User Performance by the Difference between Motor and Visual Widths for Small Target Pointing

Hiroki Usuba
Meiji University
Tokyo, Japan
cs172047@meiji.ac.jp

Shota Yamanaka
Yahoo Japan Corporation
Tokyo, Japan
syamanak@yahoo-corp.jp

Homei Miyashita
Meiji University
Tokyo, Japan
homei@homei.com

ABSTRACT

We investigate user performance by the difference between motor and visual widths when pointing to small targets. In recent desktop GUIs, the visual shapes of targets tend to be smaller than the motor width. For example, the motor width of a window frame in which the user can click tends to be larger than the visual width. We assume that it is difficult to point to the target when the motor and visual widths are different. Therefore, we compare the conditions where the motor and visual widths are equal and not equal. The results show that, compared with the conditions where the motor and visual widths were equal, users completed a task without any problems in terms of movement time and error rate even if the visual width was one pixel when the motor and visual widths were not equal. We also discuss the potential implications such as design for visually small targets.

Author Keywords

Difference in motor and visual widths; small target pointing; Fitts' law; human performance.

ACM Classification Keywords

H.5.2. Information interfaces and presentation (e.g., HCI): User Interfaces; Input devices and strategies.

INTRODUCTION

In graphical user interfaces (GUIs), there are many clickable components such as buttons and hyperlinks. Regarding the design of clickable components, Johnson said "Make click-targets — graphical buttons, menu items, links — big enough that they are easy for people to hit. Don't make people click on tiny targets…" [p. 189 in 19]. Johnson's design implication is based on Fitts' law [13] which can predict the movement time *MT* for pointing. Fitts' law means that as the distance to a target increases and/or the target width decreases, the movement time increases. In accordance with Johnson's design implication and Fitts' law, if designers do not want to frustrate users, the target should be large enough.

NordiCHI'18, September 29-October 3, 2018, Oslo, Norway

ACM ISBN 978-1-4503-6437-9/18/09$15.00
https://doi.org/10.1145/3240167.3240171

However, there are many small targets in GUIs, as shown in Figure 1. For such targets, users point to a window frame to resize the window, which is sometimes difficult because the frame width is, e.g., 8 pixels in Windows 8 (Figure 1a). In particular, for this target, the visual width equals the motor width[1]. Regarding Figure 1b-c, although the motor width is almost as large as that in Figure 1a, the draggable window frame in Figure 1b, for example, is indicated by the boundary between the window and the background. Hence, the visual width can be defined as one pixel. In this situation, users must aim for such a one-pixel target to click on the target. In the design shown in Figure 1b-c, although the target is not clearly drawn and assuming that pointing to the target is more difficult, such a design is currently mainstream. Actually, the window frames are clearly represented in Windows 8; however, when updated to Windows 10, the frame becomes a boundary line.

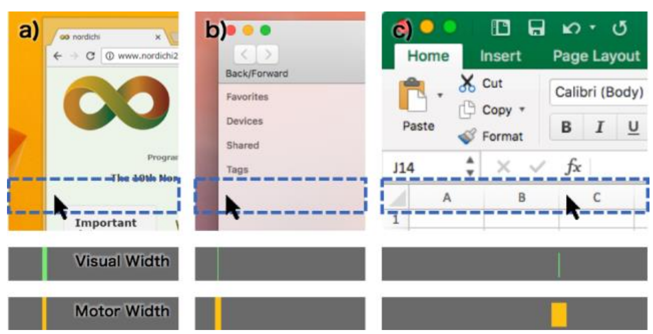

Figure 1. Top row: Users begin to perform some actions (e.g. resizing) by pointing to a target. Middle row: visual width. Bottom row: motor width. In this study, we deal with two situations: when the visual width is (a) equal to or (b, c) smaller than the motor width.

In this study, we focus on the difference between the motor and visual widths, in particular for small targets like window frames. Therefore, we conduct an experiment involving pointing operations under the conditions of differing visual sizes (Figure 1). We believe that users who are familiar with the applications roughly remember the

---

[1] Motor width is the size in which the cursor can click the target, while visual width is the size represented on the display.

motor width when using the applications. We believe that even if the visual width is one pixel, the users are aware of the motor width due to the experience; thus, users succeed in pointing to the target. In our experiment, we try to simulate participants' experience by informing them of the motor width and investigate the effects of the difference of the experience. Moreover, our experimental results show that the design update of Windows is successful and that the main stream of the user interface design, i.e., flat design, does not negatively affect pointing performance.

RELATED WORK

In this section, first, we describe Fitts' law, which is a well-accepted model for pointing. Second, we discuss how the visual effects affect user performance because there are various visual appearances of targets in our experiment. Third, we introduce techniques to improve pointing to small targets. Finally, we describe the features of the small target.

Fitts' Law

Equation 1 shows Fitts' law [13]. The model can predict the movement time of pointing with two linear regression constants (*a* and *b*). The dependent variables are the distance to target *D* and target width *W*. *ID* indicates the index of difficulty. As *ID* increases (the distance increases and/or the width decreases), it is difficult for users to click the target, and the users take a longer time to perform pointing.

$$MT = a + bID, \text{where } ID = \log_2\left(\frac{D}{W} + 1\right) \qquad (1)$$

Fitts' law is also applicable to operations other than pointing (e.g. dragging an object [15, 24] and text selection [8]) and to several input techniques [6, 8, 12, 22, 24]. In addition, researchers have extended Fitts' law to two-dimensional tasks [2, 23] and trajectory-based tasks [1]. When pointing, not only the movement time but also the error rate can be predicted [21, 28].

