Handbook of Research on User Interface Design and Evaluation for Mobile Technology

Volume I

Joanna Lumsden
National Research Council of Canada
Institute for Information Technology – e-Business, Canada

Chapter XXIV
The Design Space of Ubiquitous Mobile Input

Rafael Ballagas
RWTH Aachen University, Germany

Michael Rohs
Deutsche Telekom Laboratories, Germany

Jennifer G. Sheridan
BigDog Interactive Ltd., UK

Jan Borchers
RWTH Aachen University, Germany

ABSTRACT

The mobile phone is the first truly pervasive computer. In addition to its core communications functionality, it is increasingly used for interaction with the physical world. This chapter examines the design space of input techniques using established desktop taxonomies and design spaces to provide an in-depth discussion of existing interaction techniques. A new five-part spatial classification is proposed for ubiquitous mobile phone interaction tasks discussed in our survey. It includes supported subtasks (position, orient, and selection), dimensionality, relative vs. absolute movement, interaction style (direct vs. indirect), and feedback from the environment (continuous vs. discrete). Key design considerations are identified for deploying these interaction techniques in real-world applications. Our analysis aims to inspire and inform the design of future smart phone interaction techniques.

INTRODUCTION

Today, mobile phones are used not just to keep in touch with others, but also to manage everyday tasks, to share files, and to create personal content. Consequently, our mobile phones are always at hand. Just as Mark Weiser suggested in his vision of ubiquitous computing, the ubiquitous nature of mobile phones certainly does make them *"blend into the fabric of our everyday lives"* (Weiser, 1991).

Technology trends show an increasing number of features packed into this small, convenient form factor. Smart phones already have eyes (camera), ears (microphone), and sensors to perceive their environment. However, their real power, as Weiser pointed out, comes not just from one device, but from the interaction of all of them. Our interest is in showing how modern mobile phones, which resemble Weiser's *"tabs,"* can be used as interaction devices for our environment. Within this environment, emphasis will be placed on interactions with public and situated displays (O'Hara, Perry, & Churchill, 2004) – what Weiser called *"boards."*

The range of input and output (I/O) capabilities for modern mobile phones is broad. Keypad, joystick, microphone, display, touch-screen, loudspeaker, short-range wireless connectivity over Bluetooth, WiFi, or infrared, and long-range wireless connectivity via GSM/GPRS and UMTS all provide multiple ways of interacting with our phones. These multiple I/O capabilities have increased our ability to use mobile phones to control resources available in our environment, such as public displays, vending machines, and home appliances.

Could this ubiquity mean that mobile phones have become the default input device for ubiquitous computing applications? If so, then mobile phones are positioned to create new interaction paradigms, similar to the way the mouse and keyboard on desktop systems enabled the WIMP (windows, icons, menus, pointers) paradigm of the graphical user interface to emerge and dominate the world of personal desktop computing. However, before this potential is realized, first we must consider which input techniques are intuitive, efficient,

and enjoyable for users and applications in the ubiquitous computing domain.

EXAMINING THE DESIGN SPACE OF INPUT DEVICES

Recent research demonstrates a broad array of mobile phone input techniques for ubiquitous computing application scenarios. To make sense of the cumulative knowledge, we systematically organize the input techniques to give insights into the design space. The design space is an important tool for helping designers of ubiquitous computing applications to identify the relationships between input techniques, and to select the most appropriate input technique for their interaction scenarios. Design spaces can also be used to identify gaps in the current body of knowledge and suggest new designs (Zwicky, 1967).

Looking to Foley, Wallace, and Chan's classic paper (Foley et al., 1984), we find a taxonomy of desktop input devices that are structured around the graphics subtasks that they are capable of performing (POSITION, ORIENT, SELECT, PATH, QUANTIFY, and TEXT ENTRY). These subtasks are the elementary operators that are combined to perform higher-level interface tasks, and will be elaborated upon in later sections. In this chapter, we structure our analysis of smart phones as ubiquitous input devices using this taxonomy. This analysis builds on classic design spaces (Buxton, 1983; Card, Mackinlay, & Robertson, 1991) and extends our own previous work (Ballagas, Ringel, Stone, & Borchers, 2003, Ballagas, Rohs, M., Sheridan, J., and Borchers, 2006) on the design space of input techniques. In our analysis, we blur the line between smart phones and personal digital assistants (PDAs) because their feature sets continue to converge.

Although Foley et al.'s analysis was completed with the desktop computing paradigm in mind, the subtasks in their analysis are still applicable to ubiquitous computing today. They naturally apply to situated display interactions; however, their applicability is not limited to graphical interactions.

In the following sections, each of Foley et al.'s subtasks will be examined in the context of mobile phone interactions. Foley et al.'s taxonomy uses the following input characteristics to further classify input techniques:

- **Feedback:** Continuous interactions describe a closed-loop feedback, where the user continuously gets informed of the interaction progress as the subtask is being performed. For example, when using a mouse, the current cursor position is continually fed back to the user. Discrete interactions describe an open-loop feedback, where the user is only informed of the interaction progress after the subtask is complete. For example, when selecting an object on a touch panel, the progress of the selection is not displayed until after the finger meets the surface to complete the selection of the desired item.
- **Interaction Style:** In direct interactions, input actions are physically coupled with the user-perceivable entity being manipulated (such as an image on a display). Physical coupling can be achieved when the feedback spatially coincides with the input action, or can be achieved at a distance if the user is manipulating a 3-D ray (such as with a laser pointer) that intersects directly with the entity being manipulated. To the user, this appears as if there is no mediation, translation, or adaptation between input and output.

In indirect interactions, user activity and feedback occur in disjoint spaces (e.g., using a mouse to control an on-screen cursor). Scaling and abstraction between input actions and feedback are often necessary in indirect interactions.

Position

During a POSITION task, the user specifies a position in application coordinates, often as part of a command to place an entity at a particular position. Positioning techniques can either be *continuous,* where the object position is continually fed back to the user, or *discrete,* where the position is changed at the end of the positioning task. Positioning tasks can further be differentiated using the directness of the interaction. In *direct* interactions, input actions are physically coupled with the object being positioned; in *indirect* interactions, user activity and feedback occur in disjoint spaces. We note that position could refer to screen position, or physical position in the real world. For example, the height of motorized window blinds can be adjusted using the position subtask.

The mobile phone has been used for positioning tasks in a variety of ways:

Continuous Indirect Interactions

1. **Trackpad:** A trackpad is a touch-sensitive surface that is used as a relative pointing device, standard in modern laptops. Remote Commander (Myers, Stiel, & Gargiulo, 1998) enables individuals to use the touch screen on a PDA as a trackpad to control the relative position of a cursor on a remotely situated display. In this interaction, the user's attention is concentrated on the situated display and no application-level feedback is provided on the PDA; thus, the functionality of the PDA is essentially reduced to an input device.

2. **Velocity-controlled joystick:** A return-to-zero joystick controls the velocity of an object (such as a cursor) that is continuously repositioned on the display. Zero displacement of the joystick corresponds to no motion (zero velocity). Positioning with a velocity-controlled joystick (a temporally and spatially constrained task) has been shown to be inferior to positioning with a mouse (a spatially constrained task) for desktop pointing scenarios (Card, English, & Burr, 1978). Silfverberg et al. (Silfverberg, MacKenzie, & Kauppinen, 2001) have done an in-depth study of isometric joysticks on handheld devices to control the cursor on a situated public display. Many of today's mobile phones are shipping with simple joysticks with a push button for menu navigation.

