FingerSense - Augmenting Expressiveness to Physical Pushing Button by Fingertip Identification

Jingtao Wang, John Canny
Computer Science Division
University of California at Berkeley, Berkeley, CA 94720-1776, USA
{jingtaow, jfc}@cs.berkeley.edu

ABSTRACT
In this paper, we propose a novel method, *FingerSense* to enhance the expressiveness of physical buttons. In a *FingerSense* enabled input device, a pressing action is differentiated according to the finger involved. We modeled the human performance of *FingerSense* interfaces and derived related parameters from a preliminary usability study. Overall findings indicate that *FingerSense* is faster compared with traditional keypads when the finger switching action could be paralleled.

Categories & Subject Descriptors: H5.2 [**Information interfaces and presentation**]: Input devices and strategies, Theory and methods

General Terms: Design, Human Factors.

Keywords: Text input, input device, mobile computing, fingerprint recognition, performance modeling.

INTRODUCTION
Tapping physical buttons is one of the most frequent tasks in computer-human interaction. In a button-based input device, e.g. the QWERTY keyboard, 1/2/3-button mouse or the telephone keypad, the user's fingers act as triggers for executing commands. Although alternative input modalities such as speech and handwriting are available, button-based interfaces, especially the keyboard, are still the most widely used input device.

The emergence of handheld, cell phone and other forms of mobile computing devices, however, present unique challenges to traditional button interfaces - due to the size of human fingers and the corresponding motor control accuracy, buttons can not be made too small. It becomes increasingly difficult for a full QWERTY keyboard to fit into the ever smaller mobile devices.

In this paper, we propose an alternative method, *FingerSense*, to improve the expressiveness of pushing buttons without the cost of minimizing the button size or adding additional key strokes[1]. In a *FingerSense* enabled input device, a button will respond differently when it is pressed by different fingers. As illustrated in figure 1, when the thumb finger taps the given button, the action can be interpreted as event A. If index finger is used, the system will interpret this action as event B, similarly the middle finger will correspond to event C, etc. As a result, a single pressing action could generate as many events as the number of user's fingers. We define *FingerSense* as the method of multiplexing a physical button according to the actual finger selected in taping, despite the underlining sensing/recognition technology used to distinguish fingers.

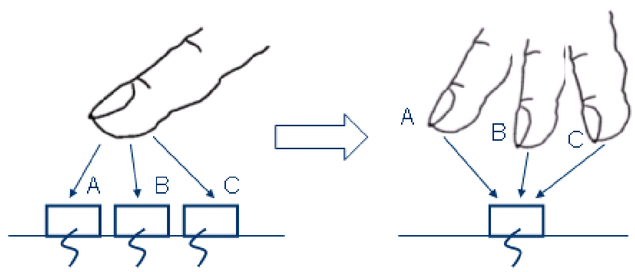

Figure 1. From classic buttons to *FingerSense* button

To verify the effectiveness of *FingerSense*, we investigate the follow three questions in this paper: 1) Is *FingerSense* technologically feasible? i.e. is it possible to classify the finger tapping at a button in real time and in a cost effective manner? 2) To use *FingerSense*, the user must select and switch to the correct finger before tapping the intended button; is this procedure a cognitive workload too high to be adopted by most of the users? 3) Is there any speed advantage for the *FingerSense* enabled text input when it is compared with the state-of-the-art?

In the next section, we give a survey of projects and sensing technology related with *FingerSense*, and then we describe the implementation of a computer-vision based prototype, which aims to demonstrate the feasibility of *FingerSense*. In the follow-on section, we present a theoretical model of *FingerSense* and quantitatively calculate the parameters in this model through a preliminary usability study.

RELATED WORKS
The key idea behind *FingerSense* is to detect and use the information implicitly encoded in specific fingers. To acquire and use such "information at your fingertips", many

[1] Here additional keystrokes also mean pressing multiple buttons at the same time.

CHI 2004, April 24–29, 2004, Vienna, Austria.
ACM 1-58113-703-6/04/0004.

potential sensing technologies are possible. In this short review, we focus on projects and methods related with detecting and identifying human fingers and hand gestures.

Visual Panel [11] uses a camera to track finger movements and translates the user's virtual typing on a paper to characters. In this project, only finger movements, not the specific finger involved is detected. Additional techniques can be applied to make the corresponding computer vision algorithms easier – cameras could be mounted under the forearm to simplify the captured images[8]; LED tags could be attached to finger joints and wrists to facilitate the segmentation of finger images[6].

