15

Touchscreen Software Keyboard for Finger Typing

Kentaro Go[1] and Yuki Endo[2]
[1]Interdisciplinary Graduate School of Medicine and Engineering, University of Yamanashi
[2]Yamanashi R&D Center, HAL Laboratory, Inc.
Japan

1. Introduction

Touchscreen interfaces have attracted attention in recent years because their flexibility, functionality, and usability for design have become better known. They have been used in public-use terminals such as automated teller machines, ticket machines, and information kiosks. Recently, they are used for small, mobile device interfaces such as Apple's iPhone and iPod touch (Apple Inc., 2006). Furthermore, they are employed as large tabletop interfaces such as Microsoft Surface (Microsoft Corp., 2007) and MERL DiamondTouch (Dietz & Leigh, 2001).

For touchscreen interfaces, touchscreen keyboards are a fundamental function because text entry is a primary and frequently used task in several applications. Physical screen size affects the touchscreen keyboard design. A keyboard might include a different type of keyboard, display fewer keys on a small screen, or enable selection of an appropriate typing device such as a finger or stylus. Fundamentally, the touchscreen keyboard is created by software. It therefore has flexibility in visual presentation and software-based technique.

In this chapter, we specifically examine touchscreen software keyboards for finger typing. We first examine criteria for analyzing and discussing touchscreen keyboards. Then the chapter presents a description of our ongoing efforts at designing a Customizable and Adaptable Touchscreen Keyboard with bubble cursor-like visual feedback (CATKey). We are particularly interested in addressing the needs of a wide range of user classes by improving its perceived usability without decreasing the text entry rate. Many touchscreen systems are targeted at the public-use market. For that reason, users' skills and experiences are expected to vary widely. Users would refuse to use a touchscreen keyboard if the text entry method were to provide low perceived usability.

2. Touchscreen Keyboard for Finger Typing

A touchscreen keyboard is a virtual keyboard: it displays a keyboard image on a computer display and is operated mainly with a finger or stylus. It is intended to be used in place of a physical keyboard. The keyboard image is created and controlled by software and displayed on a screen. Therefore, the touchscreen keyboard is categorizable as a soft keyboard (software keyboard) or onscreen keyboard. Although several alternative approaches for onscreen-based methods have been used for text entry such as handwriting recognition-based approaches and alphabetic gesture-based approaches (e.g., Graffiti, Unistrokes), we specifically examine display of a keyboard image onscreen and typing with a finger or stylus because of the user's familiarity with physical keyboards. Figure 1 shows our criteria for classifying touchscreen keyboards for text entry. Listing the criteria, we specifically examine important factors related to text entry with touchscreen keyboards.

Screen size
- Small (Mobile and smart phones, Personal Digital Assistant, Handheld computer)
- Medium (Standard PC, Tablet PC)
- Large (Tabletop, Wall-sized display, projector)

Touchscreen keyboard types
- Soft keyboard
- Gesture-based keyboard (menu-based)

Number of keys
- Alphabetical (selection keyboard)
- Numerical (10-key numeric keypad)

Typing device
- Finger typing
- Stylus typing
- Hybrid use (Finger + Stylus)

Technique
- Prediction of input word
- Customization of keyboard layouts
- Adaptation
- Visual appearance (size, shape, color, visual feedback, etc.)

Fig. 1. Criteria for classifying touchscreen keyboards for text entry

2.1 Screen Size

The touchscreen device size and the touchscreen's effective area affect the touchscreen keyboard design. The device sizes are categorizable into three groups: small, medium, and large. Small touchscreen devices include mobile and smart phones, Personal Digital Assistants (PDAs), and handheld computers. Their effective touchscreen area is small; hence onscreen objects become small. Therefore, users are assumed to use a stylus for manipulating them. The finger use of small touchscreen devices has become more popular in the research community since Apple's iPhone and iPod touch were released.

Medium-size touchscreen devices include standard PCs and tablet PCs.

Finally, large touchscreen devices contain table-top displays, wall-sized displays and projectors. Recently, researchers started to examine text entry specifically for tabletop displays (Hinrichs et al., 2007). They analyzed properties of tabletop displays and summarized existing text entry methods for tabletop use.

