Multitouch Haptic Interaction

Michael Schmidt and Gerhard Weber

Technische Universität Dresden, Institut für Angewandte Informatik,
Nöthnizer Str. 46, 01602 Dresden, Germany
`{mschmidt,gerhard.weber}@inf.tu-dresden.de`

Abstract. Gestural user interfaces designed for planar touch-sensitive tactile displays require an appropriate concept for teaching gestures and other haptic interaction to blind users. We consider proportions of hands and demonstrate gestures by tactile only methods without the need for Braille skills or verbalization. A user test was performed to confirm blind users may learn gestures autonomously.

Keywords: haptic interaction, assistive technology, gestures.

1 Introduction

Although the use of tactile displays for Braille is not new and common when browsing the web or working with a GUI while using a screen reader, a whole set of new possibilities is open to blind or visually impaired people by the appearance of large pin-matrix devices [1, 2]. Access to contextual and layout information as well as graphical notations may become available.

Unlike speech synthesis is competence in Braille the basis for reading and writing a tactile notation. However, graphical notations such as maths have to be linearized in Braille in order to support reading and writing avoiding drawn fraction bars, for example. As tactile displays convey layout information they can present graphical information such as arrows, scrollbars or window frames [3]. Tactile displays are refreshable and hence even non-verbal information may be expressed through animated tactile or vibrating patterns. Major limitations arise only when graphics are encountered, for which no accessible alternative description has been developed.

The integration of touch-sensitive sensors on a Braille display [4] seems to allow gestural input in the context of haptic interaction, even if problems like the Midas touch effect arise. By combining large pin-matrix devices with touch-sensitiveness, use of more intuitive interaction techniques appears on the horizon. Gestural touch input could improve efficient user communication with the system. The options of pointing and direct manipulation, mnemonic vs. abstract interaction are promising, but for blind users the real advantage is the locality of input. While exploring a tactile display's content through touch, switching to conventional input devices means a loss of focus and therefore extra time and effort for reorientation.

Active tactile interaction takes the direction and temporal structure of movements into account [2]. Feedback from tactile output while moving and feedback from other media ensures perception of information and enhanced navigation. This has led to

C. Stephanidis (Ed.): Universal Access in HCI, Part II, HCII 2009, LNCS 5615, pp. 574–582, 2009.

non-visual multimodal systems such as the Nomad [5], which provide verbal and non-verbal acoustic feedback for touch input on relief placed on a touch- or pen-sensitive surface.

The HyperBraille project[1] aims to allow exploring the screen beyond text and plain widgets. This is enabled by conversion of information to a tactile representation shown on the BrailleDis9000 [2]. Besides information retrieval from screen content [3] and creation of tactile pendants to graphical user interfaces [6], part of this work is to enable the user to control a multimodal system while also reading with the fingers and the hand simultaneously.

In this paper we investigate the implications for designing tactile interaction while the computer may guide the necessary movements. We outline some concepts and problems as well as explicitly target the problem of teaching gestures to blind persons. We present a prototype and our findings from an evaluation.

2 Related Work

Gestures have been proposed for use by blind people for graphical applications with a non-visual user interface, in the domain of mobile devices, and for access to graphical user interfaces with a Braille display. Complexity of non-visual utilization of gestures has increased considerably over time. A blind user of the Nomad device receives auditory feedback after touching some tactile shape [5]. Such audio-haptic interaction techniques create an affordance to tap with a single finger by the type of shape. Many more different gestures may be memorized if the user is guided mechanically. Hill and Grieb show gestures may control a word processor and allow flexible text editing through an audio-haptic user interface [7]. An interactive tactile map of stars on a planar tactile display has shown that loss of overview after gestural input may occur [8]. In this application fingers explore a tactile display sequentially but both hands can be active simultaneously and one hand may be used for gesturing. Gestures change the scale of the map through circling and control the selected region of the sky by forming a caret into the intended direction of panning. After such gestural input users have to restart to orient themselves on the map.

Although touch screens on mobile devices lack tactile output, they can be used in an audio-haptic user interface, if some vibration can be generated while audio is played back. Moving fingers along the edges on a touch screen of a PDA has been evaluated successfully with blind children for navigating geographical information [9]. This application intends to utilize gesturing with off-the-shelf mobile devices for cost-effectiveness. Only simple gestures with mechanical support from the casing can be formed. This study showed that users memorize between one and three strokes in various combinations, if auditory verbal feedback follows completion of gestural strokes. Multi-touch gestures have also been demonstrated [10] to be useful for blind people, if used for audio-haptic interaction on a mobile phone. Single strokes by a single finger, angular strokes by a single finger and a single stroke by two fingers simultaneously have been evaluated. The authors point out the need for error robustness and propose to confirm gestural input by tapping with a second hand.

