the user on both LPD, and mobile phone screens. Using these two displays at the same time has, as an overview, an advantage that the general information can be shown on the LPD, e.g., the geographical map of the earth, and when the user clicks on a special part of his interest, the detailed information, e.g. the population of the region and climate, is shown on the mobile phone's screen. This detailed information can directly be saved in the mobile phone and the user can carry it with himself and even continue working on it on the move just using his mobile phone.

Our suggested interaction concept is called Finteraction. It uses the camera, which is integrated on the backside of the mobile phone, for interacting with an LPD. The connec-

Figure 1: Finger Interaction using the camera integrated in to a mobile phone.

tion between the mobile phone and LPD is accomplished via Bluetooth. Using a computer vision approach, there is no need for adding special hardware to this set. As illustrated in figure 1, user just needs to move his index finger in front of the mobile phone's camera, while holding the phone in the same hand. For designing this interaction concept, we have chosen our set of primitive tasks, namely: pointing, selecting, dragging, zooming, and scrolling, which are usually needed for controlling applications. The tasks of pointing, selecting, and dragging are in common with the ISO 9241-400 task primitives, but zooming and scrolling are additional. The reason is that we have defined an atomic interaction specific for scrolling and zooming that are used quite often, in order to make the interaction more efficient for the users; i.e. the user does not need to point to the scrollbar in oder to scroll, but an immediate scroll gesture will scroll the document pages. According to the Foley, et al.[3] definition of interaction task types, pointing, dragging, zooming, and scrolling are POSITION and selecting is the SELECT type.

Primitive tasks:
Pointing with the mobile phone will take place by moving the index finger to the left, right, up or down directions, which causes the relative movement of the pointer on the screen. For example when the user wants to move the pointer to the left on an LPD, he just needs to move his index finger from the center to the left. If he wants that the pointer continues to the left, he just needs to keep his finger on the left side, without any need for movement. Keeping the finger further from the center to the left would make the pointer movement quicker (similar to the functionality of joysticks). The movement on the other directions will be similar. The interpretation of the finger gesture can be defined differently in applications; e.g., in a geographic map or a photo album, moving the finger to the left or right, will move the digital objects on the screen or pan the map. In a music player application the same movement will play the "Next" or "Previous" music. The interpretation of the finger gestures is application-dependent.

Selection is easily done by taping the finger on the camera that basically means your finger covers the camera lens for a short time. Doing this tapping two times is considered a double click. Tapping may leave fingerprints on the camera lens, but our algorithm will be robust enough to continue the tracking correctly.

Dragging consists of these steps: pointing to the object, long selection (keep the finger longer at the camera, dwelling) and moving the object. That is accomplished by moving the finger to the direction that object is located on the display.

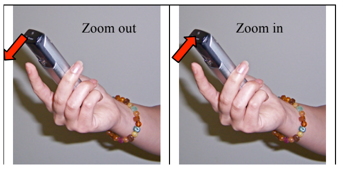

Figure 2: Zooming interaction is illustrated.

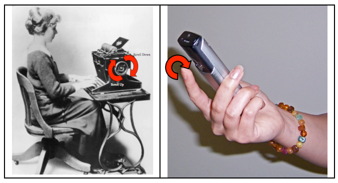

Figure 3: Shows a traditional typing machine. Turning the mechanical turning tool clockwise would move the paper upwards, and counter clockwise downward. We have used the same interaction.

Then clicking the object and again moving the finger to the direction of new location. The dwelling is necessary, since with a normal selection there would be no difference recognized between dragging an object on the screen and a simple click, which continues with a movement of the cursor to the other side. To avoid this ambiguity the selection in the dragging task needs to take longer. In the case of dragging the user gets a tactile feedback from the system after keeping his finger at the camera for a longer time, such as a vibration of the mobile phone.

Zooming is done by moving the index finger close to the camera for zoom in, and moving the finger away from the camera for a zoom out interaction (illustrated in figure 2). The relative movement of the finger will be mapped to zooming objects or images. This task is applicable e.g. in a geographic map or a photo album for zooming.

Scrolling is simply a cycling movement of the finger towards the camera and again away from the camera. Clockwise movement will scroll up and counter clockwise will scroll down. This is a standard movement of a mouse scroller for scrolling up and down. Even the same movement is used on old typing machines for pushing the paper up and down (see figure 3). Replacing the writing and typing tasks with gestures is not necessary, since the keypad gives a better opportunity to users to write. Furthermore switching between the primitive tasks will be automatically recognized by the algorithm; i.e., users do not need to specify it implicitly.

Finteraction uses just one hand for the interaction. The user holds the phone in one hand and move only his index finger of the same hand to interact with the system. The other hand will be free for parallel tasks. For example in the meeting scenario, mentioned in the previous section, the user can move to the next slides using his mobile phone with one hand, while keeping his slide notes in the other hand, or show some interesting parts on the display with the other hand. Finteraction is a mobile, eye-free interaction that solves the problems of occlusion and screen-absence. This continues, indirect interaction technique performs relative positioning, such as a mouse with three degrees of freedom(DOF), by determining (x,y,z) dimensions with the finger movements. Giving feedback to the user is important in any interactive system, but especially in the recognition systems, so that the user knows in which status the system is, whether it has recognized the gesture already, or has missed some gestures. Therefore we will provide visual and tactile feedback. E.g. when the user's finger gets out of the camera's view, a tactile feedback will warn the user to correct the interaction.

