

Issues and Techniques in Touch-Sensitive Tablet Input

William Buxton
Ralph Hill
Peter Rowley

Computer Systems Research Institute
University of Toronto
Toronto, Ontario
Canada M5S 1A4

(416) 978-6320

Abstract

Touch-sensitive tablets and their use in human-computer interaction are discussed. It is shown that such devices have some important properties that differentiate them from other input devices (such as mice and joysticks). The analysis serves two purposes: (1) it sheds light on touch tablets, and (2) it demonstrates how other devices might be approached. Three specific distinctions between touch tablets and one button mice are drawn. These concern the signaling of events, multiple point sensing and the use of templates. These distinctions are reinforced, and possible uses of touch tablets are illustrated, in an example application. Potential enhancements to touch tablets and other input devices are discussed, as are some inherent problems. The paper concludes with recommendations for future work.

CR Categories and Subject Descriptors: I.3.1 [Computer Graphics]: Hardware Architecture: Input Devices. I.3.6 [Computer Graphics]: Methodology and Techniques: Device Independence, Ergonomics, Interaction Techniques.

General Terms: Design, Human Factors.

Additional Keywords and Phrases: touch sensitive input devices.

Permission to copy without fee all or part of this material is granted provided that the copies are not made or distributed for direct commercial advantage, the ACM copyright notice and the title of the publication and its date appear, and notice is given that copying is by permission of the Association for Computing Machinery. To copy otherwise, or to republish, requires a fee and/or specific permission.

1. Introduction

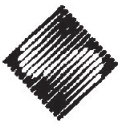
Increasingly, research in human-computer interaction is focusing on problems of input [Foley, Wallace & Chan 1984; Buxton 1983; Buxton 1985]. Much of this attention is directed towards input technologies. The ubiquitous Sholes keyboard is being replaced and/or complemented by alternative technologies. For example, a major focus of the marketing strategy for two recent personal computers, the Apple Macintosh and Hewlett-Packard 150, has been on the input devices that they employ (the mouse and touch-screen, respectively).

Now that the range of available devices is expanding, how does one select the best technology for a particular application? And once a technology is chosen, how can it be used most effectively? These questions are important, for as Buxton [1983] has argued, the ways in which the user *physically* interacts with an input device have a marked effect on the type of user interface that can be effectively supported.

In the general sense, the objective of this paper is to help in the selection process and assist in effective use of a specific class of devices. Our approach is to investigate a specific class of devices: touch-sensitive tablets. We will identify touch tablets, enumerate their important properties, and compare them to a more common input device, the mouse. We then go on to give examples of transactions where touch tablets can be used effectively. There are two intended benefits for this approach. First, the reader will acquire an understanding of touch tablet issues. Second, the reader will have a concrete example of how the technology can be investigated, and can utilize the approach as a model for investigating other classes of devices.

2. Touch-Sensitive Tablets

A touch-sensitive tablet (touch tablet for short) is a flat surface, usually mounted horizontally or nearly horizontally, that can sense the location of a finger pressing on it. That is, it is a tablet that can sense that it is being touched, and where it is being



touched. Touch tablets can vary greatly in size, from a few inches on a side to several feet on a side. The most critical requirement is that the user is not required point with some manually held device such as a stylus or puck.

What we have described in the previous paragraph is a *simple* touch tablet. Only one point of contact is sensed, and then only in a binary, touch/no touch, mode. One way to extend the potential of a simple touch tablet is to sense the degree, or pressure, of contact. Another is to sense multiple points of contact. In this case, the location (and possibly pressure) of several points of contact would be reported. Most tablets currently on the market are of the "simple" variety. However, Lee, Buxton and Smith [1985], and Nakatani [private communication] have developed prototypes of multi-touch, multi-pressure sensing tablets.

We wish to stress that we will restrict our discussion of touch technologies to touch tablets, which can and should be used in ways that are different from touch screens. Readers interested in touch-screen technology are referred to Herot & Weinsapfel [1978], Nakatani & Rohrlich [1983] and Minsky [1984]. We acknowledge that a flat touch screen mounted horizontally is a touch tablet as defined above. This is not a contradiction, as a touch screen has exactly the properties of touch tablets we describe below, as long as there is no attempt to mount a display below (or behind) it or to make it the center of the user's visual focus.

Some sources of touch tablets are listed in Appendix A.

3. Properties of Touch-Sensitive Tablets

Asking "Which input device is best?" is much like asking "How long should a piece of string be?" The answer to both is: it depends on what you want to use it for. With input devices, however, we are limited in our understanding of the relationship between device properties and the demands of a specific application. We will investigate touch tablets from the perspective of improving our understanding of this relationship. Our claim is that other technologies warrant similar, or even more detailed, investigation.

