



Interaction with Mobile Devices
Safiza Suhana Binti Kamal Baharin

Delft, October 2007

RGI 149 Report Nr 1



Interaction with Mobile Devices

Safiza Suhana Binti Kamal Baharin

Delft, October 2007

RGI 149 Report Nr 1

Summary

The focus of this report is on how to interact on mobile devices and why it is necessary to embrace a paradigm shift in terms of interaction techniques for mobile devices which allow a user to communicate with mobile technology using as little visual attention as possible to access the effectiveness of such paradigms. The literature covered is on current input and output methods, and the advantages and disadvantages of each method with a view to answering the question "What is the best input-output method for mobile devices?", and could be used as guidelines for input and output methods for mobile devices. This report will contribute to the RGI-149 project, of which the main focus is to push a proper amount of geo-information to the mobile user's device. Generally speaking, the user of this project application is an environmental officer responsible for environmental protection. Their job specification requires them to always be moving from one point to another, dependent on their tasks. The application of 'Buiten Beter' is to support surveillance for environmental officers, to obtain information on rubbish dumping, pollution and other happenings, for example. In order to carry out inspections, for instance, they should be supported with suitable maps. The key problem here is how to present this amount of information within the limitations of a small screen size. A previous report on Cartographic Representation in Small Display Screens mentioned applying a context-aware technology to this application.

This report attempts to explain how to interact with a mobile device and why it is necessary to embrace a paradigm shift in terms of interaction techniques for mobile devices that could be used by dedicated users with as little visual attention as possible to access information efficiently and effectively based on their profile, identity, location and behaviour.

Project leader	: Section GIS Technology OTB Research Institute for Housing, Urban and Mobility Studies Delft University of Technology
Address	: Jaffalaan 9 2628 BX Delft
Contact	: ir. E. Verbree
Tel.	: +31 (0)15 278 2584
Fax	: +31 (0)15 278 2745
E-mail	: e.verbree@tudelft.nl
Website	:

Contents

1 Introduction	1
2 User Interfaces: Desktop Computers and Hand-Held Devices	2
3 Fundamentals of Input/Output Methods for Mobile Devices	4
3.1 Output Mechanisms	7
3.2 Input Mechanisms	9
3.2.1 Multi-tap.....	12
3.2.2 Two-key Disambiguation	12
3.2.3 Linguistic Disambiguation (T9)	13
3.3 Key Press Techniques	13
3.4 Tilt-based Interaction.....	19
4 Conclusion	26
References	27

1 Introduction

Today, mobile devices such as mobile phones, PDAs and smartphones are ubiquitous. Mobile phone technology is rapidly evolving towards a personal information assistant that is capable of managing many other personal needs beyond communication with other humans, offering other assorted services from personal calendars, address books, photography and music to navigation tools and location information. Mobile phones can also support humans in communicating or interacting, or even for exchanging data between devices. This is becoming more widespread as users are starting to move their computer usage from the workplace and occasional home use onto the street and into their everyday lives. Carrying a mobile device has become as important as wearing a watch, if not more useful: right now, the functionality of mobile devices exceeds that of the watch in terms of usability. For example, more than simply telling us the time and date and having an alarm function, a mobile device can display a task list, navigation guide, health information or shopping list.

Unfortunately, parallel with the rapid growth of the mobile computing market, there are issues raised in terms of the usability of interactive mobile devices, it being hard to design interfaces and modes of interaction for devices that have small screens and limited memory resources. In response to this, as we know, once a device becomes mobile, less is more: the device must keep users in touch without stretching the lining of the pocket or purse. A space-saving device is better for the user as long as its size does not compromise its usability or efficiency. For this reason, the average size of the mobile device is in decline. The question here could therefore be as follows: "How can mobile phones with a limited screen size interact effectively with their users?"

One major challenge faced in the design of mobile devices is that devices are typically used when the user has limited physical resources and increased content must be accessed via shrinking input and output channels. Compared to a desktop system, mobile devices have restricted input and output capabilities that typically reduce their usability. Often with a very limited amount of screen size or space, the visual display can become cluttered with information and widgets. Data input is also limited by small keypads or simple handwriting recognition. Speech recognition would appear not to be an ideal option in this case because of problems recognising input in noisy environments, which require improvement. New interaction techniques are therefore needed to access services whilst on the move.

The focus of this report is on how to interact on mobile devices and why it is necessary to embrace a paradigm shift in terms of interaction techniques for mobile devices which allow a user to communicate with mobile technology using as little visual attention as possible to access the effectiveness of such paradigms. The literature covered is on current input and output methods, and the advantages and disadvantages of each method with a view to answering the question "What is the best input-output method for mobile devices?", and could be used as guidelines for input and output methods for mobile devices. This report will contribute to the RGI-149 project, of which the main focus is to push a proper amount of geo-information to the mobile user's device. Generally speaking, the user of this project application is an environmental officer responsible for environmental protection. Their job specification requires them to always be moving from one point to another, dependent on their tasks. The application of 'Buiten Beter' is to support surveillance for environmental officers, to obtain information on rubbish dumping, pollution and other happenings, for example. In order to carry out inspections, for instance, they should be supported with suitable maps. The key problem here is how to present this amount of information within the limitations of a small screen size. A previous report on Cartographic Representation in Small Display Screens mentioned applying a context-aware technology to this application.

This report attempts to explain how to interact with a mobile device and why it is necessary to embrace a paradigm shift in terms of interaction techniques for mobile devices that could be used by dedicated users with as little visual attention as possible to access information efficiently and effectively based on their profile, identity, location and behaviour.

2 User Interfaces: Desktop Computers and Hand-Held Devices

Users are stationary in the desktop environment. Users sit at a desk and normally devote all their visual resources to the application with which they are interacting, with the result that the interfaces of desktop-based applications are typically very graphical and extremely detailed and utilise a standard mouse and keyboard as interaction mechanisms. An example of a desktop system is shown in Figure 2.1, with a monitor and speakers as output devices and a keyboard and mouse as input devices. All this equipment, or at least the functionality of each device, must be provided in a mobile device as well.



Figure 2.1 Desktop Environment

In contrast, users are typically in motion whilst using a mobile device. This means that users are not devoting all or any of their visual resources to interacting with their device because they are restricted to focusing on their primary task (driving, walking, navigating, etc.), otherwise they expose themselves to the risk of accident. As we know, mobile devices have limited screen area, and traditional input and output capabilities are restricted in general to the use of a keyboard, keypad or simple handwriting recognition, depending on the type of mobile device. Uotila (2000) states that there are three categories of mobile device: mobile communication devices, mobile computer devices and mobile information devices. Each category has three parameters: supported input device, capabilities of screen configuration and type of network access. Mobile communication devices include devices that are small in size and thus have limited input and output capabilities, and are also capable of peer-to-peer text or voice communication and limited information retrieval. Figure 2.2 shows an example of various mobile communication devices.