It is possible to identify fingers by capturing and recognizing the associated fingerprints. Sugiura *et al* [7] uses fingerprint scanner and an out-of-the-box fingerprint recognizer to create a "finger aware" user interfaces. Different fingers are used as shortcuts to launch common applications or invoke commands in that project.

It's also possible to attach sensors to fingers to capture and identify finger movements. Actually, most of the methods used in building digital gloves could be applied to identify fingers. See[6] for a comprehensive survey of different sensing technologies behind digital gloves. Sensors which could be attached to fingers or palms include Infra-red sensor, pressure sensor, acceleration sensor[2], or Electromyographic (EMG) sensor[9] *etc*.

PROTOTYPE

In the initial stage, we build a computer vision based prototype as a proof-of-concept. In this prototype, we attach color tags on different fingertips and use images captured from a CMOS camera to detect colors, so as to identify the corresponding finger.

We used a PrimaScan iCam 320 camera with USB interface to capture still images at a resolution of 320x240. Five color tags – (pink, blue, green, yellow, purple) were attached to the corresponding fingers on a hand.

In Figure 2, the first row shows sample images captured by our camera. Thanks to the usage of color tags, the image segmentation algorithm becomes very straightforward, we first convert the captured image from RGB color space to the HSV color space. In addition, we only use the Hue and Saturation information and ignore V(brightness) in order to minimize the influence of shadow and uneven lighting. The second row of figure 2 are the image segmentation results after applying global thresholds on H-S space for the 5 pre-registered color tags. We can see that after attaching color tags to fingertips, it's computationally efficient to identify the finger used in tapping buttons.

Although the camera-based prototype is relatively large and not comfortable to use due to the additional color tags, it provides a starting point for us to identify potential usability problems related with *FingerSense.* We plan to build the second prototype by mounting *Frustrated Total Internal Reflection* (FTIR) fingerprint sensors on buttons and detect finger used by matching the partial fingerprint collected in the tapping process through a modified minutiae-matching algorithm [4].

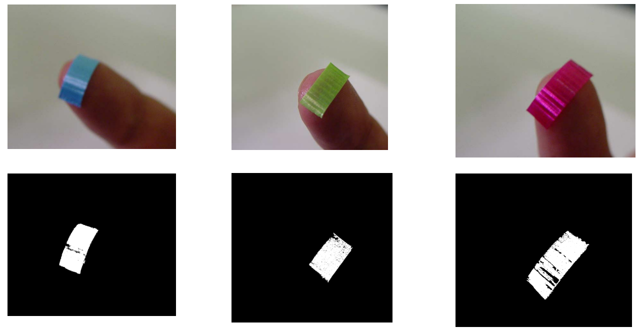

Figure 2. Threshold in H-S space to segment predefined color tags. Color tags are (from left to right) – blue (index finger), light green (middle finger), pink (the third finger).

USABILITY STUDY

As mentioned in the first section, several concerns exist with *FingerSense*, such as, is it natural for the user to switch and tap fingers to reach enhanced functions via the *FingerSense* interface? How fast can the user performance be? We feel that both theoretical analysis and usability study are necessary to answer these questions.

Modeling of Finger Switching Task

It is evident that hitting a *FingerSense* enabled button is an interaction task composed of a series of sub-tasks, including:

t1 – withdraw the formerly used finger (*prev-finger*) from the button face

t2 – cognitively determine the intended finger (*cur-finger*) that maps to the anticipated function.

t3 – extend the intended finger

t4 – visually search the target button from a list of button candidates

t5 – move the intended finger from its starting position to the target button and press it

In the worst case, the total time used to hit one button is:

$$T = t1 + t2 + t3 + t4 + t5 \qquad (1)$$

Note that the time consumed in t1 and t3 are not constant - it will depend on the *prev-finger* and the *cur-finger* involved and some other existing conditions, which we will discuss in detail later.

t2 can be improved by practice. The power law of practice [1] models such learning effect.

t4 corresponds to the choice reaction time of a user. Reaction time can be modeled by Hicks' law [1]. Since the mapping from functions to fingers is fixed in our system, according to [5], the choice reaction time should only be considered for the performance of novice users, for expert users who are familiar with the keyboard layout, time consumed in t4 can be ignored.

t5 is also called target acquisition, which can be modeled by Fitts' law [1], note that the index of performance (IP) in Fitts' law equation might not be deemed as a const in our system because the five fingers may have different performance on target acquisition. Essentially, sub-task t5 is the same as other pointing tasks (e.g. [10]) modeled by Fitts' law.