4. CONCLUSION AND OUTLOOK

We have described a new interaction concept using camera-based mobile phones as input devices, called Finteraction. The mobile phone plays the role of an input device for interacting with LPDs. This method maps the finger movement to commands (including pointing, selecting, zooming, dragging, and scrolling). An advantage of this method is that the user does not need to look at the screen or navigate through a specific menu for controlling a computer application on an LPD. This helps the user to focus on his intended task. The user does not even need to face the LPD. He may walk around the room and perform some finger gestures; i.e., the user experiences a fluid and mobile interaction. This method occupies just one hand and lets the other hand to be free for parallel tasks. Additionally the problem with display occlusion that exists with touch displays is solved here, since the interaction is done using the camera at the backside of the mobile phone without occluding the phone's screen with the finger.

Our next step would be to run a Wizard of Oz experiment to test the gestures and come up with a gesture collection that is confirmed by users. We will test 15 users, by showing them how each gesture can be performed with the device and giving them the chance to try out the gestures. Another person will act in the role of "wizard" by using a mouse to quickly perform the same interaction the user has done , but with a mouse, in a way that user sees the correct feedbacks from the system and has the feeling that his interaction with the mobile phone got this feedback. A camera will be installed as a "spy" to show the actions of the user to the wizard, who is invisible to the user. Questionnaires for usability evaluation will be prepared according to ISO 9241-410. Afterwards we will implement this concept on Nokia N96 mobile phone, which has a 5 megapixel high quality integrated camera, taking a computer vision approach, by extending an already existing algorithm[4]. This algorithm works for the finger tracking in front of the mobile phone's camera with further distances from the finger to the camera (interaction with extended arm). In their method they embed kalman filter and Expectation Maximization algorithms for tracking finger, and recognize the gesture using Hidden Markov Models (HMM). Therefore with extracting new features for HMMs, we will achieve a proper algorithm for finger tracking in closer distances to the camera for our Finteraction concept.

Using mobile phone as an input device gives us the advantage of one-time calibration; i.e., the calibration process, needed for tracking, is done for the owner of the mobile phone once and used by him all the time. For the input devices that are used by a larger user group the calibration needs to take place for each user separately, which can be time consuming and may still perform with higher error rate for individuals. After the implementation part, the prototype will be tested with users and the result will be analyzed, in order to see, if the hypothesis are fulfilled.

5. ACKNOWLEDGMENTS

This work is supported by DFG Research Training Group GK-1042 "Explorative Analysis and Visualization of Large Information Spaces", University of Konstanz and the project "Information at your fingertips – Interactive Visualization for Gigapixel Displays", which is supported by the "Information Technology Baden-Württemberg (BW-FIT)" funding program.

6. REFERENCES

[1] R. Ballagas. *Bringing Iterative Design To Ubiquitous Computing: Interaction Techniques, Toolkits, and Evaluation Methods*. PhD thesis, RWTH Aachen University, 2007.

[2] R. Ballagas, M. Rohs, and J. G. Sheridan. Sweep and point and shoot: phonecam-based interactions for large public displays. In *CHI '05: CHI '05 extended abstracts on Human factors in computing systems*, pages 1200–1203, New York, NY, USA, 2005. ACM.

[3] J. D. Foley, V. L. Wallace, and P. Chan. The human factors of computer graphics interaction techniques. *IEEE Comput. Graph. Appl.*, 4(11):13–48, 1984.

[4] J. Hannuksela, M. Barnard, P. Sangi, and J. Heikkilä. Adaptive motion-based gesture recognition interface for mobile phones. In *Computer Vision Systems*, pages 271–280. Springer Berlin / Heidelberg, 2008.

[5] T. R. Hansen, E. Eriksson, and A. Lykke-Olesen. Mixed interaction space: designing for camera based interaction with mobile devices. In *CHI '05: CHI '05 extended abstracts on Human factors in computing systems*, pages 1933–1936, New York, NY, USA, 2005. ACM.

[6] P. Keir, J. Payne, J. Elgoyhen, M. Horner, M. Naef, and P. Anderson. Gesture-recognition with non-referenced tracking. In *3DUI '06: Proceedings of the 3D User Interfaces*, pages 151–158, Washington, DC, USA, 2006. IEEE Computer Society.

[7] A. J. Maunder, G. Marsden, and R. Harper. Snapandgrab: accessing and sharing contextual multi-media content using bluetooth enabled camera phones and large situated displays. In *CHI '08: CHI '08 extended abstracts on Human factors in computing systems*, pages 2319–2324, New York, NY, USA, 2008. ACM.

[8] C. Schwesig. What makes an interface feel organic? *Commun. ACM*, 51(6):67–69, 2008.

[9] C. Schwesig, I. Poupyrev, and E. Mori. Gummi: a bendable computer. In *CHI '04: Proceedings of the SIGCHI conference on Human factors in computing systems*, pages 263–270, New York, NY, USA, 2004. ACM.

[10] D. Wigdor, C. Forlines, P. Baudisch, J. Barnwell, and C. Shen. Lucid touch: a see-through mobile device. In *UIST '07: Proceedings of the 20th annual ACM symposium on User interface software and technology*, pages 269–278, New York, NY, USA, 2007. ACM.