DistScroll - A new One-Handed Interaction Device

Matthias Kranz, Paul Holleis,Albrecht Schmidt
Research Group Embedded Interaction
University of Munich
Amalienstraße 17
80333 Munich, Germany
{matthias, albrecht, paul}@hcilab.org

Abstract

We present an interaction device for navigating data structures or browsing menus using only one hand. It is especially designed for situations in which the user wears gloves that renders direct input too difficult or applications in which it is impossible or impractical to use both hands. Besides a detailed description of the technical realization of first prototypes, we compare application potentials and advantages of our system over existing devices.

1 Introduction

Mobile devices of pad and tab size as described by Weiser in [16] are available nowadays in a great variety, but there are only very few ways of interacting with such a device. The three most prominent ways are inputting information with or without a stylus on a touch-sensitive display or by entering data via a keypad, using techniques like multitap or T9 and finally by fine movements, e.g. a finger on a mobile phone joystick. Recently there are research efforts on other sensor-based interaction techniques for such devices, like [17], [8], [11] or [9].

In this paper we present a new interaction technique that utilizes the distance between the user and the device as an additional input parameter. We investigate how this parameter can be used for browsing hierarchical data structures such as menus.

The paper starts with a short overview of related work and motivates our approach based on previous findings. We then introduce our research prototype, which we named DistScroll, after our initial experiment - distance-based scrolling. We show possible application areas where distance-based scrolling appears a suitable method of input. Some findings from initial user interviews are presented in this section, too. Finally, a short outline of further research work on the DistScroll is given.

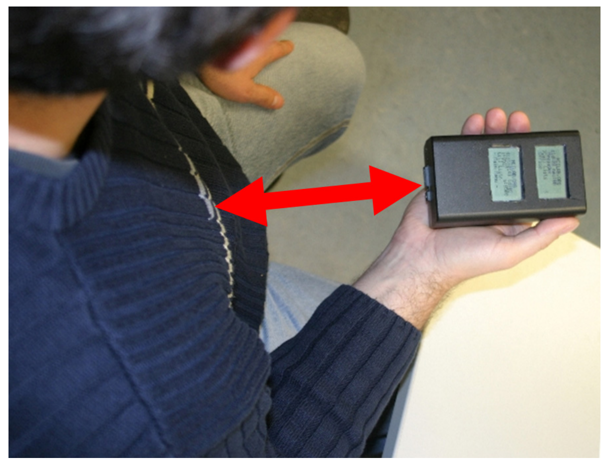

Figure 1. A user interacting with the DistScroll. By moving device as the arrow indicates, he scrolls through the menu entries of a fictive application depicted on the top display. On the bottom display additional state information is presented.

2 Related Work

Providing input to mobile devices has been a research focus in the area of mobile and wearable user interfaces over the recent years.

Besides the mainstream approaches of using keyboard or touch screen, input techniques using movements and gestures have been investigated as potential alternatives for text and navigation input to small devices.

Some of them are in direct analogy to traversing linear or hierarchical data structures: [17] and [11] use tilting to remove the ambiguity of the mapping of several letters to the same button on standard mobile phone keyboards. TiltText in [17] still requires the user to press small buttons. Additionally, by tilting the device the user also changes the viewing angle on the display significantly which in certain circumstances can be problematic. We also argue that using this input method for a longer period of time is fatiguing.

The Unigesture approach ([11]) uses gestures only. Letters are grouped into zones that can be selected by tilting the device. A 'T9' like algorithm is used to disambiguate entered words. However, this puts a high load on the wrist, allows linear searches only and is highly specialized for word input of a specific language.

GestureWirst [10] supports hands-free operations, but it needs to be attached to the wrist. This may interfere with tight (in a laboratory scenario) or thick clothing (in winter environments) when worn underneath.

The gestures required for using FreeDigiter [7] are easy to be made when wearing no restrictive clothing such as parkas and we think of them as unsuitable when e.g. dealing with chemicals as there may be hazardous fluids on the gloves.