Note that some sub-tasks are not executed in a uniform and sequential manner. There are at least two variations. First, if the *prev-finger* and the *cur-finger* are the same, t1 and t3 will take less time than usual since it is not necessary for the user to fully withdraw and spread the same finger in order to hit the target button. In this case we rename the sub-tasks as t1' and t3' correspondingly. Second, t1 and t3 can be performed in parallel if the *prev-finger* and the *cur-finger* do not hit the same button. In addition, when t3 is finished, the finger is ready to perform follow-on sub-tasks even if t1 is not completely finished. A user must perform t1 and t3 in a fully sequential manner only if 1) the *pre-finger* and the *cur-finger* are not the same, and 2) the *pre-finger* and the *cur-finger* hit the same button.

To measure the performance of *FingerSense*, we need to understand the corresponding time consumed in each sub-tasks. As described above, time for t2, t4, t5 can be modeled and measured with existing knowledge in human-computer interaction. Unfortunately there is no existing method to measure t1 and t3 in our system, so estimating t1 and t3 is the primary goal of our initial usability study.

Using the conditions described above, we can decompose t1 and t3. For example, when an expert user is hitting different buttons with different fingers, and if no target acquisition is necessary (i.e. the button to push is right below the cur-finger), the task conducted here can be represented as:

$$T_{e1} = t2 + t3 \quad (2)$$

Note that t2 here is the cognitive generation time for expert users, an expert user spends less time than novice user due to practice effects. As an example, in random character entry tasks, the probability that we use two different fingers to hit two different buttons on a *FingerSense* enabled telephone keypad can be estimated [2] as 27*(27-3)*(2/3)/(27*27) = 0.59. So this is the most common "actual" task for expert users.

Similarly, the task for expert users to use different fingers to hit the same button can be represented as:

$$T_{e2} = t1 + t2 + t3 \quad (3)$$

This event has a probability of 27*2/(27*27) = 0.07 to occur on random character input.

In addition, the task of expert users to use the same finger to hit the same button can be represented as:

$$T_{e3} = t1' + t2 + t3' \quad (4)$$

This event has a probability of 27*1/(27*27) = 0.04 to occur.

Lastly, the task of expert users to use the same finger to hit different buttons has a probability of 27*(27-3)*(1/3)/(27*27) = 0.30 to occur. Since this task was measured and analyzed by previous research [3], we did not measure the performance of this task in the following study.

Based on the three conditions represented in equations (2) – (4), we designed an experiment to measure the user performance parameters of *FingerSense*.

Experimental Subjects

Three subjects participated in this preliminary usability study - two males and one female with an average age of 26. Two of them are graduate students at UC Berkeley. All of them are right-handed and had former experiences with mobile devices such as cell phone and PDA. A within subject test was conducted. The three conditions are T_{e1}, T_{e2} and T_{e3}. Each subject was presented with all three conditions. The order of the conditions presented was counter balanced in a Latin Square pattern across the three subjects.

Experimental Task

In the experiment, each user was tested under all three conditions in the experiment. For conditions T_{e1} and T_{e2}, we measure the usages of all five fingers as the *pre-fingers* and the *cur-fingers* respectively so 5x4 = 20 potential finger transitions are measured. For condition T_{e3} we measure the performance of the usage of all five fingers, i.e. 5 sub-conditions. We tested each condition at least 20 times.

Result

The results of conditions T_{e1} and T_{e3} are shown as a bi-tap transition matrix in Table 1 below (the unit is millisecond). The diagonal cells represents results related with T_{e3} and all the other cells are results for equations T_{e1}.

	Thumb	Index	Middle	Third	Little
Thumb	172.37	159.22	115.13	113.56	107.50
Index	144.56	130.16	116.38	103.20	126.67
Middle	122.63	115.13	137.53	114.30	159.11
Third	105.00	100.11	146.78	141.79	143.67
Little	132.88	116.75	189.13	206.38	145.47

Table 1. Finger switching speed matrix (ms) for condition Te1 and Te3. Diagonal cells represent t1' + t2 + t3', other cells represent t2 + t3

[2] To simplify the estimation, we assume each character has equal probability to appear (random character). Character transition digram [10] generated from large corpus is a more accurate method to estimate real world character frequencies

Similarly, the bi-tap transition matrix for conditions T_{e2} and T_{e3} are shown in table 2 below. Similar as table 1, the diagonal cells represents results related with equation (4) and all the other cells are results for equations (3)

