

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

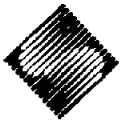
Increasingly, research in human-computer interaction is focusing on problems of input [Foley, Wallace & Chan 1984; Buxton 1983; Buxton 1985]. Much of this attention is directed towards input technologies. The ubiquitous Sholes keyboard is being replaced and/or complemented by alternative technologies. 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 & Rohrlach [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.

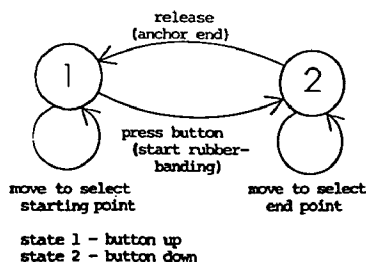


Figure 1. State diagram for rubber-banding with a one-button mouse.

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.

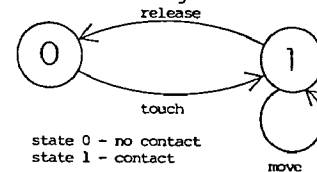


Figure 2. Diagram for showing states of simple touch-tablet.

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

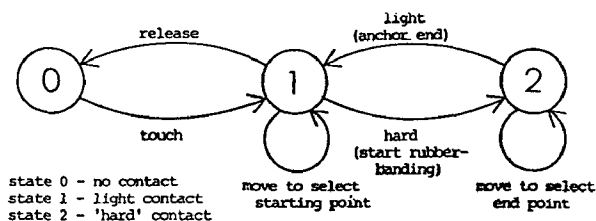


Figure 3. State diagram for rubber-banding with pressure sensing touch tablet.

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.

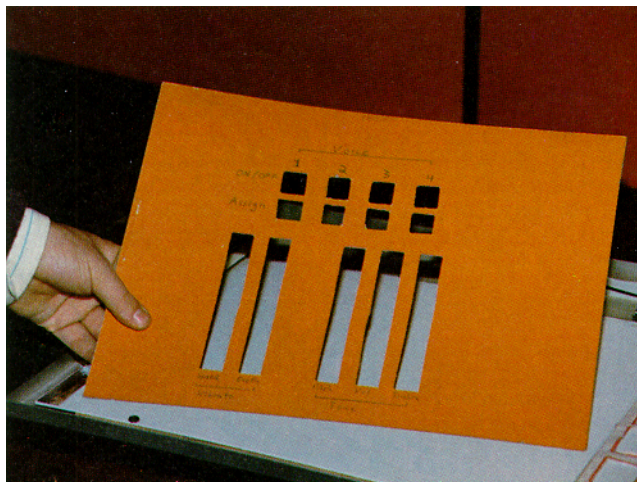


Figure 4a. Sample template.



Figure 5. Main display for paint program.

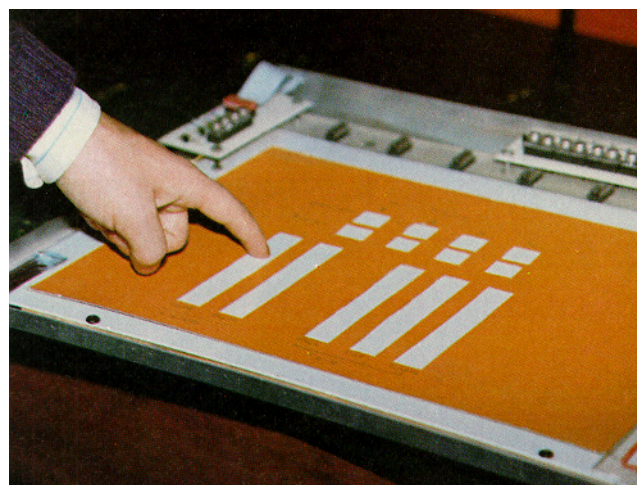


Figure 4b. Sample template in use.

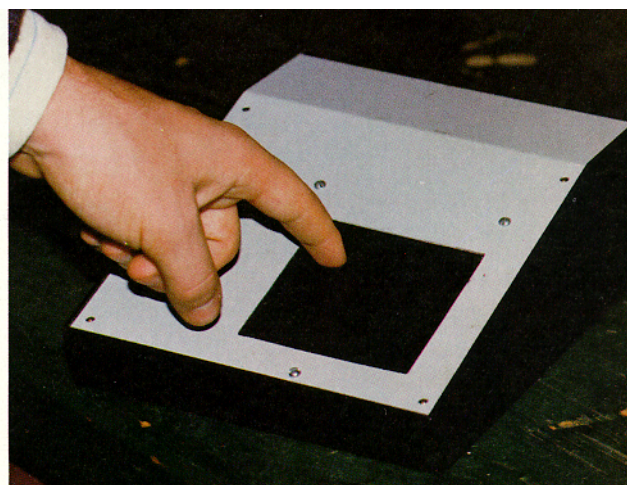


Figure 6. Touch tablet used in demonstrations.

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).

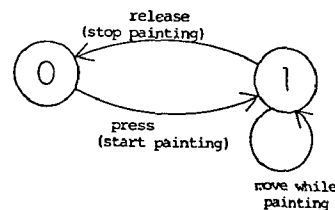


Figure 7. State diagram for drawing portion of simple paint program.

Explore Litigation Insights

Docket Alarm provides insights to develop a more informed litigation strategy and the peace of mind of knowing you're on top of things.

Real-Time Litigation Alerts



Keep your litigation team up-to-date with **real-time alerts** and advanced team management tools built for the enterprise, all while greatly reducing PACER spend.

Our comprehensive service means we can handle Federal, State, and Administrative courts across the country.

Advanced Docket Research



With over 230 million records, Docket Alarm's cloud-native docket research platform finds what other services can't. Coverage includes Federal, State, plus PTAB, TTAB, ITC and NLRB decisions, all in one place.

Identify arguments that have been successful in the past with full text, pinpoint searching. Link to case law cited within any court document via Fastcase.

Analytics At Your Fingertips



Learn what happened the last time a particular judge, opposing counsel or company faced cases similar to yours.

Advanced out-of-the-box PTAB and TTAB analytics are always at your fingertips.

API

Docket Alarm offers a powerful API (application programming interface) to developers that want to integrate case filings into their apps.

LAW FIRMS

Build custom dashboards for your attorneys and clients with live data direct from the court.

Automate many repetitive legal tasks like conflict checks, document management, and marketing.

FINANCIAL INSTITUTIONS

Litigation and bankruptcy checks for companies and debtors.

E-DISCOVERY AND LEGAL VENDORS

Sync your system to PACER to automate legal marketing.