3. **Accelerometers:** Accelerometers are beginning to emerge in handheld devices. For example, Samsung's SCH-S310 mobile phone comes with an integrated 3-D accelerometer. Several researchers (Bartlett, 2000; Harrison et al., 1998; Hinckley and Horvitz, 2001) have proposed interactions that allow users to scroll (e.g., through an electronic photo album) by tilting the handheld device. The scrolling is typically activated through a clutch mechanism, such as squeezing the sides of the device (Harrison, Fishkin, Gujar, Mochon, & Want, 1998). The degree of tilting controls the speed of scrolling, making this a temporally constrained positioning task similar to the velocity-controlled joystick. Although these techniques were used to interact with an application directly on the device, they could clearly be extended to positioning tasks in ubiquitous computing environments.

4. **Camera tracking:** C-Blink (Miyaoku, Higashino, & Tonomura, 2004) rapidly changes the hue of a color phone screen to allow an external camera system to track the phone's absolute motion for cursor control on a large public display. The hue sequence encodes an ID to allow multiple users to interact simultaneously and control independent cursors.

The Smart Laser Scanner uses a laser combined with a wide-angle photo detector (see Figure 1) to detect relative finger motion in 3-dimensional space (Cassinelli, Perrin, & Ishikawa, 2005). The laser beam is steered with a two-axis micro-mirror. The tracking principle is based on the backscatter of a laser beam. When the backscatter is disrupted, the motion is deduced from the angle of the backscatter, and the laser is repositioned for the next measurement. Like other tracking techniques, it is possible for the device to lose track if the finger moves too fast, but input can easily be resumed by repositioning the finger to the laser. The research prototype of the tracker is fast enough to track the motion of a bouncing ping-pong ball.

5. **Motion detection:** With the Sweep (Ballagas, Rohs, Sheridan, & Borchers, 2005) interaction technique, the phone is waved in the air to control relative cursor motion on a remote screen (see Figure 2). This is accomplished using motion detection, an image processing technique involving rapidly sampling successive images from the phone's camera and sequentially comparing them to determine relative motion in the (x, y, θ) dimensions. No visual tags are required. The screen on the phone can be ignored, and the camera does not even need to be pointed at the display. A

Figure 1. The Smart Laser Scanner: A 3-D input technique for mobile devices using laser tracking (Cassinelli et al., 2005). Reprinted with permission from the authors.
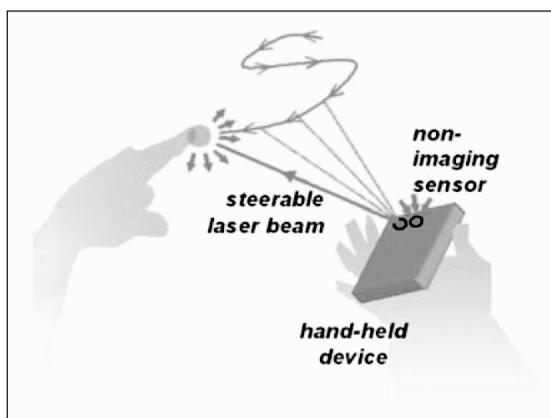
clutch mechanism, such as a button press, is used to activate the Sweep interaction. The clutch can be used to reposition the arm, similar to the way a mouse can be lifted to be repositioned without additional cursor motion.

6. **Location detection:** Location of the phone can also be used as input, where the user moves through physical space. Mogi (Licoppe & Inada, 2006), for instance, is a phone-based persistent item collection and trading game where the absolute geo-position of a subscriber correlates to the position in the game world. Mogi combines GPS (global positioning system) technology built into the phone with information from different mobile infrastructure towers from the network service provider to determine the player's position.

Continuous Direct Interactions

7. **Camera tracking:** Madhavapeddy, Scott, Sharp, and Upton (2004) present camera-based interactions involving tagging interactive GUI elements such as sliders and dials (see Figure 3). In manipulating the position and orientation of the phone camera, the user can position a graphical slider, or orient a graphical dial. Similarly, Direct Pointer (Jiang, Ofek, Moraveji, & Shi, 2006) uses a handheld camera to track the standard cursor on the display. An analogy

can be drawn to the classic light pen with a tracking cross. As the light pen moves to a new position, the cross follows the motions of the pen. Tracking may be lost if the pen is moved too fast, but can be easily resumed by repositioning the pen back to the tracking cross. Madhavapeddy et al.'s interactions rely on the tagged GUI widget instead of a cross for tracking; in Direct Pointer, the mouse cursor is the modern equivalent of the tracking cross.

In these tracking examples, the handheld device is responsible for tracking. An alternative is to use a tracker in the environment to track the output from a handheld device. For example, smart phones have been augmented with laser pointers, as in Patel and Abowd (2003), making them suitable for positioning tasks, described by Dan, Olsen, and Nielsen (2001), that use a camera in the environment to track the laser.

The mobile phone can also be passively tracked using a camera in the environment, such as in VisionWand (Cao & Balakrishnan, 2003). The user holds a passive handheld device that is augmented with distinctive markings (such as colored balls) at each end. Using two fixed cameras to perform stereo tracking, a 3-D ray can be deduced from the orientation of the markings in the stereo view, assuming the distance of the markings on the device is known *a priori*. This allows using a projection of the ray as a pointing device for a fixed remote screen.

Figure 2. The Sweep technique uses camera input and optical flow image processing to control a cursor (Ballagas et al., 20*05). © 2006 IEEE. Adapted with permission.*

The result is an interaction that is very similar to pointing using a laser pointer, except the ray is not a visible beam of light. This technique has an advantage over the standard laser pointer in that it provides an extra dimension of information: the distance to the display. The disadvantage of this interaction is that it is vulnerable to occlusion (e.g., by the users' own body) bringing into question the robustness of tracking in practical scenarios, although different camera configurations (such as from overhead facing downward) may solve these issues for certain interaction scenarios.

Discrete Indirect Interactions

8. **Directional step keys:** The location of an object is controlled using up, down, left, and right step keys for 2-D applications, plus in and out for 3-D. In the Blinkenlights project (Chaos Computer Club, 2002), users played the arcade classic "Pong" using the side of a building as a large public display. Each window equaled one pixel on the 18x8 pixel display. Players connected to the display by making a standard voice call to a phone number. Pressing the number 5 on the phone keypad moved the paddle up, and the number 8 moved it down. The server controlling the "Pong" application would decode the tones generated from the key activity during the phone call and use them as application

input. One of the notable things about this interaction is that it used the lowest common denominator of phone technologies. The communications channel was the standard voice channel, and the input was the numeric keypad, requiring no additional hardware or software besides what standard phones provide.