	Thumb	Index	Middle	Third	Little
Thumb	172.37	275.00	264.78	280.56	281.44
Index	278.30	130.16	234.67	270.22	271.56
Middle	245.40	248.40	137.53	257.11	273.67
Third	286.30	268.60	281.30	141.79	287.11
Little	277.50	309.40	268.40	301.40	145.47

Table 2. Finger switching speed matrix (ms) for condition Te2 and Te3. Diagonal cells represent t1' + t2 + t3' and are duplicated from table 1. All other cells represent t1 + t2 + t3

The four findings in the usability study are -

1. The performance of t1 and t3 (Table 2) depend on the actual *prev-finger* and *cur-finger* pair and the performance is asymmetric among any two fingers.

2. In most of the testing cases (i.e. T_{e1} represented by equation (2)), t1 and t3 can be paralleled and finger switching is faster than single finger tapping no matter which finger is involved. This finding yields the insight that that *FingerSense* systems should be designed to facilitate parallel typing in order to get better performance.

3. If t1 and t3 must be carried out in a sequential order, the time for finger switching will be significantly slower than single finger tapping. In this case, some combinations such as the third finger + the little finger, are especially inefficient. The worst finger combination is about 100% slower than single finger tapping.

4. The results of single finger, same button condition T_{e3} ranges from 130ms to 172ms, which accords with the performance 200ms per keystroke of skilled user quite well. (the difference should be considered as t5 - horizontal movement for target acquisition)

CONCLUSION AND FUTURE WORK

In this paper, we proposed a novel technology named *FingerSense* to enhance the expressiveness of physical pushing buttons by fingertip identification. We surveyed potential technologies which can be used by *FingerSense*. We created a computer-vision based prototype which uses color tags to facilitate finger identification as a proof-of-concept. After a GOMS analysis of *FingerSense,* we derived the related bi-taping matrixes in a preliminary user study.

This paper is an initial step of *FingerSense*. We plan to create a second prototype based on capturing and matching the partial fingerprint on users' fingertip and conduct a larger usability study to verify our hypothesis statistically.

ACKNOWLEDGMENTS

We thank Tye Rattenbury, Matthew Kam, Zile Wei, Shumin Zhai, and Jia Li for their help and valuable feedbacks.

REFERENCES

1. Card, S.K., Moran, T.P. and Newell, A. *The Psychology of Human-Computer Interaction*. 1983, Hillsdale, New Jersey:Lawrence Erlbaum Associates Publishers.
2. Fukumoto, M., and Tonomura, Y., Body Coupled FingeRing: Wireless Wearable Keyboard, *Proc. of ACM CHI 1997*.
3. MacKenzie, I. S.and Soukoreff, R. W.. Text entry for mobile computing: Models and methods, theory and practice. *Human-Computer Interaction, 17*, 2002, 147-198.
4. Maltoni, D., Maio, D., and Jain, A., Prabhakar, S., *Handbook of Fingerprint Recognition*, ISBN 0-387-95431-7, Springer, 2003
5. Soukoreff, R. W., and MacKenzie, I. S. Theoretical upper and lower bounds on typing speed using a stylus and soft keyboard. *Behaviour and Information Technology, 14*, 1995, 370-379
6. Sturman, D.J. and Zeltzer, D. A survey of glove-based input. *IEEE Computer Graphics and Applications, 14*(1), January 1994, 30-39
7. Sugiura, A. and Koseki, Y., A User Interface Using Fingerprint Recognition: Holding Commands and Data Objects on Fingers, *ACM Symposium on User Interface Software and Technology (UIST) 1998*, 71-79
8. Vardy, A., Robinson, J., and Li-Te Cheng. The WristCam as input device. *Proc. Int. Symp. on Wearable Computers 1999 (ISWC 99)*, 199-202
9. Wheeler, K.R.; Jorgensen, C.C., Gestures as input: neuroelectric joysticks and keyboards, Pervasive Computing, IEEE , Volume: 2 Issue: 2 , April-June 2003 , 56 -61
10. Zhai, S., Hunter, M., and Smith, B. A. Performance optimization of virtual keyboards. *Human-Computer Interaction, 2002*, 17, 229-269
11. Zhang, Z. Wu, Y., Shan, Y. and Shafer, S., Visual panel: Virtual mouse keyboard and 3d controller with an ordinary piece of paper, *In Workshop on Perceptive User Interfaces*, ISBN 1-58113448 -7, 2001