A Comparison of Three Selection Techniques for Touchpads

I. Scott MacKenzie
Dept. of Computing & Information Science
University of Guelph
Guelph, Ontario Canada N1G 2W1
+1 519 824 4120 x8268
smackenzie@acm.org

Aleks Oniszczak
Dept. of Computing & Information Science
University of Guelph
Guelph, Ontario Canada N1G 2W1
+1 519 824 4120
aoniszcz@uoguelph.ca

ABSTRACT
Three methods of implementing the select operation on touchpads were compared. Two conventional methods — using a physical button and using "lift-and-tap" — were compared with a new method using finger pressure with tactile feedback. The latter employs a pressure-sensing touchpad with a built-in relay. The relay is energized by a signal from the device driver when the finger pressure on the pad surface exceeds a programmable threshold, and this creates both aural and tactile feedback. The pressure data are also used to signal the action of a button press to the application. In an empirical test with 12 participants, the tactile condition was 20% faster than lift-and-tap and 46% faster than using a button for selection. The result was similar on the ISO-recommended measure known as throughput. Error rates were higher with the tactile condition, however. These we attribute to limitations in the prototype, such as the use of a capacitive-sensing touchpad and poor mechanical design. In a questionnaire, participants indicated a preference for the tactile condition over the button and lift-and-tap conditions.

Keywords
Touchpads, pointing devices, input devices, tactile feedback, Fitts' law

INTRODUCTION
Since notebook computers are usually operated in constrained spaces, mice are generally not used as the systems' pointing device. Until recently, most notebooks included either a trackball or an isometric joystick as a pointing device. Apple was the first company to incorporate a touchpad in a notebook computer, and many other companies have since chosen touchpads over joysticks or trackballs. A touchpad implements the select operation either using physical buttons (as with mice) or using a "lift-and-tap" technique.

This paper presents an empirical evaluation of a new selection technique for touchpads that is based on tactile feedback. The work is a continuation of a design described in an earlier short paper [11].

TOUCHPADS VS. MICE
Although touchpads are also available for desktop computers, most people prefer to use a mouse. So, why is a mouse a better pointing device than a touchpad when space is not an issue? The answer may lie in the separation of selection from positioning. Using a mouse, the pointer is positioned by moving the mouse on a mousepad. The device is gripped between the fingers and thumb and movement occurs via the wrist and forearm. With a touchpad, pointer movement is accomplished by sliding a finger along the touchpad's surface. Both are generally used as "relative positioning" devices, where the pointer moves relative to its previous position when the device or finger moves.

For a mouse, selecting is the act of pressing and releasing a button while the pointer is over an icon or other screen object. Double clicking and dragging are related operations that also require pressing a button. There are two common implementations for selecting with touchpads: (a) using physical buttons, or (b) using lift-and-tap. Both inherit problems we are attempting to correct in our tactile touchpad.

Physical Buttons
Most touchpads include physical buttons that are typically operated with the index finger or thumb. If an index finger is used, the finger must move frequently between the touchpad and the buttons and this impedes performance compared with the same procedure using a mouse. If the thumb is used, then positioning and selecting proceed in concert, as with a mouse; however, the result may be sub-optimal because of interference between the muscle and limb groups engaged. A similar problem has been noted for trackballs [12], wherein high error rates (particularly for dragging tasks) are attributed to the "closeness" of the muscle and limb groups required

for the separate acts of positioning and selecting. With a mouse, on the other hand, positioning occurs primarily via the wrist and forearm, while selecting occurs primarily through the fingers. Thus, the limbs and muscle groups are separate for each task and tend not to interfere.

Lift-and-Tap

Because of the problem noted above, most touchpads also support "lift-and-tap" as an alternative to pressing buttons. However, this is perhaps replacing one problem with another. We'll illustrate this by considering the basic transactions with computer pointing devices. According to Buxton's three-state model of graphical input [4], these can be modeled by three states:

State 0 out-of-range (the device/finger is elevated)
State 1 tracking (pointer movement)
State 2 dragging (movement with button depressed)

These are identified in Figure 1, annotated for mouse interaction.