Discrete Direct Interactions

9. **Camera image:** Using the Point and Shoot (Ballagas et al., 2005) interaction technique, the user can specify an absolute position on a public display using a crosshair drawn over a live camera image on the mobile phone. To make a selection, the user presses a button while aiming at the desired target.[1] The button press triggers a brief overlay of a grid of 2-D tags over the large display contents, as can be seen in the middle of Figure 4. The grid allows the phone to derive a perspective-independent coordinate system on the large display that is enabled by the special properties of the Visual Code tags (Rohs, 2005a). Only one visual tag is required to establish a coordinate system, but a grid is used to increase the probability of having one tag entirely in the camera view. The drawback of the current implementation is that the tag grid is disruptive in multiuser scenarios, but future implementations could, for example, display the tags in infrared so

Figure 3. Using the phone to manipulate tagged widgets such as buttons, dials, and sliders (Madhavapeddy et al., 2004). Reprinted with permission from the authors.

that they are visible to the camera but not to other users.

Point and Shoot is related to the classic light pen, where position is discretely determined by displaying a raster scan when the user clicks a button on the light pen. When the raster scan is temporally sensed by the pen, the position of the pen is known because of a tight coupling between the pen clock and display clock. In Point and Shoot, a visual tag grid replaces the functionality of the raster scan except its mechanics are spatial rather than temporal. The lack of temporal dependencies makes Point and Shoot robust to different display technologies and the loose coupling between camera and display.

The breadth of positioning techniques is relatively large, making it difficult to choose which technique is most appropriate for a particular application scenario. To help with this selection, it is important to examine different figures of merit for each device.

Evaluating Positioning Techniques

There have been only a handful of thorough evaluations of the different ubiquitous mobile input techniques (Ballagas et al., 2005; Myers, Bhatnagar, Nichols, Peck, Kong, Miller, & Long, 2002; Silfverberg et al., 2001; Wang, Zhai, & Canny, 2006), as the field is still relatively new. These studies are difficult to directly compare, since they each used different experimental parameters, and some evaluations were not done in the context of ubiquitous computing interaction scenarios. Therefore, rough estimates for a variety of ergonomic measures are used to create a high-level comparison table for the positioning task presented in Figure 6. These rough estimates are derived using our knowledge of the interaction techniques for mobile phones and the collective knowledge of their desktop computing counterparts. The ergonomic parameters are mostly borrowed from Foley et al.'s survey of interaction techniques.

The evaluation measures are grounded in psychological and physiological foundations. Card et al. (Card, Newell, & Moran, 1983) provide an integrated survey of the various fundamental theories in a way that makes them more accessible and easier to use during analysis. Central to this work is the human processor model, which brings knowledge of the perceptual, cognitive, and motor processes of a human together under a single model. Ideally, a user interface minimizes the work required for each of these basic psychological processes.

The comparison table also incorporates various ergonomic measures designed to capture the efficiency of users executing the subtask, the accuracy they can achieve, and the pleasure the user derives from the process. The individual measures used in our comparison table are as follows:

- **Perceptual load** refers to the difficulty for the user to recognize, with their own senses, the physical stimuli and feedback of the interaction. For example, in the Point and Shoot interaction, users need to shift their perceptual attention between a large display and the phone screen to isolate a target in the phone camera view, leading to a comparatively high perceptual load.
- **Cognitive load** refers to the difficulty for the users to organize and retrieve information related to the interaction technique.
- **Motor load** refers to the number of motor steps required to execute the action after the appropriate action has been determined in the cognitive process. For example, Mogi is classified as a high-motor-load technique because the user needs to physically move at the city scale to specify the necessary position.
- **Motor acquisition time** characterizes the amount of time for the processes involved in the interaction technique (i.e., reaching for an object, moving to a certain target area, rotating to a certain orientation, etc.)
- **Visual acquisition time** characterizes the amount of time it takes to perceive the physical stimuli of the interaction technique.

Figure 4. Point & Shoot interaction: (Left) The phone screen is used to aim at a puzzle piece on a large situated display. (Middle) Pressing the joystick indicates a selection and a Visual Code grid is briefly superimposed to compute the target coordinates in the captured photo. (Right) The grid has disappeared and the target puzzle piece is highlighted on the large display, indicating successful selection (Ballagas et al., 2005). © 2006 IEEE.

Figure 5. Summary of positioning techniques using a smart phone as an input device. © 1984, 2006 IEEE. Adapted by permission.

	Foley	Mobile Phone Interactions	
	In Environment	On Phone	In Environment
Direct Pick	Search for Light Pen (Raster Scan)	Camera + On-Screen Cursor [Point & Shoot]	Visual Tags
		Laser Pointer + Camera [Olsen]	
	Light Pen Tracking	Camera Tracking + [Madhavapeddy]	Visual Tags
		Camera Tracking + On-Screen Cursor [Direct Pointer]	
		Vision Markers + Camera [VisionWand]	
		Camera + Touch-Screen	
Direct with Locator Device	Touch Panel		
Indirect with Locator Device	Mouse	Camera [Sweep]	
		Display + Stationary Camera [C-blink]	
	Joystick (Velocity)	Joystick (Velocity) [Silfverberg]	
	Tablet	Trackpad [Remote Commander]	
	Trackball		
	Cursor Control Keys with Auto-Repeat	Cursor Control Keys with Auto-Repeat	
	Joystick (absolute)	Accelerometer [Harrison + Rock 'n' Scroll]	
		Steerable Laser + Wide Angle Photodetector [Cassinelli]	
Indirect with Directional Commands/ Button Push	Up-Down-Left-Right Arrow Keys	Directional Step Keys [Blinkenlights]	
	(See Selection)	(See Selection)	
Location Detection		GPS + Cell Network Towers [Mogi]	
Numerical Value/ Numerical Coordinates/ Character String Name	(See Text Input)	(See Text Input)	

Figure 6. Rough estimates of ergonomic measures to compare mobile-phone-based positioning techniques (small circle = low, medium circle = medium, large circle = high). © 2006 IEEE. Adapted by permission.

Project / Author	Reference	Interaction Type	Ergonomic Measures								
			Cognitive Load	Perceptual Load	Motor Load	Visual Acquisition	Motor Acquisition	Ease of Learning	Fatigue	Error Proneness	Distance Sensitivity
Remote Commander	1	Continuous Indirect	•	•	•	•	⬤	•	•	•	·
Silfverberg et al.	2		•	•	·	•	•	•	·	•	·
Rock 'n' Scroll, SqueezeTilt	3		•	•	•	•	•	•	•	•	·
Smart Laser Scanner	4		•	•	•	•	•	•	•	•	·
C-blink	4		•	•	⬤	•	•	•	⬤	•	•
Sweep	5		•	•	⬤	•	⬤	•	⬤	•	·
Mogi	6		⬤	•	⬤	•	⬤	•	⬤	⬤	·
Madhavapeddy et al.	7	Continuous Direct	•	•	•	•	•	•	•	•	⬤
Direct Pointer	7		•	•	•	•	•	•	•	•	⬤
Olsen et al.	7		·	•	•	•	•	•	•	⬤	⬤
VisionWand	7		•	•	•	•	•	•	•	⬤	⬤
Blinkenlights	8	Discrete Indirect	⬤	⬤	·	⬤	•	•	·	•	·
Point & Shoot	9	Discrete Direct	•	⬤	•	⬤	⬤	•	•	⬤	⬤

- **Ease of learning:** Characterizes the level of skill that is required to use the device.
- *Fatigue* characterizes how tiring the interaction technique is to perform.
- **Error proneness:** Characterizes the susceptibility for errors of the input technique, the degree to which the interaction technique, by its design, allows/avoids errors, for example, if possible movement trajectories match the degrees of freedom of the required input then certain errors can be avoided.
- **Sensitivity to distance:** Users in ubiquitous computing scenarios typically have freedom of motion, making the amount of separation between the user and the target in the environment (such as a large display or other device) dynamic and unpredictable. Thus, the range of distances the interaction will support is an important design consideration. Interactions that are based on aiming, such as laser pointers, become more difficult to perform when further away, where targets are perspectively smaller. Other techniques, such as the Sweep technique, are not significantly affected by distance of interaction.

