Informing the Design of Direct-Touch Tabletops

Chia Shen, Kathy Ryall, Clifton Forlines, and Alan Esenther
Mitsubishi Electric Research Labs

Frédéric D. Vernier
University of Paris IX

Katherine Everitt
University of Washington

Mike Wu and Daniel Wigdor
University of Toronto

Meredith Ringel Morris
Stanford University

Mark Hancock and Edward Tse
University of Calgary

Tables provide a large and natural interface for supporting direct manipulation of visual content for human-to-human interactions. Such surfaces also support collaboration, coordination, and parallel problem solving. However, the direct-touch table metaphor also presents considerable challenges, including the need for input methods that transcend traditional mouse- and keyboard-based designs.

The term *display* suggests a device used solely to output visual information, untouched for fear of occluding or dirtying the screen. In contrast, a *surface* is free of this burden—it's part of the physical environment and invites touch. By superimposing input and visual output spaces onto surfaces, we can merge both ideas, creating touchable, interactive surfaces. Such surfaces have numerous uses; one exciting example is a horizontal, interactive, computationally augmented tabletop.

Compared with traditional displays, interactive tables provide three potential benefits. First, because the table is both the display and direct input device, it can take as input natural hand gestures and intuitive manipulations. Such inputs can improve the fluidity and reduce the cognitive load of user–content interactions. Second, by leveraging people's tendency to gather around a table for face-to-face interactions, a horizontal tabletop surface provides opportunities for building and enhancing collocated collaborative environments. Third, large tabletop surfaces have a spacious work area that can positively influence working styles and group dynamics. Users can also employ the surfaces' larger visual field as an external physical memory (thereby extending their working memory capacity); it can further serve as an external cognitive medium for new forms of visual representation and direct manipulation.

Over the past few years, we've sought to exploit direct-touch surfaces' advantages and affordances. To this end, we've designed, implemented, and studied a variety of tabletop user interfaces, interaction techniques, and usage scenarios. We've also empirically evaluated our work and obtained preliminary findings on:

- how people use a story-sharing table with digital photos;
- how nonspeech audio feedback affects multiuser interaction on tabletops, and
- how group size affects different aspects of multiuser tabletop collaboration.

Here, we explore tabletop advantages by examining the techniques we've developed to leverage those advantages. In addition to presenting six basic challenges we've encountered in our efforts, we discuss the experiences gained and lessons learned on this research journey.

Usability challenges

Direct-touch tabletops are a new interaction form factor, so researchers don't yet well understand appropriate user interfaces and interaction techniques for their widespread use. Existing research surveys include Scott and colleagues, who summarize tabletop systems and design approaches,[1] and Kruger and colleagues, who cover orientation approaches on a traditional meeting table.[2]

Tables are commonly found in homes, offices, command-and-control centers, cafés, design centers, showrooms, waiting areas, and entertainment centers. As such, they provide a convenient physical setting for people to examine documents, lay out and navigate maps, sketch design ideas, and carry out tasks that require face-to-face collaboration. In contrast, digital documents are still commonly used only on desktop/laptop computers, vertical plasma or projected displays, and handheld devices. Making such documents available on direct-touch interactive tabletop surfaces involves several design and usability challenges, including tabletop content orientation, occlusion and reach, gestural interaction, legacy application support, group interaction, and walk-up/walk-away use issues.

Tabletop content orientation

In contrast to computer monitors or projected displays, people seated around digital tabletops don't share

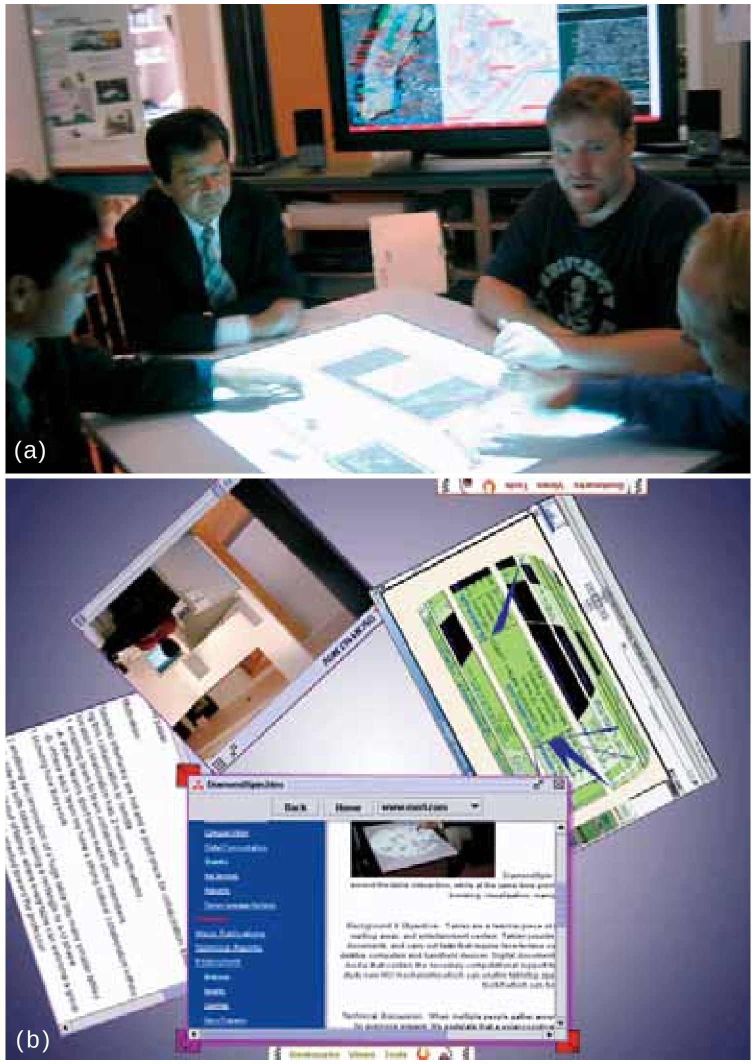
 Published by the IEEE Computer Society

a common perspective on information. That is, information presented right-side up to one participant might be upside down for another. Content orientation has implications for group social dynamics,[2] readability,[3] and performance.[4] Because content-orientation solutions let researchers evaluate other tabletop applications and interaction techniques, it's an extensively studied issue.

To address content orientation, we developed the DiamondSpin Tabletop Toolkit,[5] a set of interaction techniques. Figure 1 shows one popular layout technique in which the toolkit constrains documents to always face the tabletop's closest outside edge. People can use a single-finger touch and slide to move documents, which automatically turn to face them or the users they're passing documents to. A separate corner widget lets users perform arbitrary orientation, in which the constraint of documents always facing the tabletop's outside edge is removed. DiamondSpin also provides two other document orientation options:

- a lazy-Susan tabletop background that lets users rotate all documents together, and
- a magnetizer that reorients all the documents to face the same direction.