Touch tablets have a number of properties that distinguish them from other devices:

- They have no mechanical intermediate device (such as stylus or puck). Hence they are useful in hostile environments (e.g., classrooms, public access terminals) where such intermediate devices can get lost, stolen, or damaged.
- Having no puck to slide or get bumped, the tracking symbol "stays put" once placed, thus making them well suited for pointing tasks in environments subject to vibration or motion (e.g., factories, cockpits).
- They present no mechanical or kinesthetic restrictions on our ability to indicate more than one point at a time. That is, we can use two hands or more than one finger simultaneously on a single tablet. (Remember, we can manually control at

most two mice at a time: one in each hand. Given that we have ten fingers, it is conceivable that we may wish to indicate more than two points simultaneously. An example of such an application appears below).

- Unlike joysticks and trackballs, they have a very low profile and can be integrated into other equipment such as desks and low-profile keyboards (e.g., the Key Tronic Touch Pad, see Appendix A). This has potential benefits in portable systems, and, according to the Keystroke model of Card, Newell and Moran [1980], reduces homing time from the keyboard to the pointing device.
- They can be molded into one-piece constructions thus eliminating cracks and grooves where dirt can collect. This makes them well suited for very clean environments (eg. hospitals) or very dirty ones (eg., factories).
- Their simple construction, with no moving parts, leads to reliable and long-lived operation, making them suitable for environments where they will be subjected to intense use or where reliability is critical.

They do, of course, have some inherent disadvantages, which will be discussed at the close of the paper.

In the next section we will make three important distinctions between touch tablets and mice. These are:

- Mice and touch tablets vary in the number and types of events that they can transmit. The difference is especially pronounced when comparing to simple touch tablets.
- Touch tablets can be made that can sense multiple points of contact. There is no analogous property for mice.
- The surface of a tablet can be partitioned into regions representing a collection of independent "virtual" devices. This is analogous to the partitioning of a screen into "windows" or virtual displays. Mice, and other devices that transmit "relative change" information, do not lend themselves to this mode of interaction without consuming display real estate for visual feedback. With conventional tablets and touch tablets, graphical, physical or virtual templates can be placed over the input device to delimit regions. This allows valuable screen real estate to be preserved. Physical templates, when combined with touch sensing, permit the operator to sense the regions without diverting the eyes from the primary display during visually demanding tasks.

After these properties are discussed, a simple finger painting program is used to illustrate them in the context of a concrete example. We wish to stress that we do not pretend that the program represents a viable paint program or an optimal interface. It is simply a vehicle to illustrate a variety of transactions in an easily understandable context.

Finally, we discuss improvements that must be made to current touch tablet technology, many of which we have demonstrated in prototype form. Also, we suggest potential improvements to other devices, motivated by our experience with touch technology.

4. Three Distinctions Between Touch Tablets and Mice¹

The distinctions we make in this section have to do with suitability of devices for certain tasks or use in certain configurations. We are only interested in showing that there are some uses for which touch tablets are not suitable, but other devices are, and vice versa. We make no quantitative claims or comparisons regarding performance.

Signaling

Consider a rubber-band line drawing task with a one button mouse. The user would first position the tracking symbol at the desired starting point of the line by moving the mouse with the button released. The button would then be depressed, to signal the start of the line, and the user would manipulate the line by moving the mouse until the desired length and orientation was achieved. The completion of the line could then be signaled by releasing the button.²

Figure 1 is a state diagram that represents this interface. Notice that the button press and release are used to signal the beginning and end of the rubber-band drawing task. Also note that in states 1 and 2 both motion and signaling (by pressing or releasing the button, as appropriate) are possible.

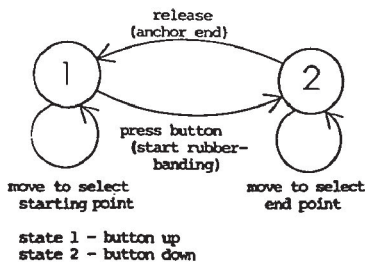


Figure 1. State diagram for rubber-banding with a one-button mouse.

Now consider a simple touch tablet. It can be used to position the tracking symbol at the starting point of the line, but it cannot generate the signal needed to initiate rubber-banding. Figure 2 is a state diagram representation of the capabilities of a simple touch tablet. In state 0, there is no contact with the tablet.³ In this state only one action is possible:

the user may touch the tablet. This causes a change to state 1. In state 1, the user is pressing on the tablet, and as a consequence position reports are sent to the host. There is no way to signal a change to some other state, other than to release (assuming the exclusion of temporal or spatial cues, which tend to be clumsy and difficult to learn). This returns the system to state 0. This signal could not be used to initiate rubber-banding, as it could also mean that the user is pausing to think, or wishes to initiate some other activity.

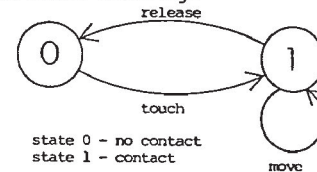


Figure 2. Diagram for showing states of simple touch-tablet.

This inability to signal while pointing is a severe limitation with current touch tablets, that is, tablets that do not report pressure in addition to location. (It is also a property of trackballs, and joysticks without "fire" buttons). It renders them unsuitable for use in many common interaction techniques for which mice are well adapted (e.g., selecting and dragging objects into position, rubber-band line drawing, and pop-up menu selection); techniques that are especially characteristic of interfaces based on *Direct Manipulation* [Shneiderman 1983].