In the pointing tasks in those studies, the motor width of the target equals its visual width. In this study, we conduct an experiment under the condition that the motor width does not match the visual width. If the *W* in Fitts' law indicated the visual width, our experimental results under the condition of different motor and visual widths would not show a good fit for the model.

Visual Effects when Pointing

Previously, the relationship between the visual condition and the pointing performance has been explored. Murphy [25] found that high contrast targets required a shorter movement time, but the accuracy was lower than that of low contrast targets. Yu et al. [31] found that visual post-selection feedback in touch pointing tasks increases the movement time; however, the visual feedback improves the accuracy. However, Akamatsu et al. [3] and Appert et al. [4] found that the visual feedback of a cursor hovering on a target does not affect the movement time. In our experiment, under the condition that the visual width is one pixel, we assume that it is very difficult for the participants to succeed in clicking a target with no visual feedback because the users have to judge whether the cursor is exactly on a target. Therefore, we provide visual feedback to the participants by changing the color of the cursor. In addition, Cockburn and Brock found that visually expanding targets affects the pointing performance [11]. In comparison, in our experimental conditions, the motor width is also larger than the visual width; however, these widths are static, not dynamically expanding.

Improving Small Target Pointing

According to Fitts' law, it is difficult for users to click a target if the width is small. Therefore, pointing techniques for small targets have been explored. There are two approaches to the target width. One approach is enlarging the activation area of the cursor by using, for example, an area cursor [20, 29] and a bubble cursor [17]. In these techniques, by extending the activation area of the cursor, the *W* in Fitts' law is enlarged (and the distance to the target is shortened). Therefore, even if the target width is small, users can click the target by capturing the target in the enlarged activation area. The other approach is enlarging the target itself by using, for example, the birdlime icon [27] and bubble targets [10]. With the birdlime icon, by enlarging the target, users are prevented from overshooting it. With bubble targets, when the cursor is close to the target, the target is enlarged like a bubble, and the user can click it even if the target is small. Additionally, the nudging technique [30] has a different approach and allows users to drag a target by pushing it.

Semantic pointing [7] is a technique to dynamically adjust the control-display (C-D) gain depending on the distance between the cursor and the target. As the cursor becomes closer to the target, the gain decreases to slow down the cursor speed. The use of sticky icons [29] is similar in that the C-D gain decreases as the cursor approaches the target. These techniques have the same effect as if *W* increases in the motor space; thus, users can more easily point to the target. Note that the visual width does not change. Our experimental conditions also allow unequal visual and motor widths, but these sizes do not dynamically change.

Acquisition of Small Target

Based on Fitts' law [13], Johnson [19] said that developers should make targets large enough so that users can easily click the target; however, many targets are smaller than 10 pixels [10]. Additionally, Chapuis and Dragicevic [9] mentioned that "little is known about the reasons why very small targets are so difficult to acquire." They conducted an experiment using various scales (i.e. motor scale, visual scale, and quantization), and their results showed that both visual and motor scales affect the pointing performance even if the *ID* of Fitts' law was the same. Note that, in their experiment, motor scale means the above-mentioned C-D gain; it is not the same as the motor width of a target as in this study. Therefore, our focus and findings have new contributions compared to their study. In addition, we believe that our findings contribute to understanding small target acquisition and developing techniques to improve user performance as much as the above-mentioned studies.

EXPERIMENT

We conducted an experiment to investigate the effects of the difference between the motor and visual widths in small target pointing. Roughly speaking, the task conditions were divided into two types: the visual and motor widths were equal (Figure 2a) or not equal (Figure 2b). Additionally, to remove the effects of other factors (e.g. approach angle) and simplify the experimental task, the task was 1D pointing.

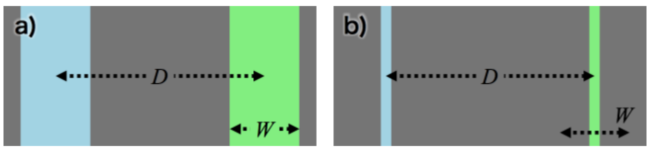

Figure 2. Experimental task outline. Visual and motor target widths are equal (a, *Normal*) or not equal (b, *Line* and *Line-Unknown*).

Apparatus

We used an Apple MacBook Pro laptop (Intel Core i5, 2.4 GHz, 2 cores, Intel Iris 1536 MB, 8 GB RAM, macOS Sierra). The display scaling resolution was 1680 × 1050 pixels (the actual size was 13.3 inches, 286.47 × 179.04 mm). The input device was an optical mouse, Logitech M100R (1000 dpi). The cursor speed was the macOS's default. The full-screen experimental system was developed by using JavaScript.

