Augmenting Interactive Tables with Mice & Keyboards

Björn Hartmann[1], Meredith Ringel Morris[2], Hrvoje Benko[2], Andrew D. Wilson[2]
1 – Stanford University HCI Group
Computer Science Dept, Stanford, CA 94305
bjoern@cs.stanford.edu
2 – Microsoft Research
One Microsoft Way
{merrie, benko, awilson}@microsoft.com

ABSTRACT

This note examines the role traditional input devices can play in surface computing. Mice and keyboards can enhance tabletop technologies since they support high fidelity input, facilitate interaction with distant objects, and serve as a proxy for user identity and position. Interactive tabletops, in turn, can enhance the functionality of traditional input devices: they provide spatial sensing, augment devices with co-located visual content, and support connections among a plurality of devices. We introduce eight interaction techniques for a table with mice and keyboards, and we discuss the design space of such interactions.

ACM Classification: H5.2: User Interfaces – Input devices

General terms: Design, Human Factors

Keywords: Interactive tabletops, surface computing

INTRODUCTION

Interactive tabletops are a compelling platform offering new interaction methods for multi-user collaboration. Many initial tabletop applications have been limited to casual interactions of short duration, such as entertainment scenarios and photo browsing. As tabletop technologies mature, researchers are beginning to explore their value for professional productivity scenarios such as graphic design [6], office work [10], and intelligence analysis [8].

Certain known problems of direct touch input may limit this expansion into the productivity realm. Text entry on interactive surfaces is cumbersome [7, 10], and the sensing resolution of tabletop interfaces limits the precision and responsiveness of touch input [4, 5]. The use of styli [6] and custom physical devices [4, 14] to replace or augment touch input on tables have been proposed to address these issues. In this paper, we consider the design possibilities that arise by reintroducing keyboards and mice on interactive tabletops. This combination has been largely unexplored, perhaps because physical input devices tended to obscure much of the active display area of many early tabletops. However, more recent surfaces (*e.g.*, [4, 6]) are growing to match the size of traditional tables. These larger tabletops can accommodate multiple input devices, much as standard conference tables support multiple laptops. Occlusion-avoidance techniques [1, 3] can ensure that digital content is not unintentionally obscured by these devices.

Figure 1: Large surfaces enable use of keyboards & mice.

In exploring multi-user collaboration with a shared application through mice and keyboards, we build on prior research on using personal computing devices to interact with horizontal displays [11, 12] and augmenting physical input devices with graphic output [4].

OVERVIEW OF BENEFITS

Mice and keyboards can enhance the functionality of tabletops in three ways: they provide high precision, high performance input; they enable interaction with distant objects while minimizing physical movement; and the devices can serve as proxies for user identity and position on tables that are not able to automatically identify or locate their users.

Interactive tabletops can enhance the capabilities of mice and keyboards in several ways, as well: tabletops can provide absolute location and orientation sensing of devices; they can augment devices with co-located displayed information and additional “soft” inputs; and they can enable flexible association patterns among multiple devices.

The following scenario, enabled by the interaction techniques introduced in this note, illustrates how a combination of traditional input devices and tabletops can aid collaborative information work.

SCENARIO

College students Adam, Betty, and Carl meet in a group study space equipped with an interactive surface to prepare a class presentation on green technologies (Fig. 1). They each place their wireless keyboards on the tabletop. A login box displayed above each keyboard asks them to identify themselves to provide access to their files in the cloud.

The students independently begin collecting material by typing search queries (Fig. 2a). Soft configuration buttons displayed next to each keyboard allow them to search the web, search personal documents, or create a new document

UIST’09, October 4–7, 2008, Victoria, BC, Canada.