One solution to the problem is to use a separate function button on the keyboard. However, this usually means two-handed input where one could do, or, awkward co-ordination in controlling the button and pointing device with a single hand. An alternative solution when using a touch tablet is to provide some level of pressure sensing. For example, if the tablet could report two levels of contact pressure (i.e., hard and soft), then the transition from soft to hard pressure, and vice versa, could be used for signaling. In effect, pressing hard is equivalent to pressing the button on the mouse. The state diagram showing the rubber-band line drawing task with this form of touch tablet is shown in Figure 3.⁴

As an aside, using this pressure sensing scheme would permit us to select options from a menu, or

mice. With conventional tablets, this corresponds to "out of range" state.

At this point the alert reader will wonder about difficulty in distinguishing between hard and soft pressure, and friction (especially when pressing hard). Taking the last first, hard is a relative term. In practice friction need not be a problem (see Inherent Problems, below).

⁴ One would conjecture that in the absence of button clicks or other feedback, pressure would be difficult to regulate accurately. We have found two levels of pressure to be easily distinguished, but this is a ripe area for research. For example, Stu Card [private communication] has suggested that the threshold between soft and hard should be reduced (become "softer") while hard pressure is being maintained. This suggestion, and others, warrant formal experimentation.

¹ Although we are comparing touch tablets to one button mice throughout this section, most of the comments apply equally to tablets with one-button pucks or (with some caveats) tablets with styli.

² This assumes that the interface is designed so that the button is held down during drawing. Alternatively, the button can be released during drawing, and pressed again, to signal the completion of the line.

³ We use state 0 to represent a state in which no location information is transmitted. There no analogous state for mice, and hence no state 0 in the diagrams for

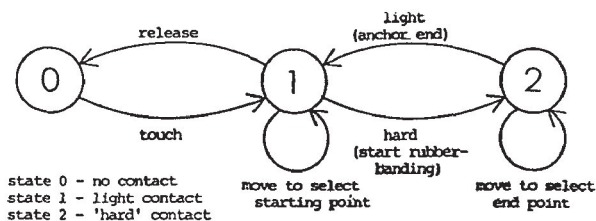


Figure 3. State diagram for rubber-banding with pressure sensing touch tablet.

activate light buttons by positioning the tracking symbol over the item and "pushing". This is consistent with the gesture used with a mouse, and the model of "pushing" buttons. With current simple touch tablets, one does just the opposite: position over the item and then lift off, or "pull" the button.

From the perspective of the signals sent to the host computer, this touch tablet is capable of duplicating the behaviour of a one-button mouse. This is not to say that these devices are equivalent or interchangeable. They are not. They are physically and kinesthetically very different, and should be used in ways that make use of the unique properties of each. Furthermore, such a touch tablet can generate one pair of signals that the one-button mouse cannot — specifically, press and release (transition to and from state 0 in the above diagrams). These signals (which are also available with many conventional tablets) are very useful in implementing certain types of transactions, such as those based on character recognition.

An obvious extension of the pressure sensing concept is to allow continuous pressure sensing. That is, pressure sensing where some large number of different levels of pressure may be reported. This extends the capability of the touch tablet beyond that of a traditional one button mouse. An example of the use of this feature is presented below.

Multiple Position Sensing

With a traditional mouse or tablet, only one position can be reported per device. One can imagine using two mice or possibly two transducers on a tablet, but this increases costs, and two is the practical limit on the number of mice or tablets that can be operated by a single user (without using feet). However, while we have only two hands, we have ten fingers. As playing the piano illustrates, there are some contexts where we might want to use several, or even all of them, at once.

Touch tablets need not restrict us in this regard. Given a large enough surface of the appropriate technology, one could use all fingers of both hands simultaneously, thus providing ten separate units of input. Clearly, this is well beyond the demands of many applications and the capacity of many people, however, there are exceptions. Examples include chording on buttons or switches, operating a set of slide potentiometers, and simple key roll-over when touch typing. One example (using a set of slide potentiometers) will be illustrated below.

Multiple Virtual Devices and Templates

The power of modern graphics displays has been enhanced by partitioning one physical display into a number of virtual displays. To support this, display window managers have been developed. We claim (see Brown, Buxton and Murtagh [1985]) that similar benefits can be gained by developing an input window manager that permits a single physical input device to be partitioned into a number of virtual input devices. Furthermore, we claim that multi-touch tablets are well suited to supporting this approach.

Figure 4a shows a thick cardboard sheet that has holes cut in specific places. When it is placed over a touch tablet as shown in Figure 4b, the user is restricted to touching only certain parts of the tablet. More importantly, the user can *feel* the parts that are touchable, and their shape. Each of the "touchable" regions represents a separate virtual device. The distinction between this template and traditional tablet mounted menus (such as seen in many CAD systems) is important.