Participants

Twelve graduate and undergraduate university students (of which two were females) participated in the experiment (*mean age* = 22.92, *SD* = 1.56 years). All participants were right-handed and operated the mouse with the right hand (three participants usually used a mouse, the others usually used a trackpad).

Visual and Motor Widths

The visual width of the target was determined by target type *T*. There were 3 target type conditions: *Normal*, *Line*, and *Line-Unknown*. In the *Normal* condition, the visual width and the motor width *W* were equal; they were determined by the value of *W* (Figure 1a, Figure 2a). In the *Line* and *Line-Unknown* conditions, however, the visual width was always one pixel, and the motor width was determined by the value of *W* (Figure 1b-c, Figure 2b). In the *Line* condition, the pixel value of *W* was displayed on the screen as text information (Figure 3). In familiar applications, even if the visual target width is drawn as one pixel, as shown in Figure 1b-c, users succeed in clicking the target because they are roughly aware of the motor width by getting used to the application. *Line* simulates the users' operations when using familiar applications. In contrast, in the *Line-Unknown* condition, we did not inform the participants of the motor width. This simulates participants using unfamiliar applications. To be fair, in the *Normal* condition, we also gave the pixel value of *W* to the participants.

Figure 3. The pixel value of motor target width *W* was shown for *T* = *Normal* and *Line* conditions.

Task

Under *Normal* or *Line* conditions, the *W* of the pixel was displayed as text at the upper left of the screen (Figure 3) and spoken aloud using a recorded voice. In the *Line* condition, because the visual width did not match the motor width *W* and the participants could not estimate the motor width from the visual width, we asked the participants to predict motor width *W* from the given text and audio information.

First, the participants clicked any position, and then the cursor automatically moved to the middle of the start area (blue)[2]. When the participants clicked the start area, a sound was played to inform them of the trial beginning, and the measurements began. Then, the participants had to aim for the end area (green) as fast and accurately as possible. If the click was in the motor width, a success sound was played. However, if the click was outside the motor width, the failure sound was played, and the trial was regarded as an error.

In the *Line-Unknown* condition, because the participants did not know the motor width, it was difficult to click the target. In addition, in unfamiliar applications, users grasp the motor width by the change of the cursor shape as visual feedback and succeed in clicking the target. Therefore, in all conditions, we changed the cursor color from black to yellow when it was in the motor width, as shown in Figure 4. The change allowed the participants to grasp whether the cursor is within the motor width even in *Line* and *Line-Unknown* conditions. To prevent the participants from predicting the motor width before starting a trial, the cursor color change only occurred in the end area. In addition, we informed the participants of the change of cursor color as described above. In the measurements, the participants cannot clutch the mouse, but they can perform this operation before starting the trial.

Design and Procedure

Figure 2 shows an outline of the experimental tasks. Distance *D* was 480 or 640 pixels (81.85 or 109.13 mm,

[2] In the *Line* and *Line-Unknown* conditions, it is difficult to click the start area because its visual width is also one pixel. Therefore, we move the cursor to the center of the start area automatically before starting a trial. To be fair, the initial automatic cursor movement was performed in all conditions.

respectively). Motor width *W* was 3, 7, 11, or 13 pixels (0.51, 1.19, 1.88, or 2.22 mm, respectively). The value of *W* was decided with reference to the value of the window frame of the macOS Finder and Windows Explorer and the value of the cell border of MS Office Excel. *ID* was consistently more than 3.00 (5.25-7.74 bits), and we assumed that the participants' movements were visually controlled [14]. The order of the three *T* conditions was balanced among the 12 participants by using a Latin square pattern, and the order of *D* and *W* was randomized. One *set* consisted of *D* (2) × *W* (4) = 8 trials. Before starting the experiment, each participant received a brief explanation. Under each *T* condition, after an introductory exercise set, each participant completed 20 sets to produce experimental data. After completing all the sets, we asked each participant about the strategy that they were using to try to complete the task under each condition. A total of 5,760 trials (i.e., *T* (3) × *D* (2) × *W* (4) × 20 sets × 12 participants) were carried out, and the time needed was approximately 30 min.

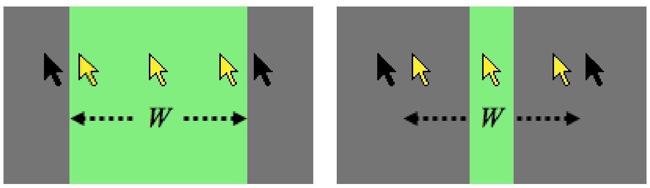

Figure 4. Change of cursor color. If the cursor is within the motor target width, it turns yellow.

Measurements

The cursor coordinates were recorded at approximately 100 Hz. The dependent variables were the movement time *MT* (the time from clicking the start area to clicking the target excluding the error trials), spread of hits on the x-axis SD_x (the standard deviation of the x-coordinate of the cursor position including the error trials), error rate, and reaction time *RT* (the time from the cursor color change to the trial end excluding the error trials).

Hypotheses

We made the following two hypotheses:

1. The ranking regarding *MT* is *Normal* (shortest), *Line* (middle), and *Line-Unknown* (longest)
2. With *Line-Unknown*, *MT* is independent of *W*
3. The results for *Line-Unknown* are not a good fit for Fitts' law.