There are many possible document orientation and movement schemes. We classified and compared five different rotation and translation techniques for objects displayed on a direct-touch digital tabletop display.[6] We then analyzed their suitability for interactive tabletops given their respective input and output degrees of freedom, as well as their consistency, completeness, GUI integration for conventional window-based documents, and support for coordination and communication. Our comparative usability analysis results indicate that DiamondSpin's Polar-Coordinate-based orientation and translation schemes are especially effective for usage scenarios that require consistency and GUI integration.



1 **Content orientation in DiamondSpin Tabletop Toolkit. (a) Users meet around a multitouch, multiuser interactive digital table. (b) A bird's eye view of a DiamondSpin application, which constrains documents to face the tabletop's outside edge.**

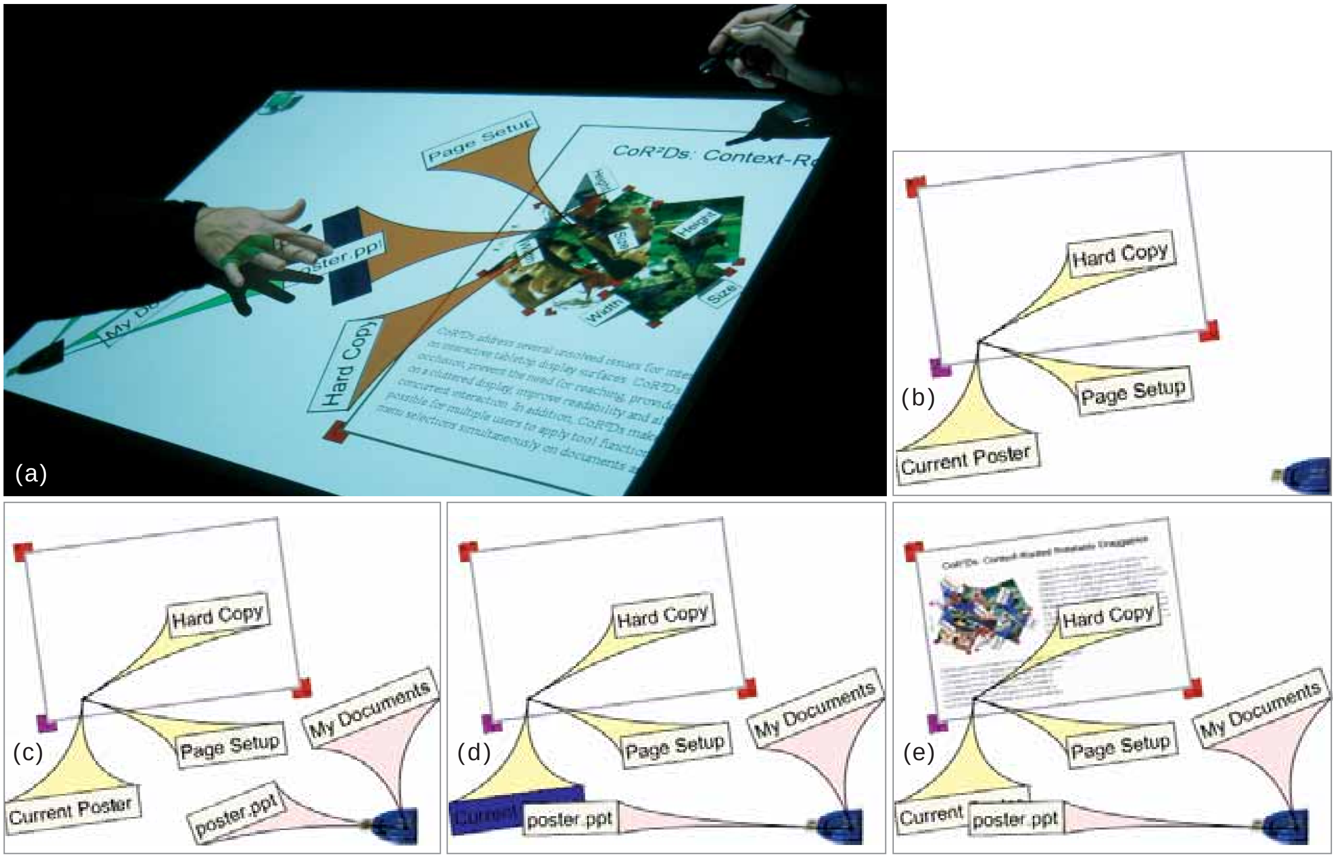
Occlusion and reach

When users interact with displayed information through direct touch, they might visually obscure the information immediately below their hand, arm, or stylus. Furthermore, the tabletop's large workspace makes many display regions either uncomfortable to work in or completely unreachable. To contend with these issues, we developed three techniques: Context-Rooted Rotatable Draggables (CoR^2Ds),[7] ExpressiveTouch puppetry,[8] and occlusion-aware visual feedback.

Context-rooted rotatable draggables. We designed CoR^2D interactive popups for multiuser direct-touch tabletop environments. As Figure 2 (next page) shows, translucent, colored swaths visually root CoR^2Ds to objects. Users can employ CoR^2Ds to issue commands or display information. Users can freely move, rotate, and reorient CoR^2Ds on a tabletop display surface using their fingers or hands; pointing devices (such as mice); or marking devices (such as a stylus or light pen).

