An Extended Menu Navigation Interface Using Multiple Pressure-Sensitive Strips

Gábor Blaskó Steven Feiner
Department of Computer Science, Columbia University
New York, NY 10027
[gblasko, feiner]@cs.columbia.edu

Abstract

We present extensions and modifications that we have made to a cursorless menu navigation interface that is controlled by multiple pressure-sensitive linear strips. Our approach is based on detecting pressure thresholds and dual-finger motions, allowing us to overcome the physical limitations of the prototype input device and virtually more than triple the number of linear strips that can be used for menu navigation. This allows the user to directly access and navigate in up to fourteen independent multiple-depth menu trees. Our system allows this breadth and depth of navigation without the need for an on-screen cursor, nor the need to navigate a 2D input device, thereby reducing the need for visual feedback. We have tried to design the on-screen graphical interface so that it can be easily used with very small field-of-view display devices, such as eyeglass displays. Additionally, since our input device is physically compact and can be easily positioned on different places on the body, the menu navigation system is especially appropriate for wearable computing systems.

Introduction

In previous work [1], we introduced a method of controlling menus with multiple touch-sensitive linear strips. A user uses this multi-strip input device, shown in Figure 1(a), by placing each of the four fingers of one hand above a corresponding strip and simultaneously using all four fingers for interaction.

We have created a prototype of the multi-strip device with a Synaptics TouchPad [4] that has been programmed to treat the touch-sensitive surface as four independent linear strips in landscape mode. A hand has only four fingers with the same range of motion that can move up and down on a surface when placed side-by-side, while the thumb has a different range of motion. Therefore, there is a biomechanical limit to the number of linear strips that can be used simultaneously for control. Since our prototype is approximately four fingers wide, there is also a physical hardware limitation. Consequently, our original menu interface [1] allowed the user to directly access and navigate only four independent menu trees. In this work, we address these limitations and present methods that, without physical or electrical hardware modifications, allow the virtual use of up to fourteen linear strips. We also present additional improvements that we have made, based on preliminary user feedback we collected during informal tests with about 30 coworkers and lab visitors.

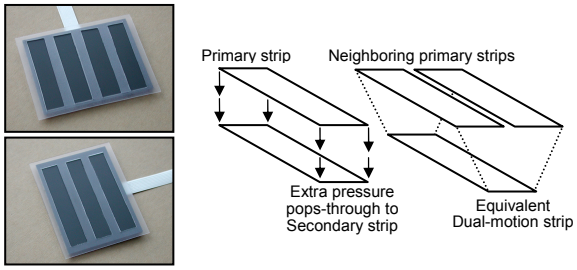

Figure 1a (left): Prototype in landscape and vertical orientation.
1b (right): Illustration of pop-through and dual-motion functionality.

Pressure-Based Extensions

The TouchPad datastream reports total finger capacitance value (*Z*) in addition to the horizontal *X* and vertical *Y* coordinates of the point of contact. This *Z* value is a function of the finger's contact area, which is affected by the finger's contact pressure and by the angle at which the finger is held [4]. As more pressure is applied, the contact area increases, producing higher *Z* values.

In the work of Zeleznik et al. [5] on "pop-through buttons", applying soft or firm pressure to two physically layered buttons allows a user to gain access to a more detailed version of a menu or a different underlying menu in a traditional mouse-controlled desktop graphical user interface (GUI). Inspired by this work, we have tried to explore the possibility of using pressure thresholds to extend the number of directly accessible menu trees.

We adapt the idea of "pop-through" menu expansion to the multi-strip based menu-navigation system and introduce *secondary menu trees,* which may be accessed by applying firm pressure to a given linear strip. We call the menus accessible with light pressure *primary menu trees.* Once a preset pressure threshold is exceeded, the user pops through to an underlying secondary menu tree. In informal tests, we have found that users quickly learn what the threshold pressure level is and directly access the needed menu with a relatively small error rate. Nonetheless, we think that if the pressure decreases after popping through, the user should not be allowed to step back to the primary menu tree, but should instead be required to break contact with the strip's surface and reset the menu system as explained below. Navigation among the

Figure 2: Sample screenshots and finger gestures used:
a, top row: primary and secondary menus for the leftmost linear strip.
b, middle row: primary and secondary menus for the middle dual-motion linear strip (middle two linear strips together).
c, bottom row: indication of depth with tabs for secondary menu of the rightmost single linear strip.

menu items happens as in [1], using a combination of two adjacent fingers: the finger corresponding to the menu tree highlights menu items, and the adjacent finger contacts the lower half of the adjacent strip to select the item or the upper half of the adjacent strip to step back in the menu hierarchy. By using this pop-through technique, the number of menu trees that are directly accessible from the base state can be doubled from four to eight.