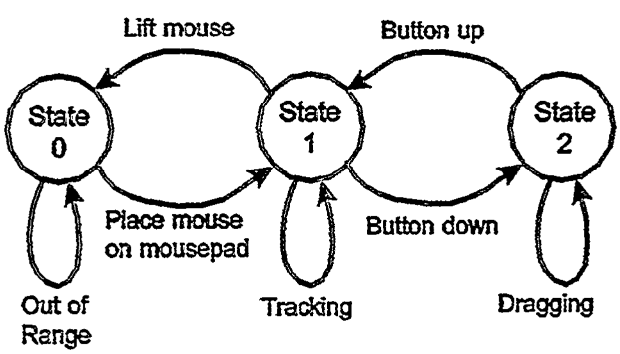


Figure 1. Buxton's three state model of graphical input with labels appropriate for mouse interaction

For touchpads and mice, pointer motion occurs in state 1, the tracking state. The comparison becomes interesting when we consider the state transitions required for clicking, double clicking, dragging, and clutching. (Clutching is the act of lifting the mouse or finger from the mousepad or touch surface and repositioning it.) Figure 2 identifies the state transitions for the most common operations for a mouse and a lift-and-tap touchpad. A few observations follow. In general, operations require more state transitions with a lift-and-tap touchpad than with a mouse. A simple click on a mouse begins and ends in state 1, whereas on a touchpad it begins in state 1 and ends in state 0. To return to pointer positioning (state 1), the finger must resume contact with the pad, and if this occurs too quickly a dragging operation occurs. Note as well that clutching on a lift-and-tap touchpad is confounded with clicking and dragging. This is not the case with a mouse.

Operation	Mouse	Lift-and-tap Touchpad
Pointer Positioning	1	1
Single Click	1-2-1	1-0-1-0
Double Click	1-2-1-2-1	1-0-1-0-1-0
Dragging	1-2	1-0-1-0-1
Clutching	1-0-1	1-0-1

Figure 2. State transitions for common operations using a mouse and a lift-and-tap touchpad.

THE TACTILE TOUCHPAD

In view of the preceding discussion, it is worth exploring alternate, perhaps better, implementations for state transitions. One possibility is to implement them by pressing harder with the pointing/positioning finger. A mouse button provides aural and tactile feedback when it is pressed, and this is an important component of the interaction. Similar feedback may be elicited from a touchpad by means of a mechanical solenoid or relay positioned under the pad and activated with an electrical signal to create a "click" sensation in the fingertip. Since a mouse button clicks both when pressed and when released, the same response is desirable for a tactile touchpad to achieve a more natural feel.

To prevent spurious clicks, the transitions should include hysteresis. That is, the state 1-2 pressure level that maps to the button-down action should be higher than the state 2-1 pressure level that maps to the button-up action. This is illustrated in Figure 3. The correct thresholds must be determined in user tests.

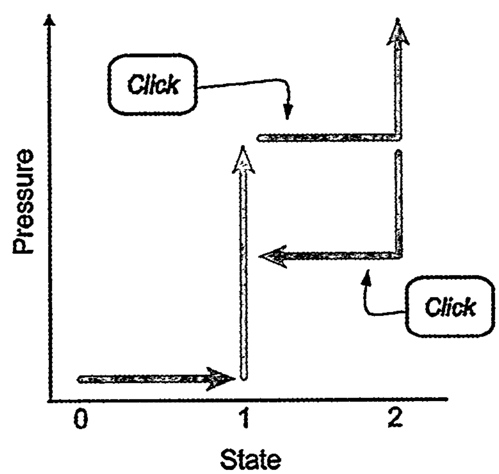


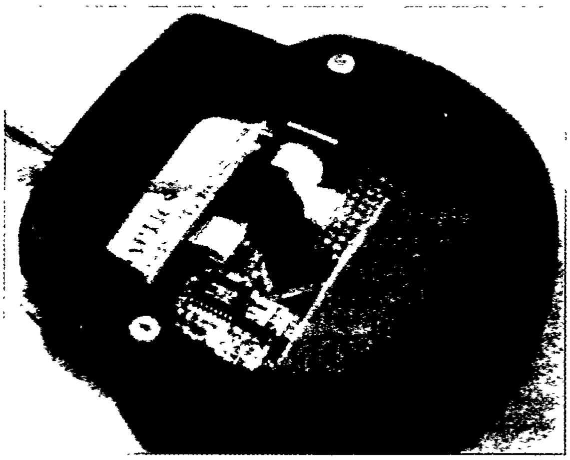
Figure 3. Pressure-state function. A click is generated for state 1-2 transitions and for state 2-1 transitions.