---

[1] http://www.hyperbraille.com

Another development is based on a Braille display with one line of 80 Braille modules. Gestural input is generated by moving a single finger forward or backward [4] while touching Braille pins. These gestures may be distinguished by their speed of movement. Speech output, for example, is triggered by a speed-up of movement towards the end of the line.

In summary, strokes, combined strokes, strokes from multiple fingers, circles and carets have been used even at differing speed by blind users. But learning additional and possibly multi-touch gestures involving multiple fingers appears to be not achievable without a proper approach for training gestures.

3 Teaching Gestures to the Blind

To compensate some drawbacks of gestural input and get ductile interaction and therefore a high degree of efficiency, a set of gestures should meet several requirements, like being intuitive and memorizable. But intuition is often misleading, if gestures have no deictic nature while at the same time people tend to forget gestures [11]. In terms of usability we need self-explanatory and learnable gestures. As most common gestures are not self-explanatory we propose as a precondition for learnability its teachability to describe its execution. A printed manual's constricted way of communicating information on interface design is the bottleneck of learnability. Through teachability, a gesture becomes graspable from appropriate use of a planar tactile display.

When it comes to utilization of tactile interfaces by blind persons involving scenarios that include gestural input several methods of explaining are possible:

1. Keep gestures simple enough to describe them verbally.
2. A second person demonstrates gestures by guiding the user's hand.
3. Illustrations are provided for each gesture.

For many applications the first method would apply well. For instance, on mobile devices a small invariable set of strokes does fine. For evaluations under lab conditions, as they are done within HyperBraille, the second and third methods are adequate, too. If targeting at an interaction concept on a tactile display capable of substituting mouse input and adding the option of users creating their own gestures for specific tasks, one cannot rely on these methods anymore. Autonomous work with such a system includes the ability to not only initially learn gestures, but also recall self-defined gestures the same way as system defined ones, if needed. A method is required that is capable to serve as fallback in case of missing memorization, needs no second person, would work with gesture sets rather flexibly and, as a side effect, provides an instrument to measure a gesture's teachability.

4 System Evaluation

For a first insight into the possibilities of showing the concept of gesture input for tactile interaction a set of gestures was defined involving single and multiple fingers of the same hand. Our objective was to communicate non-verbal information on several more or less complex spatial gestures performed with tactile only feedback by multiple fingers.

4.1 Participants

Tests were performed by six subjects, three female and three male. Two of them were legally blind, the other four sighted but blind-folded during tests. Due to organizational issues it was not possible for the congenital blind person to go through the whole testing procedure. Nevertheless, we include her results as they give some useful indications.

Table 1. Test persons participating in our tests

Subject	Gender	Degree of Impairment	Handedness
1	female	congenital blind	dextral
2	male	gone blind	sinistral
3	male	blind-folded	sinistral
4	female	blind-folded	dextral
5	male	blind-folded	dextral
6	female	blind-folded	dextral

4.2 Experimental Set-Up

The BrailleDis 9000 [2] is a planar tactile display with 720 touch-sensitive modules arranged in 12 lines per 60 modules. Each module contains 10 pins in 5 rows of 2 pins each, resulting in a matrix of 7200 pins arranged in 60 rows. Touch intensity is measured in 256 steps but for our purposes cutting them down to binary steps by some predefined threshold was sufficient.

Our prototype showed some sensitivity to normal hand perspiration, causing modules reporting further touch input some time after fingers left them. For this reason we covered the planar display by a foil, thin enough to allow proper sensing of pins. While now hands were not easily sliding on that foil, four subjects made usage of magnesium carbonate (liquid chalk). Our modification reasonably improved recognition rates of all gestures.

Application. A gesture guide was developed capable of detecting a comfortably laid down hand on the planar display. The program offers a compact graphical user interface for displaying blobs caused by the hand's touch and ensures random selection of specific tasks. Each task involves guiding the hand to an initial position and, on arrival, displaying gestures in a tactile form. The gesture guide recognizes different fingers as well the size of the hand. It generates dynamically prototypical gestures mostly as static relief pattern. Some gestures included an animated relief pattern built up according to the progress of the user's movement.