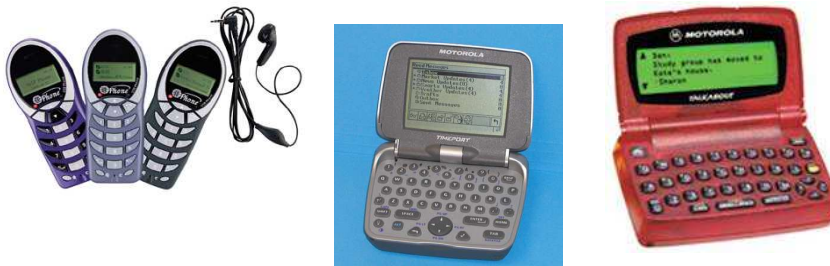


Figure 2.2 Mobile Communication Devices

Mobile computer devices can perform more general and personalised user tasks through the installation of third party applications such as prayer time alerts, mobile dictionaries, maps, pictures, etc. They usually have better user interface capabilities than mobile communication devices. Figure 2.3 shows some examples of mobile computer devices. Environmental officers require at least this kind of device to carry out their work.



Figure 2.3 Mobile Computer Devices

Mobile information devices are a hybrid of the other two categories. They provide better access to network information systems than mobile computer devices by providing a larger display screen and better interaction devices. They have similar functionality to mobile computer devices but they have the upper hand when it comes to accessing networks and browsing internet data. Figure 2.4 shows examples of mobile information devices.

Table 2.1 gives a classification of mobile devices.



Figure 2.4 Mobile Information Devices

Table 2.1 Classification of Mobile Devices (source: Uotila, 2000).

Category	Input	Screen Configuration	Network Access
Mobile communication devices - Standard phones - PIM phones - Two way pagers	Keyboard	< 100*100 pixels < 5cm ² Single colour	Low-end network access
Mobile computer devices - PDA & PIM devices - Pads	Touch screen (Keyboard)	> 100*100 (≤ 640*480) pixels > 100cm ²	No direct access

- Sub-notebooks		≥ 16 colours	
Mobile information devices - Communicators - Smartphones	Touch screen	As mobile computer Landscape display Half VGA resolution Portrait display	High-end network access

Differences relating to output and input capabilities between desktop and mobile device user interfaces can be divided into two parts: output and input.

3 Fundamentals of Input/Output Methods for Mobile Devices

Before we examine input/output methods for mobile devices, we should look at fundamental issues concerning the input/output methods of mobile clients. One approach to addressing the issues associated with this mobile computing technology is to look at the constraints associated with the human user. An input/output approach defines the issues as follows:

The user's method of interaction with the device: what type of mobility does it allow?
Where and when may the user download and upload information to the mobile client?

These issues may be summarised as mobility and connectivity. Figure 3.1 shows the two dimensions of mobility and connectivity. Mobility dimension defines the "mobility" of a mobile client.

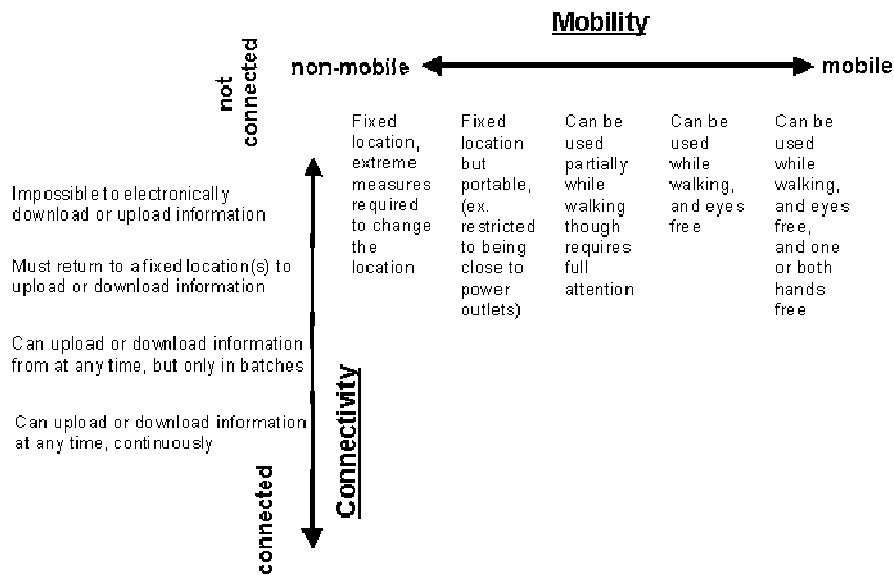


Figure 3.1 Mobility versus Connectivity
(source: Uotila, 2000)

According to Uotila (2000), based on the figure above there are two dimensions, mobility and connectivity. Mobility is concerned with how tethered a user is when using a mobile device. The definition of mobility is not restricted to the ability to walk about, but refers instead to the mobility of the whole body. The major components of mobility consist of freedom of the legs, the arms, and the eyes. The requirement is that the arms and legs be free for a device to be considered to be one that allows full mobility. The point of requiring the eyes to be free for true mobility comes from the fact that if a device requires the full attention of the eyes, then the rest of the body is not completely free either.

The dimension of connectivity concerns the electronic downloading and uploading of information. This aspect differentiates a mobile device from a mobile client in that a device becomes something much more when it has the power to communicate with the rest of the world. The degree of connectivity can vary from none at all, to being able to download/upload only when synced with another device, to full connectivity at any time. The connectivity of a device has clear implications for how the device may be used to accomplish tasks and the actions that then become required of the operator.

Allowing the user greater freedom of movement caused the shift in interface methods from fixed layouts, fixed keyboards and visual displays to wearable systems, chording keyboards, and auditory interface methods. The ultimate goal of this shifting of interface methods is to allow the user to accomplish the same tasks while mobile that were originally performed at a fixed location system. Figure 3.2 shows how mobile computing shifts interface methods from fixed location hardware to wearable devices.