There is prior work on embedding a solenoid under a mouse button to create tactile feedback. A study by

Akamatsu and MacKenzie [1] found significant reductions in movement times for target selection tasks using a modified mouse incorporating tactile feedback as compared to an unmodified mouse. Using a Fitts' law analysis of the data, it was found that the tactile condition produced the highest throughput of all tested conditions. It was surmised that similar results would be achievable with the tactile touchpad. One can provide aural feedback through the computer's existing sound system. However, we feel the combination of spatially-placed aural and tactile feedback at the finger tip is preferable to spatially-displaced audio-only feedback using the system's loudspeaker, although the latter is worthy of investigation.

Our tactile touchpad is illustrated in Figure 4. For our prototype, we cut a hole in the bottom of a Synaptics T1002D capacitive touchpad and installed a Potter & Brumfield T90N1D12-5 relay. A wooden platform attached to base provides space for the relay. The relay is controlled by signals sent from the host's parallel port.

(a)



(b)

Figure 4. The tactile touchpad. (a) top view. (b) bottom view.

The Synaptics touchpad includes an *x-y-z* mode in which the *z*-axis information is the applied pressure. Our software uses *z*-axis information to determine when to energize and de-energize the relay. In informal tests with pilot subjects we determined that, of the 256 pressure levels detected by the touchpad, a value of 140 with a hysteresis value of 5 produced an acceptable response — one similar to the feel of a mouse button.

ISO TESTING OF POINTING DEVICES

Although there is an abundance of published evaluations of pointing devices in the disciplines of human-computer interaction and human factors, the methodologies tend to be ad hoc, and this greatly diminishes our ability to interpret the results or to undertake between-study comparisons. Fortunately, there is an emerging ISO standard that addresses this particular problem [8]. The full standard is ISO 9241, "Ergonomic design for office work with visual display terminals (VDTs)". The standard is in seventeen parts, and some have received approval as a DIS (draft international standard). Part 9 of the standard is called "Requirements for non-keyboard input devices". As of this writing it is in the CD (committee draft) stage.

ISO 9241-9 describes, among other things, quantitative tests to evaluate computer pointing devices. The procedures are well described and will allow for consistent and valid performance evaluations of one or more pointing devices.

The standard quantitative test is a point-select task. The user manipulates the on-screen pointer using the pointing device and moves it from a starting position to a target and selects the target by pressing and releasing a button on the device. There are many variations on this test; however, a simple reciprocal selection task is easiest to implement and allows for a large quantity of empirical data to be gathered quickly. The task is "reciprocal" because the user moves the pointer back and forth between targets, alternately selecting the targets. The selections are "blocked" with multiple selections per task condition.

As the point-select task is carried out, the test software gathers low-level data on the speed and accuracy of the user's actions. The following three dependent measures form the basis of the subsequent quantitative evaluation:

Movement Time. Movement time (*MT*), or task completion time, is the mean time in seconds or milliseconds for each trial in a block of trials. Since the end of one trial is the beginning of the next, the movement time is simply the total time for a block of trials divided by the number of trials in the block.

Error Rate. Error rate (*ER*) is the percentage of targets selected while the pointer is outside the target.

Throughput. Throughput (*TP*) is a composite measure, in "bits per second", based on both the speed and accuracy of performance. The measure was introduced in 1954 by Paul Fitts [5], and it has been widely used in human factors and experimental psychology ever since.[1] See [16] [9] for extensive reviews.

Throughput, as specified in the ISO draft standard, is calculated as follows:

$$Troughput = \frac{ID_e}{MT} \tag{1}$$

where

$$ID_e = \log_2\left(\frac{D}{W_e}+1\right) \tag{2}$$

The term ID_e is the effective index of difficulty, and carries the unit "bits". It is calculated from D, the distance to the target, and W_e, the effective width of the target.

The term MT is the movement time to complete the task, and carries the unit "seconds". Thus, throughput carries the unit "bits per second", or just "bps".