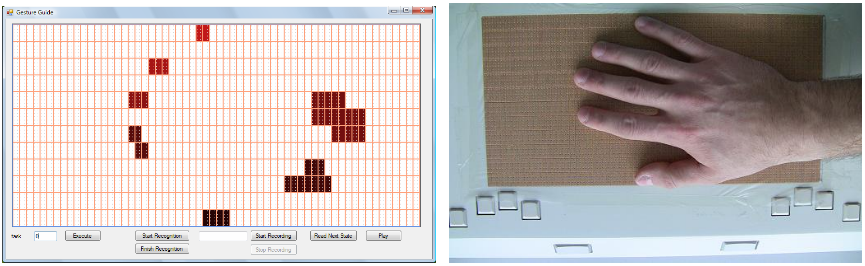
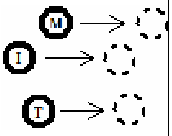
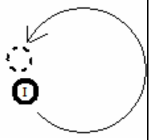
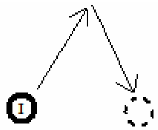
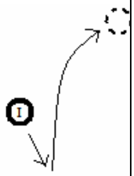
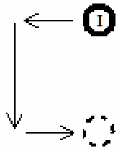


Fig. 1. Gesture Guide's data (left) of a hand on the display (right)

Table 2. Gesture set made of thumb (T), index finger (I) and middle finger (M)

Gesture					
Name	panning	zoom out	zoom in	drag & drop	undo
Abbr.	P	ZO	ZI	DD	U
Gesture	caret 	check 	left squared bracket 	two finger check (T+I)	two finger caret (T+I)
Name	minimize Window	close area	focus	full screen (close areas)	desktop (minimize all windows)
Abbr.	MW	CA	F	FS	D

Additionally the program includes a modified $1 classifier [12] to support multitouch input. This classifier came along with a set of user defined gesture prototypes. The gesture guide fits them to match the display's resolution. Hence in our tests we used gestures as they could occur after a user defined them himself and not as perfect geometric shapes.

Gestures. The provided gestures are all embedded in basic scenarios that could occur while working with a tactile user interface [6]. Our classifier is simple, and improved techniques have been developed elsewhere [13, 14], allowing to assist users' further input and thus may reduce error rates additionally. To support such a system, early cut-down of possible gestures would be of advantage. This can be achieved by incorporating multiple modalities, interaction context (history), and static (fingers to use) or dynamic (locality) features. Diversification of gestures has been applied in this study.

Keeping this in mind we created gesture prototypes to open/close regions/windows, rearrange objects, perform panning, zooming in and out, marking an area and going back in interactions history. Table 2 describes single and parallel finger movements by arrows.

4.3 Procedure

Eeach test consisted of 41 test runs, each made of three phases. In phase 1 subjects were asked to comfortably lay down their preferred hand on the tactile display such that their fingertips and palm touch the surface. The program identified palm as well as each finger and chose fingers needed to draw the gesture with respect to the task. Selecting fingers means elevating all pins except the ones under the chosen finger tips. In other words, fingers involved in a gesture start to move from within a groove formed by lowered pins. In phase 2 under each selected finger the grove is extended and leads to an initial position. Subjects move their selected fingers to the initial position where they are recognized to prepare for phase 3. Phase 3 shows the actual gesture as a path consisting of pins (width is one pin) for every selected finger starting at its initial position while all other pins are lowered. The subject now follows these possibly multiple paths of pins to their end.

The 41 test runs are made up of four sets A, B, C and D. The first five test runs (set A) serve as an introduction, explaining procedures and phases to the subjects. They consist of simple gestures of type DD (straight lines involving different initial points and targets). Additionally information is encoded representing two execution speeds (slow and fast). Each speed is coded in two ways, either by fast or slow blinking pins (alternating every second pin) or by gaps between every two set pins where a gap of one pin indicates fast movement while a gap of two pins indicates slow movement. Repetition during this introduction was possible. Subjects were asked what kind of speed coding they preferred.

The following test runs, taken from sets B, C and D, were presented in randomized order. Set B includes only panning and zooming gestures with variations of speed. Subjects were asked to identify the gesture (which includes number of fingers used) and the speed information sensed in each case. We tested five variations of panning (normal and one of each speed coding) plus three versions (normal plus both blinking variants) of each zooming gesture. Part C contains a selection of partly multi touch, but single stroke gestures (U, MW, CA, F, FS and D, see table 2). Subjects were only asked to describe the gesture's figure along with the number of fingers they were using. If they couldn't describe them for sure (some even recognized the check gesture as a check mark) subjects were asked to draw the gesture three times on the tactile display where it was classified by our recognizer. If at least two out of three times the gesture was classified correctly (and was drawn with the right number of contact points), it was treated as properly learned. To resemble real world conditions, a collection (part D) of 20 DD gestures was included randomly.