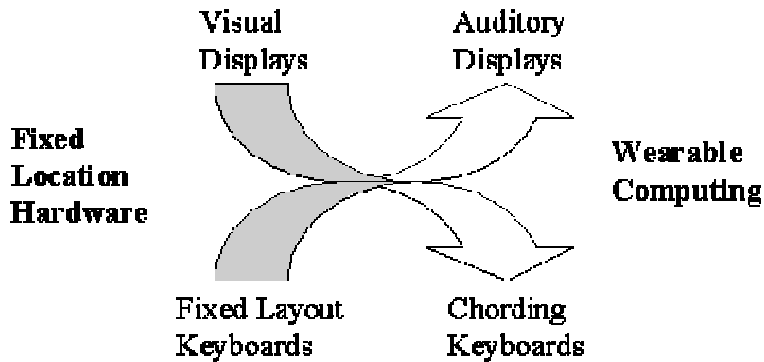


Figure 3.2 Computing shifts interface methods from fixed location hardware to wearable devices (source: Uotila, 2000)

There are three most prominent ways of interacting with a mobile device, which are the following:
 Inputting information with or without a stylus on a touch-sensitive display.
 Entering data using a keypad by using multi-tap or predictive text T9 technology.
 Making (fine) movements, e.g. using a finger on a mobile phone joystick or moving the whole device.

Unlike desktop computers, input devices for mobile devices are often restricted by the limited size of the device. As a result, multiple functions must be assigned to the same buttons and this makes the user interface complicated and unpredictable. In this sense, the way or method with which the user interacts with the mobile device also contributes to the efficiency of the mobile device user interface design. The way or method in which the user currently interacts can be categorised in two ways: two-hand and one-hand.

Pascoe et al. (2000) report that interface design for fieldworkers using a mobile device (normally PDA) for data collection tasks is based on two general principles: Minimal Attention of User Interface (MAUI) and context awareness. The MAUI seeks to minimise the attention required, but not necessarily through the number of interactions required from the user in operating the device. Context awareness enables the mobile device to provide assistance based on knowledge of its environment or means that it has the capacity to sense its environment.

We can assume that environmental officers use a mobile device in the same way as fieldworkers, as Pascoe et al. (2000) mention that there are four characteristics of fieldworkers that should be considered in user interface design for mobile devices. These are listed below.

Dynamic User Configuration: The fieldworker's task is to collect data whenever and wherever they need, but without a chair or a desk on which to set up their computing apparatus. Nevertheless, fieldworkers still need to record data during observations whether they are standing, crawling or walking. Likewise, environmental officers need to obtain information whenever and wherever they like in whatever conditions (walking, running, driving, etc).

Limited Attention Capacity: Data collection tasks are oriented around observing the subject. Level of attention is dependent on the nature of the subject. For example, when observing animal behaviour, fieldworkers must keep constant vigil on the subject to note any changes in state, so he/she has limited time to interact with the recording mechanism. Similarly, mobile environmental officers also need a device that can minimise the time devoted to interacting with the mobile device in order to obtain the relevant information.

High-Speed Interaction: Taking observation of animal behaviour as an example, subjects requiring time-dependent observation are highly animated and fieldworkers must enter high volumes of data very quickly and accurately or it may be lost. Similarly, environmental officers may sometimes need to make high-speed interaction to obtain relevant information.

Context-Dependency: The fieldworker's activities are intimately associated with their context or the subject's context. For example, in recording an observation of an elephant, its location or the point of observation will be recorded too. In this way, the data recorded is self-describing of the context from which it was derived. The same could be said of an environmental officer observing a polluted area, which may self-describe the context from which it was derived.

Although the relative importance of these four factors can vary with different fieldworker or mobile users like the environmental officer, almost all factors are needed by them in mobile application. To solve the problem of limited attention capacity and dynamic configuration of the user, much research has been carried out in which different approaches have been highlighted and constructed involving input devices and mechanisms, from simple and low-cost hardware and software to high-end and additional hardware requirements. Starting from the assumption that small devices are used under the same conditions as computing devices, which require an input and output device, so the design of input and output modes is also important to mobile applications to make interaction smoother.

3.1 Output Mechanisms

In terms of output, the most significant difference between any two devices is the size of the screen. Although today's devices have more than enough computing power to offer complex GUIs, their limited screen size means that the presentation of complex GUIs is worsened. It is often impractical to simply apply the same user interface as a desktop in a mobile device such as a PDA, mobile phone or smartphone because they have a small screen. Hardware development solutions will not solve the problem entirely until hardware development does not require the capabilities of the user for it to function. Advances in technology cannot solve all these problems either, because human factors remain constant. Screen configuration affects the physical size of the device. For example, in order to address screen size issues, concepts have been designed which use a head-mounted display with a mobile device. However, this approach can be obtrusive and hard to use, especially in bright daylight, and could occupy the user's visual attention. Despite this new interaction, techniques are needed to enable users to access particular services whilst on the move effectively and safely. From various sources it can be concluded that there are several factors that should be taken into consideration when designing the interactive user interface for mobile devices. These are listed below.

User Needs: This is to ensure safe navigation through the environment whilst interacting with the mobile application. The approach that could be used in interaction techniques is "eyes-free" or "hands-free", giving the user freedom in his/her actions.

Social Context: To define the situation in which the technique is to be employed. For example, which gesture interactions are socially acceptable, to what extent speech-based interaction is appropriate, etc.

Physical Context: To do with mobile device situations that are constantly changing, including changes in ambient temperature, noise levels, lighting levels, privacy, etc.

Task Context: Any given task might require different interaction techniques and depend on the context in which the task is being performed, such as while walking, navigating, standing, sitting, etc.

For the most part, research into mobile device interaction has been focused on designing a user interface to involve as little visual attention as possible from its user. According to Sawhney & Schmaridt (2000) and Pirhonen et al. (2002), non-speech audio is an effective approach in mobile device interaction. Relevant information is fed to the user's ears while the user is still able to maintain their visual focus whilst navigating the world around them. This approach requires users to wear an additional listening device, which should not prove to be a problem since the user could wear the same device as for listening to music. Thus a combination of simulated 3D sound and multidimensional gestures was introduced by Begault (1994).

According to Begault, 3D sound allows a sound source to appear as if it is coming from anywhere in space around the user (as listener). For example, in research by Cohen & Ludwig (1991), users could obtain information about different items in different areas by wearing a headphone-based 3D audio. Sawhney & Schmandt (2000), in their research on Nomadic Radio application, investigate the use of 3D sound on a mobile device. This application is a wearable personal messaging system that delivers information and messages to the user on the move via speech and non-speech sounds. The user wears a microphone and shoulder-mounted speakers that provide a planar 3D environment. Nomadic Radio uses an auditory user interface which synchronises speech recognition, speech synthesis, non-speech audio and spatial presentation of digital audio. Figure 3.3 shows the primary wearable audio device in Nomadic Radio.