Orient

The ORIENT subtask involves specifying a heading or direction instead of a position. Like POSITION, ORIENTATION is also not limited to graphics subtasks as it can relate to physical orientation in the real world, such as a security camera, a spotlight, or a steerable projector. Some of Foley et al.'s original graphics interactions carry over directly to ubiquitous computing, including *indirect continuous orientation with velocity-controlled joystick* and *discrete orientation with angle type-in*. The remaining techniques observed in our survey include

Continuous Indirect Interactions

1. **Locator device:** The user can specify the angle of orientation by using a continuous quantifier or one axis of a positioning device. The Sweep technique supports detection of rotation around the Z-axis (perpendicular to the display), allowing interactions like rotating a puzzle piece in a jigsaw puzzle

application, where the phone is used like a ratchet to adjust orientation. The image processing used by Sweep also detects rotation around the X and Y-axis. However, for better performance as a positioning device, rotation around the Y-axis is mapped to translation along the X-axis, and rotation around the X-axis is mapped to translation along the Y-axis.

2. **Camera tracking:** VisionWand (Cao & Balakrishnan, 2003) uses a set of cameras in the environment to track the absolute orientation of a marked handheld device. The technique requires that at least two markers are visible in at least two camera viewpoints to determine the orientation in 3-dimensional space.

Continuous Direct Interactions

3. **Camera tracking:** Madhavapeddy's tagged GUI dials (Madhavapeddy et al., 2004) can be oriented using the phone camera to track rotation movement. Similar to the Sweep technique, the phone is used like a rachet to adjust orientation.

4. **Compass:** Electronic compasses, such as the Honeywell HMC1052 magnetometer, can be used to detect the physical orientation of the phone with a +/-3° error, enabling a continuous and direct ORIENT task. This or similar sensors could be easily incorporated into future mobile phone applications.

Discrete Direct Interactions

5. **Camera image:** The Point & Shoot technique supports discrete orientation along the Z-axis. As the user aims at a target, they rotate the phone to specify the desired Z-orientation using the aiming cross-hair as an axis of rotation.

Select

In many interaction scenarios, the user must choose from a set of alternatives, such as a menu of icons. The SELECTION subtask addresses this style of interaction. The SELECTION subtask is commonly accomplished by arranging the items spatially in a graphical user interface, allowing the user to complete the selection using a cursor controlled through the positioning subtask. Instead of icons, the set of alternatives might be a list of commands. However, selection is not limited to graphical interactions, as a user may select a physical object

Figure 7. Summary of ORIENT techniques using a smart phone as an input device. © 1984, 2006 IEEE. Adapted by permission.

	Foley	Mobile Phone Interactions	
	In Environment	On Phone	In Environment
Direct Pick		**Camera** [Point & Shoot]	
		Camera Tag Tracking + [Madhavapeddy]	**Visual Tags**
		Vision Markers + [VisionWand]	**Camera**
Indirect with Cursor Match/ Locator Device		**Camera** [Sweep]	
	Joystick (Velocity)	**Joystick** (Velocity)	
	Joystick (absolute)		
Numerical Value/ Numerical Coordinates/ with Character String Name	(See Text Input)	(See Text Input)	

to operate upon, such as selecting a lamp to adjust its setting. Many selection techniques carry over directly from Foley et al.'s earlier analysis, such as *character string name type-in* common for command prompts, or *button push–soft keys,* where buttons are located on the edge of the display area with their labels displayed on screen. The remaining selection techniques are as follows:

Continuous Indirect Interactions

1. **Gesture recognition:** The user makes a sequence of movements with a continuous positioning device such as the joystick, camera, trackpad, or accelerometers. For example, Patel, Pierce, & Abowd (2004) used gesture recognition of accelerometer data from the handheld device to authenticate users that wanted to access data on their mobile phone through an untrusted public terminal. Using this technology, users could securely bring up data on the public terminal from their phone without removing it from their purse.

Continuous Direct Interactions

2. **Tagged objects:** RFIG Lamps (Raskar, Beardsley, van Baar, Wang, Dietz, Lee, Leigh, & Willwacher, 2004) allows a handheld projector to be used to select objects with photosensitive RFID tags in the physical world. The handheld projector emits a gray-code pattern that allows the tags to determine their relative position in the projected view. Waving the handheld projector around, you can navigate a cursor in the center of the projected view to select individual physical objects.

Discrete Indirect Interactions

3. **Voice recognition:** The user speaks the name of the selected command, and a speech recognizer determines which command was spoken. The Personal Universal Controller (Nichols & Myers, 2006) supports automatic generation of speech interfaces (as well as graphical interfaces) to issue commands to objects in the real world.

VisionWand (Cao & Balakrishnan, 2003) also demonstrates a rich gesture vocabulary using stereovision to track a passive wand. For example, a tapping gesture is used to allow selection of the current cursor position specified by the orientation of the wand. It should be noted that information from any continuous positioning technique can be used for gesture recognition, as long as there is a mechanism to specify when a gesture begins and ends.

Discrete Direct Interactions

4. **Tagged objects:** Tagged objects can be used to present information on a wireless mobile computer equipped with an electronic tag reader, as demonstrated by the early E-tag project (Want, Fishkin, Gujar, & Harrison, 1999). For example, selecting a book by scanning its embedded RFID tag would activate a virtual representation of the object on the screen, such as a Web reference to the book allowing it to be purchased. Similar interactions have also been proposed for visual tags in the environment (Rohs, 2005a) and tagged GUI elements (Madhavapeddy et al., 2004; Rohs, 2005b), where a camera is used to acquire an image to decode the selected tag. Patel and Abowd (2003) present a physical world selection method for mobile phones in which a modulated laser pointer signal triggers a photosensitive tag placed in the environment, allowing users to bring up a menu to control the object on their handheld device.

5. **Laser pointer:** Myers et al. (2002) proposed a multilayer selection technique, called "semantic snarfing," that combines multiple devices in consecutive actions. First, a laser pointer integrated with a handheld computer is used to make a coarse-grained selection of a screen region on a display in the environment. A camera, also in the environment, detects

Figure 8. Summary of SELECTION *techniques using a smart phone as an input device (Continued on next page). © 1984, 2006 IEEE. Adapted by permission.*

	Foley	Mobile Phone Interaction	
	In Environment	On Phone	In Environment
Direct Pick		**Camera + On-Screen Cursor** [Point & Shoot]	
		Laser Pointer + [Olsen, Semantic Snarfing, Patel]	**Light Sensor** (e.g. camera)
	Light Pen Tracking	**Camera +** [Madhavapeddy] **Camera Tracking +** [Direct Pointer] **Camera +**	**Visual Tags** **On-Screen Cursor** **Pen Input**
		Vision Markers + [VisionWand]	**Camera + Tapping** **Gesture Recognition**
		Handheld Projector + RFID Reader [RFIG]	**Light Sensitive** **RFID Tags**
		RFID Reader + [Want]	**RFID Tags**
	Touch Panel		
Indirect with Cursor Match/ Locator Device	**Mouse**	**Camera** [Sweep]	
		Display + [C-blink]	**Stationary Camera**
	Joystick (Velocity)	**Joystick** (Velocity) [Silfverberg]	
	Tablet	**Trackpad** [Remote Commander]	
	Trackball		
	Cursor Control Keys	**Cursor Control Keys**	
	Joystick (absolute)	**Accelerometer** [Harrison, Rock 'n' Scroll]	
		Steerable Laser + **Wide Angle Photodetector +** **Button Push** [Cassinelli]	

Figure 9. Summary of SELECTION techniques using a smart phone as an input device (Continued from previous page). © 1984, 2006 IEEE. Adapted by permission.