The use of the "effective" width (W_e) is important. W_e is the width of the distribution of selection coordinates computed over a block of trials. Specifically,

$$W_e = 4.133 \times SD_x \tag{3}$$

where SD_x is the standard deviation in the selection coordinates measured along the axis of approach to the target. This implies that W_e reflects the spatial variability or accuracy that occurred in the block of trials. As a result, throughput is a measure of both the speed and the accuracy of the user's performance. In some sense, throughput reflects the overall efficiency with which the user was able to accomplish the task given the constraints of the device or other aspects of the interface.

It is important to test the device on difficult tasks as well as easy tasks; so, multiple blocks of trials are used, each with a different target distance and/or target size.

METHOD

Participants

Twelve participants (5 male, 7 female) were used in the study. All participants were right handed, and all used computers with graphical user interfaces on a daily basis. Two participants had prior experience with touchpads.

Apparatus

A 166 MHz Pentium-class system with a 17" color monitor was used. The *Ctmouse* mouse driver for DOS, version 1.2, was used for all but the tactile touchpad condition. For the latter, a custom driver was written to implement the special features of the tactile condition.

[1] Fitts used the term "index of performance" instead of throughput. The term "bandwidth" is also used.

The experiment used custom software known as the *Generalized Fitts' Law Model Builder* [15]. The software executes under DOS and interacts with the system's pointing device through the installed mouse driver.

All three selection techniques used the same device, a modified Synaptics T1002D touchpad, as described earlier. Standard features of the touchpad include two physical buttons and a lift-and-tap button emulation in firmware.

For each block of trials the experimental software presented a new target condition. Two rectangles of width W separated by distance D appeared. A crosshair pointer appeared in the left rectangle and a red X appeared in the opposite rectangle denoting it as the current target (see Figure 5.)

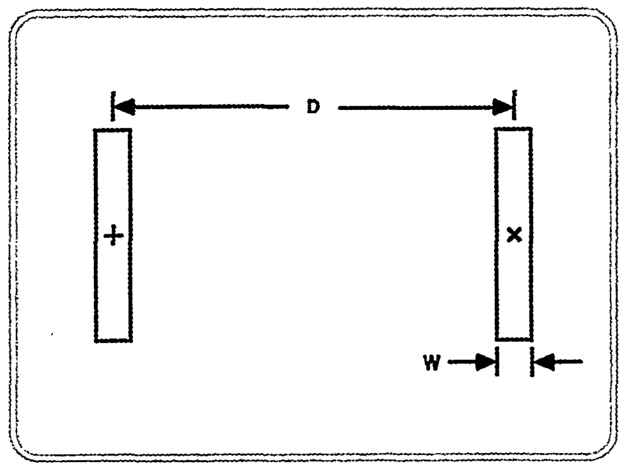


Figure 5. Experimental condition.

Procedure

Participants were instructed to move the pointer by moving their index finger on the touchpad surface. Specifically, they were instructed to move the pointer as quickly and accurately as possible from side to side alternately selecting the target using the current selection technique.

As each target was selected the red X disappeared and reappeared in the opposite rectangle. This helped synchronize participants though a block of trials. If a select operation occurred while the pointer was outside the target, a beep was heard to signal an error. Participants were instructed to continue without trying to correct errors. For each task condition, participants performed 20 selections.

Before gathering data, the task and the selection technique were explained and demonstrated to the participants. Participants were given a block of warm-up trials prior to data collection.

Design

The experiment was a 3 × 3 × 3 × 3 × 20 within subjects design. The independent variables were as follows:

Selection Technique	button, lift-and-tap, tactile
Block	1, 2, 3
Target Distance	40, 80, 160 pixels
Target Width	10, 20, 40 pixels
Trial	1, 2, 3 ... 20

The conditions above combined with 12 participants represent a total of 19,440 trials. To minimize skill transfer, the presentation of the selection techniques was counter balanced. The target distance/size conditions were blocked. Each block consisted of nine distance/size combinations presented in random order. For each condition, participants performed 20 trials in succession.