All test persons could choose repetitions of sequences if they were unsure about their answers. Every repetition was recorded as a new trial, if it contained all phases. This was due to limitations of the software when canals of the second phase crossed positions of unselected fingers and thus lead to misinterpretations. In these cases the test supervisor repeated the sequence before it reached the third phase and the test person was asked to slightly rotate the hand.

4.4 Results

In the second part of our test procedure (panning and zooming) only two erroneous gestures were recognized as one subject didn't realize he had to use three fingers for two of the panning gestures. Two panning sequences (normal and one pin gap lines) weren't attempted by subject S1 due to a busy schedule. If asked for the preferred speed coding, four out of the six subjects said blinking is the better choice, although one of those subjects made two mistakes in its recognition. Nevertheless two other persons (one preferred blinking) made one mistake in recognition of the encoding by gaps. We conclude gaps seem to be more challenging than blinking to indicate speed of movement. Results of the third part are shown in Figure 2.

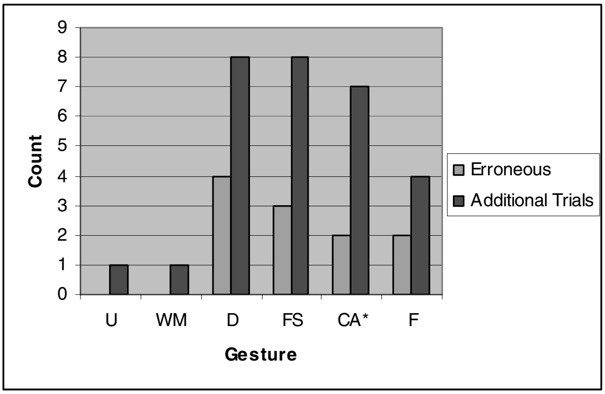


Fig. 2. Number of trials and errors in performing gestures. *Note that only five subjects performed Close Area (CA).

There were 35 single tests at an average of 1.8 trials and 0.3 errors for each gesture per subject. As Figure 2 shows by the number of additionally needed trials, our two finger single stroke gestures were the most challenging, followed by the "close area" gesture due to its bended form. Approximately one third of the total number of single tests was not properly recognized but that number is to be modified if inspected closer. For instance, there were four errors for two types of the check gestures (CA, FS) that were classified as a “v” with the correct number of fingers used. Interestingly the double caret gesture was described like an arrow. Anyhow, it challenged some subjects as they were misled to twist their two fingers around each other for input. Changes in palm orientation and use of multiple fingers indicate a lower teachability of such gesture types.

5 Conclusions and Future Work

As presented, the work could only scrape the surface of problems concerning non-visual haptic interaction on planar tactile displays. Further investigation is to be done to enable blind users to define their own gestures. Nevertheless we showed gestures can be learned from tactile only feedback.

Our method suggests learning gestures in a way which requires no knowledge of Braille. This has several advantages like possibly addressing deaf-blind users or Braille-illiterate people. More parameters could be included such as width of reliefs, tactons and Braille. But it has its drawbacks, too, if gestures are difficult to teach due to crossing one’s own path. An audio-haptic approach may create audio feedback on automatic identifiable features like the number of fingers used or execution speed should confirm the user’s own recognition. In addition to this an audio-haptic approach may resolve issues where the user is not capable of touching with his hand due use of a pen or grip [15].

Furthermore, the software could not convey that the user could decide which fingers to use and how large the amount of variation is that would be allowed. The decision how to input a specified gesture is not taken by the system. In fact, the user should find an ergonomic way to induce the touch input that is necessary for a specific gesture to work. Even at the size of an A4 sheet the tactile display’s size is limited. It is not always possible to guide the user to the most comfortable initial position. A slightly twisted hand or thumb complicates proper sensing of multiple lines and, for instance, made following the zooming gestures somewhat fiddly.

The presented work did not include self-crossing gestures like the single stroke x. With reliable touch-sensitiveness on elevated pins amending the software to dynamically draw the guiding lines should be easily possible to a certain degree. Of course, this approach comes to its limits, too, at more complex gestures, but would offer the option of teaching writing or sketching with fingers. Far more challenging would be teaching of multistroke and compound gestures by the available instruments.

Finally a very important question is, whether it would be possible to allow the second hand touching and reading, too, while the first hand is monitored. This may become the future way of avoiding the Midas touch effect while allowing gesture input.