Dual-Finger Extensions

Touch-sensitive devices that can detect multiple finger contacts and can be used for multi-finger gesturing include the Tactex MTC Express [2] and the Fingerworks iGesture Pad [3]. Unfortunately, these devices are too large for many wearable applications and their software APIs are not yet mature. Even though our compact prototype TouchPad reports only a single point of contact, fortunately it is possible to detect the motions of two neighboring fingers reliably to extend the number of linear strips. When two fingers contact the TouchPad simultaneously, the position of the centroid of the contact surface and a larger *Z* value are reported. Additionally, there is a flag in the data stream that is set if the contact area is very wide and, with high probability, two fingers are contacting the surface [4]. By monitoring the reported position, pressure and this flag bit, we can reliably detect dual-finger motions and therefore introduce *dual-motion linear strips* and *dual-motion menu trees*.

In the base state of the menu system, two neighboring fingers activate a dual-motion menu tree and both fingers, moving side-by-side, are used to highlight a menu item. A menu item is then selected by contacting the lower half of the right adjacent strip. Taking a step back is done by contacting the upper half of the right adjacent strip. After taking the first step into the dual-motion menu tree, it is possible to allow either the leftmost finger or both fingers to be used for selection.

At any point during menu navigation, it is possible to return to the base state by simultaneously placing three or more fingers on the pad at one, since this event can also be detected by monitoring another flag in the data-stream.

By pairing up fingers for dual-motion gestures, the number of primary menu trees may be increased from four to seven. In combination with the pop-through technique, this number may be doubled to fourteen, offering the user a wide range of directly accessible menu trees.

Further Improvements and Discussion

We have introduced physical partitions in the form of a thin plastic stencil overlaid on the touchpad as shown in Figure 1(a). This not only visually conveys that the device should be used as multiple independent strips, but also helps avoid erroneous and ambiguous input.

In our initial tests, multiple users had difficulty in directly highlighting a desired item from among more than four menu items, because precisely contacting the vertically rescaled targeted subportion of the strip became difficult. This can be explained with the short length (ca. 4cm) of the strips. When the pad is used vertically with a different stencil as illustrated in Figure 1(a), with fewer but longer strips, direct highlighting becomes easier if a strip needs to be divided into more than four subsections.

We have made minor changes to the GUI elements relative to our earlier work [1], as shown in Figure 2. We have moved the indicators for the "Select" and "Cancel" functionality to the inside of the menu selection field, represented by "S" and "C" virtual buttons. We have also tried to use more extensive variations in brightness and contrast to convey both depth and active/inactive state of on-screen elements. The GUI is monochromatic (shades of blue), so it can be easily used with monochrome displays.

Conclusions and Future Work

With the adaptation of the pop-through technique and dual-finger motions we extended a cursorless menu navigation interface, increasing the number of the directly accessible menu trees to fourteen. As we refine our prototype, we intend to design and run a formal user study to compare it with alternatives that target wearable platforms.

Acknowledgements

This research is funded in part by ONR Contracts N00014-99-1-0249, N00014-99-1-0394, and N00014-99-0683; NSF Grants IIS-00-82961 and IIS-01-21239; and gifts from Synaptics, Intel and Microsoft.

References

[1] G. Blaskó and S. Feiner. A Menu Interface for Wearable Computing *Proc. ISWC '02*, Seattle, WA, 2002, p.164–165
[2] http://www.tactex.com and CA. Pat. 2273113.
[3] http://www.fingerworks.com and US. Pat 6323846.
[4] Synaptics TouchPad Interfacing Guide. http://www.synaptics.com
[5] R. Zeleznik, T. Miller, and A. Forsberg. Pop through Mouse Button Interactions. *Proc. ACM Symp. On User Interfaces and Tech. (UIST '01)*, November, 2001, Orlando, FL, 195–196.