The WristCam by Vardy et al. in [15] provides one-handed interaction but needs a camera for recognizing the gestures made with the fingers. We think that any usage scenario involving gloves would render input difficult if not even impossible.

Methods specialized for scrolling through larger sets of data have been devised for a large variety of applications and input devices. The Radial Scroll Tool ([14]) was designed to navigate through documents. It could also, however, be applied to data structures of any sort as well. Whenever the user makes rotational gestures, a small wheel shaped graphical widget is displayed that the user can turn to advance or reverse the document. Unfortunately, this works only on touch screens because otherwise, the display is no longer static and the user would have to refocus.

The Rock 'n' Scroll input method by Bartlett (described in [2]) includes a method to navigate a list of thumbnails. By tilting the device around two axis, it is possible to scroll a matrix in the two directions. Bartlett uses several more complicated gestures for indicating that a picture should be shifted, zoomed into or displayed in landscape or portrait mode. Each gesture is recognized as a discrete event and no continuous operation for browsing structures is suggested.

The TUISTER ([3]) provides an interface where the user can turn part of a device thus exploring one level of a menu structure. Turning the second part with the other hand, an entry can be selected or you can move up one level in the menu. For many application areas one limitation is that both hands have to be used. Furthermore it may be difficult to build a device that can be used equally well by right and left-handed people.

Our approach is based on a similar idea as the YoYo-interface described by J. Rantanen in [9]. They have investigated the use of wearable computers and smart clothing in the context of the arctic environment. They built a prototype for experienced snowmobile users to prevent accidents and to help survival in case an accident occurs. For this they needed a user interface that was capable of controlling several different features built into the snowmobile suit. It had to meet the restriction that persons are wearing thick gloves. They suggested a YoYo-like device attached to the garment. It can be pulled with one hand and retracts automatically using a spring. By pulling, a wheel is turned and this is translated as an input parameter. Selection was done by pressing the device.

Our approach differs from Rantanen's idea in that our device is not attached to any part of the clothing or the environment. The sensing can be built into any handheld device. We further do not use mechanical parts as that could be vulnerable to defects of any kind, e.g. fluids penetrating the case of the device. We also hope that by removing wires and additional mechanical parts users are more willing to use this new interaction method. Furthermore, the reduction of mechanical parts reduces costs. Other input techniques like [10], [15] or [7] use gestures in the form of acceleration or image/video data as input. Those gestures have to be detected and interpreted with more or less costly algorithms. In our approach, the input parameter can be directly derived from the sensor without the need of heavy input processing.

To explore several new possibilities we built the DistScroll hardware which is described in more detail in Section 4.

3 DistScroll – a new Interaction Technique

The basic idea of DistScroll is to sense the distance between the user's body and the mobile device he or she is holding. The basic idea is depicted in Figure 1.

3.1 Design Goals

In comparison to the approaches described above, the DistScroll concentrates on the following issues:

- an easy way to use without a long learning phase
- an intuitive mapping between data structure and interaction method
- one handed input, potentially wearing a glove

To make use easy and to avoid long learning phases we looked for an interaction technique that is easy to do and does not require learning of special gestures. One obvious parameter that can be easily varied is the distance between the user's body and the device. A device is always held at a certain distance and by moving the device close or away, the user can control a parameter. To keep this interaction comfortable, the distance has to be kept within a certain range.

A further concern to us was the mapping between the interaction performed and the resulting interaction in the user interface. We first investigated scrolling as the motion performed by the user which is directly mapped to the action in the user interface.

A further requirement we wanted to investigate was that interaction should be possible using one hand only. In many situations where people use mobile devices, such as mobile phones, people are doing something else, too. Therefore, often only one hand is available for inputting data. Using one hand only, we additionally wanted to find out how a device would have to be constructed so that it also can be operated while wearing gloves. By not having to input data as precise as with a stylus, the not-preferred hand can also be used easily.