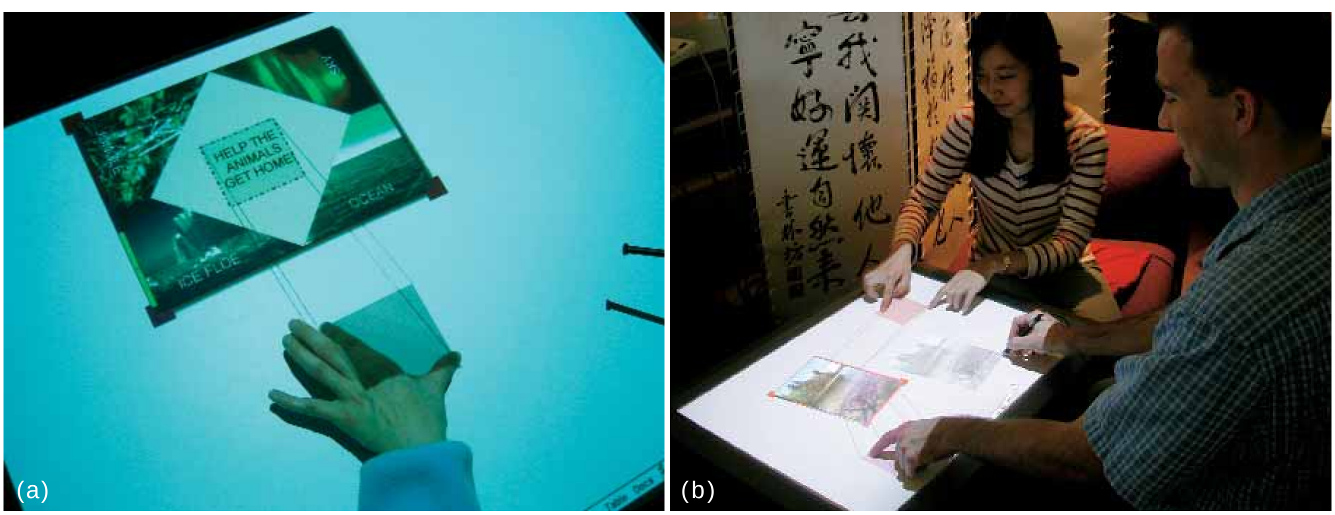
CoR^2Ds address five key interaction issues in tabletop systems: occlusion, reach, establishing context on a cluttered display, readability, and concurrent, coordinated multiuser interaction. Also, multiple people can use a single CoR^2D or a pair of CoR^2Ds to cooperatively complete a task (as in Figures 2a through 2e). For example, one person can drag a CoR^2D to a different part of the display surface while another user operates on it, thus facilitating multiuser operations. CoR^2Ds let operators and operands function across a visual distance, eliminating on-object occlusion. This also lets users operate objects at a distance without losing the visual cue of the objects' menus or tools, which might be across the display and partially hidden among display clutter.

Puppetry. In our ExpressiveTouch puppetry technique, users can apply operations—such as copy and paste[8]—on a distant document. This can be important when target objects are obscured from view. In a copy-and-paste operation, for example, users select objects by touching a document region. Then, while still touching the table, users can slide their hands away from the document to transition to indirectly adjusting the selec-

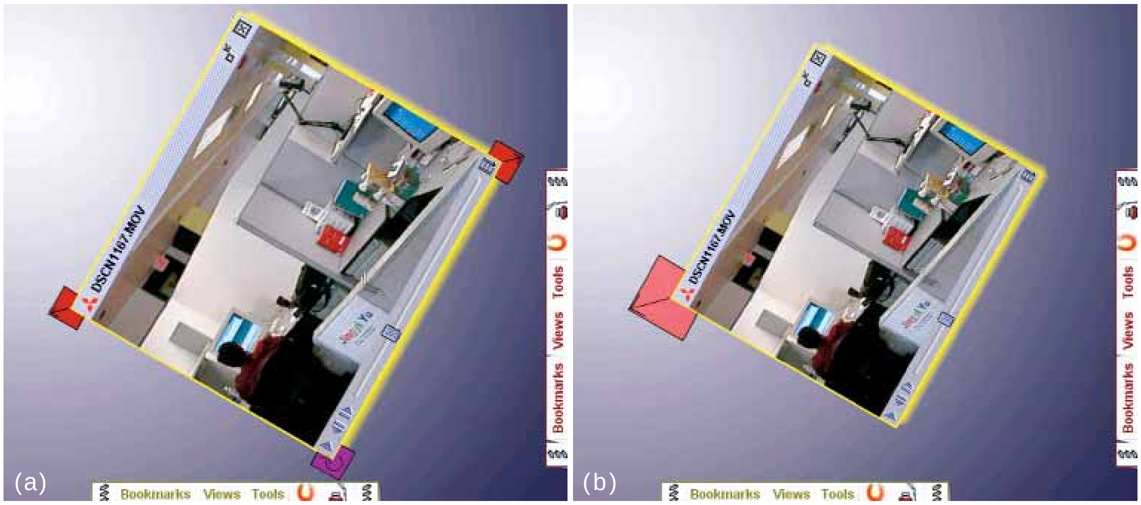


2 **Concurrent interaction on a large tabletop. (a) Two users employ CoR?Ds in a poster design scenario on an interactive tabletop. (b) A user double taps inside an empty poster document on the tabletop to launch its associated CoR?Ds. (c) A user double taps the USB reader icon, launching its associated CoR?Ds. (d) and (e) Users stretch and overlap "Current poster" and "Poster.ppt" from two separate CoR?Ds, completing the task of copying the Power Point file from the USB reader into the poster document on the table. This operation can be carried out bimanually by one user or by two separate users.**

tion box's location and size. As Figure 3a shows, four visual lines provide feedback to indicate the relationships between the control and display regions. This is a visually tethered, indirect distant operation that solves the occlusion problem on direct-touch surfaces. Our technique lets users comfortably control documents from various locations. The technique also mitigates physical interference, letting multiple people simultaneously use the same document from different sides of a table (see Figure 3b).



3 **Bimanual, multitool gestural interaction. (a) The tool decouples control and display spaces. (b) Two users simultaneously perform copy-n-paste operations on the same image object.**



4 **Occlusion-aware visual feedback. (a) A user's finger lands on a DiamondSpin object's Resize widget, occluding part of it. (b) The widget's size temporarily enlarges and grows transparent, to provide visual feedback and indicate its ephemeral nature, respectively.**

Occlusion-aware visual feedback. When users' fingers are larger than the target object—say, a menu item or a button—it's difficult for them to hit the target of the selection. Compounding this is the issue of occlusion; traditional windows, icons, menus, and pointing device (WIMP) systems generally offer in-place visual feedback on an action's target pixel. On a direct-touch interface, this feedback will always be occluded by the user's hand, thus removing much-needed feedback on target selection accuracy. Thus, traditional visual feedback—such as highlighting or drop shadows—isn't always effective in direct-touch tabletop interactions.

To address this, we developed occlusion-aware visual feedback. Our solution provides both in-place and visible feedback on a direct-touch interface. Figure 4 shows an example: when the user successfully selects the target, it enlarges. This change in visual state clearly indicates activation and offers in-place feedback without occlusion.

Gestural interaction

In a graphical user interface, users must manage a plethora of tools and interaction methods, typically using only a keyboard and mouse as input devices. Direct-touch surfaces that permit fluid, bimanual input could provide a more natural mapping between input and commands. For this to happen, we must first address several major questions:

- How should gestures map to various system functions?
- Should gestures map to the most common tasks or the most complex?
- Does each command require its own gesture—and, if so, how many gestures can we reasonably expect users to learn?
- How can hand gestures coexist with familiar point-based, mouse-like interaction?