Traditionally, the options have been:

- Save display real estate by mounting the menu on the tablet surface. The cost of this option is eye diversion from the display to the tablet, the inability to "touch type", and time consuming menu changes.
- Avoid eye diversion by placing the menus on the display. This also make it easier to change menus, but still does not allow "touch typing", and consumes display space.

Touch tablets allow a new option:

- Save display space and avoid eye diversion by using templates that can be felt, and hence, allow "touch typing" on a variety of virtual input devices. The cost of this option is time consuming menu (template) changes.

It must be remembered that for each of these options, there is an application for which it is best. We have contributed a new option, which makes possible new interfaces. The new possibilities include more elaborate virtual devices because the improved kinesthetic feedback allows the user to concentrate on providing input, instead of staying in the assigned region. We will also show (below) that its main cost (time consuming menu changes) can be reduced in some applications by eliminating the templates.

5. Examples of Transactions Where Touch Tablets Can Be Used Effectively

In order to reinforce the distinctions discussed in the previous section, and to demonstrate the use of touch tablets, we will now work through some examples based on a toy paint system. We wish to stress again that we make no claims about the quality of the example as a paint system. A paint system is a common and easily understood application, and thus, we have chosen to use it simply as a vehicle for discussing interaction techniques that use touch tablets.

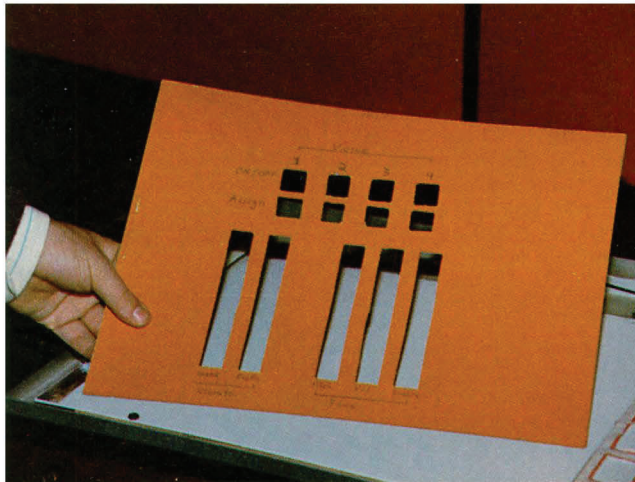


Figure 4a. Sample template.



Figure 5. Main display for paint program.

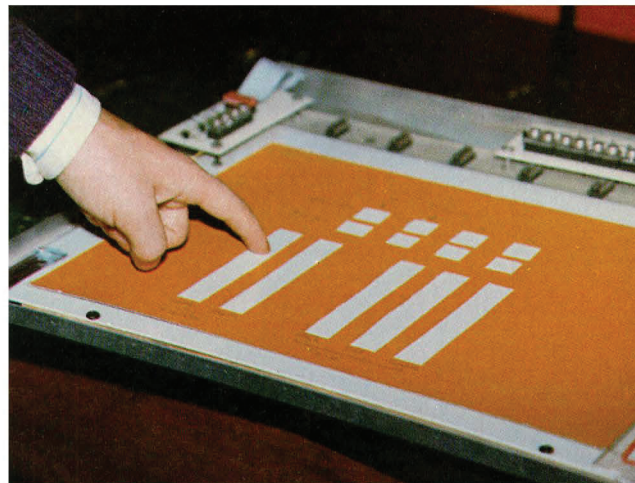


Figure 4b. Sample template in use.

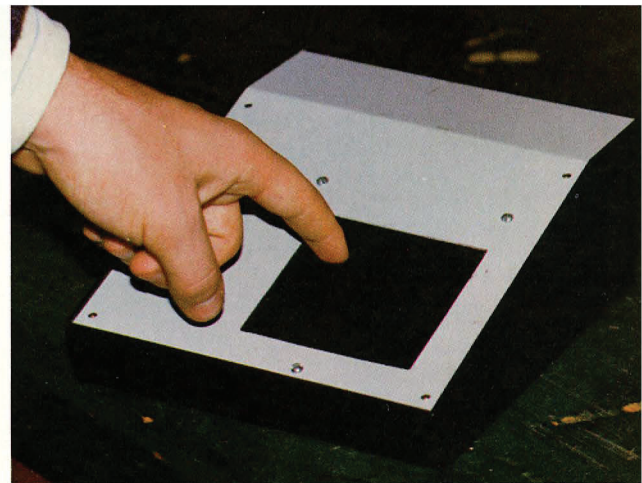


Figure 6. Touch tablet used in demonstrations.

The example paint program allows the creation of simple finger paintings. The layout of the main display for the program is shown in Figure 5. On the left is a large drawing area where the user can draw simple free-hand figures. On the right is a set of menu items. When the lowest item is selected, the user enters a colour mixing mode. In switching to this mode, the user is presented with a different display that is discussed below. The remaining menu items are "paint pots". They are used to select the colour that the user will be painting with.