3.2 Research Approach

Based on the initial idea we considered various options for conducting the research. From talking to people – colleagues, students, and potential users – it became obvious that it is important to have a prototype to try out this new form of interaction. Three basic options were considered:

- simple distance measuring prototype attached to a PC
- simple distance measuring prototype attached to a mobile phone / PDA
- self contained interaction device that can be wirelessly linked to a PC

We opted for the third option as we felt it could provide us with the most freedom for the research with the least effort for prototyping. When we know some of the parameters (e.g. where to fit the sensor) we plan to re-implement the prototype as additional component to a PDA or mobile phone. Having a device connected to the PC and using the PC screen for visual feedback would have been the easiest way to implement this. However, issues related to the movement of the device carrying the displays would not be taken into account. We also felt that a device connected by wire to a PC would have been used less freely and would detract the user's attention.

In the next section we describe the prototype built. Additionally to the distance sensor, it also includes most common sensors we found when reviewing related work.

4 Prototype

The DistScroll prototype is implemented based on Smart-Its technology ([4],[12]), using a Microchip PIC 18F452 8-bit microcontroller with 32 kBytes of flash memory and 1,5 kBytes RAM. The prototypical design comprises two distance sensors (only one is used in our experiments so far), a sensor for two axes of acceleration, three push buttons and two displays. The DistScroll prototype is implemented as an add-on board to the Smart-Its platform. The hardware and battery are fitted in a small case, similar in size to a PDA. The device is powered by a 9 Volt block battery. The interior of the prototype is depicted in Figure 3 and in Figure 1 the prototype can be seen in use. The code for the microcontroller in the DistScroll device is programmed in C. The overall hardware architecture is depicted in Figure 2.

4.1 Hardware Overview

The interior of the DistScroll is shown in Figure 3. At the top half of the device is the Smart-Its add-on board with the displays and the distance sensor (black cables on the left going to the top, '1' in Figure 3). The add-on board ('2' in Figure 3) is connected to the Smart-Its base board using standard add-on board connectors. To allow an opening of the device for battery changes and code downloads, these connectors were elongated with ribbon cable (in the middle on the right side).

The battery is situated on the right ('4' in Figure 3). Display brightness can be adjusted with a potentiometer ('4' in Figure 3). Next to the Sharp distance sensor ('5' in Figure 3) on the left hand side is the base Smart-Its board with serial and programmer connector.

4.2 Distance Sensors

The integral part of the presented hardware is the distance sensor at the bottom of the DistScroll device (see Figure 3). We used a Sharp distance sensor GP2D120.

The distance between the DistScroll device and an object before the distance sensor is measured. The sensor sensitivity is depicted in Figure 4 and Figure 5. We chose this special sensor as its measurement range fits perfectly for the predicted normal usage of the DistScroll device of about 4 to 30 cm. It starts immediately in front of the sensor and

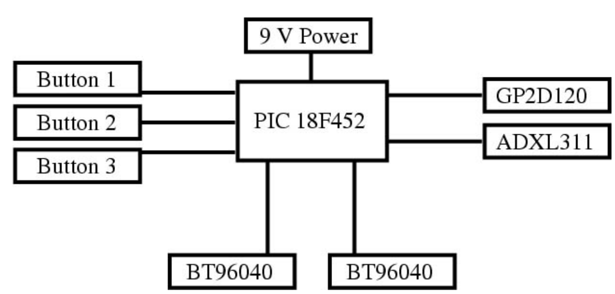

Figure 2. DistScroll system architecture.