Regarding hypothesis 1, there are three reasons. First, in *Line* and *Line-Unknown*, because the visual width is one pixel, we assumed that it is difficult and time-consuming for the participants to confirm that the cursor is in the motor width of a target; thus, their *MT*s are longer than that of the *Normal*. Second, if the *W* in Fitts' law indicated the visual target width, *ID* would be high. Tasks under the *Line* and *Line-Unknown* conditions are more difficult than those under the *Normal* condition. Finally, we assumed that users who were familiar with the applications (*Normal* and *Line*) complete a task faster than users who were unfamiliar with the applications (*Line-Unknown*) because of knowing the motor width. Therefore, we assumed that the ranking of *MT* is as described above. Regarding hypothesis 2, with *Line-Unknown*, the participants do not know the motor width and have no choice but to aim for a one-pixel line at the beginning of a trial. Therefore, we assumed that the movement was independent of the visual width because the visual width is always the same. Regarding hypothesis 3, motor width *W* cannot accurately capture the variability of the participants' aiming motion; thus, the results do not show a good fit for Fitts' law.

Results

In 5,760 trials, 321 pointing errors occurred (5.72%). We analyzed the data via repeated-measure ANOVA and the Bonferroni post hoc test. The independent variables were *T*, *W*, and *D*. The dependent variables were the measurements described in the previous section. In the graphs, the error bars represent standard error.

Movement Time

We observed the main effects for *D* ($F_{1, 11}$ = 127.39, $p <$ 0.001) and *W* ($F_{3, 33}$ = 220.52, $p < 0.001$). The post hoc test showed that increasing *D* and/or reducing *W* slowed down *MT*. Figure 5 shows that there was no significant difference in *MT* between *T* ($F_{2, 22}$ = 2.62, p = 0.094). Therefore, hypothesis 1 and hypothesis 2 were rejected.

Reaction Time

We observed the main effects for *W* ($F_{3, 33}$ = 24.63, $p <$ 0.001) and *T* ($F_{2, 22}$ = 28.91, $p < 0.001$). The post hoc test showed that reducing *W* slowed down *RT* and that the *RT* of *Normal* (352.26 ms) was smaller than that of *Line* (400.47 ms, $p < 0.001$) and *Line-Unknown* (391.41 ms, $p < 0.001$). Additional interaction was found for *W* × *T* ($F_{6, 66}$ = 68.51, p < 0.001, Figure 6). Increasing *W* widened the gap between *Normal* and the other conditions. There were significant differences between *Normal* and *Line* at *W* = 3 ($p < 0.01$) and between *Normal* and the others at *W* = 11 and *W* = 13 (for both, $p < 0.001$).

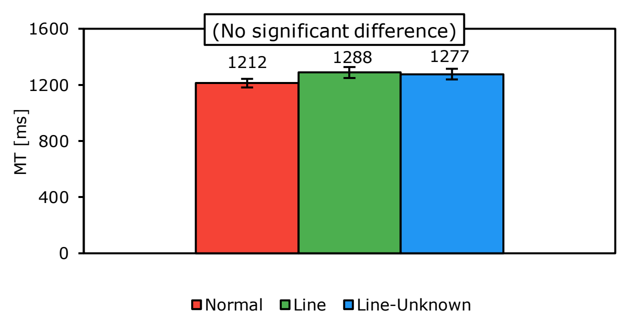

Figure 5. *MT* versus *T*

Error Rate

We observed the main effect for *W* ($F_{3, 33}$ = 66.33, $p <$ 0.001). The post hoc test showed that increasing *W* decreased the error rate. Additional interaction was found for $W \times T$ ($F_{6, 66} = 4.52$, $p < 0.01$, Figure 7). Increasing W narrowed the gap between the conditions. There were significant differences between *Normal* and *Line-Unknown* at $W = 3$ ($p < 0.05$).

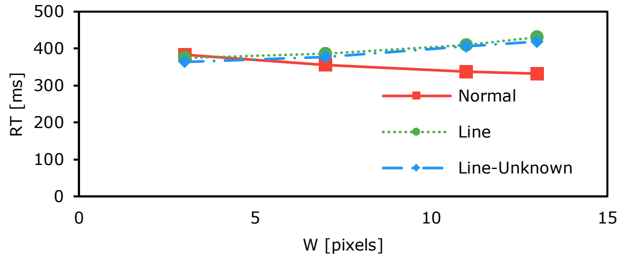

Figure 6. *W* versus *RT* under each *T*

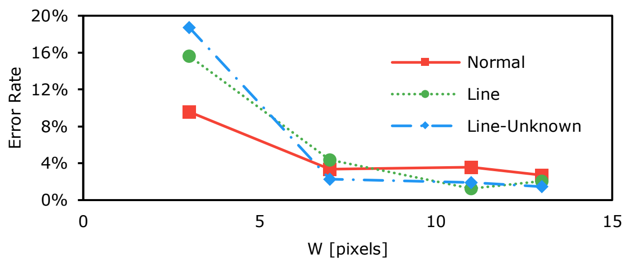

Figure 7. *W* versus the error rate under each *T*

Standard Deviation for X-coordinate

We observed the main effects for W ($F_{3, 33} = 355.25$, $p < 0.001$). The post hoc test showed that reducing W reduced SD_x. Additional interactions were found for $D \times W$ ($F_{3, 33} = 2.97$, $p < 0.05$) and $W \times T$ ($F_{6, 66} = 4.04$, $p < 0.01$, Figure 8). There were significant differences between *Normal* and *Line-Unknown* at $W = 3$ ($p < 0.001$) and $W = 11$ ($p < 0.01$). As examples of cursor hit positions on the x-axis, Figure 9 shows those when $W = 13$ under each T. While the differences were significant, they were slight in terms of pixels (less than 0.36 pixels). In addition, the distributions under each T were similar, so we assume that T has only a little effect on SD_x.