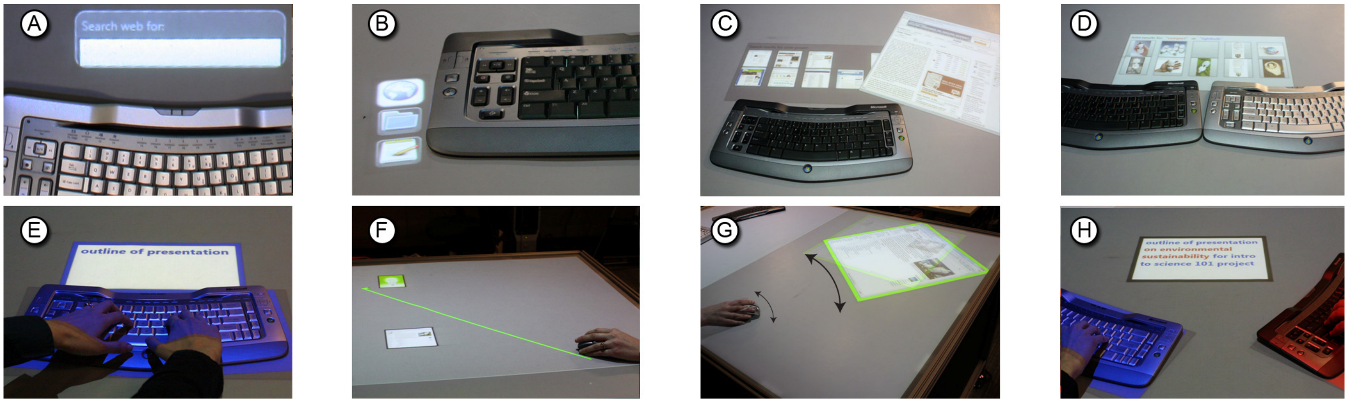

Figure 2: (a) entering a search query with a command prompt; (b) "soft" inputs select command modes; (c) command output tracks the originating keyboard; (d) conducting a joint Web search; (e) document docked to keyboard; (f) leader line locates cursor; (g) rotation of the mouse reorients the selected document; (h) projected light and colored text reveal authorship.

(Fig. 2b). Results for each query are displayed above the originating keyboard and track it when it moves (Fig. 2c).

Adam has trouble finding relevant images for energy-efficient light bulbs. To help him, Betty picks up her keyboard and places it next to his. Entering additional terms into Adam's original search box, they perform a joint search with a broader scope (Fig. 2d). Through this joint search, they find a suitable image.

After their search generates enough material, the students create an outline of the presentation. Carl uses his fingers to drag an empty document to his keyboard, thereby docking the document to his device (Fig. 2e), and starts writing.

Adam found a relevant encyclopedia article that he wants to share with Carl. He uses a nearby mouse to select the item and push it over to Carl, who is out of his direct reach. To find his cursor initially, he right-clicks to display a leader line connecting the location of his mouse to its associated cursor (Fig. 2f). Since Carl is standing at a different table edge, Adam reorients the article for Carl by rotating his mouse (Fig. 2g).

Betty independently worked on a separate part of the outline on saving water resources. She pastes her text into Carl's document. Different text colors in the outline match the colored auras surrounding each user's input devices, indicating authorship (Fig. 2h). With their outline completed and graphics assembled, the students arrange images and text into a presentation.

SYSTEM INFRASTRUCTURE

Our techniques were implemented on *FourBySix*, a large (120cm × 180cm), standing height (91cm) interactive table. Graphics are top-projected with two tiled XGA projectors (1024 × 1536 pixels, 21dpi). Touch input and object locations are sensed by two tiled VGA cameras under the table through diffused IR illumination (640 × 960 pixels, 13 dpi). FourBySix supports multiple sets of wireless input devices. To track identity, position, and orientation, each device is augmented with a unique optical tag on its base. White rectangular cardboard pieces with unique individual measurements served as tags; using common fiducial markers would be straightforward. Events are linked to physical device locations on the table by mapping optical tags to hardware identifiers reported by the devices to the operating system using the Windows RawInput API.

INTERACTION TECHNIQUES

This section describes the techniques from our scenario in more detail. Our video figure demonstrates each technique.

Keyboard + Tabletop Techniques

Physical keyboards offer improved input performance over projected keyboards, tactile feedback, and eyes-free operation for experts. For clarity, we highlight the targets of each keyboard's output by projecting a colored aura on the keyboard, and re-coloring the target's border to match that aura. We establish the target and semantics of typed text using the following techniques.