	Foley	Mobile Phone Interaction	
	In Environment	On Phone	In Environment
Indirect with Directional Commands/ Button Push/ Time Scan	**Programmed Function Keyboard**	**Programmed Function Keyboard**	
	Soft Keys	**Soft Keys**	
	Alphanumeric Keyboard		
Gesture Recognition/ Sketch Recognition		**Camera + Button Push Clutch** [TinyMotion]	
		Display +	**Camera**
		Accelerometer [Patel]	
	Light Pen	**Pen Input**	
		Steerable Laser + Wide Angle Photodetector + Button Push for Clutch [Cassinelli]	
		Laser Pointer +	**Light Sensor** (e.g. camera) [Olsen, Semantic Snarfing, Patel]
		Vision Markers +	**Camera** [VisionWand]
	Tablet + Stylus		
Location Detection		**GPS +**	**Cell Networks Towers** [Mogi]
Voice Input		**Microphone +**	**Voice Recognizer** [PUC]
Numerical Value/ Numerical Coordinates/ with Character String Name	(See Text Input)	(See Text Input)	

laser activity on the display. The system then transmits the details of the selected screen region to the handheld device, which composes a GUI on the handheld screen to make the fine-grained selection with a stylus.

Path

The PATH subtask involves specifying a series of positions and orientations over time. The PATH subtask has different requirements than POSITION and ORIENT because the movement is governed by

the speed-accuracy tradeoff (Schmidt, Hawkins, Frank, & Quinn, 1979). Despite this, PATH adheres to the same taxonomy as the corresponding POSITION and ORIENT techniques, because a PATH task can be specified using the more primitive subtasks.

Quantify

The QUANTIFY task involves specifying a value or number within a range of numbers. This technique is used to specify numeric parameters such as time

or speaker volume. In ubiquitous applications, Quantify tasks using phone input were typically accomplished through the GUI using 1-D Position or Orient subtasks.

Text

Text entry for mobile phones is a well-studied area (MacKenzie & Soukoreff, 2002) as it is central to text-based mobile communications like SMS (short messaging service) and personal information management functionality. Text entry also has many applications for ubiquitous applications, for example, the Digital Graffiti (Carter, Churchill, Denoue, Helfman, & Nelson, 2004) project seeks to annotate public content on large public displays. This section is not intended to be a comprehensive survey of mobile text entry techniques, but we have selected a few examples to illustrate the design space. All of the techniques listed were originally designed for text input directly on the mobile phone, but could clearly be used for text entry for a ubiquitous computing application.

Keyboard

Although some mobile phones and handheld devices feature a full QWERTY keyboard (albeit much smaller than their desktop counterparts), miniaturization trends make this type of keyboard impractical in a majority of mobile phone form factors. The most well known text entry techniques for mobile phones use a standard numeric keypad. For text entry from a 26-character alphabet using this keyboard, a mapping with more than one character per button is required. Following the classification by Wigdor and Balakrishnan (2004), there are two fundamental types of disambiguation: *consecutive*, where the user first selects a letter grouping and then an individual letter, or *concurrent*, where the user simultaneously selects the letter grouping and the individual letter. Consecutive approaches are the most common today. One approach to disambiguate text entry is MultiTap, which requires users to make multiple presses to select a single letter from the characters associated with a certain key. Another solution is

to use a two-key disambiguation, where the first key selects the letter group, and the second key specifies the letter in the group. Dictionary-based techniques, such as T9[2], deduce the word being typed, based on the different possibilities for combining the groups of characters. When multiple words match the key sequence, the user selects the intended word from a list (typically ordered by probability or frequency of use).

Concurrent approaches, however, demonstrate a lot of promise. For example, TiltText (Wigdor & Balakrishnan, 2003) combines the standard 12-key keypad with an accelerometer. To disambiguate which character is intended when a key is pressed, TiltText uses the tilt orientation of the handset. A keypress with the phone tilted to the left enters the first character on the key, forward tilt enters the second character, right tilt enters the third character, tilting towards the user enters the fourth character (if one exists for the key), and no tilt enters the numeric character.

ChordTap (Wigdor & Balakrishnan, 2004) combines the standard numeric keyboard with additional "chording" buttons on the back of the phone. A user selects an individual letter by selecting the key group on the numeric keyboard while pressing the appropriate "chord" key on the back of the phone.

If miniaturization trends continue, Tilt-Type (Partridge, Chatterjee, Sazawal, Borriello, & Want, 2002) represents an interesting point in the design space that combines chord button presses to specify a letter grouping and tilting to allow the user to specify a particular character within that grouping. Using only four buttons and a two-axis accelerometer, the technique supports an alphabet of 55 characters in a watch-sized form factor. Expert users can memorize the character positions, allowing the letter grouping and individual character within the grouping to be specified concurrently.

Speech Recognition

Text entry by speech recognition is not yet technically viable on mobile platforms, but we list it here for completeness. Technology is making

rapid advances in the realm of speech processing. For example, system on a chip designs for speech processing (Ravindran, Smith, Graham, Duangudom, Anderson, & Hasler, 2005) have the potential to bring speech input to interactive text entry on mobile phones. Karpov et al. (Karpov, Kiss, Leppanen, Olsen, Oria, Sivadas, & Tian, 2006) have developed a short message (SMS) dictation system for Symbian phones with a vocabulary of 23,000 words. The language model is adapted to words typically used in SMS messages.

Speech recognition could also be achieved in a compound architecture where the speech is recognized through an external computer (i.e., connected through a voice call) and sent back to the mobile phone.

Stroked Character Recognition

Pen-based techniques, such as Graffiti, are very common in the PDA form factor, and are also available on a small portion of the handsets on today's market. However, any of the continuous positioning tasks discussed earlier are capable of generating stroke information necessary for stroked-character recognition. For example, TinyMotion (Wang et al., 2006) demonstrates both English and Chinese stroked character recognition using camera-based motion estimation (similar to the Sweep technique).

Menu Selection

On-screen keyboards are common for touch sensitive displays, where the letters of the alphabet are displayed as a menu of buttons, commonly in a spatial layout similar to the QWERTY keyboard. If the screen size of the mobile phone is not large enough to depict a keyboard layout, items in the environment could be used to display the menu, where users select the characters using the SELECTION subtask previously discussed.