The distance/size conditions were chosen to create a set of tasks covering a range of task difficulties. The easiest task combines the largest target (40 pixels) with the shortest distance (40 pixels). The index of task difficulty is

$$ID = \log_2\left(\frac{D}{W}+1\right) = \log_2\left(\frac{40}{40}+1\right) = 1.00 \text{ bits} \qquad (4)$$

The most difficult task combines the smallest target (10 pixels) with the largest distance (160 pixels):

$$ID = \log_2\left(\frac{D}{W}+1\right) = \log_2\left(\frac{160}{10}+1\right) = 4.09 \text{ bits} \qquad (5)$$

Rest intervals were permitted between blocks of trials. The duration of rest intervals was based on participants' discretion. All three selection techniques were tested in a single session lasting about an hour. At the end, participants were given a brief questionnaire on their impressions of the three selection techniques.

RESULTS AND DISCUSSION

Since the experiment employed a within-subjects design, a Latin Square was used to balance potential learning effects. However, there remained the possibility of asymmetrical skill transfer [14] from one selection technique to the next based on the order of presentation. This was tested for and was found not to have occurred, as the effect for order of presentation was not statistically significant on all three dependent measures (movement time, error rate, throughput, $F_{2,9} < 1$).

The grand means on the three primary dependent measures were 1641 ms for movement time, 6.6% for error rate, and 1.17 bps for throughput. The interaction technique and block effects on these measures are reported in the following sections.

Speed and Accuracy

The tactile selection technique had the lowest movement time per trial at 1345 ms. The other conditions were slower by 20% for lift-and-tap (1611 ms) and by 46% using the physical button (1967 ms). These differences were statistically significant ($F_{2,18} = 47.6, p < .0001$).

Exactly the opposite ranking was observed on error rates, however. Using a button for the select operation, the error rate was 4.1%. It was 1.4× higher using lift-and-tap (5.8%) and 2.4× higher using the tactile condition (9.9%). However, these differences were not statistically significant ($F_{2,18} = 2.27, p > .05$).

The results for speed and accuracy are shown in Figure 6. Overall performance is better toward to bottom-left of the figure.

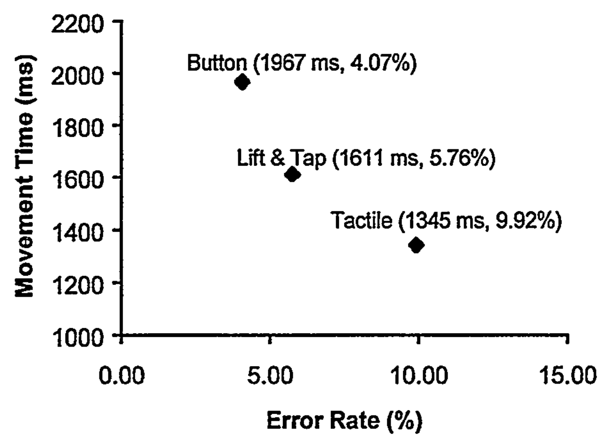


Figure 6. Results for speed and accuracy

Throughput

A strong analysis of the effect of selection technique is obtained by the dependent measure throughput, because it reflects both the speed and accuracy of performance and because it is the measure recommended in the ISO draft standard, 9241-9. The highest throughput was observed in the tactile condition at 1.43 bps. The other conditions exhibited lower throughputs by 25% for lift-and-tap (1.07 bps) and by 31% using a button (0.99 bps). See Figure 7. The differences were statistically significant ($F_{2,18} = 18.0$, $p < .0001$).