Model Fitting

According to the result of the movement time, the movement time was changed by the motor width, not the visual width. Therefore, we believed that the W in Fitts' law is indicated by the motor width (W in the experiment). Overall, Fitts' law showed a reasonably good fit for the 24 data points ($3T \times 2D \times 4W$): $MT = 262.94 + 159.45ID$ with $R^2 = 0.90$. This is in accordance with the typical threshold [16, 26]. If we merged the three T conditions, the combined condition is a good fit for Fitts' law for the eight data points ($2D \times 4W$): $MT = 262.94 + 159.45ID$ with $R^2 = 0.99$. Figure 10 shows that, in each T, the values of R^2 were greater than 0.95. We confirmed that our experimental conditions were suitable for the pointing task. Considering the fitness and the movement time, hypothesis 3 was rejected.

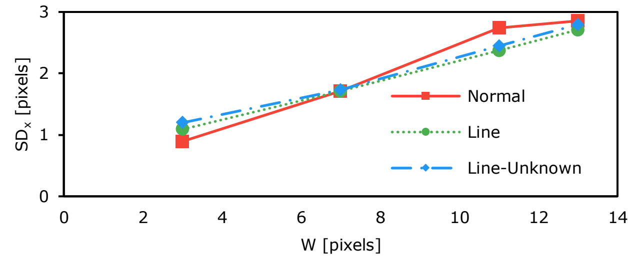

Figure 8. *W* versus SD_x under each *T*

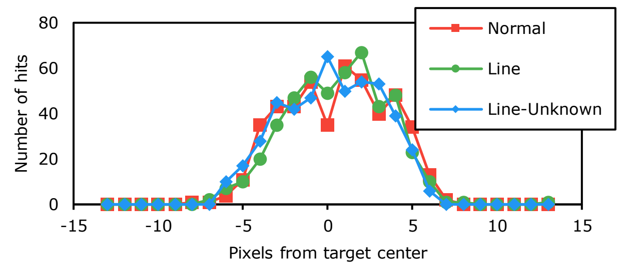

Figure 9. Distribution of x-coordinates of cursor hit position at *W* = 13 under each *T*

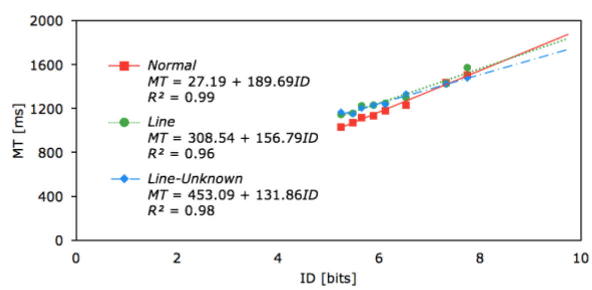

Figure 10. Model fitting under each *T*

DISCUSSIONS

Regarding target type T, the results showed that there were significant differences in the reaction time, but there were no significant differences in the movement time. In addition, each target type was a good fit for Fitts' law. In short, our hypotheses have been rejected. We discuss the reasons behind them below.

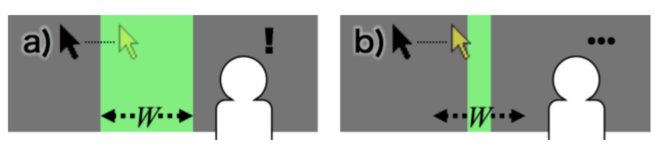

Figure 11. a) When motor and visual target widths are equal, users can predict that cursor enters the motor width by the next operation. b) When motor and visual widths are not equal, users cannot predict this.

Difference in Reaction Time

Under the condition where motor and visual target widths are equal (*Normal*), we believe that it is easy for the