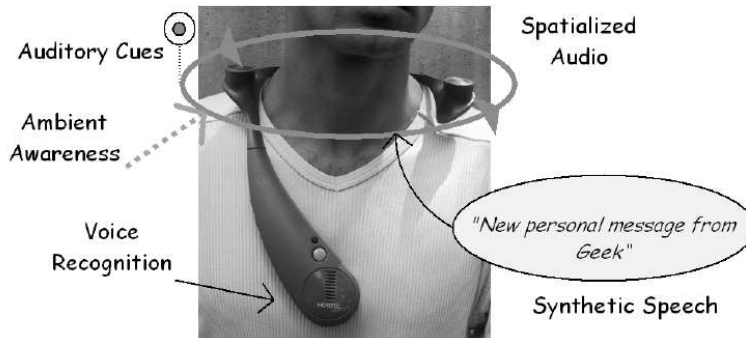


Figure 3.3 Nomadic Radio
(source: Begault, 1994)

Cocktail Party, an application by Arons (1992), uses 3D audio that allows users to listen to multiple sound streams consecutively but still being able to separate each one. The spatial position of the sound around the head can also give information about the file of occurrence. The process of separation is based on factors that can create a "filter" that can separate voices.

According to the research of Brewster & Murray (2000) and Brewster (2002), non-speech audio can improve interaction when the sound allows users to keep their visual attention focused on navigating the world around them. Pirhonen, Brewster & Holguin (2002) use a gesture and non-speech audio to improve the user interface of a mobile music player, where the user may use a player without having to look at its controls while they are walking, running, exercising, etc. This research concerns design and involves a music player controlled with simple gestures, with feedback given via non-speech sound. The sounds used to give feedback on the gestures were sent through earphones. Here, it is important to provide feedback on gestures as users would not see anything but would need to know the state of the device. In other words, gesture and audio feedback are closely related. So, in order to design the user interface for a mobile device, we must take into account the way in which the user may obtain information on the state of his/her action, either through feedback or cues.

However, besides the designing of the GUI and the device itself, the application of adaptive systems and context awareness also play an important role that must be taken into consideration when designing the application in order to push the 'proper' amount of geo-information to the right person, at the right time and in the right way. These approaches ensure that the user gets the possibility to specify the information he/she really needs for his/her purpose, or he/she can specify his/her purpose and then the system could select the necessary data and leave out unimportant details. The basic concept of this approach is that if the system knows enough from the surroundings of the usage situation and current user activity, then the map can be adapted to the context so the user will get an appropriate or proper amount of information that is suitable for the situation and the specific user and for that device. From here, we can see the importance of 'task context' in the design phase of mobile devices, as well as GUI presentation and output devices that could be used with them.

3.2 Input Mechanisms

The traditional mobile phone keypad dedicated to digit input also features letter input (see Figure 3.4). The device is often held in one hand and the thumb is used as an operator to enter digits or letters. The digits (0–9) are mapped to each key in a one-to-one fashion. Letters are mapped to those particular 9 keys (keys 1–9) in a many-to-one fashion, depending on the device model. Each model may have a different approach to generating a letter. The mobile phone letter keypad is normally used for sending short text messages (SMS).

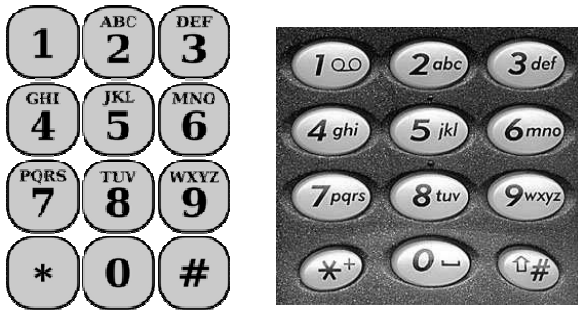


Figure 3.4 Normal Mobile Phone Keypad

However, there are some differences in numerical layout between phones and keyboards (see Figure 3.5). The telephone layout locates digit 1 at the upper left, while the keyboard layout locates digit 1 at the lower left. In the interests of preventing typing errors, various researchers have studied why the telephone keypad is arranged differently from the calculator keypad. There is a theory that phone companies intentionally reverse the calculator configuration to slow down operating speeds, especially those of people who are already fast at operating a calculator, to allow the phone signal to register. Tone recognition technology could not operate effectively at the speeds at which these experienced users and specialists could dial the numbers. So telephone designers figured that if they reversed the layout, the dialling speeds would decrease and the tone recognition would be able to do its job more reliably. Theoretically, when we pick up the phone, the exchange senses the completion of our loop and it plays a dial tone sound so we know that the exchange and our phone are working. The dial tone sound is simply a combination of a 350Hz tone and a 440Hz tone (see Figure 3.6, Keyboard Trivia).



Figure 3.5 Numerical layout of (a) telephone and (b) computer keyboard

	1,209 Hz	1,336 Hz	1,477 Hz
697 Hz	1	2	3
770 Hz	4	5	6
852 Hz	7	8	9
941 Hz	*	0	#

Figure 3.6 The Dial Tone Sound
 (source: <http://communication.howstuffworks.com/telephone5.htm>)

Another theory refers to a study carried out by Bell Labs in 1960. This study involved testing several different telephone keypad layouts in order to find out which was easiest to use. After testing several layouts, including one that used two rows of five numbers each and another that used a circular positioning, it was determined that the three-by-three matrix that had 1, 2 and 3 across the top was the easiest for people to use. Another theory is based on the layout of a rotary telephone or push-button dialling (see Figure 3.7). On a rotary dial, 1 is at the top right and zero is at the bottom. However, it did not make sense to design the new keypad (touch-tone dialling) with digit 1 on the top right because Western writing is read from left to right. Putting 1 on the top left, and the subsequent numbers to the right, did make sense, with zero having its own row at the bottom. There was another reason as well. If we stick with the keyboard layout, when it came to matching the letters of the alphabet up with the numbers, putting 1-2-3 across the top made a lot more sense because it was the most natural way to get ABC in the top row. If 7-8-9 had been at the top, the letters and numbers would have run in opposite directions, and PQRS would have been

the first set of letters. Either arrangement would have seemed very odd and complicated indeed. So some telephone companies designed their phone keypads to mimic the QWERTY computer keyboard, as illustrated in Figure 3.8.



Figure 3.7 Rotary telephone or push-button dialing



Figure 3.8 Telephone keypads with QWERTY style keyboard

The Velotype keyboard could provide an alternative solution to the prevention of typing errors. The Velotype is a specially adapted computer developed in the Netherlands that allows a trained typist to input text at speaking rate. It is a portable wireless text telephone with an extra microphone and text display in its later versions. Velo-typing is an easy, natural and ergonomic way to type three times faster than normal QWERTY, AZERTY or QWERZ typing in all EU languages, offering very favourable applications for the benefit of the hearing impaired in (real-time) telecaptioning applications. For example, when a deaf person visits the doctor, meets someone in the street, sits in a classroom or any of the thousands of situations in which a deaf person would need to have access to spoken information, he/she only need dial the "velotypists relay centre", place the microphone near the sound source, and read a real-time transcription of what is being said from the display. A copy of the text can be read by a single user, or by multiple users in a single or in several locations. Telecaptioning allows hearing impaired persons to have access to and participate in teleconferences, satellite conferences and other speech-based telematics applications. Figure 3.9 shows the Velotype keyboard.