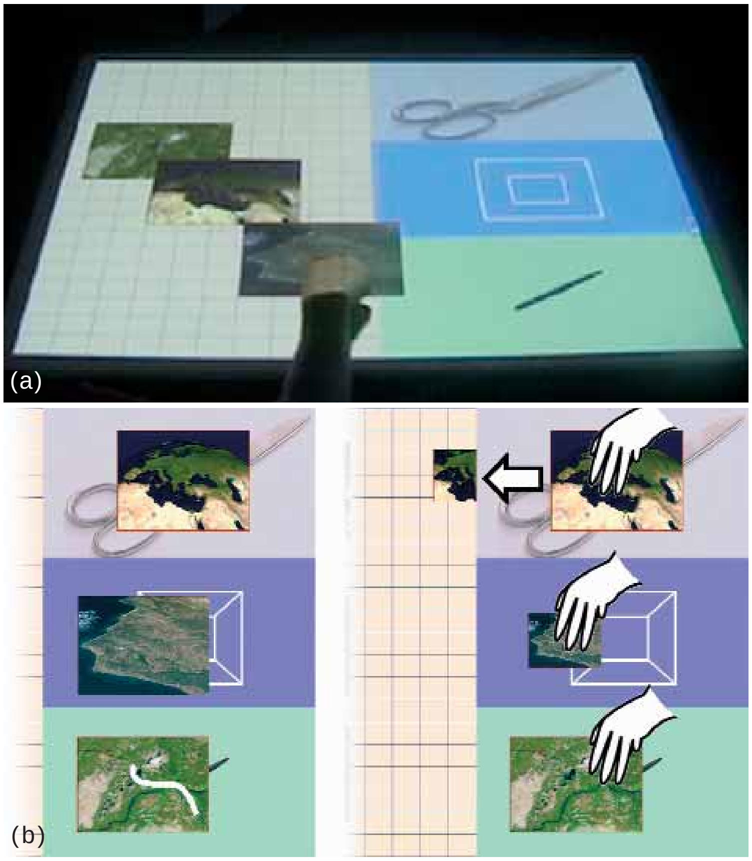
Designers must develop gestures that address their applications' specific needs. However, guidelines for introducing new gestural commands into an application can help designers avoid overly complicated systems. Here, we offer guidelines for gesture reuse within tabletop applications that begin to address some of these research questions.

Registration and gesture reuse. In our solutions, each gesture operation begins with gesture registration,[8] which sets the context for all subsequent interactions. A gesture registration action can be a hand posture, a simple touch, a dwell action, or a specific number of finger taps. Registration occurs when the system recognizes a distinctive gesture registration action on the table.

Registration lets users reuse gestures during other gesture phases. The same hand movements can thus produce different results, depending on which gesture action the user employs in the registration phase. For example, our desktop publishing application uses gesture registration to change stylus action modes. In normal operation, the stylus moves documents around the table and behaves like a mouse. However, if the user places two fingers on a document—as if to hold it in place—the stylus behaves like a pen, letting the user mark up and annotate the document. Through gesture registration, users can map the same stylus movements to multiple system commands. Gesture registration combined with gesture reuse is a powerful idea that lets designers define a variety of gestures using a small set of building blocks.

Modal spaces and gesture reuse. Our Modal Spaces solution[9] enhances conventional modal interfaces to permit reuse of gestures for different commands. It also clearly indicates the system's mode and lets users seamlessly change modes.

Modal Spaces divides the table into multiple workspaces, called *modal regions*. The system recognizes commands based on the target object, the user's gestural input, and the table's input region. As Figure 5 (next page) shows, location mediates user input, and docu-



5 **Modal spaces for an image editing application. (a) The touch-sensitive tabletop surface has four modal spaces: cutting, resize, annotation, and layout. (b) A before-and-after look at the effects of three annotations: bounding box cut, shrinking, and clearing.**

ments respond differently to the same gestures depending on where users execute them. In one modal region, for example, a touch might open a popup menu; in another, it might launch a stroke operation.

Legacy application support

As the tabletop interfaces field matures, developers will likely design more applications from scratch to take full advantage of multiuser horizontal workspace characteristics. Still, legacy applications are widely deployed and many are indispensable for real-world tasks. A digital tabletop environment must therefore address issues related to using preexisting applications on a horizontal workspace. We've developed solutions here in two areas: one enables mouse-emulation on touch surfaces; the other enables bimanual gestures for mouse-based applications.

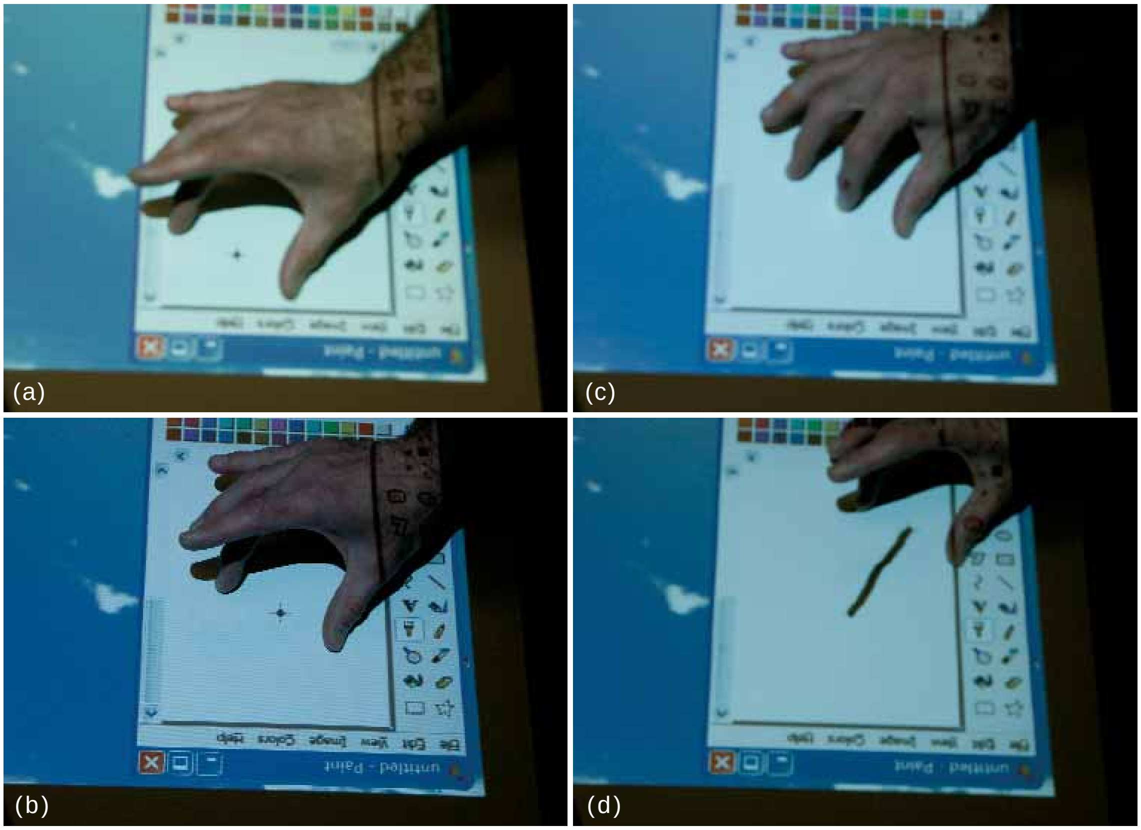
Mouse support for touch-based interactions. To support existing mouse-based applications, we need a finger-touch mechanism that emulates a computer mouse. This entails several issues. How does a user indicate mouse dragging versus mouse hovering (moving the mouse without pressing any buttons)? How does the user right-click? We also face a finger-resolution issue. A fingertip has a relatively large area, so how does the user specify a particular pixel, especially when his or her fingertip is obscuring the mouse cursor? Finally, traditional desktop applications assume (and support) only a single mouse. What happens if multiple users touch at the same time?