2.2 Touchscreen Keyboard Types

Touchscreen keyboards of two types have been investigated: soft keyboards and gesture-based keyboards. Soft keyboards can have various keyboard layouts. The QWERTY layout is the standard for soft keyboards, but an alphabetical layout is used as a selection keyboard in some cases. These two layouts are suitable for walk-up-and-use scenarios (Hinrichs et al., 2007). For example, with a public terminal, people might come to use it briefly then leave. Consequently, we cannot assume their prior knowledge and experience of sophisticated tools and methods. Users can transfer their knowledge related to alphabetical order if we use a selection keyboard with alphabetical order for a touchscreen keyboard on a public terminal. Users can transfer their skills and experience gained using common hardware keyboards if we were to use QWERTY layouts for touchscreen keyboards (MacKenzie et al., 1995). Notwithstanding, previous studies have shown that the text entry rate for a touchscreen keyboard is considerably lower than that for a hardware QWERTY keyboard, even for expert typists (Shneiderman, 1991; Sears et al., 1993).

Although public terminals usually use these layouts for touchscreen keyboards, private devices might implement any keyboard layout. Because the primary emphasis of text entry is its performance, several efforts have been conducted to find a keyboard layout with higher performance for text entry: the Dvorak layout, the Metropolis layout (Zhai, Hunter & Smith, 2000), and so forth.

On the other hand, with a gesture-based keyboard, the user inputs a gesture, drawing a line without lifting up the finger or stylus. Gesture-based approaches might display specialized menus (Venolia & Neiberg, 1994; Guimbretiere & Winograd, 2000), provide a dedicated input area (Perlin, 1998; Mankoff & Abowd, 1998), or a dedicated soft keyboard for gesture input (Zhai & Kristensson, 2003). Finally, Masui (1998) provides combined usage of a soft keyboard and a gesture-based keyboard.

2.3 Number of Keys

Onscreen keyboards present several advantages over physical keyboards. The main advantage is their flexibility to keyboard design. For example, keyboard layouts, even the number of keys, can be modified easily and tested for actual use. Two typical cases for the number of keys include alphabetical (full-size) keyboards such as the 101 keyboard for standard PCs and the numerical 10-key pad for mobile phones.

MacKenzie and Tanaka-Ishii (2007b) described the number of keys for text entry from the concept of a *key-ambiguity continuum*. In their concept, if each symbol is assigned to a dedicated key, it has no ambiguity. If some symbols are assigned to a key, such as a key with upper and lower case letters, it creates ambiguity. The ambiguity increases if we assign more symbols to a key. A typical example is a 10-key pad with the assignment of English letters: three or four letters are assigned to each key. In general, the fewer keys we provide, the greater the ambiguity, given set of letters.

A technique to resolve the ambiguity must be used when we design a keyboard with ambiguity. Two approaches exist: multi-tap method and prediction-based method. Using the multi-tap method, the user presses a key one or more times to specify a desired letter from a cyclic sequence of letters. The multi-tap method is a standard for numeric 10-key pad text entry. Using a prediction-based method, the user presses a series of keys; then, the system creates a list of possible words from the combinations of assigned letters to the keys using a dictionary. A typical example of this method is T9 (Tegic Communications).

2.4 Typing Device

Although a soft keyboard might be used with a mouse or other pointing device, a touchscreen keyboard is typed in mainly with a finger or stylus. For large keys or buttons on screen, finger typing is a reasonable approach. Sears et al. (1993) investigated the effect of touchscreen keyboard size on text entry speed. Results showed that expert typists achieve 32.5 wpm using the largest keyboard (24.6 cm wide) out of four levels of keyboard size condition. The narrower the keyboard, the fewer words typed per minute. Nevertheless, expert typists achieved 21.1 wpm using the smallest keyboard: 6.8 cm wide.

For small screen devices such as mobile and smart phones, PDAs, and handheld computers, stylus typing is used. However, in some daily situations, people feel that taking out a stylus is cumbersome, especially in brief use. Vogel & Baudisch (2007) proposed a technique for operating stylus-based interfaces with touch: a hybrid approach. It displays an offset call-out when the finger occludes small dense targets; the user operates the offset cursor in the call-out area to select a target. Tsurumi et al. (2008) applied this concept for a touchscreen keyboard to implement a hybrid approach.