Figure 3.9 The Velotype Keyboard

When using a standard keyboard (such as QWERTY, AZERTY or QWERTZ), we must enter text from a 26-character alphabet using the keypad. This task forces a mapping of more than one character per button of the keypad. A typical mapping has keys 2–9 representing either three or four characters, with space and punctuation mapped to the other buttons. All text input techniques that use this standard keypad have to somehow resolve the ambiguity that arises from this multiplexed mapping. There are three main techniques for overcoming this ambiguity: multi-tap, two-key and linguistic disambiguation (Wigdor & Balakrishnan, 2003).

3.2.1 Multi-tap

The most common style of text entry using a mobile phone number pad is multi-tap (Mackenzie, 2002). This style uses multiple letters that are mapped to the same key (see Figure 3.4). Users can enter text into the keypad with one or both hands, using one or two fingers or thumbs, pressing a key to cycle through the letters until the desired one appears onscreen, and proceeding to the next letter if it is located on a different key. The problem arises when the desired letter is on the same key. For example, tapping the 2 key three times could result in either *c* or *ab*. To overcome this, multi-tap employs a time-out of typically 1–2 seconds on the button presses, so that not pressing a button for the length of the time-out indicates that you are done entering that letter. Entering *ab* under this scheme means the user must press the 2 key once for *a*, wait for the time-out, then press 2 twice more to enter *b*. To overcome the time overhead this incurs, many implementations add a “time-out kill” button that allows the user to skip the time-out. If we assume that 0 is the time-out kill button, this makes the sequence of button presses for *ab*: 2,0,2,2. In other words, users must wait a short time for the phone to accept the letter or recognise the “time-out kill”. Alternatively, some phones use the “#” button to force the phone to move to the next letter. For example, the word “good” can be written by entering “4666#6663” (9 key presses).

3.2.2 Two-key Disambiguation

The two-key technique requires the user to press two keys in quick succession to enter a character. The first key press selects the appropriate group of characters, while the second identifies the position of the desired character within that group. For example, to enter the character *e*, the user presses the 3 key to select the group, followed by the 2 key, since *e* is in the second position within the group. This technique is quite simple but has failed to gain popularity for Roman alphabets.

3.2.3 Linguistic Disambiguation (T9)

Another common typing method on mobile phones is T9 (predictive text), which uses a dictionary to disambiguate the user's input. Just like multi-tap, multiple letters correspond to the same key. However, the user is required to press keys with the corresponding letters only once. As the user enters the word, the phone continually guesses what word the user is typing. Users can cycle through potential words by pressing the "#" key. For example, if the user types "3323" for "deaf", the first word T9 would guess is "dead", but the user may press the "#" key until the phone displays the proper word. However, T9 can be frustrating and slow when entering words the phone does not recognise.

3.3 Key Press Techniques

There are several techniques under research concerning key press methods for efficient input of data that could also benefit interactive techniques in mobile phones. Small screen devices often do not have QWERTY keyboards and mouse-like devices like the desktop computer. Therefore, to address issues of size and portability some input modes have been developed to mimic the keyboard and the mouse for use with small devices. Many small portable electronic products, like chord keyboards, have also been proposed as input devices. A chord keyboard is a keyboard that takes simultaneous multiple key presses to form a character in the same way that a chord is played on a piano. In chord keyboards, the user presses multiple key combinations, mainly two-letter combinations, to enter an input instead of using one key for each character. Since chord keyboards require a small number of keys, they do not require much space, or as many keys as regular keyboards such as the QWERTY keyboard.

For example, the Twiddler2 (see Figure 3.10) is a one-handed chord keyboard with only 12 fingertip keys. The Twiddler2 is a pocket-sized mouse pointer plus a full-function keyboard in a single unit that fits neatly in the user's right or left hand and a ring of control keys under the thumb. The Twiddler2's mouse pointer is the IBM TrackPoint. When controlling the mouse pointer with the thumb, the fingers in front control the click functions. For example, when pressing the mouse button with the thumb, the "A" button on the front is your LEFT CLICK, "B" is a dedicated DOUBLE LEFT CLICK, and so on. The Twiddler2 incorporates a radical new ergonomic keypad designed for "chord" keying. This means that the user presses one or more keys at a time. Each key combination generates a unique character or command. With 12 finger keys and 6 thumb keys, the Twiddler2 can emulate with ease the 101 keys on the standard keyboard.



Figure 3.10 Twiddler2
(source: <http://twiddler2.com/>)

The BAT Personal Keyboard (see Figure 3.11) is a one-handed compact keyboard that replicates all the functions of a full-size keyboard. As a "chording" keyboard, letters, numbers, commands and macros are created with key combinations or "chords". The BAT is a good typing solution for people with physical or visual impairments. It can be used as a one- or two-handed chording keyboard.



Figure 3.11 BAT Personal Keyboard

(source:http://www.maxiaids.com/store/prodView.asp?utm_source=Froogle&utm_medium=organic&utm_campaign=Froogle&idAff=15225&idproduct=3583)

The Virtual Keyboard (see Figure 3.12) (<http://www.virtualdevices.net/Products.htm>) provides similar input capabilities as a desktop keyboard. The difference is that it does not require external space to balance the keyboard and is appropriate for mobile situations where space is at a premium. It can be projected onto and used on any surface. The keyboard watches your fingers move and translates that action into keystrokes in the device. Most systems can also function as a virtual mouse.



Figure 3.12 Virtual Keyboard

The Senseboard (see Figure 3.13) (<http://www.senseboard.com>) is a unique design that allows standard text input in practically any environment. The Senseboard consists of two hand-worn components which use Bluetooth technology to connect with the designated computing device. Senseboard utilises sensor technology to recognise the characters a user is typing. Text input is based on movement of the users' hands and fingers.



Figure 3.13 Sensorboard

The Lightglove (see Figure 3.14, and demo session at <http://www.lightglove.com/demofr.htm>) is a new human interface device that requires the user to wear a device on the underside of his/her wrist. Light from the device (visible or infrared) scans the palm and senses wrist, hand and finger motion. This data is translated into either on-screen cursor control or key closures and in addition works as a long distance on/off switch for virtually all electronics. Wireless operation offers cordless freedom, thereby reducing the occurrence of repetitive strain injuries such as carpal tunnel syndrome.