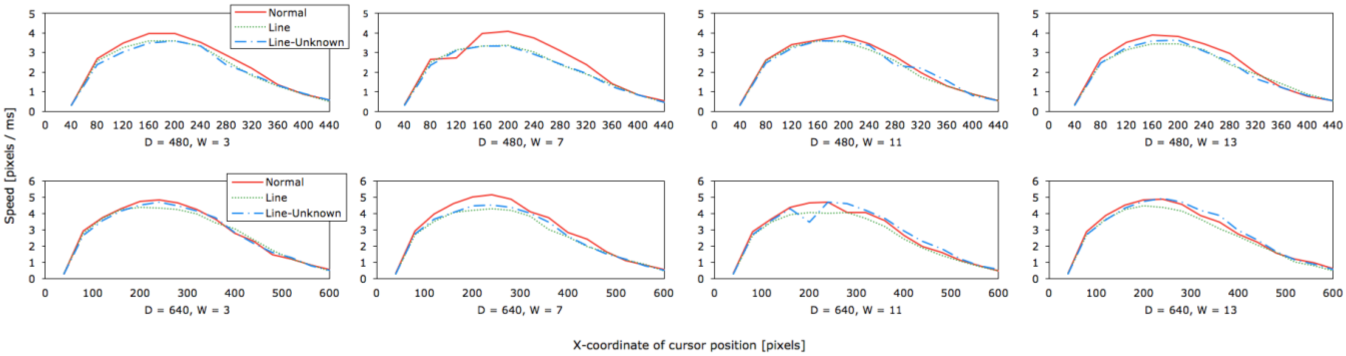

Figure 12. Average speed per 40 pixels versus x-coordinates of cursor position at all *D* × *W* under each *T*.

participants to predict whether a cursor enters the motor width. When pointing, users move the cursor while aiming for the target. When the cursor is close to the target, the users see that the cursor is close to the visual width, and they can predict that the cursor will enter the motor width soon (Figure 11a). Therefore, when the cursor enters the motor width, the prediction allows the users to click the target immediately by reacting to the cursor color change. However, under other conditions (*Line* and *Line-Unknown*), the visual target width is always one pixel. Therefore, because it is difficult for the participants to grasp that the cursor is close to the motor width, they cannot predict the remaining distance to it (Figure 11b). The absence of prediction makes the participants late for clicking the target. We believe that the gap between *Normal* and the other conditions (Figure 6) depended on whether there is a prediction. Additionally, under the conditions of smaller target width (e.g. *W* = 3 pixels), because the cursor movement by a single operation (submovement) may sometimes be more than 3 pixels, the cursor may overshoot the target. Because it is difficult for the participants to predict that the next submovement will allow the cursor to ride on the three-pixel target, they are late for clicking under all target type conditions. Therefore, in Figure 6, at *W* = 3 and 7 pixels, the significant gaps in *RT* were not observed.

Movement Time and Model Fitting

Regarding the movement time, we believe that the reason there are no significant differences in the movement time is that the movements to a target under all conditions are the same. Figure 12 shows the average speed at all *D* × *W* under each target type. Note that the speeds are resampled every 40 pixels. In pointing tasks, the participants occasionally overshoot the target, so the cursor may be moved to-and-fro. Therefore, averaging the speed at any x-coordinate is difficult, so Figure 12 shows the movement just before the target. The maximum value of the x-coordinate of the cursor position in Figure 12 is the value obtained by subtracting 40 pixels from *D*.

According to Figure 12, in any condition, the participants' cursor movement is: 1) go quickly up to the half distance and 2) go carefully to the target. These tendencies are consistent with related studies [e.g., 5]. There does not seem to be a remarkable difference in the speed under any conditions. In addition, the strategy that most participants have mentioned is move the cursor while watching the change of the cursor color. The participants succeeded in performing pointing while relying on the change of the cursor color. Because the cursor color was changed depending on the motor width *W*, the *T* conditions did not affect the movement time (Figure 5); hence, there were good fits for Fitts' law regardless of *T*.

Design Implication

The results showed that if the motor target width is more than 7 pixels, the target is large enough for pointing in terms of error rates (approximately 4% [22, 26], Figure 7). In addition, we found that the visual condition of the target and knowledge of the motor target width have little effect on the pointing performance.

In Windows 8, the motor width is clearly represented by the frame (*Normal* in the experiment, Figure 1a). In Windows 10, the motor width is indicated by a one-pixel boundary between the window and the background (*Line* or *Line-Unknown* in the experiment). However, because the motor width of the frame is equal to or greater than 7 pixels, according to our experimental results, the update does not affect the performance. In addition, according to our findings, users, who are using Windows 10 for the first time and do not know the motor width of the frame, can operate the mouse without the problems (the movement time is too long and the error rate is too high). Regarding pointing, the update is successful. In summary, if developers dislike clarifying a small target such as a window frame (Figure 1a), they should set the motor width of the target to more than 7 pixels.

LIMITATIONS AND FUTURE WORK

First, Figure 13 shows the navigation bar of the website[3] of NordiCHI 2018. When a cursor is outside the motor width of an item in the navigation bar, users who visit this page for the first time are uncertain of the motor width. The users may think that the motor width equals the text length

[3] http://www.nordichi2018.org/papers

(Figure 13a). However, when the cursor hovers on the item, users know that the motor width is larger than the text height/length (Figure 13b). Users must move the cursor near the item without knowledge of the actual motor width. Hence, this situation is regarded as *Line-Unknown* in our experiment. However, the item has an even larger width than the window frame on which we focused in this study. Because our findings have not been verified where the motor width is larger than 13 pixels, we cannot directly apply our results and discussion (e.g., *T* did not affect *MT*, high Fitts' law fitness, etc.) to such GUIs. Our future work includes conducting an experiment with such larger target widths.