Early work attempted to mitigate the effects of visual occlusion and limited pointing accuracy by introducing a fixed offset between touch location and cursor position. This approach breaks the direct-touch input paradigm. In contrast, our solution detects multiple concurrent touches from the same user,[10] allowing the user's hand posture to define the offset more logically. When a user touches the table with one finger, the left mouse button is activated to simulate dragging. When a user touches with two fingers at once, the mouse cursor jumps to the center point between the touches; no mouse button is activated. Once in contact with the table, moving either or both fingers moves the mouse. As Figure 6 shows, this precision-hover mode gives users an unobscured view of the precise mouse cursor location between their fingers. This two-fingered control provides precision unobtainable with single-finger input. Unlike with tool tips and image rollovers, our method lets users move the mouse without activating mouse buttons. While in precision-hover mode, tapping with a third finger in between the first two will toggle the left mouse button up or down. Users can thus fluidly switch between dragging and moving, without inadvertently moving the mouse cursor.

This technique is natural and intuitive if users employ the thumb and middle finger of one hand for the first two touches, and use the index finger to toggle the left mouse button. They press the right mouse button by placing one finger down at the desired location and then quickly tapping anywhere else with a second finger. Users can then either drag with the right mouse button held down, or—to generate a right click—let go with the first finger, too. We can use variations of this basic technique to support other mouse buttons. We can, for example, support multiple users by letting the first toucher win: the system ignores subsequent touches by other users until the first toucher stops touching the table. Also, we under projected the display inside the touch surface just enough so that it's easy to use the technique in display corners, too. Our initial user experiences with these touch-based mouse-emulation schemes have shown encouraging user acceptance and fast learnability.

Gestural interactions for mouse-based applications. Meaningful gestures on a tabletop can improve group awareness in ways that simple mouse emulation can't achieve. For example, using a whole hand to pan a map might be more intuitive than selecting a pan mode and then panning with a single finger.

To allow multiuser gestural interaction with desktop applications, we map between gestures that a gesture engine recognizes and the keyboard and mouse events that legacy applications expect.[11] Using the Microsoft Send Input API, we can map a single gestural act to a series of keyboard and mouse events to provide a more natural mapping between tabletop gestures and legacy input. Turn-taking protocols let us manage system responses



6 Using precision-hover mode to transition between moving the mouse and drawing. (a) The user makes contact with the display. (b) The mouse remains centered between the user's thumb and middle finger as he moves his fingers and drags the mouse. (c) The user taps briefly with the index finger to toggle the left mouse button's state, engaging the drawing function. (d) The cursor remains between his two fingers, but by moving his thumb and middle finger, the user engages the left mouse button and the interface draws a line.

when multiple people gesture simultaneously. As a result, legacy applications appear to directly understand freehand gestures from multiple people. We've also integrated support for speech input using the same techniques, creating multimodal tabletop applications.

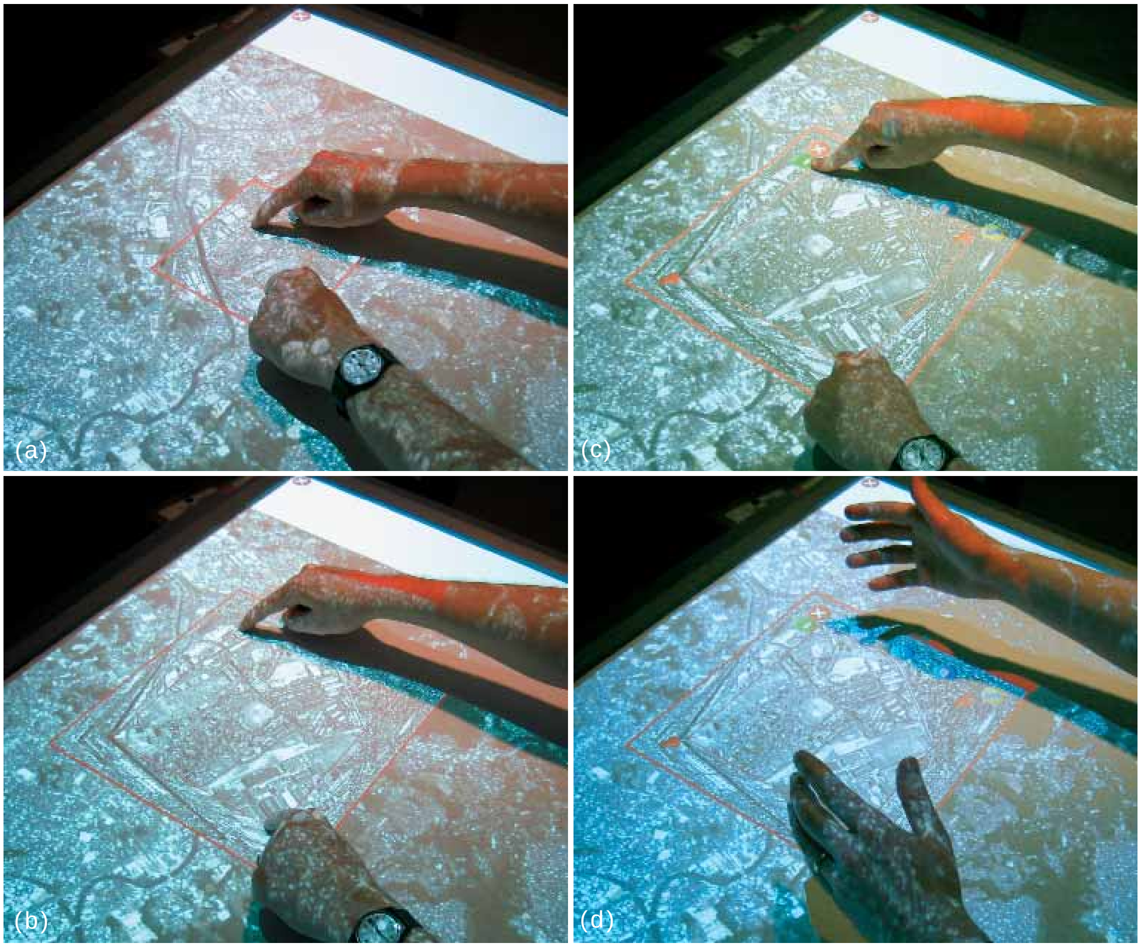
Group interaction techniques

Collocated, multiuser activities present many new challenges for UI designers. Among the problems are: How can multiple users access conventional, single-user menubars and toolbars? How can multiple users simultaneously explore detailed image or geospatial data without interfering with the global context view?