Figure 3.14 Lightglove

KITTY (see Figure 3.15 and Figure 3.16, demo at <http://www.kittytech.com>), an acronym for Keyboard-Independent Touch-TYPing, is a finger-mounted keyboard that uses touch-typing as the method of data entry. The device is targeted at the portable computing market and in particular wearable computing systems, which are in need of a silent, "invisible" data entry system based on touch-typing. The new device combines the idea of a finger-mounted chording device (such as the Twiddler2) with the advantages of a system that uses touch-typing (such as the Virtual Keyboard, Senseboard, Lightglove, or the Scurry by Samsung (<http://www.time.com/time/magazine/article/0,9171,220015,00.html>)).



Figure 3.15 KITTY (Keyboard-Independent Touch-TYping)



Figure 3.16 Finger-mounted keyboard

Several types of glove-based devices that recognise hand gestures or contact gestures directly have been widely proposed as computer input devices. These devices are well suited for use in a mobile environment because the gloves can be worn instead of held, are lightweight, and take up little carrying space. It is, however, difficult to recognise enough separate gestures to allow useful text input. Rosenberg & Slater (1999) proposed a glove-based input device called the chording glove to combine the portability of a contact glove with the benefits of a chord keyboard. In their chording glove, the keys of a chord keyboard were mounted on the fingers of a glove and the characters were associated with all the chords, following the keymap. The Chording Glove is shown in Figure 3.17.



Figure 3.17 Chording Glove

The Pinch keyboard (Bowman, Ly & Campbell, 2001) is a virtual keyboard that uses Pinch Gloves (see Figure 3.18), lightweight gloves with conductive cloth on each fingertip that sense when two or more fingers are touching. The Pinch keyboard uses the same key layout as the QWERTY keyboard, and some “inner” characters were selected by a wrist-rotating motion that could be detected by motion trackers attached to the gloves. Pinching is basically the motion of making contact between the tip of the thumb and a fingertip of the same hand.



Figure 3.18 Pinch Glove

According to Pratt et al. (1998), “Thumbcode” is also a one-handed device-independent chording sign language with a glove-based input device similar to the Pinch keyboard. The Thumbcode paper can be viewed at <http://boole.stanford.edu/thumbcode/>.

Figure 3.19 below shows how to spell “C A T” in Thumbcode. The Thumbcode glove is shown in Figure 3.20.



Figure 3.19 How to spell "C A T" in Thumbcode



Figure 3.20 Thumbcode Glove

Unfortunately, when the user is frequently mobile, distracted or busy with other tasks, a keyboard becomes problematic when the user stops keying in the data, and a stylus pen might easily be dropped and lost. Static input methods for mobile devices have had a negative effect in terms of dynamic usage. This is because those methods (stylus or keyboard usage) often rely on using two hands, one to hold the device and one to enter the data. Pascoe et al. (2000) identified the most comfortable grip for one-handed operations, shown in Figure 3.21, and Dong et al. (2005) the one shown in Figure 3.22.

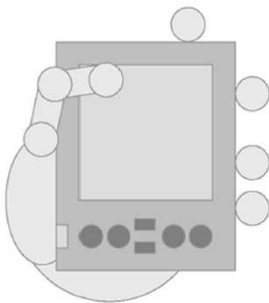


Figure 3.21 Comfortable grip in one-handed operations (source: Pascoe et al., 2000)



Figure 3.22 Comfortable grip in one-handed operations
(source: Dong et al., 2005)

Such a hand grip leaves only one thumb free to operate the device. In other words, this limits the modes of task input. In answer to this, Rekimoto (1996) created another innovative input mechanism, a device that could handle the input mechanism using only single-hand operation, using tilt as the input method. In spite of the disadvantages of having a stylus, touch screen devices have advantages such as providing direct manipulation simply by touching a visual display of choices, easier hand-eye coordination than a mouse or keyboard, no extra workspace and durability in terms of public access and high-volume usage. But small screen size again restricts the usage of a touch screen. These devices cannot be used without a stylus due to the fact that user element interfaces are often smaller than the size of a human fingertip. Selecting text blocks, for example, becomes difficult without a stylus.

3.4 Tilt-based Interaction

Several researchers have recently proposed interesting interaction techniques that are enabled by incorporating a low-cost tilt sensor within mobile devices. Partridge et al. (2002) and Eslambolchilar et al. (2004) proposed the TiltType (see Figure 3.23) to provide fast and reliable text entry to small watch or pager-like devices. It works by detecting the tilt of the device with a small onboard accelerometer. To enter text the user simply presses a button and holds it down, while tilting the device in the direction of the desired letter. When the desired letter is displayed, the user stops pressing the button and the character is entered. The device is then ready for the next command. The letters are laid out in a tic-tac-toe fashion for each button, thus each button can access nine letters (e.g. right-forward, centre-backward). Three buttons therefore give the English alphabet. By pressing a combination of buttons, the user can write numbers, special characters, backspace and space.

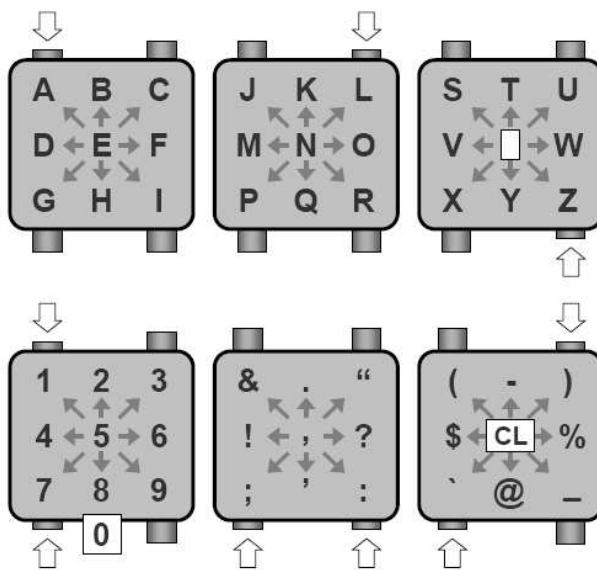


Figure 3.23 TiltType

TiltText (Wigdor & Balakrishnan, 2003) is a new technique for entering text into a mobile phone. TiltText uses a combination of a button press and tilting of the device to determine the desired letter. It uses the same concept as the TiltType technique, using tilt to disambiguate button presses. The difference between TiltType and TiltText is that TiltType uses eight tilt directions, while TiltText uses a maximum of four tilt directions, in order to reduce the accuracy demands on the user when tilting. TiltText is not language dependent but it can be used only by experts because it does not require the user to pay visual attention to the display screen. TiltText uses the orientation of the phone along two axes to disambiguate the meaning of button presses. Tilting the phone to the left selects the first letter of the key, away from the body the second, to the right the third, and, if present, towards the body the fourth (see Figure 3.24). Pressing a key without tilting results in entering the numeric value of the key. Space and backspace operations are carried out by pressing unambiguous single-function buttons (as in multi-tap).