Figure 10. Summary of TEXT ENTRY *techniques using a smart phone as an input device. © 1984 IEEE. Adapted by permission.*

	Foley	Mobile Phone Interactions	
	In Environment	On Phone	In Environment
Keyboard	**Alphanumeric**	**Alphanumeric** [Multitap, T9]	
		Alphanumeric + **Accelerometer** [TiltText]	
	Chord	**Alphanumeric +** **Chord** [ChordTap]	
		Chord + **Accelerometer** [TiltType]	
Stroked Character Recognition	**Tablet with Stylus**	(See Continuous Positioning)	
Voice/Speech Recognition	**Voice Recognizer**	**Speech Recognizer**	
		Microphone +	**Speech Recognizer**
Direct Pick from Menu with Locator Device	**Light Pen**	(See Selection)	
	Touch Panel	(See Selection)	
Indirect Pick from Menu with Locator Device	(See Positioning)	(See Positioning)	

SPATIAL LAYOUT OF THE DESIGN SPACE

Our interaction taxonomy is summarized in Foley-style graphs in Figures 5, 7, 8, 9, and 10. Card et al. (1991) point out that this format is somewhat ad hoc and lacks a notion of completeness. Card then builds on the work of Buxton (1983) to create a systematic spatial layout of the design space of input devices that captures the physical properties of manual devices very well. However, it does not capture many aspects that are relevant to ubicomp interactions such as modality or feedback (Ballagas et al., 2003).

Using Foley et al.'s taxonomy, we propose a five-part spatial layout, shown in Figure 11, for mobile phone interaction tasks discussed in our survey including supported subtasks (POSITION, ORIENT, and SELECTION), dimensionality, relative vs. absolute movement, interaction style (direct vs. indirect), and feedback from the environment (continuous vs. discrete). Feedback and interaction style have been previously defined in the introduction to Foley et al.'s taxonomy. We describe the remaining dimensions in more detail in the remainder of this section.

Supported Subtasks

When choosing the most appropriate input device for a particular interaction scenario, the subtasks an interaction supports are the primary consideration. By including the subtask directly in the design space, it becomes more useful as a design tool.

Dimensionality

Dimensionality refers to the number of dimensions the interaction supports. Dimensionality can indicate spatial dimensions (X,Y,Z) or rotational dimensions (rX,rY,rZ). This distinction is visible in our design space by observing the subtask of the dimension. Following Card et al. (1991), if a particular interaction uses a combination of dimensions across different points in the design space, the relationship is indicated using a merge composition operator (a solid line). In contrast to Card's notation, our merge composition operators are connecting subtasks, not spatial sensor dimensions.

Relative vs. Absolute

Relative input is specified with respect to interaction history: the input technique provides information about the amount of change from the previous state. Relative input can be specified regardless of the current physical properties, such as position and orientation. For example, standard desktop mouse input is specified through motion across the desktop regardless of the physical position of the mouse on the desktop.

Absolute input is specified with respect to current physical properties, and can be specified independently of any interaction history. For example, stylus input can be used to provide absolute positional information on a screen space.

Other Relevant Attributes of Interaction Devices

It should be noted that this set of dimensions is not comprehensive, and other dimensions, such as resolution, direction (input vs. output), and modality, may provide further insights into the design space. However, the design space depicted in Figure 11 does provide an interesting overview of the interaction techniques covered in this chapter. Using this graphical layout, we are able to pinpoint gaps in the breadth of the interaction techniques surveyed, and can anticipate opportunities for future work. For example, our space shows no interaction that supports 3-dimensional relative direct orientation. An alternative layout might include direction and modality, which would demonstrate the sparse usage of auditory and haptic feedback in these techniques.

Designing for Serendipity

One key design consideration is the ease and speed of setting up a data connection between the phone and the environment or the device it is controlling. In some of the interactions surveyed,

Figure 11. Classification of different mobile phone interactions that have been implemented in the projects surveyed. Inspection of the diagram reveals opportunities for future work, for instance, developing interaction techniques that support 3-D relative direct orientation. In the listing of techniques, (P) indicates capabilities of the phone, and (E) indicates capabilities of the environment. © 2006 IEEE. Adapted by permission.
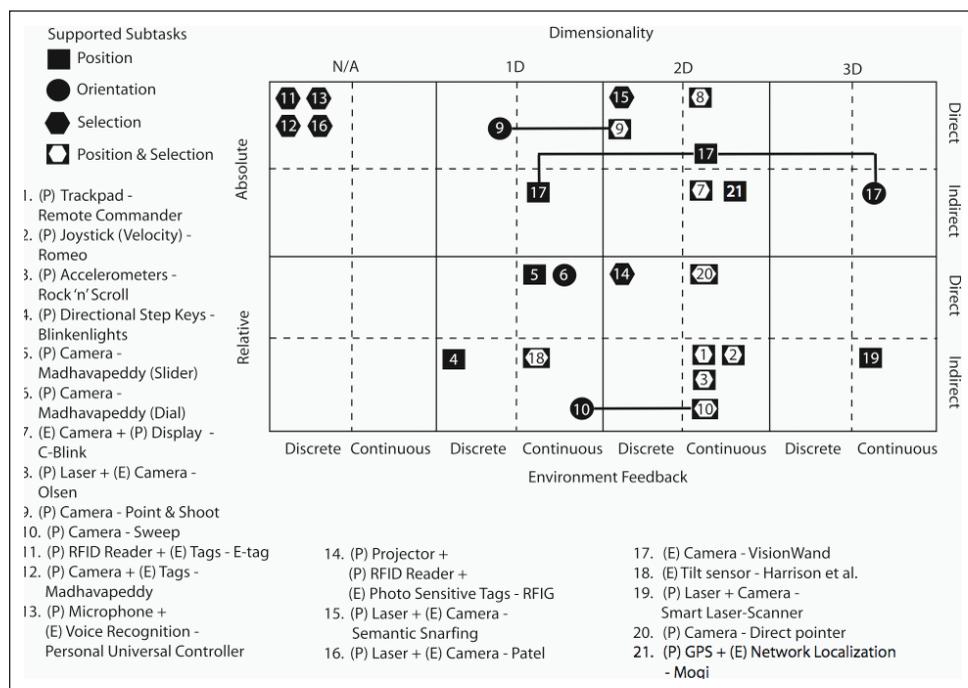
the data connection is inherent in the physical properties of the device. The VisionWand, for example, is a completely passive system, so it requires no additional action on the user's part to start the interaction.

The C-Blink interaction is classified as highly serendipitous, as the users merely launch an application on their mobile phone to interact with a display; no network connection or handshaking is required. The RFIG Lamps project also falls into this category because RFID tags are so simple in terms of communications protocol that no connection need to be established before data can be transferred.

For projects that use short-range wireless communications models, such as Bluetooth, visual or RFID tags can be used to encode the connection information for the environment, creating a very low threshold of use.

Social Acceptance

Smart phones today are social devices. While smart phone ubiquity seems inevitable, social acceptance will influence the success of these new interactions. Remind yourself, for example, of the first time you came across a person using a wireless headset to communicate via their mobile phone. For many people, this communication technique is still awkward and strange, particularly in public places. Smart phone interaction will require users to perform particular actions and behaviors that might feel unintuitive and awkward to them. Furthermore, they will perform these actions in the presence of passive or active others, both familiars and strangers. On one hand, outside observers might find these interactions disturbing or embarrassing, but on the other hand, these kinds of interaction have the potential to raise your social

status, similar to the way phones themselves are status symbols for part of our society.