Conventional menubar access. Menubars and tools are popular UI widgets, and we must facilitate both shared and personal use of them in a group setting. In DiamondSpin, we provide four types of menubar usage patterns:

- *Draggable for sharing*. With a single finger movement, a user can slide a group menubar along the tabletop's edge to any parking position. Multiple users can thus share a single menubar, passing it among themselves.
- *Lockable for private use*. On a multitouch, multiuser direct-touch tabletop, it's easy for one user to select items from another user's menubar. Consequently, we built in to the menubar a touch-to-lock user-controlled option that lets users selectively prevent all other collocated users from operating their personal menubars.
- *Replicated for convenience*. Users can replicate their private menubars and give copies to other collaborators at the table. Figure 1b shows a usage scenario in which the user's replicated menubar appears on both the top and bottom edges of the tabletop. This is a convenient usage pattern, but it requires social protocols to mitigate conflicting menu operations among collocated users.
- *Subset for restricted access*. Finally, rather than duplicating an entire menubar, users can replicate and distribute a menubar subset to other users. Making limited menubar actions available could be useful in situations with one power user, as in a teacher–student setting. The teacher's menubar might have full functionality, for example, while the students' menubars contain a smaller set of options.

Group exploration of geospatial data. High-resolution satellite photographs, maps, and blueprints are often presented from a bird's eye view so that intelligence analysts, architects, and city planners can gather around the rolled-out paper documents. Such experts are accustomed to viewing documents from a variety of



7 **Exploring a satellite image using DTLens on a multitouch tabletop. (a) Using both hands, the user launches the DTLens tool and (b) stretches open the view. (c) The user presses his hands to tack the lens in place, which (d) frees his hands for other work.**

perspectives as they work face-to-face around a table. In contrast, groups that work with high-resolution digital files on a tabletop display are hindered by the surfaces' inability to display a document's full resolution.

Tiling displays or projectors offer a promising solution, but they're currently prohibitively expensive. The single-user solution of zooming in and panning around the document is inappropriate in groups, as members might want to see a detailed view of different document regions at the same time. Furthermore, people often get lost in a data set when using pan and zoom interfaces, which sacrifice larger context when they zoom in for detailed views.

Our DTLens[12] is a multiuser tabletop tool that lets groups explore high-resolution spatial data without the panning and zooming drawbacks. As Figure 7 shows, DTLens gives each group member an independent, zoom-in-context fisheye lens that they manipulate through multihand gestures performed directly on the document. Any group member can thus reach onto the table, grab a document region, and stretch the area to reveal more detail. By allowing simultaneous exploration of document details, DTLens lets group members move naturally from collaborative to independent activities as they work face to face around the tabletop display. DTLens also gives users a consistent interaction set for lens operations, thus minimizing tool switching during spatial data exploration.

Walk-up and walk-away usage issues

With traditional tables—in airports, cafés, and conference rooms, for example—people often spontaneously approach a table and collaborate with people already seated. In such scenarios, people generally bring their own material and documents. Therefore, if we're to develop walk-up, kiosk-like digital tabletops, we must consider appropriate user interfaces. Although researchers have actively explored how to share personal data on public vertical displays, a key difference between shared use of a tabletop and that of a vertical display is that when people sit around a table, a particular table region is in their immediate physical proximity. Moreover, such regions are not visually equivalent for all users on all sides of a table. These physical and perceptual properties make these individual areas ideal choices for private work spaces.[13] To this end, we developed a user interface design solution called UbiTable.[14]

As Figure 8 shows, UbiTable lets users dynamically connect personal laptops, cameras, and USB devices to an interactive tabletop so they can fluidly share, manipulate, exchange, and mark up data. At the same time, each user maintains explicit control over their document's accessibility. We divided the UbiTable tabletop into two regions—personal and public—that have distinct access control and document interactibility properties. Personal region documents are visible to all users, but can only be manipulated and moved by the document owner. Public region documents are accessible to all users, but only the owner can move it into a personal region. This gives the owner explicit control over how the document is distributed to other meeting participants. Documents that are displayed on the tabletop use colored borders to provide feedback information for owners and users. Green and pink borders, for example, indicate personal document regions, while gray borders indicate public document regions. In addition to the public and personal areas, UbiTable designates personal devices, such as laptops, as private regions for user's data. It therefore offers three information-sharing levels: public, personal, and private.

Evaluations, experiences, and reflections

We've learned many lessons and gained many insights in developing, using, and testing our solutions. Our experiences fall into three general categories that we've observed across numerous application prototypes and evaluation sessions: orientation side effects, input precision, and nonspeech audio feedback for group interaction.

Orientation side effects

Providing interface-level support for flexible document orientation and positioning on a large tabletop has emerged as an important foundation for our work. The DiamondSpin toolkit has supported many research prototypes and applications. In one project,[4] we evaluated and analyzed various orientation techniques' performance and differences. Our findings suggest that a more objectively precise technique doesn't necessarily translate into high qualitative ratings from users. Indeed, each technique seems to have a different feel for users, related to interaction fluidity, how a technique behaves under the user's touch, and the technique's perceived naturalness.

We also observed two noticeable operational side effects of rotating tabletop documents that are independent of the rotation and translation methods. First, while users want the ability to reorient documents to suit tabletop collaboration, some orientations can severely affect a document's resize range. This problem occurs when a document is rotated out of alignment with a rectangular or square tabletop's canonical edges, as seen in three out of the four documents in Figure 1b. Second, text readability can degrade when a document is rotated at an angle with respect to the canonical Cartesian *x* and *y* axes, due to non-antialiased text rendering for rotated text.

To solve these usability problems, we built a Table-for-*N* feature in the current DiamondSpin version. When a user rotates a document, Table-for-*N* automatically snaps the document's bottom edge to align parallel with one of the tabletop's *N* edges. This function is convenient

8 **Walk-up options for worksharing. (a) Walk-up usage of a conventional table. (b) Walk-up usage of UbiTable, which offers users a private region for laptop displays, and personal or public regions on the tabletop. (c) The UbiTable interface. Green and pink borders indicate personal areas, blue portals are for data movement to and from private regions (laptops or other personal devices), and the center gray area is for document sharing.**

when several tabletop users are working independently on documents or images.