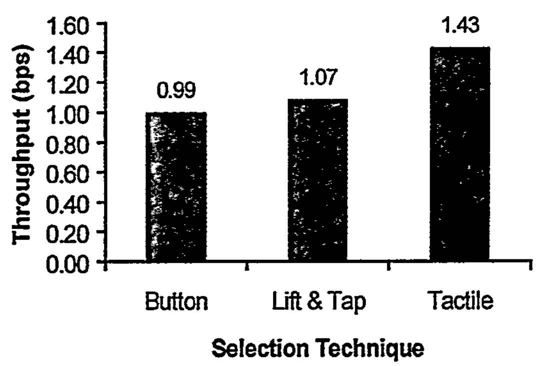
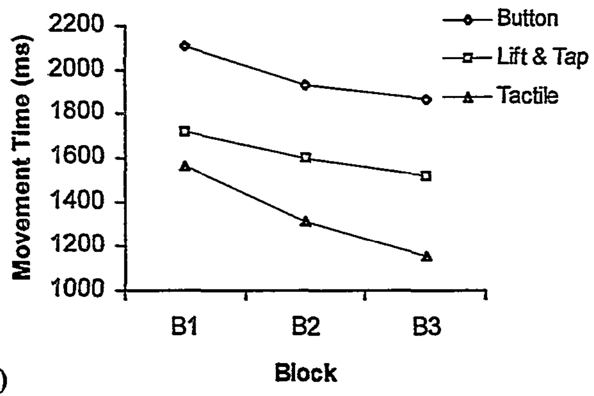
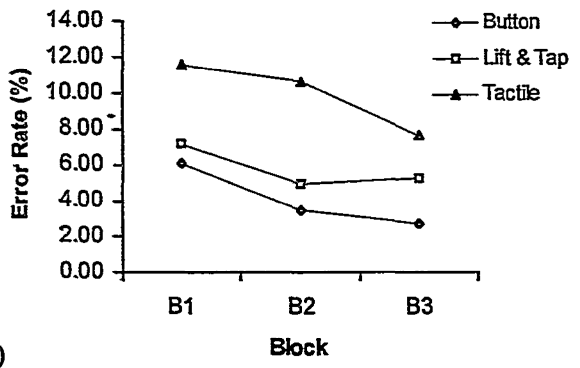


Figure 7. Throughput by selection technique

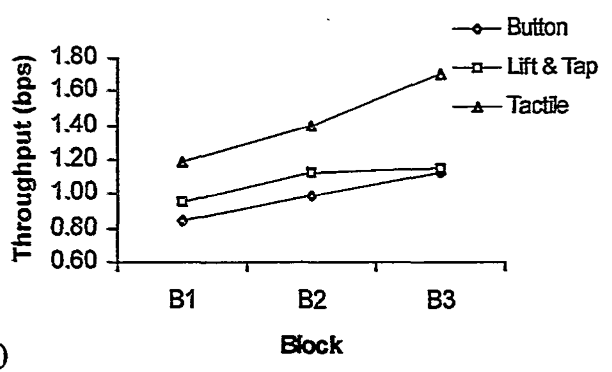
These measures for throughput are on the low side when compared to other pointing devices. We have conducted other unpublished studies using the same experimental conditions, and have obtained measures in the range of 3.0-4.5 bps for mice and 2.0-3.5 bps for trackballs. Published figures for throughput are also higher, in general. A 1991 study reported 3.3 bps for a Kensington trackball, 4.5 bps for an Apple mouse, and 4.9 bps for a Wacom stylus [12], while a 1993 study found throughput equal to 4.3 bps for the mouse [13]. Rates less than 4 bps are not uncommon, however (e.g., [2, 7, 10, 3, 6]).



(a)



(b)



(c)

Figure 8. Block by interaction technique for (a) movement time, (b) error rate, and (c) throughput.

Learning Effects

For each selection technique, participants performed three blocks of trials in succession. Each block consisted of 20 trials on each of the nine randomly presented target conditions (180 total trials). It is worthwhile, therefore, to examine the effect of "block" on the three dependent measures, since this reflects the extent to which participants improved with practice. As well, a block × selection technique interaction effect may be present, indicating different learning patterns across devices.

The main effect of block was statistically significant for movement time and throughput, but not for error rate. The reverse pattern emerged for the block × selection technique interaction, which was significant for error rate, but not for movement time or throughput. These patterns are best illustrated through figures (see Figure 8).

The pattern in all three parts of Figure 8 looks favorable for the tactile selection condition. The improvement in performance is clearly seen in each figure, and it is most dramatic from block 2 to block 3 (although the block × interaction technique effect was not statistically significant). With continued practice, the tactile condition is likely to improve. On error rate — the only measure on which the tactile condition faired poorly — it might even "catch up", although this could only be determined in a prolonged study.

Outliers

Since the error rates were somewhat high, we decided to investigate further. We identified a category of response called "wrong-side outliers". These are selections that occurred on the wrong side of the display. For example, if the goal was to select the target on one side of the display and the selection occurred before the pointer was halfway to the target, the selection was on the wrong side of the display. This is a gross error. We call these "outliers" because they are outside the normal range of variations expected in participants' behavior. A wrong-side outlier can occur for several reasons, such as double-clicking on a target or inadvertent lifting or pressing with the finger during pointer motion.