Link-by-Docking: Users can establish associations between a keyboard and digital objects on the table through physical proximity. Moving a digital object (*e.g.*, a text document) near the keyboard causes it to snap to the keyboard. Thereafter, keyboard input is routed to the object (Fig. 3a). Dragging the object away breaks the association.

Link-by-Placing: Another method of association supported by our system is to place the keyboard on top of an existing digital object (Fig. 3b). In both Link-by-Docking and Link-by-Placing, collision detection in display space between the digital object and the bounding box of the physical device can be used to detect the association event.

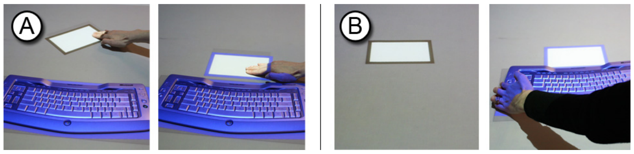

Figure 3: Link-by-Docking and Link-by-Placing techniques.

Contextual Command Prompt: In the absence of an established target, entered text is treated as a potential command and is shown in a text box directly above the keyboard (Fig. 2a). This text box tracks the position of the keyboard on the surface. Interpretation of the entered text depends on projected configuration buttons next to the keyboard (Fig. 2b). In our implementation, buttons determine if the text is to be used for web search, local machine search, or document creation. In each case, the result of the command is shown in the position previously occupied by the command box.

Pose-Based Input Modification: To enable collaboration, our system uses device pose as an implicit input channel. Multi-user joint actions can be initiated by bringing multiple devices into proximity. Moving two keyboards close to each other initiates a joint search query (Fig. 2d), or joint text entry into a shared document. Position and orientation information of one device relative to the other can implicitly parameterize the joint action. For example, collaborative editing of a single document view only makes sense if two keyboards share orientation. If orientations diverge, not all users will be able to read the text. For tabletop games, keyboard angle can be used to change the interaction dynamic from cooperative (side-by-side) to competitive (facing).

Mouse + Tabletop Techniques

Indirect input can enable manipulation of objects at a distance (*e.g.*, to bring them within reach of direct touch), and avoid touch actions that involve awkward proxemics (*e.g.*, in a collaborator's personal zone [13]). Using mice as input devices also affords more precise input than touch interfaces can provide [5]. Our mouse techniques explore different remote manipulation mappings, clarify cursor ownership, and manage associations between mice and keyboards.

Remote Object Manipulation: On multi-touch tabletops, direct translation, rotation, and scaling of digital objects have become customary. How might one offer multi-touch functionality in a single point-of-control device?

One option we support is to map individual manipulation operations to available input affordances of the mouse: orientation of a digital object can be mapped to orientation of the mouse itself (a benefit also realizable by using a custom, two-ball mouse [9]), while scale can be controlled with the scroll wheel (Fig. 2g). This method has the advantage of keeping all controls on a single device.

Another option for object manipulation is to treat cursors as touch points: whenever the left mouse button is held down, a touch event at the cursor location is added to the input event queue Working with two mice, one per hand, then enables use of existing multi-touch algorithms without modification. Users may also freely mix touch and cursor input on the same object (Fig. 4), or collaboratively manipulate a distant object.

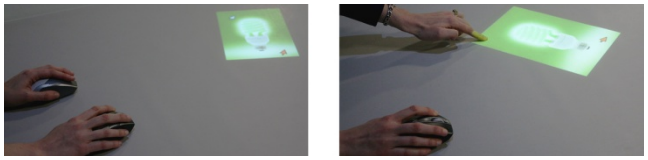

Figure 4: Remote, "multi-touch" manipulation with two cursors (left) and a combination of finger and cursor (right).

Leader Line Locator: To facilitate finding one's cursor among other cursors on a large tabletop, a leader line (Fig. 2f,) from the tracked absolute position of the device to its cursor may be displayed on demand (*e.g.*, by pressing the right mouse button). These dynamic leader lines draw upon the always visible connectors between laptops and tabletop cursors introduced in [12].