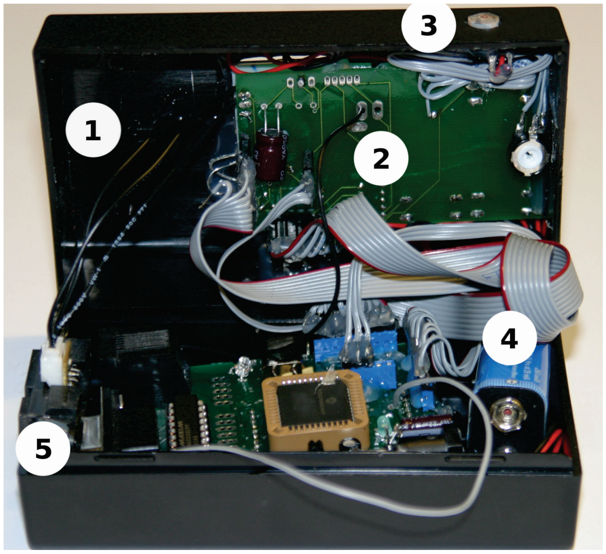

Figure 3. The open DistScroll device. The display and sensor add-on board is in the top of the case. The Smart-Its base board is in the lower part of the case. In the middle of the base board is the PIC 18F452 microcontroller.

stretches far enough. So, when the device is moved maximally away while holding it with the hand, this distance is still measurable and the display on the DistScroll device is still readable for a user.

Another important characteristic of the Sharp infra red distance sensor is, that the color (the reflectivity) of the object in front of the sensor does nearly not matter. The device can be used with arbitrary colored clothing (see data sheet of the device, [13]). Potentially problematic could be reflective surfaces with clear boundaries between the parts of the surface. This could distract the emitted light so that no correct measurement could be made.

If the device is moved towards the user, the values returned by the sensor rise, if moved away the decline until no measurement can be made, that is if the distance is larger than 30 cm. If the user moves the device too close, the values decline again. This undesired behavior occurs if the device is closer than 4 cm to the user. In this case it therefore cannot be detected if the device is moved away (> 4cm) or towards the user (< 4 cm). However, since it is hard and uncomfortable to read a small display that is very close to the front of the body, this can be tolerated. Additionally, initial tests show that users are aware of this sensor characteristic and learn how to avoid this behavior. It is also possible - because of the much faster declining sensor values between 0 and 4 cms - that this sensor characteristic is exploited by advanced users for faster scrolling or browsing.

The sensor values are not linear in the measurement range of the sensor. Therefore, we could not choose a linear mapping between sensor values and structure entities: When moving the sensor close to an object, many entities would be scrolled with only a small amount of movement. Moving the sensor farther away, only a few entries are mapped to a large range.

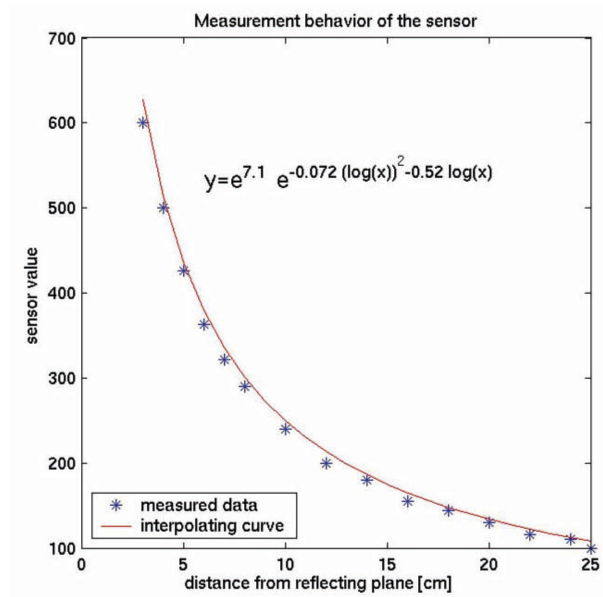

Figure 4. Visualization of the sensor values (measured analog voltage at Smart-Its input port). The measured values (asterisks) and an idealized curve fitted through these is displayed. This value distribution comes close to the distribution in the data sheet of the GP2D120 sensor.