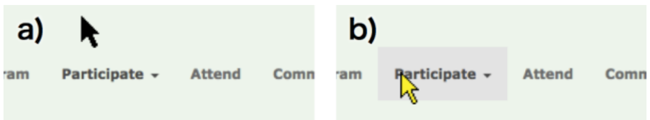

Figure 13. Example of navigation bar. a) Cursor leaves from an item. b) Cursor hovers on an item.

Second, we dealt with only targets whose motor width was larger than the visual width. Yet, in GUIs, there are also targets whose motor width is smaller than the visual width. In Figure 14, the motor width for selecting a column is smaller than the visual width. Measuring the user performance difference owing to such inconsistencies is included in our interests. Previously, expanding and shrinking targets have been explored [11, 18]. However, in the experimental condition of our future work, the change of the visual width is not dynamic.

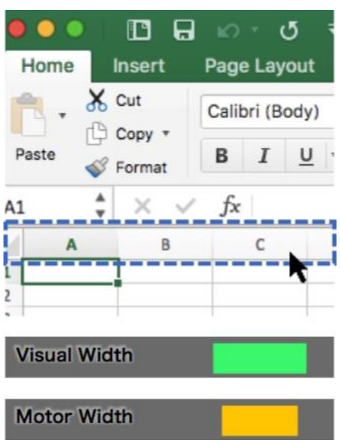

Figure 14. Visualizing visual (green) and motor (orange) widths of selecting a column.

Finally, in our experiment, we only gave the participants the change of cursor color as visual feedback. However, as shown in Figure 13, the visual feedback is not only changing the cursor color but also highlighting the target. We have not verified the effect of the perception of the motor width by visual feedback yet.

CONCLUSION

In this study, we conducted an experiment on whether the motor target width and visual target width were equal. Our results showed that users perform pointing without the negative problems (the movement time is too long and the error rate is too high) even if the visual target width was one pixel such as at the boundary between the window and the background. Additionally, based on the results, we considered the mainstream user interface design. For example, a window frame is clearly represented in Windows 8, and when updated to Windows 10, the frame is indicated by a single-pixel boundary between the window and the background. According to the results, regarding pointing, the update was successful because the visual target width did not affect the performance. In this way, we believe that our findings contribute to the understanding of the features of a small target, and we hope that our findings facilitate developing novel techniques for small target acquisition.

REFERENCES

1. Johnny Accot and Shumin Zhai. 1997. Beyond Fitts' law: models for trajectory-based HCI tasks. In *Proceedings of the ACM SIGCHI Conference on Human factors in computing systems* (CHI '97), 295-302. http://dx.doi.org/10.1145/258549.258760
2. Johnny Accot and Shumin Zhai. 2003. Refining Fitts' law models for bivariate pointing. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (CHI '03), 193-200. http://dx.doi.org/10.1145/642611.642646
3. Motoyuki Akamatsu, I. Scott MacKenzie, and Thierry Hasbroucq. 1995. A comparison of tactile, auditory, and visual feedback in a pointing task using a mouse-type device. *Ergonomics*, 38, 4, 816-827. https://doi.org/10.1080/00140139508925152
4. Caroline Appert, Olivier Chapuis, and Michel Beaudouin-Lafon. 2008. Evaluation of pointing performance on screen edges. In *Proceedings of the working conference on Advanced visual interfaces* (AVI '08), 119-126. https://doi.org/10.1145/1385569.1385590
5. Takeshi Asano, Ehud Sharlin, Yoshifumi Kitamura, Kazuki Takashima, and Fumio Kishino. 2005. Predictive interaction using the delphian desktop. In *Proceedings of the 18th annual ACM symposium on User interface software and technology* (UIST '05), 133-141. http://dx.doi.org/10.1145/1095034.1095058
6. Xiaojun Bi, Yang Li, and Shumin Zhai. 2013. FFitts law: modeling finger touch with fitts' law. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (CHI '13), 1363-1372. https://doi.org/10.1145/2470654.2466180
7. Renaud Blanch, Yves Guiard, and Michel Beaudouin-Lafon. 2004. Semantic pointing: improving target acquisition with control-display ratio adaptation. In *Proceedings of the SIGCHI Conference on Human*

Factors in Computing Systems (CHI '04), 519-526. http://dx.doi.org/10.1145/985692.985758