In each of the following versions of the program, the input requirements are slightly different. In all cases an 8 cm x 8 cm touch tablet is used (Figure 6), but the pressure sensing requirements vary. These are noted in each demonstration.

5.1. Painting Without Pressure Sensing

This version of the paint program illustrates the limitation of having no pressure sensing. Consider

the paint program described above, where the only input device is a touch tablet without pressure sensing. Menu selections could be made by pressing down somewhere in the menu area, moving the tracking symbol to the desired menu item and then selecting by releasing. To paint, the user would simply press down in the drawing area and move (see Figure 7 for a representation of the signals used for painting with this program).

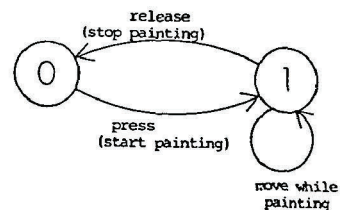


Figure 7. State diagram for drawing portion of simple paint program.

There are several problems with this program. The most obvious is in trying to do detailed drawings. The user does not know where the paint will appear until it appears. This is likely to be too late. Some form of feedback, that shows the user where the brush is, without painting, is needed. Unfortunately, this cannot be done with this input device, as it is not possible to signal the change from tracking to painting and vice versa.

The simplest solution to this problem is to use a button (e.g., a function key on the keyboard) to signal state changes. The problem with this solution is the need to use two hands on two different devices to do one task. This is awkward and requires practice to develop the co-ordination needed to make small rapid strokes in the painting. It is also inefficient in its use of two hands where one could (and normally should) do.

Alternatively, approaches using multiple taps or timing cues for signalling could be tried, however, we have found that these invariably lead to other problems. It is better to find a direct solution using the properties of the device itself.

5.2. Painting with Two Levels of Pressure

This version of the program uses a tablet that reports two levels of contact pressure to provide a satisfactory solution to the signaling problem. A low pressure level (a light touch by the user) is used for general tracking. A heavier touch is used to make menu selections, or to enable painting (see Figure 8 for the tablet states used to control painting with this program). The two levels of contact pressure allow us to make a simple but practical one finger paint program.

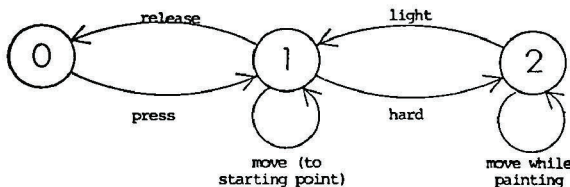


Figure 8. State diagram for painting portion of simple paint program using pressure sensing touch tablet.

This version is very much like using the one button mouse on the Apple Macintosh with MacPaint [Williams, 1984]. Thus, a simple touch tablet is not very useful, but one that reports two levels of pressure is similar in power (but not feel or applicability) to a one button mouse.⁵

5.3. Painting with Continuous Pressure Sensing

In the previous demonstrations, we have only implemented interaction techniques that are common using existing technology. We now introduce a technique that provides functionality beyond that obtainable using most conventional input technolo-

⁵ Also, there is the problem of friction, to be discussed below under "Inherent Problems".

gies.

In this technique, we utilize a tablet capable of sensing a continuous range of touch pressure. With this additional signal, the user can control both the width of the paint trail and its path, using only one finger. The new signal, pressure, is used to control width. This is a technique that cannot be used with any mouse that we are aware of, and to our knowledge, is available on only one conventional tablet (the GTCO Digipad with pressure pen [GTCO 1982]).

We have found that using current pressure sensing tablets, the user can accurately supply two to three bits of pressure information, after about 15 minutes practice. This is sufficient for simple doodling and many other applications, but improved pressure resolution is required for high quality painting.

5.4. "Windows" on the Tablet: Colour Selection

We now demonstrate how the surface of the touch tablet can be *dynamically* partitioned into "windows" onto virtual input devices. We use the same basic techniques as discussed under templates (above), but show how to use them without templates. We do this in the context of a colour selection module for our paint program. This module introduces a new display, shown in Figure 9.

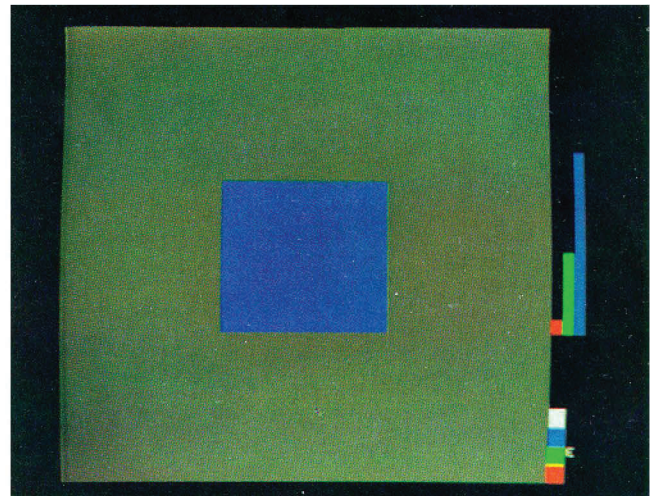


Figure 9. Colour-mixing display.