2.5 Technique

As noted already, the touchscreen keyboards are created and displayed onscreen with software. For that reason, they have flexibility of design and functionality as opposed to hardware keyboards. A gesture-based keyboard described in Section 2.2 is a typical approach for software-based dynamic control. A menu can be created and displayed easily on a touchscreen. It is easy to record a sequence of typed characters, predicting subsequent characters based on a dictionary, and showing a menu of candidate words.

A customization function can be realized easily when we design a touchscreen keyboard. For example, its appearance––the size, shape, color, font, and borderline––can be set. Visual feedback when the user types can be modified if we design the software as it is. In addition, the keyboard layout itself can be modified to fit the size and shape of the keyboard to the user's hands. The concept of adaptation is realized if we frequently customize the keyboard layout. Himberg et al. (2003) proposed online personalization of the keyboard layout.

3. CATKey: Customizable and Adaptable Touchscreen Keyboard using Bubble Cursor-Like Visual Feedback

3.1 Motivation

Text entry tasks undertaken using touchscreen keyboards are cumbersome because (1) the user cannot touch, feel, or distinguish the positions of the keys displayed on the screen; and (2) it is difficult to feel the visual and tactile feedback from the keys, especially with conventional touchscreen devices, even though tactile displays have developed rapidly (Benali-Khoudja et al., 2004). However, as described in the previous section, touchscreen keyboards present practical advantages because (1) they can be designed as customizable and adaptable keyboards; and (2) they can be designed to display the visual feedback of they keys and on the keys.

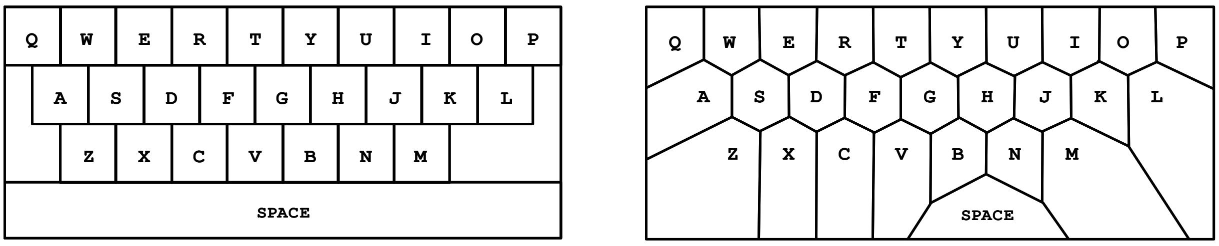
Our approach to designing touchscreen keyboards is to address the weak points using customizable and adaptable functions and effective visual feedback to improve their perceived usability. We designate the method as Customizable and Adaptable Touchscreen Keyboard with Bubble Cursor-Like Visual Feedback: CATKey (Go & Endo, 2007).

3.2 Customizable Key Layout

Figure 2 portrays the concept of Voronoi key area of our keyboard design. Figure 2(a) depicts a typical QWERTY layout keyboard, consisting of rectangular keys. Figure 2(b) is the initial key layout of our CATKey. Each key has a maximum area; consequently, the overall key area forms a Voronoi diagram (Aurenhammer, 1991). Himberg et al. (2003) employ this approach for an onscreen numeric 10-key pad. In the keyboard, each key consists of a set of points.

$$V_i = \{ x \mid d(x, c_i) < d(x, c_j), \text{ for all } j \neq i \} \tag{1}$$

In that equation, x is the coordinate of a point, d is the Euclidean distance, and c signifies the coordinate of the key centroid.



(a) QWERTY layout of rectangular keys

(b) QWERTY layout of keys with Voronoi area

Fig. 2. The Voronoi key area concept

Figure 3 presents a key layout customized by a user in the customization mode of CATKey. In this mode, the user adjusts each key by touching and manipulating it directly so that the user feels that the layout is easy to use. The customized key area also forms a Voronoi diagram.