These properties depicted in the Sharp GP2D120 data sheet were verified in different light conditions and with different clothing as surfaces in front of the sensor. Therefore we chose the following mapping between sensor values and the scrolling behavior through data structures 4.

The mapping of sensor values to elements proceeded as follows. We first chose how many entities lie in a given data structure and then distributed these entities as described over the sensor range. We calculated the expected sensor values by inserting the distance from the object in front of the sensor in the function in Figure 5. This function is the connection between the sensor characteristic provided by Sharp and the analog voltages effectively measured by the Smart-Its. We then defined islands around the calculated sensor values in such a manner that in this interval a specific entry is selected.

These islands do not cover the complete spectrum of possible values, there are intervals in which no entry is selected. By this, we provide the user with the perception that the entries are equally spaced on the complete scrollable distance. No selection or change happens if the device is held in a

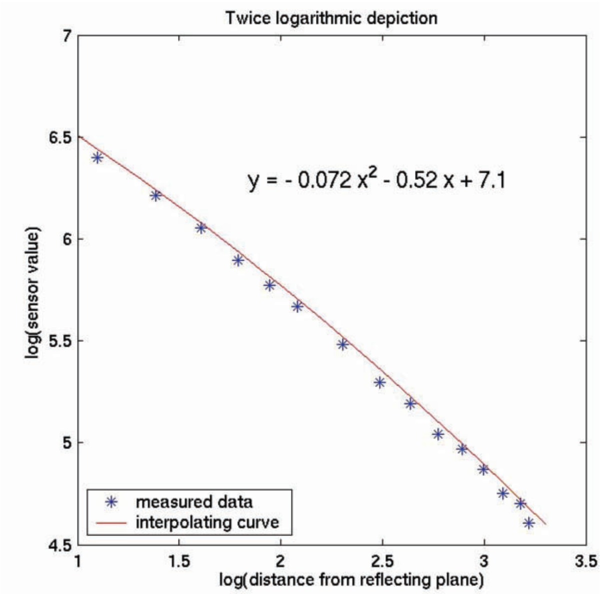

Figure 5. Visualization of the sensor values using logarithmic axis. The measured values (asterisks) nearly perfectly fit the curve.

distance between two of those islands.

4.3 Acceleration Sensor

Our design also comprises the two-axes acceleration sensor ADXL311JE from Analog Devices [1]. The sensor is located on the add-on board. In the current implementation, the sensor is unused. However, the inclusion of such additional sensors allows us to reproduce results published by others. We plan to include the acceleration sensor in the final version of the DistScroll to get information about the orientation of the device in 3D space and exploit this values for context determination. The analog acceleration sensor provides fine-granular resolution.

4.4 Displays

The displays we use for the prototype are Barton BT96040 Chip-on-glass displays. They are connected to the Smart-Its via the I2C-bus.

In our design, we include two displays with a resolution of 40x96 pixels each (5 lines in text mode). Having two displays makes the initial design easier and gives us the flexibility to investigate different scenarios with different screen real estate.

The display contrast can be adjusted using the potentiometer shown to the right of the '2' in Figure 3.

4.5 Buttons

The buttons are for additional input. We anticipated them for selecting data entities such as menu entries and sub menus.

Our initial prototype has three buttons, two of them situated in the middle area of the device on the left side and one button situated near the top on the right side. The layout provides a convenient right-handed usage of the DistScroll (see Figure 1).

We included more buttons to enable us to study what an appropriate number of buttons will be and what layout users will prefer.

5 Usage and Applications

Before discussing potential application areas we briefly explain the implementation of the device.

5.1 Usage

The DistScroll works as follows. It is to be held with one hand. By moving the DistScroll towards oneself, the values of the distance sensor change and are mapped to the current data structure, in our initial study a menu. Moving the DistScroll further in the same direction causes further scrolling through the entries. In Figure 1 you can see a user scrolling through a hierarchical menu by moving the DistScroll towards and away from himself. The menu entries are selected by clicking a specified button, here the top right button which is most conveniently operated with the thumb. We are currently analyzing whether it is more intuitive to move the DistScroll towards oneself to scroll down or to scroll up through the hierarchical data structure.