8. Stuart K. Card, William K. English, and Betty J. Burr. 1978. Evaluation of mouse, rate-controlled isometric joystick, step keys, and text keys for text selection on a CRT. *Ergonomics*, 21, 8, 601-613. https://doi.org/10.1080/00140137808931762
9. Olivier Chapuis and Pierre Dragicevic. 2011. Effects of motor scale, visual scale, and quantization on small target acquisition difficulty. *ACM Trans. Comput.-Hum. Interact*. 18, 3, 3, 32 pages. https://doi.org/10.1145/1993060.1993063
10. Andy Cockburn and Andrew Firth. 2004. Improving the Acquisition of Small Targets. *People and Computers XVII — Designing for Society*, 181-196. https://doi.org/10.1007/978-1-4471-3754-2_11
11. Andy Cockburn and Philip Brock. 2006. Human on-line response to visual and motor target expansion. In *Proceedings of Graphics Interface* 2006 (GI '06), 81-87.
12. Brian W. Epps. 1986. Comparison of six cursor control devices based on Fitts' law models. In *Proceedings of the Human Factors and Ergonomics Society Annual Meeting*, 30, 4, 327-331. https://doi.org/10.1177/154193128603000403
13. Paul M. Fitts. 1954. The information capacity of the human motor system in controlling the amplitude of movement. *Journal of Experimental Psychology*, 47, 6, 381-391. http://psycnet.apa.org/doi/10.1037/h0055392
14. Khai-Chung Gan and Errol R. Hoffman. 1988. Geometrical conditions for ballistic and visually controlled movements. *Ergonomics*, 31, 5, 829-839. https://doi.org/10.1080/00140138808966724
15. Douglas J. Gillan, Kritina Holden, Susan Adam, Marianne Rudisill, and Laura Magee. 1990. How does Fitts' law fit pointing and dragging?. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (CHI '90), 227-234. http://dx.doi.org/10.1145/97243.97278
16. Julien Gori, Olivier Rioul, Yves Guiard and Michel Beaudouin-Lafon. 2018. The Perils of Confounding Factors: How Fitts' Law Experiments can Lead to False Conclusions. In *Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems* (CHI '18). http://dx.doi.org/10.1145/3173574.3173770
17. Tovi Grossman and Ravin Balakrishnan. 2005. The bubble cursor: enhancing target acquisition by dynamic resizing of the cursor's activation area. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (CHI '05), 281-290. http://dx.doi.org/10.1145/1054972.1055012
18. Errol R. Hoffmann, Alan H.S. Chan and Coskun Dizmen. 2013. Capture of shrinking targets with realistic shrink patterns. *Ergonomics*, 56, 11, 1766–1776. http://dx.doi.org/10.1080/00140139.2013.829924
19. Jeff Johnson. 2014. *Designing with the Mind in Mind, Second Edition: Simple Guide to Understanding User Interface Design Guidelines*, 189-191.
20. Paul Kabbash and William A. S. Buxton. 1995. The “prince” technique: Fitts' law and selection using area cursors. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (CHI '95), 273-279. http://dx.doi.org/10.1145/223904.223939
21. Byungjoo Lee and Antti Oulasvirta. 2016. Modelling Error Rates in Temporal Pointing. In *Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems* (CHI '16), 1857-1868. https://doi.org/10.1145/2858036.2858143
22. I. Scott MacKenzie. 1992. Fitts' law as a research and design tool in human-computer interaction. *Hum.-Comput. Interact*. 7, 1, 91-139. http://dx.doi.org/10.1207/s15327051hci0701_3
23. I. Scott MacKenzie and William Buxton. 1992. Extending Fitts' law to two-dimensional tasks. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (CHI '92), 219-226. http://dx.doi.org/10.1145/142750.142794
24. I. Scott MacKenzie, Abigail Sellen, and William A. S. Buxton. 1991. A comparison of input devices in element pointing and dragging tasks. In *Proceedings of the SIGCHI Conference on Human Factors in Computing* Systems (CHI '91), 161-166. http://dx.doi.org/10.1145/108844.108868
25. Robert C. Murphy. 2006. The Effect of Target Color and Contrast on Movement Times in Aimed Movement Tasks. In *Proceedings of the Human Factors and Ergonomics Society Annual Meeting*, 50, 21, 2340-2344. https://doi.org/10.1177/154193120605002109
26. R. William Soukoreff and I. Scott MacKenzie. 2004. Towards a Standard for Pointing Device Evaluation, Perspectives on 27 Years of Fitts' Law Research in HCI. *International Journal of Human-Computer Studies*, 61, 751-789. http://dx.doi.org/10.1016/j.ijhcs.2004.09.001
27. Takayuki Tsukitani, Kazuki Takashima, Motoo Asahi, Yuichi Itoh, Yoshifumi Kitamura and Fumio Kishino. 2009. The birdlime icon: improving target acquisition by dynamically stretching target shape. *In Proceedings of the 22th Annual ACM Symposium on User Interface Software and Technology* (UIST '09), 13-14.
28. Jacob O. Wobbrock, Edward Cutrell, Susumu Harada, and I. Scott MacKenzie. 2008. An error model for pointing based on Fitts' law. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (CHI '08), 1613-1622. https://doi.org/10.1145/1357054.1357306

29. Aileen Worden, Nef Walker, Krishna Bharat, and Scott Hudson. 1997. Making computers easier for older adults to use: area cursors and sticky icons. In *Proceedings of the ACM SIGCHI Conference on Human factors in computing systems* (CHI '97), 266-271. http://dx.doi.org/10.1145/258549.258724

30. Shota Yamanaka and Homei Miyashita. 2013. The nudging technique: input method without fine-grained pointing by pushing a segment. In *Proceedings of the adjunct publication of the 26th annual ACM symposium on User interface software and technology* (UIST '13 Adjunct), 3-4. https://doi.org/10.1145/2508468.2514927

31. Chun Yu, Hongyi Wen, Wei Xiong, Xiaojun Bi, and Yuanchun Shi. 2016. Investigating Effects of Post-Selection Feedback for Acquiring Ultra-Small Targets on Touchscreen. In *Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems* (CHI '16), 4699-4710. https://doi.org/10.1145/2858036.2858593