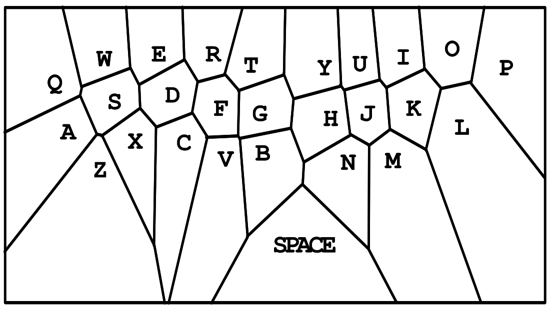
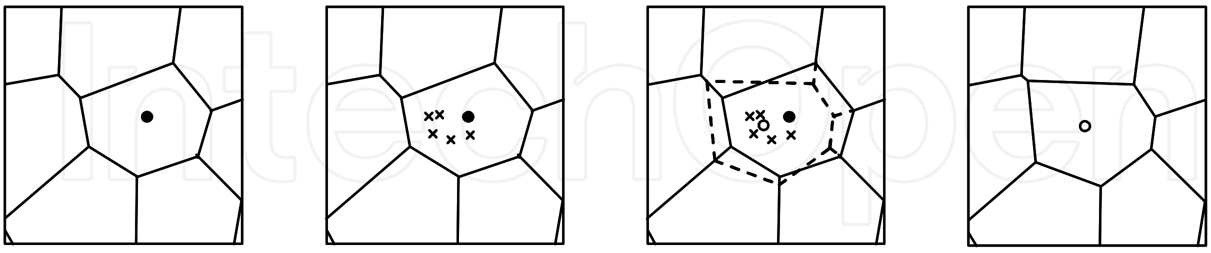


Fig. 3. A key layout customized by a user

3.3 Adaptable Keys

We designed an adaptable function similar to that presented by Himberg et al. (2003). In the adaptive mode of CATKey, the key centroid is moved to the centroid of recorded keystroke points in each key (Fig. 4). The keystroke coordinates are recorded for the period of the pre-specified number of keystrokes.



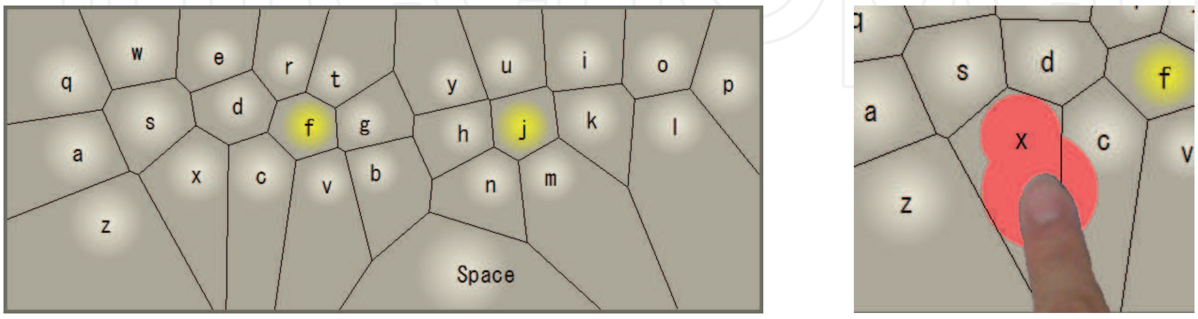
(a) Original key area (b) User's typing points (X-mark) (c) Recalculated centroid of the typing points (d) Adjusted key area

Fig. 4. The key adaptation mechanism of CATKey: Black point, the original centroid. X-mark: keystroke points; White circle, the centroid of the keystroke points and the weighted original centroid; Dotted line, recalculated key area.

3.4 Visual Feedback

Figure 5 depicts the design of visual feedback in CATKey. Figure 5(a) portrays an actual view of CATKey. We assigned a round key area to communicate the touched location to the user. The yellow f and j keys show home positions of the index fingers. The presence of the line and color of keys can be modified using the parameter-setting dialogue.