In this display, the large left side consists of a colour patch surrounded by a neutral grey border. This is the patch of colour the user is working on. The right side of the display contains three bar graphs with two light buttons underneath. The primary function of the bar graphs is to provide feedback, representing relative proportions of red, green and blue in the colour patch. Along with the light buttons below, they also serve to remind the user of the current layout of the touch tablet.

In this module, the touch tablet is used as a "virtual operating console". Its layout is shown (to scale) in Figure 10. There are 3 valuator (corresponding to the bar graphs on the screen) used to control

colour, and two buttons: one, on the right, to bring up a pop-up menu used to select the colour to be modified, and another, on the left, to exit.

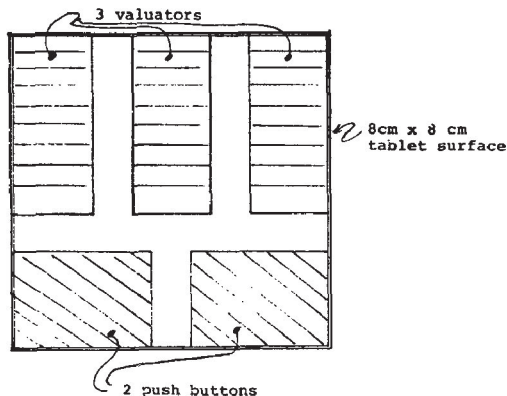


Figure 10. Layout of virtual devices on touch tablet.

The single most important point to be made in this example is that a single *physical* device is being used to implement 5 *virtual* devices (3 valuator and 2 buttons). This is analogous to the use of a display window system, in its goals, and its implementation.

The second main point is that there is nothing on the tablet to delimit the regions. This differs from the use of physical templates as previously discussed, and shows how, in the absence of the need for a physical template, we can instantly change the "windows" on the tablet, without sacrificing the ability to touch type.

We have found that when the tablet surface is small, and the partitioning of the surfaces is not too complex, the users very quickly (typically in one or two minutes) learn the positions of the virtual devices relative to the edges of the tablet. More importantly, they can use the virtual devices, practically error free, without diverting attention from the display. (We have repeatedly observed this behaviour in the use of an application that uses a 10 cm square tablet that is divided into 3 sliders with a single button across the top).

Because no template is needed, there is no need for the user to pause to change a template when entering the colour mixing module. Also, at no point is the user's attention diverted from the display. These advantages cannot be achieved with any other device we know of, without consuming display real estate.

The colour of the colour patch is manipulated by *dragging* the red, green and blue values up and down with the valuator on the touch tablet. The valuator are implemented in relative mode (i.e., they are sensitive to changes in position, not absolute position), and are manipulated like one dimensional mice. For example, to make the patch more red, the user presses near the left side of the tablet, about half way to the top, and slides the finger up (see Figure 11). For larger changes, the device can be repeatedly stroked (much like stroking a mouse). Feedback is provided by changing the level in the bar graph on the screen and the colour

of the patch.

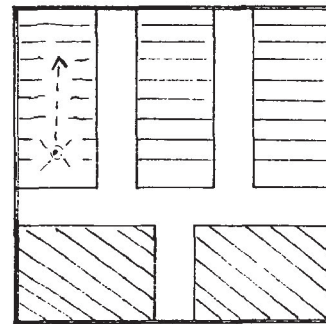


Figure 11. Increasing red content, by pressing on red valuator and sliding up.

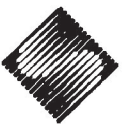
Using a mouse, the above interaction could be approximated by placing the tracking symbol over the bars of colour, and dragging them up or down. However, if the bars are narrow, this takes acuity and concentration that distracts attention from the primary task — monitoring the colour of the patch. Furthermore, note that the touch tablet implementation does not need the bars to be displayed at all, they are only a convenience to the user. There are interfaces where, in the interests of maximizing available display area, there will be no items on the display analogous to these bars. That is, there would be nothing on the display to support an interaction technique that allows values to be manipulated by a mouse.

Finally, we can take the example one step further by introducing the use of a touch tablet that can sense multiple points of contact (e.g., [Lee, et al. 1985]). With this technology, all three colour values could be changed at the same time (for example, fading to black by drawing all three sliders down together with three fingers of one hand). This simultaneous adjustment of colours could *not* be supported by a mouse, nor any single commercially available input device we know of. Controlling several valuator with one hand is common in many operating consoles, for example: studio light control, audio mixers, and throttles for multi-engine vehicles (e.g., aircraft and boats). Hence, this example demonstrates a cost effective method for providing functionality that is currently unavailable (or available only at great cost, in the form of a custom fabricated console), but has wide applicability.

5.5. Summary of Examples

Through these simple examples, we have demonstrated several things:

- The ability to sense at least two levels of pressure is a virtual necessity for touch tablets, as without it, auxiliary devices must be used for signaling, and "direct manipulation" interfaces cannot be effectively supported.
- The extension to continuous pressure sensing opens up new possibilities in human-computer interaction.