Overall, button selection had the fewest wrong-side outliers (178, 2.75%), followed by tactile (245, 3.78%) and lift-and-tap (253, 3.90%). Comparing the percentages with the overall error rates given earlier, we see that wrong-side outliers, formed a significant portion of the overall errors.

The number of wrong-side outliers, by selection technique and block is shown in Figure 9.

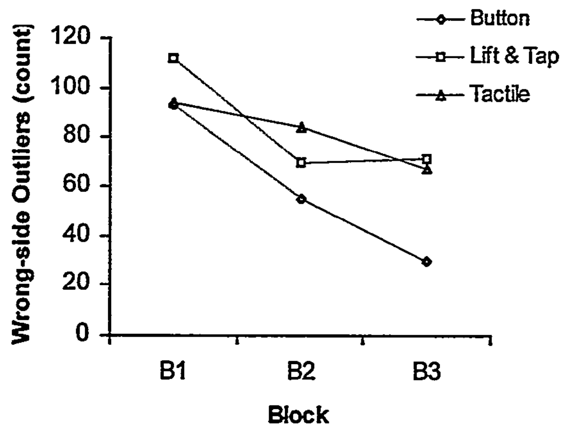


Figure 9. Wrong-side outliers by block and selection technique

The good showing of the button technique is likely due to the clear separation of pointer movement from target selection. Since movement and selection are more integrated with the lift-and-tap and tactile conditions, higher rates for wrong-side outliers are expected.

Questionnaire

At the end of the experiment, participants were given a questionnaire. For each selection technique, they were asked to provide a rating on their speed perception, their accuracy perception, and their overall preference. They entered a score from 1 (slowest, least accurate, liked the least) to 4 (quickest, most accurate, liked the most). The results are shown in Figure 10. Each cell is the total score for twelve participants, with higher scores preferred.

Selection Technique	Speed Perception	Accuracy Perception	Overall Preference
Button	3	15	5
Lift-and-tap	15	13	15
Tactile	19	15	17

Figure 10. Questionnaire results. (Note: Scores are totals of participants' ratings; higher scores are better.)

Participants liked the tactile selection technique. (This was evident in their comments, as well.) Tactile selection ranked 1st for speed perception, 1st (tied) for accuracy perception, and 1st for overall preference. It is noteworthy that on accuracy participants rated the tactile condition equal to, or better than, the other conditions even though it had the highest error rate. This could be due to the higher measures for throughput, which reflect the overall ability of participants to complete their tasks.

The Potential for a Tactile Touchpad

The Synaptics touchpad's method of deriving pressure data is indirect since it senses the capacitance between the finger and the pad. Pressure is derived from the area of the user's finger contacting the surface of the pad. Since one's finger flattens on the pad with increased pressure, the device takes advantage of this correlation. As a consequence, users with small fingers must press harder than users with large fingers. Participants with particularly large fingers required a more delicate touch than they preferred. This may account for the increased error rate of the tactile touchpad condition.

A better version of our touchpad would use true pressure-sensing technology, and such products are now available (e.g., the *VersaPad* by Interlink Electronics). A future replication of this experiment utilizing a calibration procedure at the onset would also be interesting, although this is generally not considered acceptable as a required procedure in commercial pointing devices.

Another noticeable artifact of the tactile touchpad condition was a tendency for the on-screen pointer to move down slightly as the subject pressed down to select a target. This was most pronounced with participants who held their pointing finger relatively perpendicular to the touchpad's surface. When they pressed down, the center of the finger's surface area moved towards the bottom and the onscreen pointer "dipped" with each press. As the targets were long and vertical, this most likely did not have an effect in the experiment; however, it is noteworthy. One subject suggested that the pointer freeze at a certain pressure level prior to a button press registering so that the results would be more predictable. Another possible solution would be to correct for the downward dips as the user pressed on the pad through software. That is, as the "pressure" increased, the pointer's vertical value might be slightly increased to compensate for the user's tendency to move the pointer downwards.

Our prototype's mechanical design was not of the highest quality. The relay was bulky and it was wedged-in against the bottom surface of the pad's PC board. A better design may assist in reducing error rates.