The prototype currently is to be held with the right hand, the final version of it will be designed for right and left hand use. The restriction to the right hand is introduced by the layout of the push buttons of the device (see Section 4.5).

It is to be noted, that with the current design, there is one button on the right side to be pressed by the thumb and two buttons on the left side to be pressed by one of the fingers.

5.2 Application Areas

In this section we want to present several application areas where the DistScroll can be used, maybe even more effectively or more playfully than other common devices.

We discovered three main application areas that could profit from a distance-based, one handed scrolling / seclecting approach.

The first application domain is using mobile devices when wearing gloves of any kind for security or protection reasons. This comprises arctic and alpine environments in

[9] as well as hazardous environments as can often be found in bio- or chemical laboratories. In general, gloves reduce, dependant on the thickness of the gloves, the tactile sensation of the hand and fingers and make touch and stylus interfaces harder to use. The enlargement of the target areas on the devices for easier access is not always possible or desired and would result in a change of the overall device size.

The second application domain comprises all activities that rely on a second free hand. An example here is stocktaking where one hand counts or scans the items and the second hand operates the mobile device to input data on these items. Another use case could be the comparison of items – the item in one hand and the DistScroll with related information on the item in the other hand, e.g. sorting stuff.

The last application area we think of is games on mobile devices. We think of any sort of character (e.g. aircraft) staying on a fixed position somewhere on the left side of the display. The altitude of the character is controlled by moving the DistScroll. This is done to avoid obstacles or to collect items. The speed of the character could be increased or decreased by pressing defined buttons. Firing bullets or dropping objects can also be simulated using one or more buttons.

One also could think of a DistScroll add-on for mobile devices using the power connector e.g. of mobile phones to augment the device with the ability of using an alternative input technique and thereby potentially extending its usage.

6 Initial User Study

We presented our new interaction technique to several people, students, colleagues and people without direct technical background. We handed them the DistScroll device and observed their interactions. Even when no hints were given, the manner of operation was promptly discovered. Shortly after knowing the relation between menu entry selection and distance, all users were able to nearly errorless use the device.

From this initial feedback we conclude that distance-based scrolling is indeed feasible.

In these first tests, we used the upper display of the DistScroll for data and information portrayal. We simulated a fictive mobile phone menu and used the second display to provide debug information. We later plan to provide the user with information necessary for conducting the user study itself, such as instructions which items are to be searched or selected.

As the device shall equally be usable with the left or right hand, we are currently experimenting with the number and position of the buttons. We currently favor a two button design with the buttons slidable along the sides of the device so the users can easily switch layouts between left and right hand usage. But we also think of a layout with one large button that can easily be pressed independently of which hand is used. A later user study will show which design will prove most useable.

7 Open Issues and Future Work

As distance-based scrolling is a new interaction technique, much research has to be done to prove and verify the desired properties of distance-based scrolling as well of the prototype as well.

During our research, we have identified the following issues that we will address in further work on the DistScroll.

- Is distance-based scrolling faster, equal or slower than other scrolling techniques. So far, we only know that Fitt's Law holds for scrolling([5]).
- Is the scrolling range of 4 to 30 cm appropriate?
- Is it meaningful to use distance scrolling in addition to 'normal' scrolling or exclusively?
- How to scroll long menus? A possible solution could be similiar to the one suggested in [6].
- Is it more intuitive to scroll down towards oneself or away from oneself, especially if large menus could only be accessed in chunks of e.g. 10 entries?

As with all new input devices, the success of the DistScroll depends largely on the acceptance of potential users.

