2013 Conference on Technologies and Applications of Artificial Intelligence

Tap Model to Improve Input Accuracy of Touch Panels

Takahisa Tani
The Graduate University for Advanced Studies
Tokyo, Japan
Email: tani@nii.ac.jp

Seiji Yamada
National Institute of Informatics
The Graduate University for Advanced Studies
Tokyo Institute of Technology
Tokyo, Japan
Email: seiji@nii.ac.jp

***Abstract*—In recent years, devices that use touch panels as interfaces, such as smart phones and tablet PCs, have spread. These devices have many advantages. For example, operating the panel can be done more intuitively in comparison with using conventional physical buttons, and the devices are quite more flexible than those that use a traditional fixed UI. However, mistakes frequently occur when inputting with a touch panel because the buttons have no physical boundaries and users cannot get tactile feedback with their fingers because the panels never change physically. Thus, the input accuracy of touch-panel devices is lower than that of devices with physical buttons. There are studies on improving input accuracy. Most of them use language models for typing natural language or probabilistic models to describe the errors made when users tap their fingers. However, these models are not practical, and the experiments are preliminary. Thus, in this paper, we propose a more practical model for improving input accuracy, in which the relative relationships between a target object and neighbor object that might influence error making when touching the target are tested. We consider that our model can describe important properties for designing various UIs depending on practical applications. We then make a plan to conduct experiments in order to build our model in a calibrated way and discuss our evaluation of the model.**

***Keywords*-user interface, touch panel, tap model, touch accuracy**

I. Introduction

In recent years, devices that use touch panels as interfaces, such as smart phones and tablet PCs, has spread. These have many advantages. For example, operating the panel can be done more intuitively in comparison with using conventional physical buttons, and the devices are quite more flexible than those that use a traditional fixed UI.

However, mistakes frequently occur when inputting with a touch panel because the buttons have no physical boundaries and users cannot get tactile feedback because the panels never physically change when being tapped. Thus, the input accuracy of touch-panel devices is significantly lower than those using physical buttons in a traditional way. In addition, users often make unintentional mistakes when using the panels for input. In particular, smartphones usually have a smaller screen and smaller UI in comparison with conventional large UIs on a computer display. Thus, the lack of high input accuracy is serious, and this problem with using small screens for input is called the "fat finger problem"[1]. This problem is a problem of accuracy in pointing manipulation. In the future, input devices will progress, and pointing accuracy beomes more important. Thus, improving accuracy improvement is important.

Figure 1. Example of software keyboard on smartphone

There are many studies such as on improving the accuracy of software keyboards (Figure 1). The software keyboard needs to place a lot of keys in a small area. Hence, this is a typical example of the fat finger problem because the keys are too small for a user to correctly tap them. Most of these studies have used the following two methods. The first one uses language models[2][3]. The model has language information such as a dictionary. The system can predict the next character by using the pattern of inputted characters and the dictionary. For example, when the first part of a word is input, the system can predict the next character by matching the input part with words recorded in a dictionary. Although this approach is quite effective for key-typing like tapping a software keyboard, it cannot be applied to other input UIs, including tapping simple buttons, not on a software keyboard. Thus, the applicable coverage of this language model is significantly restricted, so we need to develop more general models to improve input accuracy for various concrete UIs.

The second one uses tap models[4][5]. The model has information on the difference between the locations of buttons and the points where a user taps. The system revises points on the panel of the screen.

978-1-4799-2528-5/13 $31.00 © 2013 IEEE
DOI 10.1109/.58

There are also studies that use both methods[6], [7], [8].

There are fundamental studies in which limited and concrete applications to practical UIs are not assumed[9][10]. Although these studies might provide novel knowledge in a general aspect, it is very difficult to use this knowledge to design UIs practically. Thus, these models are not practical, and the experiments are preliminary. This means that the models may be influenced by more complex factors such as the shape or color of the interface.

Thus, in this study, we choose a method that predicts a touch point from multiple sensors that was proposed by [10]. Also, we propose a more practical model that includes the influence of neighbor objects, e.g., buttons and links.