Input precision

Although many touch-interactive surfaces provide high input resolution at single-pixel or finer precision, we've observed (along with other researchers) that pixel-accurate interaction is difficult with direct-touch interaction. This imprecision particularly manifests in terms of jitter—a shift in touched or selected pixels between input frames. Jitter is a problem in at least two cases.

First, some operations require repeated interactions, as when users must double or triple tap on the tabletop (with a single finger, multiple fingers, or the whole hand). However, such consecutive taps don't necessarily land on the exact same pixel region on the tabletop display. To solve this, we've built in a larger activation region for retouch. That is, when an interaction requires a user to repeatedly tap on the tabletop, the exact pixel area touched on the second or third landing have a tolerance. For single-finger double or triple taps, for example, we find a tolerance of 10 pixels works quite well.

Other researchers in touch-interaction precision have proposed to solve this problem through indirect (offset) or relative pointing. However, in our view, it's important to maintain direct full-handed touch and the bimanual interaction paradigm on a horizontal tabletop surface as much as possible.

Second, jitter is an issue when the operation's effect occurs when the user lifts his or her hand. During our user study for bimanual gesture design,[8] for example, we noticed that some participants were troubled by their accuracy in selecting an image region during copy-and-paste (see Figure 3). Often, as participants lifted their hands to complete the paste operation, the pasted image was slightly shifted in one or both dimensions.

To stabilize imprecise interaction jitter, we improved our gesture termination algorithm: it now looks back a few time frames from the instance that a user lifts his or her hand off the table. Specifically, our algorithm looks for a time window in which the user's hand has maintained a stable selection posture for a few frames of time. We take this selection box as the user's intended region for copy-n-paste. This improvement has offered fairly satisfactory performance.

Nonspeech audio feedback

In a multiuser, interactive tabletop setting, the table serves as both a shared display and a shared input device. Because the display is visual, designers often focus on presenting information to users through the visual channel. Although visual feedback is the primary communication modality in this environment, we believe auditory feedback can also serve an important role. However, what that role is and how it might enhance users' experiences and assist in application building is not yet clear. In particular, simultaneous actions by multiple users can both increase efficiency and create interference.

In an initial UbiTable user study,[14] we found that users were confused when the tool offered auditory feedback to indicate operational errors. Because an identical system beep sounded for both users, the common reaction was: "Who was that sound for?" In collaborative settings, users often work in parallel on individual tasks and might want to be aware of their peers' actions. Users might, for example, wish to know when another user accesses or manipulates screen objects outside their visual attention. Auditory feedback can be useful in these circumstances. While using redundant auditory feedback can increase group awareness in some cases, it can also hinder individual users' performance. Alternatively, sounds that are useful for an individual might overload the larger group's auditory channels. It's essential to consider this tradeoff when designing collocated, collaborative applications.

Our study included two experiments using nonspeech audio in an interactive multitouch, multiuser tabletop surface.[15] In our first experiment, we investigated two categories of reactive auditory feedback: affirmative sounds that confirm user actions and negative sounds that indicate errors. Our results show that affirmative auditory feedback might improve a user's awareness of group activity at the expense of awareness of the user's own activity. Negative auditory feedback might also improve group awareness, but simultaneously increase the perception of errors for both the group and the individual.

In our second experiment, we compare two methods of associating sounds to individuals in a collocated environment. Specifically, we compared localized sound, where each user has his or her own speaker, to coded sound, where users share a speaker, but the sound's waveform is varied so a different sound is played for each user. Results of this experiment reinforce the first experiment's finding: a tension exists between group awareness and individual focus. User feedback suggests that users can more easily identify who caused a localized or coded sound, and that either option lets them more easily focus on their individual work. In general, these two experiments show that, depending on its presentation, it's possible to use auditory feedback in collocated collaborative applications to support either individual work or group awareness, but not both simultaneously.

Conclusion

Interactive, direct-touch digital tables are an emerging form factor with largely immature user interface design. Our research results, along with that of other researchers, set forth interaction techniques, user experiences, and design considerations that we'll continue to expand as we exploit and explore the advantages of interactive tabletop systems.

Among our lessons so far is that, whenever we demonstrated our tabletop systems to actual users or potential customers, the most compelling moments were when the tables interoperated with vertical displays and other devices. This observation agrees with our previous finding that group interactions require supplemental vertical displays.[16]

While the digital table provides a compelling focal point for group activity, we recognize the opportunity to augment it with additional computational resources and surfaces. We've thus begun exploring multisurface table-centric interaction,[17,18] wherein all interaction occurs at a table, while ancillary surfaces provide coordinated and multiview visualization and display space. This is different from previous interactive room research, in which displays and devices are generally independent.

There are many outstanding research issues in this area. In our view, two of the most fundamental open questions are: Does a large tabletop provide spatial and perceptual cognitive advantage in helping users accomplish their tasks? If so, under what circumstances does this cognitive assistance occur, and when does it break down? To fully examine these questions, we must step back and

- analyze basic human perception and cognition;
- evaluate not only our design artifact, but the cognitive prosthesis it might enable; and
- envision how people might use this externalization device to better represent, visualize, and express their ideas.