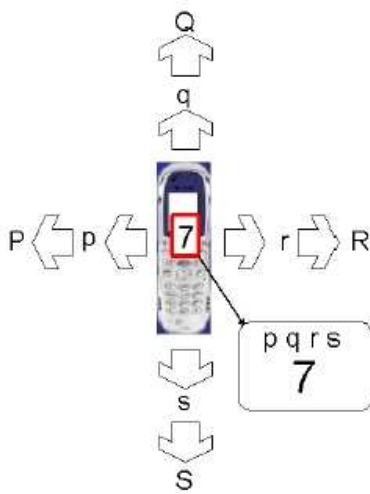


Figure 3.24 TiltText

Unigesture (Sazawal, Want & Borriello, 2002) is heavy on the wrist and allows only linear searching. If T9 is a design that saves the user from unnecessary button presses, Unigesture also applies these time-saving ideas to tilt-to-write. With Unigesture, the alphabet is divided into 7 zones. Space is allocated a zone of its own. To write a letter, the user tilts in the direction of a zone and then returns to the centre. An example is shown in

Figure 3.25. No additional gesture is needed to specify which letter is wanted within the zone. Rather, the Unigesture system will accept a sequence of tilt gestures and then attempt to infer a word. The advantages of the Unigesture method include the variety of slight tilt gestures that can be used. Each tilt gesture returns the user to a neutral centre position, which prevents strain on the arm and wrist. Slight tilts also make for quick motion, which suggests that Unigesture users should be able to write quickly.



Figure 3.25 Unigesture users must tilt a small device
(source: Sazawal, Want & Borriello 2002)

Currently, the Samsung SCH-S310 and Nokia 550 provide human-motion recognition. The Samsung SCH-S310 application can be viewed on the YouTube website at <http://www.youtube.com/watch?v=rYJ1EfpV3jI> and <http://www.youtube.com/watch?v=yqYVJ0zpDx8>, and at <http://www.youtube.com/watch?v=NXcSm94RK6A> for the Nokia 550. Using a sensor, the Samsung phone can switch from landscape to portrait when shooting video or snapping a still picture, or prevent a picture from becoming blurred from hand shake. Games can also be played by moving the phone up, down, right or left, instead of pressing buttons. The phone will also be able to sense changes in body movement and provide advice on dieting or other health tips. The HTC X7500 is also equipped with a built-in tilt sensor (see Figure 3.26).



Figure 3.26 HTC X7500

GestureWrist (Rekimoto, 2001) supports hands-free operation but needs to be attached to the wrist. It is a wristband-type input device that recognises hand gestures and forearm movements. GestureWrist recognises hand gestures by measuring changes in the arm shape on the inside of the wristband. To do this, combinations of transmitter and receiver electrodes are attached to the back of the watch dial and the inside of the wristband. However, this kind of input device must be combined with another device to present the output, such as an audio feedback function (see Figure 3.27).

FreeDigiter (Metzger, Starner & Anderson, 2004) is an interface for mobile devices which enables rapid entry of digits using finger gestures. It works via an infrared proximity sensor and a dual axis accelerometer and requires little signal processing. It supports easy gestures by recognising the number of fingers swept past its proximity sensor. For example, users may move two fingers past FreeDigiter's proximity sensor to indicate option two (see Figure 3.28).

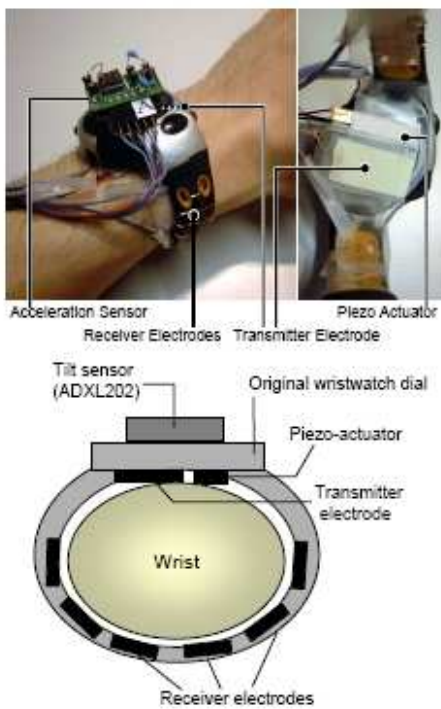


Figure 3.27 GestureWrist
(source: Rekimoto, 2001)



Figure 3.28 FreeDigiter
(source: Metzger, Starner & Anderson, 2004)

WristCam (Vardy, Robinson & Cheng, 1999) provides one-handed interaction but requires a camera to recognise gestures. This research focused on how images of a user's hand from a video camera attached to the underside of the wrist can be processed to yield finger movement information. Discrete movements of the fingers away from a rest position are translated into a small set of base symbols. These are interpreted as input to a wearable computer.

According to Yee (2003), Peephole is a movable window and spatially aware input method that tracks spatial information for use with a small screen. The screen of a small device equipped with Peephole functions as a window over a larger workspace, so that the user sees different parts of the entire workspace by physically moving the small device across the larger area. An example Peephole method is shown in Figure 3.29. In terms of an interaction method, Yee combines the possibilities of the mobile window with virtual space using both hands and stylus for interaction. For this purpose, one hand is used to move the PDA through the virtual 3D space showing its contents while the other hand uses the digital pen to interact with the representation. For example, we can write text longer than the screen by moving both hands co-ordinately whilst writing (see Figure 3.30). We can also do large drawings by moving one hand through space and the other on top of the screen. A big circle can be drawn just by holding the pen in the centre of the screen and moving it in circles in space (see Figure 3.31).