II. Method of Estimating Tap Points with Multiple Sensors.

The previous studies[10] proposed the following method. Let $\boldsymbol{s}$ be the input of the multiple sensors and (x, y) be the intended location of a user. Here, multiple sensors means the output of the touch panel (e.g., location, time, size of area, pressure), accelerator sensor, and so on. Then, they calculate the function $(x, y) = f(\boldsymbol{s})$ by regression, and the system estimates the intended location from the sensor input by f.

Next, we extend this touch model to a more practical one by introducing the relationship between a target object and a neighbor object. Furthermore, we try to introduce incremental learning to improve the touch model through user execution of the UI.

A. *Influence of Interface Shape on Tap Model*

The tap location in practical use changes with various factors. In particular, it is known that the tap model significantly changes with the differences in how the device is held and how the fingers operate the device. This influence may be solved by estimating these factors with sensors like acceleration sensors.

However, the tap model may change with the interface shape. In figure 2, let the blue square be a target. The tap location has a distribution like the blue line (a), where a Gaussian distribution is assumed for the touch model. If there is another object (the green square) on the right hand side, the distribution will move to the left side because the user is aware of the green square. In addition, there might be an influence from the color, size, or shape of the object.

The previous studies do not consider this kind of influence because the sensor inputs s do not include this information. Therefore, we propose adding interface shape $\boldsymbol{i}$ as a variable of f. We consider this information makes a touch model quite more practical. Since our touch model with interface shape is basically characterized with (x, y) coordination on the touch panel, it can be applicable to the various UIs independent of the properties of tappable objects like buttons and icons. Thus, this model has a wide coverage to be applied to the same as conventional touch models. Furthermore, this proposed touch model is very practical and precise because it effectively introduces the influence of neighbor objects in contrast with traditional touch models.

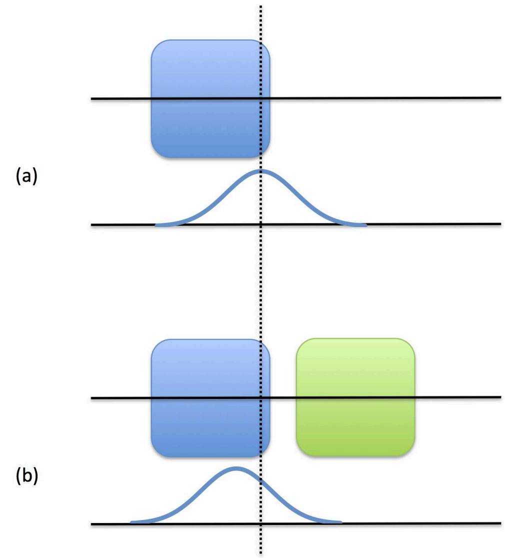


Figure 2. Tap target and tap location

III. Experiment Plan

We are currently planning the experiments to obtain a large number of training data and to evaluate the accuracy of our touch model as follows.