Figure 5(b) shows the bubble cursor-like visual feedback (Grossman & Balakrishnan, 2005). The bubble-shaped cursor in red adjusts dynamically to the closest key and provides the sensation of an offset cursor (Sears & Shneiderman, 1991). Figure 6 presents the concept of bubble cursor-like visual feedback used in CATKey. Figure 6(a) shows the Voronoi area of a key that has a colored circle with the centroid as its middle. It redraws the circle in a different color as visual feedback when the user touches inside the color circle (Fig. 6(b)). When the user touches outside the color circle, it redraws the circle in a different color, draws a circle with its keystroke point as its middle, and merges them (Fig. 6(c)). The radius of the second circle is the distance from user's keystroke point to the color circle.



(a) Overview of CATKey with lines and color circles (b) Bubble cursor-like visual feedback

Fig. 5. Design of visual feedback in CATKey

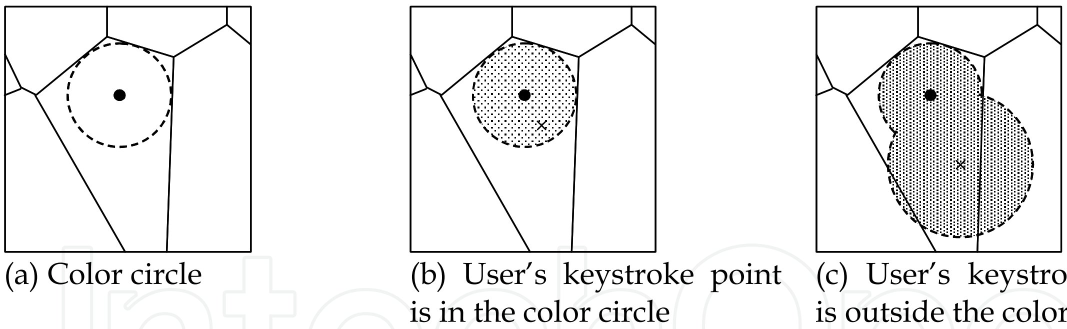

(a) Color circle (b) User's keystroke point is in the color circle (c) User's keystroke point is outside the color circle

Fig. 6. Concept of bubble cursor-like visual feedback: Black point, the centroid of the key; Dotted line, the inscribed circle of the key; X-mark, keystroke point.

4. Evaluation of CATKey

We evaluated the usability of CATKey based on ISO 9241-4 (1998). We measured the text entry speed and error rate using a hardware QWERTY keyboard, a software QWERTY keyboard, and CATKey, and assessed its comfort; finally, we conducted a debriefing interview.

Ten college students (two women and eight men) participated in the evaluation. Each uses a hardware QWERTY keyboard regularly, but seldom uses a touchscreen keyboard for daily activities.

A CAD Center NEXTRAX 15 touchscreen computer was used for evaluation. It has a 15-inch TFT monitor with 1024 × 768-pixel resolution. Also, CATKey was implemented using C# (Visual Studio.Net 2003; Microsoft Corp.). Figure 7 shows a user typing with NEXTRAX 15, which has a 20-deg mounting angle. For hardware keyboards, standard QWERTY keyboards (Dell Computer Corp.) were used. The sentences were chosen randomly from among pangrams such as "a quick brown fox jumps over the lazy dog," "how quickly daft jumping zebras vex," and "pack my box with five dozen liquor jugs."

Fig. 7. Experimental setting (NEXTRAX 15).

Each experimental session included 10 sentences for the typing task. Each participant performed two practice sessions, then completed four sessions for measurement. The presentation order of keyboard types was counterbalanced.

Results indicate that the respective mean entry speeds of the hardware QWERTY keyboard, software QWERTY keyboard, and CATKey were 33.5, 23.8, and 22.8 wpm. A significant difference was found among the keyboards ($F(2, 18) = 7.5$, $p<0.05$); but no significant difference was found between the levels of the software QWERTY keyboard and CATKey. Similarly, the respective mean error rates of the hardware QWERTY keyboard, software QWERTY keyboard, and CATKey were 3.4, 5.9, and 6.8%. A significant difference was found among the keyboards ($F(2, 18) = 6.3$ ($p<0.05$)); but no significant difference was found between the levels of software QWERTY keyboard and CATKey. Furthermore, no significant difference was found from the assessment of comfort between the software QWERTY keyboard and CATKey.

5. Conclusion

In this chapter, we examined the criteria of touchscreen keyboard text entry. We specifically emphasized the use of a touchscreen keyboard use for finger typing. We analyzed it from five aspects: screen size, touchscreen keyboard types, number of keys, typing device, and technique.