Acknowledgements. We kindfully thank all subjects for participating and supporting us with valuable hints. The HyperBraille project is sponsored by the Bundesministerium für Wirtschaft und Technologie (German Ministry of Economy and Technology) under the grant number 01MT07004 (for Technische Universität Dresden). Only the authors of this paper are responsible for its content.

References

1. Rotard, M., Bosse, K., Schweikhardt, W., Ertl, T.: Access to Mathematical Expressions in MathML for the Blind. In: Proc. of the HCI International Conference, pp. 1325–1329 (2004)
2. Völkel, T., Weber, G., Baumann, U.: Tactile Graphics Revised: The Novel BrailleDis 9000 Pin-Matrix Device with Multitouch Input. In: Miesenberger, K., Klaus, J., Zagler, W.L., Karshmer, A.I. (eds.) ICCHP 2008. LNCS, vol. 5105, pp. 835–842. Springer, Heidelberg (2008)
3. Kraus, M., Völkel, T., Weber, G.: An Off-Screen Model for Tactile Graphical User Interfaces. In: Miesenberger, K., Klaus, J., Zagler, W.L., Karshmer, A.I. (eds.) ICCHP 2008. LNCS, vol. 5105, pp. 865–872. Springer, Heidelberg (2008)
4. Kipke, S.: Sensitive Braille Displays with ATC Technology (Active Tactile Control) as a Tool for Learning Braille. In: Miesenberger, K., et al. (eds.) ICCHP 2008. LNCS, vol. 5105, pp. 843–850. Springer, Heidelberg (2008)
5. Parkes, D.: Nomad, an audio tactile tool for the acquisition, use and management of spatially distributed information by visually impaired people. In: Proceedings of the Second International Symposium on Maps and Graphics for Visually Handicapped People, pp. 24–29. A.F.&Dodds, London (1988)
6. Schiewe, M., Köhlmann, W., Nadig, O., Gerhard Weber, G.: What You Feel is What You Get: Mapping GUIs on Planar Tactile Displays. In: Stephanidis, C. (ed.) Universal Access in HCI, Part II, HCII 2009. LNCS, vol. 5615, pp. 564–573. Springer, Heidelberg (2009)
7. Hill, D.R., Grieb, C.: Substitution for a restricted visual channel in multimodal computer-human dialogue. IEEE Transactions on Systems, Man and Cybernetics 18(2), 285–304 (1988)
8. Weber, G.: Adapting direct manipulation for blind users. In: Ashlund, S., et al. (eds.) Adjunct Proceedings of INTERCHI 1993, pp. 21–22. Addison Wesley, Reading (1993)
9. Sánchez, J., Maureira, E.: Subway Mobility Assistance Tools for Blind Users. In: Stephanidis, C., Pieper, M. (eds.) ERCIM Ws UI4ALL 2006. LNCS, vol. 4397, pp. 386–404. Springer, Heidelberg (2007)
10. Kane, S.K., Bigham, J.P., Jacob, O., Wobbrock, J.O.: Slide Rule: Making Mobile Touch Screens Accessible to Blind People Using Multi-Touch Interaction Techniques. In: Proceedings of the 10th international ACM SIGACCESS Conference on Computers and Accessibility, Halifax, Nova Scotia, Canada, October 13 - 15, 2008, pp. 73–80. ASSETS. ACM, New York (2008)
11. Wolf, C.G., Morrel-Samuels, P.: The Use of Hand-Drawn Gestures for Text Editing. International Journal of Man-Machine Studies 27(1), 91–102 (1987)
12. Wobbrock, J.O., Wilson, A.D., Li, Y.: Gestures without libraries, toolkits or training: a $1 recognizer for user interface prototypes. In: Proceedings of the 20th Annual ACM Symposium on User interface Software and Technology, UIST 2007, Newport, Rhode Island, USA, October 07-10, 2007, pp. 159–168. ACM, New York (2007)
13. Rubine, D.: Specifying gestures by example. SIGGRAPH Comput. 25(4), 329–337 (1991)
14. Bau, O., Mackay, W.E.: OctoPocus: a dynamic guide for learning gesture-based command sets. In: Proceedings of the 21st Annual ACM Symposium on User interface Software and Technology, UIST 2008, Monterey, CA, USA, October 19 - 22, 2008, pp. 37–46. ACM, New York (2008)
15. Brewster, S., Brown, L.M.: Tactons: structured tactile messages for non-visual information display. In: Cockburn, A. (ed.) Proceedings of the Fifth Conference on Australasian User interface - Volume 28, Dunedin, New Zealand. ACM International Conference Proceeding Series, vol. 53. Australian Computer Society, Darlinghurst