Our future research will focus on these issues. ■

References

1. S.D. Scott et al., "System Guidelines for Co-located, Collaborative Work on a Tabletop Display," *Proc. European Conf. Computer-Supported Cooperative Work* (ECSCW), Kluwer Press, 2003, pp 129-178.
2. R. Kruger et al., "Roles of Orientation in Tabletop Collaboration: Comprehension, Coordination and Communication," *J. Computer Supported Cooperative Work*, vol. 13, nos. 5-6, Kluwer Press, 2004, pp. 501-537.
3. D. Wigdor and R. Balakrishnan, "Empirical Investigation into the Effect of Orientation on Text Readability in Tabletop Displays," *Proc. 9th Euro. Conf. Computer-Supported Cooperative Work* (ECSCW), Kluwer Press, 2005, pp. 205-224.
4. C. Forlines et al., "Under My Finger: Human Factors in Pushing and Rotating Documents Across the Table," *Proc. Int'l Conf. Human–Computer Interaction* (Interact), LNCS 3585, Springer Verlag, 2005, pp. 994-997.
5. C. Shen et al., "DiamondSpin: An Extensible Toolkit for Around-the-Table Interaction," *Proc. Int'l Conf. Human Factors in Computing* (CHI), ACM Press, 2004, pp. 167-174.
6. M.S. Hancock et al., "Rotation and Translation Mechanisms for Tabletop Interaction," *Proc. 1st IEEE Int'l Workshop Horizontal Interactive Human–Computer Systems* (TableTop), IEEE CS Press, 2006, pp. 79-78.
7. C. Shen et al., "CoR^{2}D: Context-Rooted Rotatable Draggables for Tabletop Interaction," *extended abstracts, Int'l Conf. Human Factors in Computing Systems* (CHI), ACM Press, 2005, pp. 1781-1784.
8. M. Wu et al., "Gesture Registration, Relaxation, and Reuse for Multi-Point Direct-Touch Surfaces," *Proc. 1st IEEE Int'l Workshop Horizontal Interactive Human–Computer Systems* (TableTop), IEEE CS Press, 2006, pp. 183-190.
9. K. Everitt et al., "Modal Spaces: Spatial Multiplexing to Mediate Direct-Touch Input on Large Displays," *extended abstracts, Proc. Int'l Conf. Human Factors in Computing Systems* (CHI), ACM Press, 2005, pp. 1359-1362.
10. A. Esenther and K. Ryall, "Fluid DTMouse: Better Mouse Support for Touch-Based Interactions," *Proc. Working Conf. Advanced Visual Interfaces* (AVI), ACM Press, 2006, pp. 112-115.
11. E. Tse, "Enabling Interaction with Single User Applications through Speech and Gestures on a Multi-User Tabletop," *Proc. Working Conf. Advanced Visual Interfaces* (AVI), ACM Press, pp. 336-343.
12. C. Forlines and C. Shen, "DTLens: Multi-User Tabletop Spatial Data Exploration," *Proc. 18th Annual ACM Symp. User Interface Software and Technology*, ACM Press, 2005, pp. 119-122.
13. S.D. Scott et al., "Territoriality in Collaborative Tabletop Workspaces," *Proc. ACM Conf. Computer-Supported Cooperative Work* (CSCW), ACM Press, 2004, pp. 294-303.
14. C. Shen, K. Everitt, and K. Ryall, "UbiTable: Impromptu Face-to-Face Collaboration on Horizontal Interactive Surfaces," *Proc. 5th Int'l Conf. Ubiquitous Computing*, LNCS 2864, Springer Verlag, 2003, pp. 281-288.
15. M.S. Hancock et al., "Exploring Non-Speech Auditory Feedback at an Interactive Multi-User Tabletop," *Proc. 2005 Conf. Graphics Interface,* Canadian Human–Computer Comm. Soc., 2005, pp. 41-50.
16. K. Ryall et al., "Exploring the Effects of Group Size and Table Size on Interactions with Tabletop Shared-Display Groupware," *Proc. ACM Conf. Computer-Supported Cooperative Work* (CSCW), ACM Press, 2004, pp. 284-293.
17. D. Wigdor et al., "Table-Centric Interactive Spaces for Real-Time Collaboration," *Proc. Working Conf. Advanced Visual Interfaces* (AVI), ACM Press, pp.103-107.
18. C. Forlines et al., "Multi-User, Multi-Display Interaction with a Single-User, Single-Display Geospatial Application," to appear in *Proc. ACM Conf. User Interface Software and Technology,* ACM Press, 2006

Chia Shen *is a senior research scientist at Mitsubishi Electric Research Laboratories. Her research interests include human–computer interfaces and interactions with nonconventional visual surfaces, including tables and walls. Shen has a PhD in computer science from University of Massachusetts at Amherst. Contact her at shen@merl.com.*

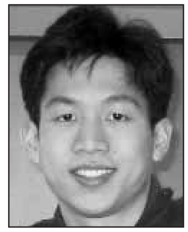
Kathy Ryall *is on the principal technical staff at Mitsubishi Electric Research Laboratories. Her research interests include human–computer interaction, computer-supported cooperative work, and information visualization. Ryall has PhD in computer science from Harvard University and is a senior member of the IEEE. Contact her at ryall@merl.com.*

Clifton Forlines *is a researcher at Mitsubishi Electric Research Laboratories. His research interests include the design and evaluation of novel user interfaces, including digital video presentation; collaborative tabletops; multiuser, multidisplay workspaces; and hand-held projectors for augmented reality. Forlines has a Masters of Human-Computer Interaction and a Masters of Entertainment Technology from Carnegie Mellon University. Contact him at forlines@merl.com.*

Alan Esenther *is a user interface software developer in the Technology Lab at Mitsubishi Electric Research Laboratories. His research interests include human–computer interaction; finding the right digital tool for the job; and making multiuser, multitouch, and large-touch surface interactions and development easier. Esenther has an MS in computer engineering from Boston University. Contact him at esenther@merl.com.*

Frédéric Vernier *is an assistant professor in the Department of Computer Science, University of Paris Sud in France, where he works in the LIMSI-CNRS laboratory. His research interests include human–computer interaction and interactive visualization. Vernier has a PhD in computer science from the University of Grenoble. Contact him at frederic.vernier@limsi.fr.*

Katherine Everitt *is a PhD student in computer science at the University of Washington. Her research interests include tabletop interfaces, computer-supported cooperative work, and multimodal interfaces. Everitt has an MS in computer science from the University of California, Berkeley. Contact her at everitt@cs.washington.edu.*



Mike Wu *is a PhD candidate in computer science in the Dynamic Graphics Project lab at the University of Toronto, where he is a Health Care, Technology, and Place Fellow. His research interests include assistive technologies and tabletop user interfaces. Wu has a MSc in computer science from the University of Toronto. Contact him at mchi@dgp.toronto.edu.*

Daniel Wigdor *is a PhD candidate in the Dynamic Graphics Project lab at the University of Toronto and a long-term intern at Mitsubishi Electric Research Laboratories. His research interests include multisurface interactive spaces, shared tabletop interaction, and text entry into mobile devices. Wigdor has an MSc from the University of Toronto. Contact him at dwigdor@dgp.toronto.edu.*

Meredith Ringel Morris *is a researcher in the Adaptive Systems and Interaction group at Microsoft Research. Her research interests include human–computer interaction, computer-supported cooperative work, and educational technologies. Morris has a PhD in computer science at Stanford University. Contact her at merrie@stanfordalumni.org.*

Mark Hancock *is a PhD student in computer science at the University of Calgary. His research interests include tabletop display interaction, including 3D interaction. Hancock has an MSc in computer science from the University of British Columbia. Contact him at msh@cs.ucalgary.ca.*

Edward Tse *is a doctoral student at the University of Calgary specializing in natural interaction over large digital displays. His research interests include supporting users' speech and gestures when collaborating on large surfaces. Tse holds a MSc in human–computer interaction at the University of Calgary. Contact him at tsee@cs.ucalgary.ca.*