For all three selection techniques, the measures for throughput were low — lower than those typically found with trackballs or mice, for example. This begs the question, why would one choose a touchpad over a trackball or mouse? Besides personal preferences, we have no definitive answer to offer. A follow-up study with experienced touchpad users, or conducted over a prolonged period of time, might shed light on this; it would help answer the question, *can* a touchpad be as good as a other pointing devices (using throughput as the criterion)?

CONCLUSION

Although touchpads are not likely to supplant mice on the desktop, our results have implications for portable computer usage, and further refinements may make the tactile touchpad closer to a mouse in performance.

The tactile touchpad was found superior to both the lift-and-tap mode touchpad and button mode touchpad in terms of movement time and throughput. Although the error rate was higher than with the other touchpad conditions, it was not generally noticed by the participants and the overall flow of information (viz., throughput) was higher even with the increased error rate. With design improvements, the use of embedded tactile feedback in a touchpad can facilitate simple interactions such as pointing and selecting.

ACKNOWLEDGEMENTS

We thank Joe Decker of Synaptics for providing the touchpads and technical documentation for our prototype. Helpful comments and suggestions were provided by members of the Input Research Group at the University of Toronto and the University of Guelph. These are greatly appreciated. This research is funded by NSERC of Canada.

REFERENCES

1. Akamatsu, A., and MacKenzie, I. S. Movement characteristics using a mouse with tactile and force feedback, *International Journal of Human-Computer Studies 45* (1996), 483-493.
2. Balakrishnan, R., and MacKenzie, I. S. Performance differences in the fingers, wrist, and forearm in computer input control, In *Proceedings of the CHI '97 Conference on Human Factors in Computing Systems.* New York: ACM, 1997, pp. 303-310.
3. Boritz, J., Booth, K. S., and Cowan, W. B. Fitts's law studies of directional mouse movement, In *Proceedings of Graphics Interface '91.* Toronto: Canadian Information Processing Society, 1991, pp. 216-223.
4. Buxton, W. A. S. A three-state model of graphical input, In *Proceedings of INTERACT '90.* Amsterdam: Elsevier Science, 1990, pp. 449-456.
5. Fitts, P. M. The information capacity of the human motor system in controlling the amplitude of movement, *Journal of Experimental Psychology 47* (1954), 381-391.
6. Gillan, D. J., Holden, K., Adam, S., Rudisill, M., and Magee, L. How does Fitts' law fit pointing and dragging? In *Proceedings of the CHI '90 Conference on Human Factors in Computing Systems.* New York: ACM, 1990, pp. 227-234.
7. Gillan, D. J., Holden, K., Adams, S., Rudisill, M., and Magee, L. How should Fitts' law be applied to human-computer interaction? *Interacting with Computers 4*.3 (1992), 291-313.
8. ISO. *Ergonomic requirements for office work with visual display terminals (VDTs) - Part 9 - Requirements for non-keyboards input devices*, International Organisation for Standardisation, 1997.
9. MacKenzie, I. S. Fitts' law as a research and design tool in human-computer interaction, *Human-Computer Interaction 7* (1992), 91-139.
10. MacKenzie, I. S. Movement time prediction in human-computer interfaces, In *Proceedings of Graphics Interface '92.* Toronto: Canadian Information Processing Society, 1992, pp. 140-150.
11. MacKenzie, I. S., and Oniszczak, A. The tactile touchpad, In *Extended Abstracts of the CHI '97 Conference on Human Factors in Computing Systems.* New York: ACM, 1997, pp. 309-310.
12. MacKenzie, I. S., Sellen, A., and Buxton, W. A comparison of input devices in elemental pointing and dragging tasks, In *Proceedings of the CHI '91 Conference on Human Factors in Computing Systems.* New York: ACM, 1991, pp. 161-166.
13. MacKenzie, I. S., and Ware, C. Lag as a determinant of human performance in interactive systems, In *Proceedings of the INTERCHI '93 Conference on Human Factors in Computing Systems.* New York: ACM, 1993, pp. 488-493.
14. Martin, D. W. *Doing psychology experiments*, 4th ed. (Pacific Grove, CA: Brooks/Cole, 1996).
15. Soukoreff, W., and MacKenzie, I. S. Generalized Fitts' law model builder, In *Companion Proceedings of the CHI '95 Conference on Human Factors in Computing Systems.* New York: ACM, 1995, pp. 113-114.
16. Welford, A. T. *Fundamentals of skill*, (London: Methuen, 1968).