Therefore we plan to do an extensive user study to see whether our expectancies for the general concept are met and to decide upon more detailed questions like number, form and layout of buttons or personalization issues. To further investigate user acceptance and possible applications, we also intend to construct a minimized version of the DistScroll as add-on for a PDA and find answers to the issues rased in Section 7.

8 Conclusions

We have introduced a new interaction technique, the navigation through a hierarchical data structure based on distance as key parameter. We built a fully working prototype and verified the feasibility of our approach and discussed its potential benefits and open issues.

9 Acknowledgement

The work has been conducted in the context of the research project Embedded Interaction ('Eingebettete Interaktion') and was funded by the DFG ('Deutsche Forschungsgemeinschaft').

References

[1] Analog Devices. Analog Devices ADXL311JE data sheet. http://www.analog.com/UploadedFiles/Data_Sheets/39398238692761ADXL311_a.pdf.

[2] J. F. Bartlett. Rock 'n' scroll is here to stay. *IEEE Computer Graphics and Applications*, 20(3):40–45, /2000.

[3] A. Butz, M. Groß;, and A. Krüger. TUISTER: a tangible ui for hierarchical structures. In *Proceedings of the 9th international conference on Intelligent user interface*, pages 223–225. ACM Press, 2004.

[4] H.-W. Gellersen, G. Kortuem, M. Beigl, and A. Schmidt. Physical prototyping with Smart-Its. *IEEE Pervasive Computing Magazine*, 3(3):74–82, July–September 2004.

[5] K. Hinckley, E. Cutrell, S. Bathiche, and T. Muss. Quantitative analysis of scrolling techniques. In *Proceedings of the SIGCHI conference on Human factors in computing systems*, pages 65–72. ACM Press, 2002.

[6] T. Igarashi and K. Hinckley. Speed-dependent automatic zooming for browsing large documents. In *UIST '00: Proceedings of the 13th annual ACM symposium on User interface software and technology*, pages 139–148. ACM Press, 2000.

[7] C. Metzger, M. Anderson, and T. Starner. Freedigiter: A contact-free device for gesture control. In *ISWC '04: Proceedings of the Eighth International Symposium on Wearable Computers (ISWC'04)*, pages 18–21. IEEE Computer Society, 2004.

[8] K. Partridge, S. Chatterjee, V. Sazawal, G. Borriello, and R. Want. 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*, pages 201–204. ACM Press, 2002.

[9] J. Rantanen, J. Impiö;, T. Karinsalo, M. Malmivaara, A. Reho, M. Tasanen, and J. Vanhala. Smart clothing prototype for the arctic environment. *Personal Ubiquitous Comput.*, 6(1):3–16, 2002.

[10] J. Rekimoto. Gesturewrist and gesturepad: Unobtrusive wearable interaction devices, 2001.

[11] V. Sazawal, R. Want, and G. Borriello. The unigesture approach. In *Proceedings of the 4th International Symposium on Mobile Human-Computer Interaction*, pages 256–270. Springer-Verlag, 2002.

[12] A. Schmidt, M. Kranz, and P. Holleis. HCILAB Smart-Its Project. http://www.hcilab.org/projects/smartits/smartits.htm.

[13] Sharp. Sharp GP2D120 data sheet. http://www.farnell.com/datasheets/38197.pdf.

[14] G. M. Smith and m. c. schraefel. The radial scroll tool: scrolling support for stylus- or touch-based document navigation. In *Proceedings of the 17th annual ACM symposium on User interface software and technology*, pages 53–56. ACM Press, 2004.

[15] A. Vardy, J. Robinson, and L.-T. Cheng. The wristcam as input device. In *ISWC*, pages 199–202, 1999.

[16] M. Weiser. The computer for the twenty-first century. (265(3):94?104), Sept. 1991.

[17] D. Wigdor and R. Balakrishnan. Tilttext: using tilt for text input to mobile phones. In *Proceedings of the 16th annual ACM symposium on User interface software and technology*, pages 81–90. ACM Press, 2003.