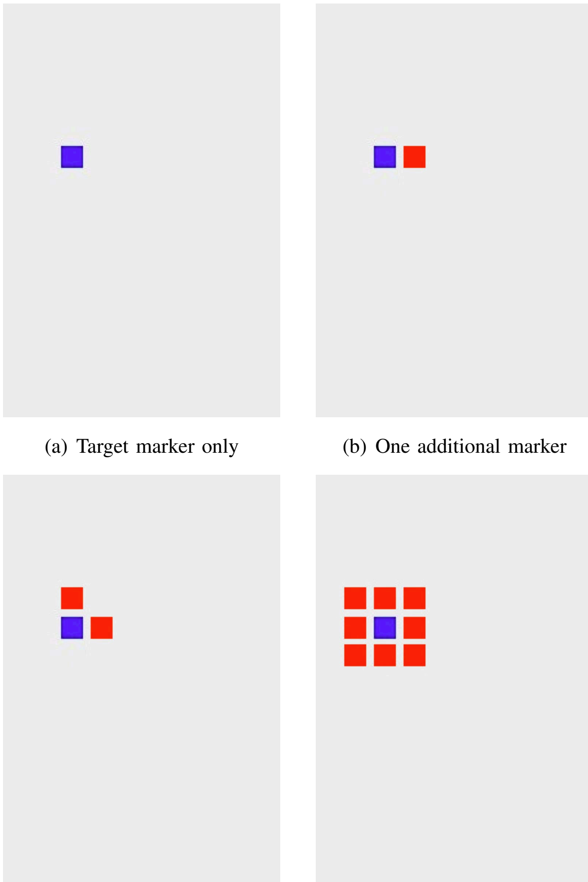
A. *Method*

We evaluate the influence of the interface shape on the tap model and the method of obtaining the training data implicitly. Participants perform a task in which they tap a marker on a touch panel. Figure 3 shows the task windows. The marker will disappear, and another one will appear in another location when it is tapped. A marker does not disappear until being tapped. We instruct participants to tap as quickly as possible.

The target color is blue or yellow, and the size is 3 $\sim$ 10mm. In addition, some additional markers appear, as shown in Figure 3(b) $\sim$ 3(d). All parameters of a tap are recorded and considered. We show a sufficient number of markers randomly in order to analyze the learning effect.

B. *Evaluation*

1) Relationship Between Interface Shape and Tap Parameters: The differences in an interface are considered for each parameter of the obtained tap data.



(a) Target marker only (b) One additional marker

(c) Two additional markers (d) Eight additional markers

Figure 3. Task windows

2) Accuracy of Input: Input data is corrected by using the tap model obtained by using the proposed technique, and input accuracy is evaluated. As a candidate for comparison, the result of not correcting and the model that does not take interface form into consideration are examined.

IV. Conclusion

In this study, we considered that our model can describe important properties for designing various UIs depending on practical applications. We therefore made a plan to conduct experiments in order to build our model in a calibrated way and discuss the evaluation of our model. In the future, we will experiment on and evaluate our method.

References

[1] K. A. Siek, Y. Rogers, and K. H. Connelly, "Fat finger worries: How older and younger users physically interact with pdas," In Proc. of Interact 2005, Tech. Rep., 2005.

[2] J. Goodman and G. Venolia, "Language modeling for soft keyboards," in *Proc. AAAI 2002*. AAAI Press, 2002, pp. 419–424.

[3] K. Al Faraj, M. Mojahid, and N. Vigouroux, "Bigkey: A virtual keyboard for mobile devices," in *Proceedings of the 13th International Conference on Human-Computer Interaction. Part III: Ubiquitous and Intelligent Interaction*. Springer-Verlag, 2009, pp. 3–10.

[4] L. Findlater and J. Wobbrock, "Personalized input: improving ten-finger touchscreen typing through automatic adaptation," in *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, ser. CHI '12. ACM, 2012, pp. 815–824.

[5] J. Himberg, J. Häkkilä, P. Kangas, and J. Mäntyjärvi, "On-line personalization of a touch screen based keyboard," in *Proceedings of the 8th international conference on Intelligent user interfaces*, ser. IUI '03. ACM, 2003, pp. 77–84.

[6] A. Gunawardana, T. Paek, and C. Meek, "Usability guided key-target resizing for soft keyboards," in *Proceedings of the 15th international conference on Intelligent user interfaces*, ser. IUI '10. ACM, 2010, pp. 111–118.

[7] D. Rudchenko, T. Paek, and E. Badger, "Text text revolution: a game that improves text entry on mobile touchscreen keyboards," in *Proceedings of the 9th international conference on Pervasive computing*, ser. Pervasive'11. Springer-Verlag, 2011, pp. 206–213.

[8] M. Goel, A. Jansen, T. Mandel, S. N. Patel, and J. O. Wobbrock, "Contexttype: using hand posture information to improve mobile touch screen text entry," in *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, ser. CHI '13. ACM, 2013, pp. 2795–2798.

[9] C. Holz and P. Baudisch, "Understanding touch," in *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, ser. CHI '11. ACM, 2011, pp. 2501–2510.

[10] D. Weir, S. Rogers, R. Murray-Smith, and M. Löchtefeld, "A user-specific machine learning approach for improving touch accuracy on mobile devices," in *Proceedings of the 25th annual ACM symposium on User interface software and technology*, ser. UIST '12. ACM, 2012, pp. 465–476.