Tap Model to Improve Input Accuracy of Touch Panels

Takahisa Tani
The Graduate University for Advanced Studies
2-1-2 Hitotsubashi, Chiyoda
Tokyo 101-8430, Japan
tani@nii.ac.jp

Seiji Yamada
National Institute of Informatics
2-1-2 Hitotsubashi, Chiyoda
Tokyo 101-8430, Japan
seiji@nii.ac.jp

Figure 1: Example of software keyboard on smartphone

HAI 2014, October 29–31, 2014, Tsukuba, Japan.
ACM 978-1-4503-3035-0/14/10.
http://dx.doi.org/10.1145/2658861.2658935

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 objects 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 also conducted preliminary

experiments in order to build our model in a calibrated way and discuss our evaluation of the model.

Author Keywords
user interface; touch panel; tap model; touch accuracy.

ACM Classification Keywords
H.5.2. [Information Interfaces and Presentation: Input devices and strategies (e.g. mouse, touchscreen)]

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*[5]. 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.

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. The some of these uses the tap models[1, 2]. 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.

There are fundamental studies in which limited and concrete applications to practical UIs are not assumed[3, 6]. 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 layout or color of the interface.

Thus, in this study, we focus on a method that predicts a touch point from multiple sensors that was proposed by [6]. Also, we propose a more practical model that includes the influence of neighbor objects, e.g., buttons and links.

Method of Estimating Tap Points with Multiple Sensors.
The previous studies[6] proposed the following method. Let 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,

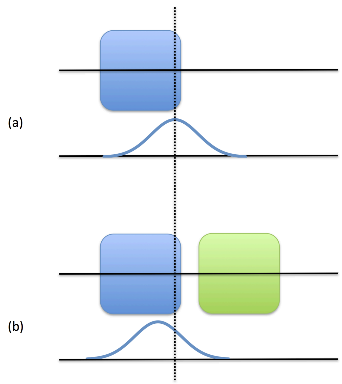

Figure 2: Tap target and tap location

time, size of area, pressure), accelerometer, 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.

Influence of Interface Layout 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. We call this difference of tap point *kinematic error* (e_k). This influence may be solved by estimating these factors with sensors like acceleration sensors.

However, the tap model may change with the interface layout. 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 squire and tries to avoid miss-touching the green squire instead of the blue squire. We call this difference of tap point *cognitive error* (e_c). In addition, there might be an influence from the position, size, color, 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 layout $\boldsymbol{i}$ as a variable of f. We consider this information makes a touch model quite more practical. Since our touch model with interface layout 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[3, 6].

Experiment

We conducted the experiment to obtain a large number of training data and to evaluate the accuracy of our touch model as follows.

Method
In order to evaluate the influence of the interface layout on the tap model, we developed the method of obtaining the training data implicitly. Participants perform a task in which they tap a marker on a touch panel. The marker disappear, and another ones appear in another position when the markers are tapped. A marker does not disappear until being tapped. The target marker which a participant should tap has a while circle in the center, and various neighbor makers appear around it. We instruct participants to tap as quickly as possible.

Result
Figure 3 shows the kinematic error(e_k) for horizontal and vertical axis. The curve is estimated by Gaussian Process Regression (GPR)[4]. Tap points (white circles) shift to the side of an operate hand($e_k > 0$) and gaps in points distant from an operate hand are larger

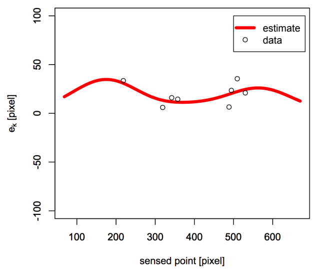

Figure 3: Tap points vs. kinematic errors for each axis

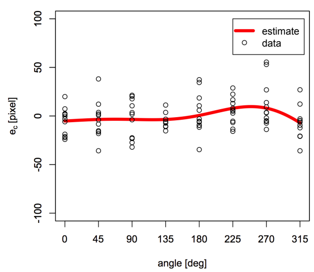

Figure 4: The angle of an additional marker vs. cognitive errors for each axis

than gaps in near points. These features are the same tendencies as previous works[3, 6].

Figure 4 shows the cognitive error(e_c) for each axis. Here, we use data which has only one neighbor marker, and the effect of e_k is eliminated. Tap points shift to upper right when a neighbor marker is at lower left side, shift to up when a neighbor marker is at lower right side and shift to lower left when additional marker is at right side. These results suggest that cognitive error exists.

Conclusion

In this study, we considered that our tap model can describe important properties for designing various UIs depending on practical applications. We conduct an experiment in order to build our model in a calibrated way and discuss the evaluation of our model. The results suggested that cognitive error existed, In the future study, we will additional experiment and evaluate our method.

References

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

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

[3] Holz, C., and Baudisch, P. Understanding touch. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '11, ACM (2011), 2501–2510.

[4] Rasmussen, C. E., and Williams, C. K. I. *Gaussian Processes for Machine Learning (Adaptive Computation and Machine Learning)*. The MIT Press, 2005.

[5] Siek, K. A., Rogers, Y., and Connelly, K. H. Fat finger worries: how older and younger users physically interact with pdas. In *Proceedings of the 2005 IFIP TC13 international conference on Human-Computer Interaction*, INTERACT'05, Springer-Verlag (2005), 267–280.

[6] Weir, D., Rogers, S., Murray-Smith, R., and Löchtefeld, M. 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*, UIST '12, ACM (2012), 465–476.