DESIGN SPACES IN THE DESIGN PROCESS

Design spaces are a particularly useful design tool as a part of a human-centered iterative design process (Nielsen, 1993). One of the pitfalls of iterative human-centered design is that if you pick a poor starting point, you may reach a peak in the usability of a particular design without reaching your desired usability goals. In this case, it may be necessary to throw the design away and start over. False starts are relatively painless early in the design process, but can be extremely expensive if determined late in the design process. In order to minimize the risk of false starts, a parallel design strategy (Nielsen & Faber, 1996) can be used, where multiple designs can be explored independently early in the design process. As the designs mature, the best design becomes clear, or the strengths of the top designs can be merged to a unified design. Using the design space, designers can more easily reason about alternative input techniques in a parallel design process.

As a concrete example, REXplorer (Ballagas, Walz, Kratz, Fuhr, Yu, Tann, Borchers, &

Hovestadt, 2007) is a pervasive spell-casting game that allows tourists to explore the history of the medieval buildings in Regensburg, Germany. The game premise is that historical spirits are trapped inside of medieval buildings. Players need to interact with the spirits to learn their stories and perform quests on their behalf to earn points in the game. The game design called for spell-casting as the primary interaction metaphor; in order to awaken a spirit, one of four spells must be cast.

Choosing one spell out of four can be characterized as a SELECT subtask. The design space was used to identify a set of design alternatives that we initially considered:

1. Four dedicated spell buttons
2. Selecting one of four spells from on-screen menu
3. Recognition of spell gestures. We noted that gestures are actually specified using the path subtask. Then we came up with gesture input alternatives including:
   a. Pen trace across a touch screen
   b. Path using camera-based motion detection to allow the phone to be used like a magic wand.

After preliminary analysis with our target group (students aged 15-25), we decided to go with the

Figure 12. REXplorer uses the Sweep technique to allow players to cast spells using the PATH subtask (Ballagas et al., 2007). Reprinted with permission from the authors.

camera-based motion detection solution (see Figure 12). Waving the phone through the air is not the most efficient technique, but is the most similar to the spell-casting metaphor. Also, this physical style of gesture was more likely to create an engaging experience (Hummels, 2000).

Later in the design process, after a working-gesture recognition system was created, we did a full playability test. Most of the test players found the gestures to be an important element of gameplay. They found it heightened the sense of magic and mysteriousness. However, we also discovered during the playability tests that a few of our players (especially our older participants) found the gestures awkward. As a compromise, we created a unified design, where an alternative gesture selection mechanism through an on-screen menu can be used anytime an invalid gesture is performed, effectively allowing people to avoid gestures altogether if desired. This final design encouraged the use of gestures for spell selection to promote engagement, but allowed an alternative selection mechanism to those who preferred to avoid gestures.

CONCLUSION

Our structured tour illustrates the state of the art in using smart phones to interact with and control our environments. The taxonomy organizes the range of techniques into families that help make functional relations between the mobile phone techniques and their desktop counterparts. The design space addresses the lack of a sense of completeness in the taxonomy, and structures the range of interactions in a way that helps visually identify gaps and predict future interaction techniques. The design space can be used as a part of a human-centered iterative design process to help generate parallel or alternative designs. These methods of thought are intended to inspire new applications that use the mobile phone for interaction with the environment, as well as inform the design of future smart phone interaction techniques.

REFERENCES

Ballagas, R., Ringel, M., Stone, M., & Borchers, J. (2003). iStuff: A physical user interface toolkit for ubiquitous computing environments. In *CHI '03 Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (pp. 537–544). New York, NY: ACM Press.

Ballagas, R., Rohs, M., Sheridan, J. G., & Borchers, J. (2005). Sweep and point & shoot: Phonecam-based interactions for large public displays. In *CHI '05 Extended abstracts of the SIGCHI Conference on Human Factors in Computing Systems* (pp. 1200–1203). New York, NY: ACM Press.

Ballagas, R., Rohs, M., Sheridan, J., & Borchers, J. (2006). The smart phone: A ubiquitous input device. *IEEE Pervasive Computing, 5*(1), 70–77.

Ballagas, R., Walz, S. P., Kratz, S., Fuhr, C., Yu, E., Tann, M., Borchers, J., & Hovestadt, L. (2007). REXplorer: A mobile, pervasive spell-casting game for tourists. In *To appear in CHI '07 Extended Abstracts on Human Factors in Computing Systems*. San Jose, CA: ACM Press.

Bartlett, J. F. (2000). Rock 'n' scroll is here to stay. *IEEE Comput. Graph. Appl., 20*(3), 40–45.

Buxton, W. (1983). Lexical and pragmatic considerations of input structures. *SIGGRAPH Comput. Graph., 17*(1), 31–37.

Cao, X. & Balakrishnan, R. (2003). VisionWand: Interaction techniques for large displays using a passive wand tracked in 3-D. In *Proceedings of the 16th annual ACM Symposium on User Interface Software and Technology* (pp. 173–182). ACM Press.

Card, S. K., English, W. K., & Burr, B. J. (1978). Evaluation of mouse, rate controlled isometric joystick, step keys and text keys for text selection on a CRT. *Ergonomics, 21*, 601–613.

Card, S. K., Mackinlay, J. D., & Robertson, G. G. (1991). A morphological analysis of the design space of input devices. *ACM Transactions Information Systems, 9*(2), 99–122.

Card, S. K., Newell, A., & Moran, T. P. (1983). *The psychology of human-computer interaction*. Lawrence Erlbaum Associates, Inc.

Carter, S., Churchill, E., Denoue, L., Helfman, J., & Nelson, L. (2004). Digital graffiti: Public annotation of multimedia content. In *CHI '04: Extended abstracts of the SIGCHI Conference on Human Factors and Computing Systems* (pp. 1207–1210). ACM Press.

Cassinelli, A., Perrin, S., & Ishikawa, M. (2005). Smart laser-scanner for 3-D human-machine interface. *CHI '05: Extended abstracts of the SIGCHI Conference on Human Factors and Computing Systems* (pp. 1138–1139).

Chaos Computer Club. (2002). *Blinkenlights*. Retrieved from http://www.blinkenlights.de

Dan, R. Olsen, J., & Nielsen, T. (2001). Laser pointer interaction. In *CHI '01 Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (pp. 17–22). New York, NY: ACM Press.

Foley, J. D., Wallace, V. L., & Chan, P. (1984). The human factors of computer graphics interaction techniques. *IEEE Computer Graphics and Applications*, *4*(11), 13–48.

Harrison, B. L., Fishkin, K. P., Gujar, A., Mochon, C., & Want, R. (1998). Squeeze me, hold me, tilt me! An exploration of manipulative user interfaces. In *CHI '98 Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (pp. 17–24). New York, NY: ACM Press/Addison-Wesley Publishing Co.

Hinckley, K., & Horvitz, E. (2001). Toward more sensitive mobile phones. In *UIST '01 Proceedings of the 14th annual ACM Symposium on User Interface Software and Technology* (pp. 191–192). New York, NY: ACM Press.

Hummels, C. C. M. (2000). *An exploratory expedition to create engaging experiences through gestural jam sessions*. Ph.D. thesis, Delft University of Technology.

Jiang, H., Ofek, E., Moraveji, N., & Shi, Y. (2006). Direct pointer: direct manipulation for large-display interaction using handheld cameras. In *CHI '06 Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (pp. 1107–1110). New York, NY: ACM Press.