Link-by-Proximity: In single-user desktop computing, the user's pointing device and keyboard are naturally associated with each other. In a multi-user setting with many devices, these associations need to be established explicitly; once established, the mouse can then select targets for the keyboard. Proximity between devices may be used as an associative cue. Bringing a mouse into contact with a keyboard initiates the association (Fig. 5), and moving the devices apart beyond a threshold distance ends the association. Projected light provides feedback about associations using color, as in the text entry target visualization.

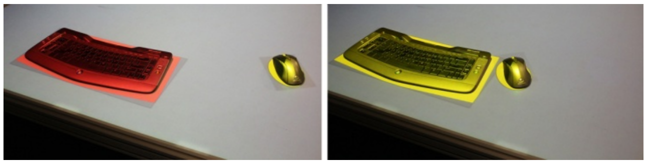

Figure 5: Link-by-Proximity (with projected-light feedback)

Link-by-Clicking: Instead of relying on proximity, the mouse cursor can be used as an intermediary connection mechanism. The user can move the cursor associated with the mouse into the area occupied by the keyboard and click to associate the devices (Fig. 6); a right-click in the same area will break the association. If desired, multiple keyboards can be simultaneously associated with a single mouse (and vice-versa). For example, linking multiple keyboards to one mouse enables them to share an output target selected by that mouse for multi-user-aware editing systems (*e.g.*, [2]).

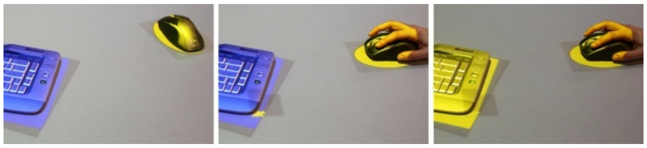

Figure 6: Link-by-Clicking (with projected-light feedback)

MULTI-USER CONSIDERATIONS

In multi-user tabletop settings, different viewing orientations and individual user identities have to be accounted for. We introduce reference frames for mouse input and treat input devices as proxies for user identity.

Reference Frame Choices

To correctly interpret mouse motion in a multi-user setting with no natural "up" direction, knowledge of each mouse's orientation and position with respect to the display coordinate system is crucial. To retain high precision input from the mouse, it is possible to fuse mouse and table data: the position and orientation of the mouse reported by the table establishes a reference frame. Relative position reported by the mouse then moves the cursor within that frame.

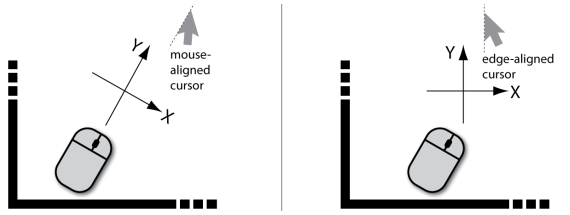

Figure 7: Mouse-aligned (left) and closest-edge-aligned reference frames (right) for calculating cursor movement.

Two conventions for mapping movement from mouse space to display space worked well in our experience (Fig. 7): adjusting reference frames continuously based on mouse orientation, thus establishing a moving local frame (mouse-aligned); or snapping to a frame aligned with the closest table edge (edge-aligned). In both cases, the cursor is rendered at an angle corresponding to the chosen frame.

As tables get large and multiple users share a table edge, mouse-aligned movement is preferable as it permits matching frame orientation with viewing direction. However, additional logic is needed to prevent unintended cursor movement during rotations of the mouse body. This shortcoming is not inherent in the edge-aligned mode.

User Identity and Location

When multiple users collaborate using a shared application, that application may need to have information about which user originated an action, and where that user is located around the table. Such metadata can be useful for applications ranging from automated meeting capture to personalization of input and output. Physical input devices can serve as a proxy for both user identity and location. To move from a unique input device ID to a user ID, the table application can query the user for account credentials whenever a new keyboard appears for the first time in a session. For the duration of the session, that keyboard is then bound to the user's account and entered text can be attributed to her. To estimate user position, the device position and orientation can be projected beyond the boundaries of the table.