- Touch tablets are superior to mice and tablets when many simple devices are to be simulated. This is because: (a) there is no need for a mechanical intermediary between the fingers and the tablet surface, (b) they allow the use of templates (including the edges of the tablet, which is a trivial but useful template), and (c) there is no need for positional feedback that would consume valuable display space.
- The ability to sense multiple points of contact radically changes the way in which users may interact with the system. The concept of multiple points of contact does not exist for, nor is it applicable to, current commercially available mice and tablets.

6. Inherent Problems with Touch Tablets

A problem with touch tablets that is annoying in the long term is friction between the user's finger and the tablet surface. This can be a particularly severe problem if a pressure sensitive tablet is used, and the user must make long motions at high pressure. This problem can be alleviated by careful selection of materials and care in the fabrication and calibration of the tablet.⁶ Also, the user interface can be designed to avoid extended periods of high pressure.

Perhaps the most difficult problem is providing good feedback to the user when using touch tablets. For example, if a set of push-on/push-off buttons are being simulated, the traditional forms of feedback (illuminated buttons or different button heights) cannot be used. Also, buttons and other controls implemented on touch tablets lack the kinesthetic feel associated with real switches and knobs. As a result, users must be more attentive to visual and audio feedback, and interface designers must be freer in providing this feedback. (As an example of how this might be encouraged, the input "window manager" could automatically provide audible clicks as feedback for button presses).

7. Potential Enhancements to Touch Tablets (and other devices)

The first problem that one notices when using touch tablets is "jitter" when the finger is removed from the tablet. That is, the last few locations reported by the tablet, before it senses loss of contact, tend to be very unreliable.

This problem can be eliminated by modifying the firmware of the touch tablet controller so that it keeps a short FIFO queue of the samples that have most recently been sent to the host. When the user releases pressure, the oldest sample is retransmitted, and the queue is emptied. The length of the queue depends on the properties of the touch tablet (e.g., sensitivity, sampling rate). We have found that determining a suitable value requires

⁶ As a bad example, one commercial "touch" tablet requires so much pressure for reliable sensing that the finger cannot be smoothly dragged across the surface. Instead, a wooden or plastic stylus must be used, thus losing many of the advantages of touch sensing.

only a few minutes of experimentation.

A related problem with most current tablet controllers (not just touch tablets) is that they do not inform the host computer when the user has ceased pressing on the tablet (or moved the puck out of range). This information is essential to the development of certain types of interfaces. (As already mentioned, this signal is not available from mice). Currently, one is reduced to deducing this event by timing the interval between samples sent by the tablet. Since the tablet controller can easily determine when pressure is removed (and must if it is to apply a de-jittering algorithm as above), it should share this information with the host.

Clearly, pressure sensing is an area open to development. Two pressure sensitive tablets have been developed at the University of Toronto [Sasaki, et al. 1981; Lee, et al. 1985]. One has been used to develop several experimental interfaces and was found to be a very powerful tool. They have recently become available from Elographics and Big Briar (see Appendix A). Pressure sensing is not only for touch tablets. Mice, tablet pucks and styli could all benefit by augmenting switches with strain gauges, or other pressure sensing instruments. GTCO, for example, manufactures a stylus with a pressure sensing tip [GTCO 1982], and this, like our pressure sensing touch tablets, has proven very useful.

8. Conclusions

We have shown that there are environments for which some devices are better adapted than others. In particular, touch tablets have advantages in many hostile environments. For this reason, we suggest that there are environments and applications where touch tablets may be the most appropriate input technology.

This being the case, we have enumerated three major distinctions between touch tablets and one button mice (although similar distinctions exist for multi-button mice and conventional tablets). These assist in identifying environments and applications where touch tablets would be most appropriate. These distinctions concern:

- limitation in the ability to signal events,
- suitability for multiple point sensing, and
- the applicability of tactile templates.

These distinctions have been reinforced, and some suggestions on how touch tablets may be used have been given, by discussing a simple user interface. From this example, and the discussion of the distinctions, we have identified some enhancements that can be made to touch tablets and other input devices. The most important of these are pressure sensing and the ability to sense multiple points of contact.

We hope that this paper motivates interface designers to consider the use of touch tablets and shows some ways to use them effectively. Also, we hope it encourages designers and manufacturers of input devices to develop and market input devices with the enhancements that we have discussed.

The challenge for the future is to develop touch tablets that sense continuous pressure at multiple points of contact and incorporate them in practical interfaces. We believe that we have shown that this is worthwhile and have shown some practical ways to use touch tablets. However, interface designers must still do a great deal of work to determine where a mouse is better than a touch tablet and vice versa.

Finally, we have illustrated, by example, an approach to the study of input devices, summarized by the credo: "Know the interactions a device is intended to participate in, and the strengths and weaknesses of the device." This approach stresses that there is no such thing as a "good input device," only good interaction task/device combinations.