Karpov, E., Kiss, I., Leppanen, J., Olsen, J., Oria, D., Sivadas, S., & Tian, J. (2006). Short message dictation on Symbian Series 60 mobile phones. In *ICMI '06 Proceedings of the 8th international conference on Multimodal interfaces* (pp. 126–127). New York, NY: ACM Press.

Licoppe, C., & Inada, Y. (2006). Emergent uses of a multiplayer location-aware mobile game: The interactional consequences of mediated encounters. *Mobilities*, *1*(1), 39–61.

MacKenzie, I., & Soukoreff, R. (2002). Text entry for mobile computing: Models and methods, theory and practice. *Human-Computer Interaction*, *17*(2), 147–198.

Madhavapeddy, A., Scott, D., Sharp, R., & Upton, E. (2004). Using camera-phones to enhance human-computer interaction. In *Sixth International Conference on Ubiquitous Computing (Adjunct Proceedings: Demos).* Springer-Verlag.

Miyaoku, K., Higashino, S., & Tonomura, Y. (2004). C-blink: A hue-difference-based light signal marker for large screen interaction via any mobile terminal. In *UIST '04 Proceedings of the 17th annual ACM Symposium on User Interface Software and Technology* (pp. 147–156). ACM Press.

Myers, B. A., Bhatnagar, R., Nichols, J., Peck, C. H., Kong, D., Miller, R., & Long, A. C. (2002). Interacting at a distance: measuring the performance of laser pointers and other devices. In *CHI '02 Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (pp. 33–40). New York, NY: ACM Press.

Myers, B. A., Stiel, H., & Gargiulo, R. (1998). Collaboration using multiple PDAs connected to a PC. In *Proceedings of the 1998 ACM Conference on Computer Supported Cooperative Work* (pp. 285–294). ACM Press.

Nichols, J., & Myers, B. A. (2006). Controlling home and office appliances with smart phones. *IEEE Pervasive Computing*, *5*(3), 60–67.

Nielsen, J. (1993). Iterative user-interface design. *Computer*, *26*(11), 32–41.

Nielsen, J., & Faber, J. (1996). Improving system usability through parallel design. *Computer*, *29*(2), 29–35.

O'Hara, K., Perry, M., & Churchill, E. (2004). *Public and situated displays: Social and inter-actional aspects of shared display technologies (Cooperative Work, 2).* Norwell, MA: Kluwer Academic Publishers.

Partridge, K., Chatterjee, S., Sazawal, V., Borriello, G., & Want, R. (2002). TiltType: Accelerometer-supported text entry for very small devices. In *UIST '02: Proceedings of the 15th annual ACM Symposium on User Interface Software and Technology* (pp. 201–204). New York, NY: ACM Press.

Patel, S. N., & Abowd, G. D. (2003). A 2-way laser-assisted selection scheme for handhelds in a physical environment. In *Ubicomp '03 Proceedings of the 5th International Conference on Ubiquitous Computing* (pp. 200–207). LNCS 2864, Springer.

Patel, S. N., Pierce, J. S., & Abowd, G. D. (2004). A gesture-based authentication scheme for untrusted public terminals. In *Proceedings of the 17th annual ACM Symposium on User Interface Software and Technology* (pp. 157–160). ACM Press.

Raskar, R., Beardsley, P., van Baar, J., Wang, Y., Dietz, P., Lee, J., Leigh, D., & Willwacher, T. (2004). RFIG lamps: interacting with a self-describing world via photosensing wireless tags and projectors. *ACM Transactions on Graphics*, *23*(3), 406–415.

Ravindran, S., Smith, P., Graham, D., Duan-gudom, V., Anderson, D., & Hasler, P. (2005). Towards low-power on-chip auditory processing. *EURASIP Journal on Applied Signal Processing*, *7*, 1082–1092.

Rohs, M. (2005a). Real-world interaction with camera phones. In H. Murakami, H. Nakashima, H. Tokuda, & M. Yasumura (Eds.), *Second International Symposium on Ubiquitous Computing Systems (UCS 2004), Revised Selected Papers* (pp. 74–89), Tokyo, Japan. LNCS 3598, Springer.

Rohs, M. (2005b). Visual code widgets for marker-based interaction. In *IWSAWC'05 Proceedings of the 25th IEEE International Conference on Distributed Computing Systems – Workshops (ICDCS 2005 Workshops* (pp. 506–513). Columbus, OH.

Schmidt, R. A., Zelaznik, H. N., Hawkins, B., Frank, J. S., & Quinn, J. T. (1979). Motor-output variability: A theory for the accuracy of rapid motor acts. *Psychological Review*, *86*, 415–451.

Silfverberg, M., MacKenzie, I. S., & Kauppinen, T. (2001). An isometric joystick as a pointing device for handheld information terminals. In B. Watson & J. W. Buchanan (Eds.), *Proceedings of Graphics Interface 2001* (pp. 119–126). Canadian Information Processing Society.

Wang, J., Zhai, S., & Canny, J. (2006). Camera phone based motion sensing: Interaction techniques, applications and performance study. In *UIST '06 Proceedings of the 19th annual ACM Symposium on User Interface Software and Technology* (pp. 101–110). New York, NY: ACM Press.

Want, R., Fishkin, K. P., Gujar, A., & Harrison, B. L. (1999). Bridging physical and virtual worlds with electronic tags. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (pp. 370–377). ACM Press.

Weiser, M. (1991). The computer for the 21st century. *Scientific American*, *265*, 94–104.

Wigdor, D., & Balakrishnan, R. (2003). TiltText: Using tilt for text input to mobile phones. In *UIST '03 Proceedings of the 16th annual ACM Symposium on User Interface Software and Technology* (pp. 81–90). New York, NY: ACM Press.

Wigdor, D., & Balakrishnan, R. (2004). A comparison of consecutive and concurrent input text

entry techniques for mobile phones. In *CHI '04 Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (pp. 81–88). New York, NY: ACM Press.

Zwicky, F. (1967). The morphological approach to discovery, invention, research and construction. In *New Methods of Thought and Procedure* (pp. 273–297). Berlin: Springer.

KEY TERMS

Continuous Interaction: Interactions with a closed-loop feedback, where the user continuously gets informed of the interaction progress as the task is being performed.

Design Space: Design spaces provide a formal or semiformal way of describing and classifying entities along different dimensions, each listing relevant categories or criteria.

Direct Interaction: Input actions are physically coupled with the user-perceivable entity being manipulated (such as an image on a display). To the user, this appears as if there is no mediation, translation, or adaptation between input and output. Physical coupling can be achieved when the

feedback spatially coincides with the input action, or at a distance if the user is manipulating a 3-D ray (such as with a laser pointer) that intersects directly with the entity being manipulated.

Discrete Interaction: Interactions with an open-loop feedback, where the user is only informed of the interaction progress after the task is complete.

Indirect Interaction: User activity and feedback occur in disjoint spaces, where scaling and abstraction between input actions and feedback are often necessary.

Input Technique: A specific way of providing data input to a computer through a combination of input devices and software for visual, auditory, or haptic feedback.

ENDNOTES

[1] An alternative implementation of the Point & Shoot technique could use pen input instead of the cross-hair image so that the user repositions the cursor by selecting the desired position directly on the live camera image displayed on the phone screen.

[2] www.tegic.com