Additionally, we described the CATKey design: it is a customizable and adaptable touchscreen keyboard with the bubble cursor-like visual feedback. Our criteria suggest that CATKey is better used for medium or larger touchscreen devices. Incorporating a soft keyboard approach, it has alphabetical keys. It specifically examines finger typing, but is useful for a stylus also. Although it has no prediction function, its keyboard layout can be customized. It has an adaptation function. Finally, its visual appearance has a bubble cursor-like visual feedback.

Early results of evaluation show that the mean text entry speed and mean input error rate of CATKey are not significantly different from those of a standard software QWERTY keyboard. This result might indicate limitations of our design: users need a compelling reason to replace their text entry device from a static standard QWERTY keyboard from a performance viewpoint. Notwithstanding, during the debriefing interview, the participants expressed their preference for the design and usability of CATKey.

6. References

Apple, Inc. (2007). iPhone and iPod touch, http://www.apple.com/ (Last accessed: May 19, 2008)

Aurenhammer, F. (1991). Voronoi diagrams--a survey of a fundamental geometric data structure, *ACM Computing Surveys*, Vol. 23, No. 3, pp. 345-405.

Benali-Khoudja, M.; Hafez, M.; Alexandre, J.M. & Kheddar, A. (2004). Tactile Interfaces: A State-of-the-art Survey, *Proceedings of 35th International Symposium on Robotics*, pp. 721-726, Paris, France, March 2004.

Dietz, P. H. & Leigh, D. L. (2001). DiamondTouch: A Multi-User Touch Technology, *Proceedings of the ACM Symposium on User Interface Software and Technology (UIST 2001)*, pp. 219-226.

Go, K. & Endo, Y. (2007). CATKey: Customizable and Adaptable Touchscreen Keyboard with Bubble Cursor-like Visual Feedback, *C. Baranauskas et al. (Eds.): INTERACT 2007, LNCS 4662*, Part I, pp. 493-496.

Grossman, T. & Balakrishnan, R. (2005). The Bubble Cursor: Enhancing target acquisition by dynamic resizing of the cursor's activation area, *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems 2005*, pp. 281-290.

Guimbretiere, F. & Winograd, T. (2000). FlowMenu: Combining command, text, and data entry, *Proceedings of the ACM Symposium on User Interface Software and Technology (UIST 2000)*, pp. 213-216

Himberg, J.; Hakkila, J.; Kangas, P. & Mantyjarvi, J. (2003). On-line personalization of a touch screen based keyboard, *Proceedings of the 8th International Conference on Intelligent User Interfaces (UIST 03)*, pp. 77-84.

Hinrichs, U.; Hancock, M.; Collins, C. & Carpendale, S. (2007). Examination of Text-Entry Methods for Tabletop Displays, *Proceedings of the Second Annual IEEE International Workshop on Horizontal Interactive Human-Computer Systems (TABLETOP 2007)*, Oct 2007, pp. 105-112.

ISO9241-4 (1998). *Ergonomic requirements for office work with visual display terminals (VDTs), Part 4: Keyboard requirements*.

Isokoski, P. (2004). Performance of menu-augmented soft keyboards, *Proceedings of the 2004 Conference on Human Factors in Computing Systems (CHI 2004)*, Vienna, Austria, April 2004, pp. 423-430.

MacKenzie, I. S. & Tanaka-Ishii, K. (2007a). *Text Entry Systems: Mobility, Accessibility, Usability*, Morgan Kaufmann.

MacKenzie, I. S. & Tanaka-Ishii, K. (2007b). Text entry using a small number of buttons, in *Text Entry Systems: Mobility, Accessibility, Usability*, Morgan Kaufmann, pp. 105-121.

MacKenzie, I. S. & Zhang, S. X. (1999) The design and evaluation of a high-performance soft keyboard, *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems: the CHI is the limit*, May 1999, Pittsburgh, Pennsylvania, United States, pp. 25-31.

MacKenzie, I. S.; Zhang, S. X. & Soukoreff, R. W. (1995). Text entry using soft keyboards. *Behaviour and Information Technology*, Vol. 18, pp. 235-244.