DISCUSSION

We described a set of eight interaction techniques and some multi-user considerations for combining multi-touch tabletops and traditional input devices. These techniques enable co-existence of direct touch input, indirect cursor interaction, and keyboard text entry, allowing users to choose the tool most appropriate for their task. The techniques also suggest a larger design space of interactions. Fig. 8 shows the salient dimensions, and the subspace we explored thus far. Our research focused on small, co-located groups racting around a large table, with fixed associations between devices and users, but flexible associations between devices and digital content. We see two interesting areas for future investigation. First, understanding how to create and interpret associations between devices and targets in multi-table systems that support remote or mixed-presence collaboration. Second, supporting more complex, dynamic combinations between mice, keyboards, and users. Initial use of our system suggests that users sometimes switch devices during a session. Effective design of such tions will require understanding how group roles and dynamics are affected by the number of available devices. Future work also includes formally evaluating our approach on group work *vis à vis* purely touch-driven interactions.

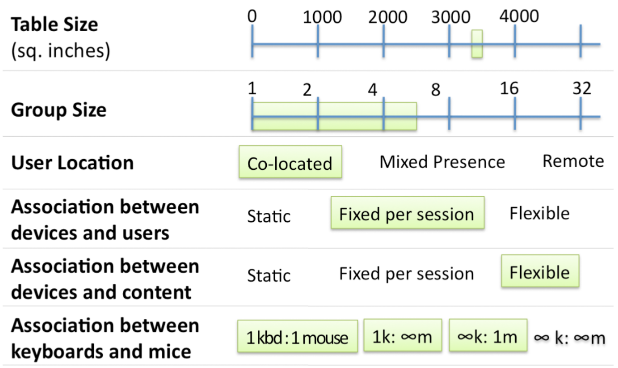

Figure 8: A design space of tabletop-device interactions.

REFERENCES

1. Bell, B. and Feiner, S. Dynamic Space Management for User Interfaces. *UIST 2000*, 239-248.
2. Bier, E., Freeman, S., and Pier, K. MMM: The Multi-Device Multi-User Multi-Editor. *CHI 1992*, 645-646.
3. Cotting, D. and Gross, M. Interactive Environment-Aware Display Bubbles. *UIST 2006*, 245-254.
4. Fiebrink, R., Morris, D., and Morris, M.R. Dynamic Mapping of Physical Controls for Tabletop Groupware. *CHI 2009*.
5. Forlines, C, Wigdor, D., Shen, C., and Balakrishnan, R. Direct-touch vs. Mouse Input for Tabletop Displays. *CHI 2007*.
6. Haller, M., Brandl, P., Leithinger, D., Leitner, J., Seifried, T., and Billinghurst, M. Shared Design Space: Sketching Ideas Using Digital Pens and a Large Augmented Tabletop Setup. *ICAT 2006*, 948-959.
7. Hinrichs, U., Hancock, M., Collins, C., and Carpendale, S. Examination of Text-Entry Methods for Tabletop Displays. *Tabletop 2007*, 105-112.
8. Isenberg, P. and Fisher, D. Collaborative Brushing and Linking for Co-located Visual Analytics of Document Collections. *EuroVis 2009*, 1031-1038.
9. MacKenzie, I.S., Soukoreff, R.W., and Pal, C. A two-ball mouse affords three degrees of freedom. *CHI 1997*, 303-304.
10. Morris, M.R., Brush, A.J., and Meyers, B. Reading Revisited: Evaluating the Usability of Digital Display Surfaces for Active Reading Tasks. *Tabletop 2007*, 79-86.
11. Pinelle, D., Barjawi, M., Nacenta, M., and Mandryk, R. An Evaluation of Coordination Techniques for Protecting Objects and Territories in Tabletop Groupware. *CHI 2009*, 2129-2138.
12. Rekimoto, J. and Saitoh, M. Augmented Surfaces: A Spatially Continuous Work Space for Hybrid Computing Environments. *CHI 1999*, 378-385.
13. Scott, S.D., Carpendale, M.S.T., and Inkpen, K. Territoriality in Collaborative Tabletop Workspaces. *CSCW 2004*, 294-303.
14. Weiss, M., Wagner, J., Jansen, Y., Jennings, R., Khoshabeh, R., Hollan J., and Borchers, J. SLAP Widgets: Bridging the Gap Between Virtual and Physical Controls on Tabletops. *CHI 2009,* 481-490.