Interactive tabletops

User Interface, Metaphors and Gestures

Chia Shen

Why Tabletops?

What’s wrong with this picture?
– “Single” point of control in “group” interaction

CHI 2001 PhotoFinder demo.
Courtesy Bill Kules, UI/MD.

Circa 2005@MERL

PDH Table Design @CHI 2001 Design Expo
### Why Tabletops?

- Touch and multi-touch
- Horizontality vs vertical displays
- Social – Collaborative vs single users

### 12 Challenges of direct-touch tabletops

<table>
<thead>
<tr>
<th>What: Physical Platform?</th>
<th>How: Social?</th>
</tr>
</thead>
<tbody>
<tr>
<td>Digital Tabletops</td>
<td>Multiple people</td>
</tr>
<tr>
<td>Interactive Rooms</td>
<td>Walk-up usage</td>
</tr>
<tr>
<td></td>
<td>Shared Environment</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Input resolution</td>
<td>Touching</td>
</tr>
<tr>
<td>Orientation</td>
<td>Perceiving</td>
</tr>
<tr>
<td>Occlusion</td>
<td>Collaborating</td>
</tr>
<tr>
<td>Bimanual (2 hands)</td>
<td></td>
</tr>
</tbody>
</table>
Tables are not desks:

**Multi-user:**
Face-to-face and simultaneous operations

**Arbitrary orientation of objects:**
Multiple rotating documents.
Anti-alias continuously, especially text

**Rotation of the entire virtual tabletop:**
Mega-pixel operation
rotation-sensitive items.

Direct-touch interfaces have two inherent characteristics:

(1) UI targets can be occluded by the fingers, and

(2) the large touch-area of a fingertip needs to be mapped to a screen-point.

System defined UI-controls, such as window manager decorations (min/max), sliders, buttons, menus, etc., need to be reengineered to allow for direct-touch input. Changes include the need to make these controls larger, to compensate for imprecise input, and reposition them to minimize occlusion of window content while in use.
When working on a tabletop, there are many fundamental differences in the user input paradigm that challenge conventional WIMP assumptions:

1. The input area of any given user/point is larger, and a different shape, than the single pixel defined by a mouse pointer.
2. Windows/widgets may be rendered off-axis (rotated).
3. A single user may be touching multiple UI-controls/screen points simultaneously.
4. Multiple users may be touching the same UI-controls/screen points simultaneously.
5. Multiple users may be touching different UI-controls/screen points simultaneously.
6. Multiple users may be entering text from different keyboards (soft or hardware) simultaneously.

**Example: A multi-user multi-touch tabletop**
Example solution for “fat fingers”:
Interactive Corner Handle Feedback

Upon touch
Before touch

Multiplicity of Tools and Hands
When Working on a Table
Postures vs. Gestures

- Postures – Static hand marks
- Gestures – Time and space varying postures
  - E.g., sliding two hands apart to “zoom out”

A Posture Engine

A possible posture recognition process:
- Mean value and covariance matrix for each pre-recorded posture
- Dynamic touch data is converted into 26 unique features
- Use univariate Gaussian algorithm to compare
- Posture detection accuracy ~ 80%
- A set of heuristics are added to improve accuracy up to ~95%
Multi-User Multi-Hand Gestural Input

How to support both conventional point-based interaction and freehand gestures within the same environment? How to multiplex tools? And multimodal interaction…?

A fundamental gesture design framework
- Gesture Registration
- Gesture Reuse
- Gesture Relaxation

Supported by a generic gesture engine

Gesture Registration

• Problem: Multiple interaction styles and tools can be present – pointers, cursors, stylus, free-hand gestures.

• Solution: Assign a “start” gesture to facilitate seamless and fluid mode change with a simple registration phase:
  - Transitioning a pen or a finger between being a pointing device and a writing device
  - Start of a particular gestural interaction with a particular context
Gesture Relaxation

• Problems:
  (1) Maintain fairly precise hand postures requires muscular tension.
  (2) High variability of sensed data due to physical affordances of a touch tabletop: height, size and reach, location, user position.

• Solution:
  Gestural interface allows relaxed hand postures and movements.

Gesture Reuse

• Problem: A large set of gesture primitives both burdens the users in memorizing gestures, and the system in having to recognize many different patterns.

• Solution: Employing the same gesture, including hand postures, finger touches or stylus, to accomplish different tasks within different context.
ExpressiveTouch Gesture Video

Gesture State Transition Diagrams

- **Annotate**
  - Asymmetric bimanual

- **Wipe**
  - Unimanual

- **Cut/Copy-n-Paste**
  - Asymmetric bimanual continuous compound

- **Pile-n-Browse**
  - Symmetric bimanual
DiamondSpin – A Java Tabletop Toolkit

**Multi-user:**
Face-to-face and simultaneous operations

**Arbitrary orientation of objects:**
Multiple rotating documents.
Anti-alias continuously, especially text

**Rotation of the entire tabletop:**
Mega-pixel operation rotation-sensitive items.

Tabletop Space Management 1/2

Black-hole
Tabletop Space Management 2/2

Fisheye

Draggable Popup menus (CoRDs)

Movable to anywhere around the table, rooted at the context.

Multi-user, or privileged.

Contextual: for Element or for Background
Using DiamondSpin: Many Polygonal Shapes of Tabletops

Using DiamondSpin: Subdivision of Tabletop into Regions

- Dividing tabletop area into separate semantic regions

- Regions can have both distinctive visual layout and semantics.
- Policy issue – As a document is moved across a region, how should the document's properties change?
UbiTab - Walk-Up Interaction

The UbiTable provides a large shared area for interaction

- The table supports the connection of multiple devices
  - Laptop, USB device, Camera, PDA
- We use personal spaces to denote a user’s area of control
- Multiple devices can be put in each personal space
- Blue portals are used to copy content and transfer access files

DiamondSpace

DiamondSpace
ACM UIST 2006
Example Interactive Tabletop Projects

Citations

http://www.diamondspace.merl.com/publications.php


Shen, C., Everitt, K., and Ryall, K. UbiTable: Impromptu Face-to-Face Collaboration on Horizontal Interactive Surfaces. in Proceedings of the 5th International Conference on Ubiquitous computing (Seattle, WA, USA, October 12-15, 2003), Springer, 281-288.