Mankoff, J. & Abowd, G. D. (1998). Cirrin: A word-level unistroke keyboard for pen input, *Proceedings of the ACM Symposium on User Interface Software and Technology (UIST98)*, pp. 213-214.

Masui, T. (1998). Integrating pen operations for composition by example, *Proceedings of the ACM Symposium on User Interface Software and Technology (UIST98)*, pp. 211-212.

Microsoft Corporation (2007) Microsoft Surface, http://www.microsoft.com/surface/ (Last accessed: May 19, 2008)

Perlin. K. (1998). Quikwriting: Continuous stylus-based text entry, *Proceedings of the ACM Symposium on User Interface Software and Technology (UIST98)*, pp. 215-216.

Shneiderman, B. (1991) Touch Screens Now Offer Compelling Uses, *IEEE Software*, March, pp. 93-94.

Sears, A.; Revis, D.; Swatski, J.; Crittenden, R. & Shneiderman, B. (1993). Investigating touchscreen typing: The effect of keyboard size on typing speed. *Behaviour and Information Technology*, Vol. 12, No. 1, pp. 17-22.

Sears, A. & Shneiderman, B. (1991). High-precision touchscreens: design strategies and comparisons with a mouse. *International Journal of Man-Machine Studies*, Vol. 34, No. 4, pp. 593-613.

T9, Tegic Communications, http://www.t9.com (Last accessed: May 19, 2008).

Tsurumi, L.; Watanabe, R.; Matsuura, Y. & Go, K. (2008) Dynamic assignment of symbols to keys for soft keyboard, *The 70th Annual IPSJ Convention (4)*, 5ZC-6, pp. 177-178 (in Japanese).

Venolia, D. & Neiberg, F. (1994). T-Cube: A fast, self-disclosing pen-based alphabet, *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI 94)*, pp. 265-270.

Vogel, D. & Baudisch, P. (2007). Shift: A Technique for Operating Pen-Based Interfaces Using Touch. *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI 2007)*, pp. 657-666.

Zhai, S.; Hunter, M. & Smith, B.A. (2000). The Metropolis Keyboard: An Exploration of Quantitative Techniques for Virtual Keyboard Design, *Proceedings of the 5th International Conference on Intelligent User Interfaces (UIST 00)*, pp. 119-128.

Zhai, S.; Hunter, M. & Smith, B.A. (2002). Performance Optimization of Virtual Keyboards, *Human-Computer Interaction*, Vol. 17, Nos. 2–3, pp. 229-269.

Zhai, S. & Kristensson, P-O. (2003). Shorthand Writing on Stylus Keyboard, *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI 2003)*, pp. 97-104.

Human Computer Interaction: New Developments
Edited by Kikuo Asai

ISBN 978-953-7619-14-5
Hard cover, 382 pages
Publisher InTech

The book consists of 20 chapters, each addressing a certain aspect of human-computer interaction. Each chapter gives the reader background information on a subject and proposes an original solution. This should serve as a valuable tool for professionals in this interdisciplinary field. Hopefully, readers will contribute their own discoveries and improvements, innovative ideas and concepts, as well as novel applications and business models related to the field of human-computer interaction. It is our wish that the reader consider not only what our authors have written and the experimentation they have described, but also the examples they have set.

Kentaro Go and Yuki Endo (2008). Touchscreen Software Keyboard for Finger Typing, Human Computer Interaction: New Developments, Kikuo Asai (Ed.), ISBN: 978-953-7619-14-5, InTech, Available from: http://www.intechopen.com/books/human_computer_interaction_new_developments/touchscreen_software_keyboard_for_finger_typing

INTECH
open science | open minds

InTech Europe
University Campus STeP Ri
Slavka Krautzeka 83/A
51000 Rijeka, Croatia
Phone: +385 (51) 770 447
Fax: +385 (51) 686 166
www.intechopen.com

InTech China
Unit 405, Office Block, Hotel Equatorial Shanghai
No.65, Yan An Road (West), Shanghai, 200040, China
中国上海市延安西路65号上海国际贵都大饭店办公楼405单元
Phone: +86-21-62489820
Fax: +86-21-62489821