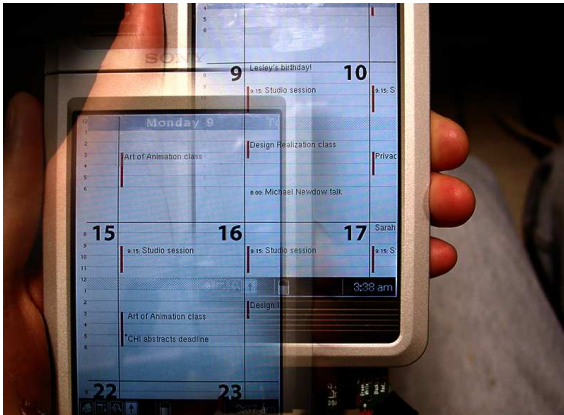


Figure 3.29 Example of peephole method to pan the calendar using only one hand



Figure 3.30 Example of peephole method to write text longer than the screen by moving both hands co-ordinately whilst writing

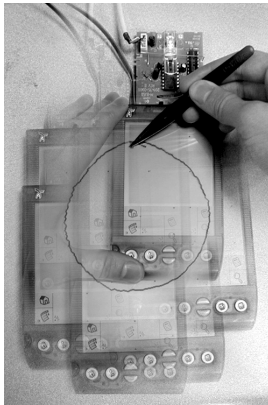


Figure 3.31 Example of doing large drawings by moving one hand through space and the other on top of the screen

There is also some research that focuses on one-handed device interaction (on the assumption that one-handed usage will minimise user attention. Examples include the AppLens and LaunchTile (Karlson, Bederson & SanGiovanni, 2005). The designs use variations on zooming interface techniques to provide multiple views of application data. AppLens uses tabular fisheye to access nine applications, while LaunchTile uses pure zoom to access thirty-six applications. Two sets of thumb gestures represent different philosophies for one-handed interaction, as shown in Figure 3.32.

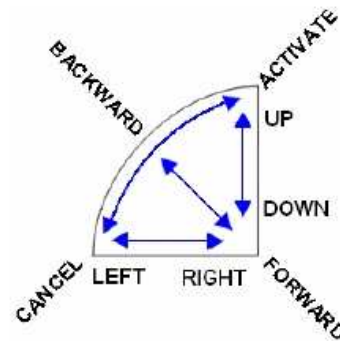


Figure 3.32 AppLens and LaunchTile

4 Conclusion

In conclusion, we can say that there are two important questions to be answered and taken into consideration when designing applications for mobile devices: "how to represent geo-information in a small device" and "how to make a small device interactive". There are also several factors that are constraints to designing the user interface, such as small and lightweight hardware, hands-free operation and no full attention demand. These three requirements could represent a guideline for designing the best interaction techniques for mobile devices.

As explained before, when users or "fieldworkers" are working outside the office (e.g. environmental officers), they will undertake tasks dynamically, whenever and wherever they like, in whatever condition. They will also have limited attention capacity, depending on the nature of their subjects, and therefore their time for interacting with the device or mechanism will be limited. While on the move, fieldworkers may sometimes need to make high-speed interaction to obtain relevant information, so this must be taken into account in application development. Context is also very important and plays an important role in answering both questions above. In order to interact efficiently with the mobile device, we can divide the interaction technique into two categories: output mechanism and input mechanism. For the output mechanism, both hardware and software development are involved and interaction techniques that might be used must be effective and safe. For example, using a head-mounted display in order to cater for screen size issues, and for software development, the device must have a sensor that can recognise the context involved, such as a sensor that could recognise the weather and alter the brightness of the display based on weather information. Besides this, the output mechanism must account for user interaction needs, such as an "eyes-free" or "hands-free" approach to give users freedom in their actions. Gesture type or to what extent speech-based interaction is socially acceptable, otherwise known as social context, are also important in choosing an interaction technique. Other than that, physical context (e.g. changes in ambient temperature, noise level, etc.) and task context are also important. There are various techniques (such as wearing headphone-based 3D audio, using a gesture and non-speech audio) and applications (such as the Nomadic Radio application, Cocktail Party) that have been used and studied to represent information to the user. For the input mechanism, there are various techniques and ways to input data such as multi-tap, two-key and linguistic disambiguation. Various devices can be used, such as Twiddler2, BAT Personal Keyboard, Senseboard, Lightglove, KITTY, Finger-mounted keyboard, Chording Glove, Pinch Glove, Thumbcode Glove, etc. and various techniques such as tilt-based interaction, TiltText, Unigesture, GestureWrist, FreeDigiter, WristCam, Peephole and AppLens and LaunchTile.

Hopefully, based on this report, the best technique for interaction with mobile devices could be chosen. The mode of interacting and representing appropriate information to the user must involve context-aware technology, so a system (software) or device (hardware) can adapt and provide relevant information in order to suit the needs of the user, of the environment, and of the equipment. Or, in other words, an application must be context-aware and capable of perceiving the user situation in its many aspects. And adaptation is a goal of context awareness which is able to drive the application without explicit tuning operations by the user. Involvement of context awareness in the RGI-149 project is very important in order to support the surveillance activities of environmental officers. By using this approach, the system can push the proper amount of geo-information based on the profile of the user, identity, location and behaviour. Thus it could indirectly solve storage limitation problems and provide only related and relevant information based on user needs. This special kind of location-aware, adaptive, context-driven device can supply geo-information to a group of users whose activities are well depicted and understood, bringing this application as a complete system to an environmental application that can aid environmental protection.

References

Arons, B. (1992) "A Review of the Cocktail Party Effects", *Journal of the American Voice I/O Society*, Volume 12, pp.35–50, July 1992.

Eslambolchilar, P. and Murray-Smith, R., *Tilt-Based Automatic Zooming and Scaling in Mobile Devices—a State-Space Implementation*, *Proceeding of Mobile HCI 2004*, Glasgow, Springer LNCS, 2004.

Begault, D.R. (1994) "3-D Sound for Virtual Reality and Multimedia", *Academic Press Professional, Inc*, p.293.

Brewster, S.A. (2002) "Overcoming the Lack of Screen Space on Mobile Computers", *Personal and Ubiquitous Computing*, Volume 6, Issue 3, pp.188–205, Springer-Verlag, London UK.

Cohen, J.M. and Ludwig, L.F., "Multidimensional Audio Window Management", *International Journal of Man-Machine Studies*, Volume 34, Issue 3, pp.319–336, Academic Press Ltd.

Keyboard Trivia, The Calculator Reference [online] <http://www.vcalc.net/keyboard.htm>

Pascoe, J., Ryan, N. and Moves, D. (2000) "Using While Moving: HCI Issues in Fieldwork Environment",

Pirhonen, P., Brewster, S.A. and Holguini, C. (2002) "Gestural and Audio Metaphors as a Means of Control in Mobile Devices", *Proceedings of the SIGCHI conference on Human factors in computing systems: Changing our world, changing ourselves*, pp.291–298, ACM Press.

Sawhney, N. and Schmandt, C. (2000) "Nomadic Radio: Speech and Audio Interaction for Contextual Messaging in Nomadic Environments", *ACM Interaction on Computer-Human Interaction*, Volume 7, Issue 3, pp.353–383, ACM Press.

Corresponding address

Delft University of Technology
OTB Research Institute for Housing,
Urban and Mobility Studies
Jaffalaan 9, 2628 BX Delft
PO Box 5030, 2600 GA Delft
The Netherlands
tel. +31 (0)15 278 30 05
fax +31 (0)15 278 44 22
e-mail: mailbox@otb.tudelft.nl
www.otb.tudelft.nl