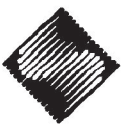
9. Acknowledgements

The support of this research by the Natural Sciences and Engineering Research Council of Canada is gratefully acknowledged. We are indebted to Kevin Murtagh and Ed Brown for their work on virtual input devices and windowing on input. Also, we are indebted to Elographics Corporation for having supplied us with the hardware on which some of the underlying studies are based.

We would like to thank the referees who provided many useful comments that have helped us with the presentation.

10. References

- Brown, E.
Buxton, W.
Murtagh, K.
1985
- Windows on Tablets as a Means of Achieving Virtual Input Devices.** Submitted for publication.
- Buxton, W.
1983
- Lexical and Pragmatic Considerations of Input Structures.** *Computer Graphics* 17.1. Presented at the SIGGRAPH Workshop on Graphical Input Techniques, Seattle, Washington, June 1982.
- Buxton, W.
1985
- There is More to Interaction Than Meets the Eye: Some Issues in Manual Input.** (in) Norman, D.A. and Draper, S.W. (Eds.), *User Centered System Design: New Perspectives on Human-Computer Interaction*. Hillsdale, N.J.: Lawrence Erlbaum and Associates. Publication expected late 1985.
- Buxton, W.
Fiume, E.
Hill, R.
Lee, A.
Woo, C.
1983
- Continuous Hand-Gesture Driven Input.** *Proceedings Graphics Interface '83*: pp. 191-195. May 9-13, 1983, Edmonton, Alberta.
- Card, S.K.
Moran, T.P.
Newell, A.
Jul 1980
- The Keystroke-Level Model for User Performance Time with Interactive Systems.** *Communications of the ACM* 23.7: pp. 396-409.
- Foley, J.D.
Wallace, V.L.
Chan, P.
Nov 1984
- The Human Factors of Computer Graphics Interaction Techniques.** *IEEE Computer Graphics and Applications* 4.11: pp. 13-48.
- GTCO
1982
- DIGI-PAD 5 User's Manual.** GTCO Corporation, 1055 First Street, Rockville, MD 20850.
- Herot, C.F.
Weinzapfel, G.
Aug 1978
- One-Point Touch Input of Vector Information for Computer Displays.** *Computer Graphics* 12.3: pp. 210-216. SIGGRAPH'78 Conference Proceedings, August 23-25, 1978, Atlanta, Georgia.
- Lee, S.
Buxton, W.
Smith, K.C.
1985
- A Multi-Touch Three Dimensional Touch-Sensitive Tablet.** *Human Factors in Computer Systems*: pp. 21-25. (CHI'85 Conference Proceedings, April 14-18, 1985, San Francisco).
- Minsky, M.R.
Jul 1984
- Manipulating Simulated Objects with Real-world Gestures using a Force and Position Sensitive Screen.** *Computer Graphics* 18.3: pp. 195-203. (SIGGRAPH'84 Conference Proceedings, July 23-27, 1984, Minneapolis, Minnesota).
- Nakatani, L.H.
Rohrlich, J.A.
Dec 1983
- Soft Machines: A Philosophy of User-Computer Interface Design.** *Human Factors in Computing Systems*: pp. 19-23. (CHI'83 Conference Proceedings, December 12-15, 1983, Boston).
- Sasaki, L.
Fedorkow, G.
Buxton, W.
Retherath, C.
Smith, K.C.
1981
- A Touch Sensitive Input Device.** *Proceedings of the 5th International Conference on Computer Music*. North Texas State University, Denton Texas, November 1981.
- Shneiderman, B.
Aug 1983
- Direct Manipulation: A Step Beyond Programming Languages.** *Computer* 16.8: pp. 57-69.
- Williams, G.
Feb 1984
- The Apple Macintosh Computer.** *Byte* 9.2: pp. 30-54.
- Appendix A: Touch Tablet Sources**
- Big Briar: 3 by 3 inch continuous pressure sensing touch tablet
Big Briar, Inc.
Leicester, NC
28748
- Chalk Board Inc.: "Power Pad", large touch table for micro-computers
Chalk Board Inc.
3772 Pleasantdale Rd.,
Atlanta, GA 30340
- Elographics: various sizes of touch tablets, including pressure sensing
Elographics, Inc.
105 Randolph Toad
Oak Ridge, Tennessee
37830
(615)-482-4100



Key Tronic: Keyboard with touch pad.

Keytronic
P.O. Box 14687
Spokane, WA 99214
(509)-928-8000

KoalaPad Technologies: Approx. 5 by 7 inch touch tablet
for micro-computers

Koala Technologies
3100 Patrick Henry Drive
Santa Clara, California
95050

Spiral Systems: Trazor Touch Panel, 3 by 3 inch touch
tablet

Spiral System Instruments, Inc.
4853 Cordell Avenue, Suite A-10
Bethesda, Maryland
20814

TASA: 4 by 4 inch touch tablet (relative sensing only)

Touch Activated Switch Arrays Inc.
1270 Lawrence Stn. Road, Suite G
Sunnyvale, California
94089