Multitouch User Input Systems

Andy Isaacson
User Interface Programming
2/4/08
What is Multitouch Input?

• Multiple points of spatial input
• Can be on a touch sensitive screen or through a touch sensitive pad
• May come from more than one user
• May be single points or a chorded gesture
New Hype, Old Technology

• 1982 - Flexible Machine Interface (Nimish Mehta, University of Toronto)
• 1983 - Video Place/Video Desk (Myron Krueger)
• 1984 - Multitouch Screen (Bell Labs, Murray Hill, NJ)

Detailed history from Bill Buxton (http://www.billbuxton.com/multitouchOverview.html)
Talk Outline

• State of Multitouch Devices Today
• Multitouch Programming through Events
• Future Possibilities
Devices from Apple
iPhone/MacBook Air

Images from Apple Computer
NYU/Perceptive Pixel
Jeff Han

Images from http://cs.nyu.edu/~jhan/
Frustrated Total Internal Reflection

Images from http://cs.nyu.edu/~jhan/ftirsense/
Mitsubishi DiamondTouch

Source: Mitsubishi Electric Research Laboratories
Microsoft Surface

Source: Microsoft and Popular Mechanics
Homebrew Multitouch

Wii Remote Hack
Johnny Chung Lee
http://www.johnnylee.net

dyeSight
Medallia
http://blog.medallia.com/dyesight.html
Multitouch Software Projects

- Multi-pointer X (MPX)
  - Adds capability for multitouch/multipointer in X Window
  - Backwards compatible with mouse events
  - Provides multitouch aware window manager, so windows can be resized with pinching motion
Multitouch Software Projects

• touchlib
  – Adds multitouch functionality for MS Windows
  – Could be ported to other systems
  – Uses an event protocol that is common to several applications like PureData and Processing
Multitouch Software Projects

• iPhone native application programming
  – SDK should be released within the next few months
  – All info is reverse engineered for now
  – Growing developer base
Event Driven Multitouch Programming

- Can have three types of events (in order of complexity):
  - Blob Events
  - Discrete Touch Events
  - Gesture Events
Blob Events

- Bitmap of touch activation

Event Driven Multitouch Programming

• Can have three types of events (in order of complexity):
  – Blob Events
  – Discrete Touch Events
  – Gesture Events
iPhone Case Study

• Input system provides discrete touch points
• Differentiates between single touches and chorded gestures
• More may be available soon when SDK is released
• Programmed in Objective-C
• Unofficial documentation in iPhone Open Application Development by Jonathan A. Zdziarski
iPhone: Single Touch Events

- Support for mouseDown, mouseUp, mouseDragged
- mouseEntered, mouseExited, mouseMoved don’t really make sense

@interface MyTable : UITable
{
}
- (void) mouseDown:(struct __GSEvent *)event;
- (void) mouseUp:(struct __GSEvent *)event;
- (void) mouseDragged: (struct __GSEvent *) event;
@end
iPhone: Single Touch Events

- Can extract x, y coordinates from event with:

```c
CGPoint point = GSEventGetLocationInWindow(event);
float x = point.x;
float y = point.y;
```
iPhone Multitouch Events

- By extending the UIView base class, multifinger gestures can be supported.
- Calls different methods:

  ```
  -(void)gestureStarted:(struct __GSEvent)event {
    CGPoint leftFinger = GSEventGetInnerMostPathPosition(event);
    CGPoint rightFinger = GSEventGetOuterMostPathPosition(event);
    [ super gestureStarted: event ];
    /* Handle gesture started event */
  }
  ```

  Pinching and dragging easily supported using the innermost and outermost finger positions.
iPhone Multitouch Events

- Multitouch events contain an array of active touch points
- If more than two finger positions are needed:

```c
for (i = 0; i < event->numPoints; i++) {
    struct GSPathPoint *point = &event->points[i];
    // Process points
}
```

Source: pocketguitar
iPhone Method Correlation

<table>
<thead>
<tr>
<th></th>
<th>Single Touch</th>
<th>Multitouch</th>
</tr>
</thead>
<tbody>
<tr>
<td>Finger is added</td>
<td>(void) mouseDown: (GSEvent *) event</td>
<td>(void) gestureStarted: (GSEvent *) event</td>
</tr>
<tr>
<td>Finger is removed</td>
<td>(void) mouseUp: (GSEvent *) event</td>
<td>(void) gestureEnded: (GSEvent *) event</td>
</tr>
<tr>
<td>Finger is dragged</td>
<td>(void) mouseDragged: (GSEvent *) event</td>
<td>(void) gestureChanged: (GSEvent *) event</td>
</tr>
</tbody>
</table>
iPhone Gesture Events

• By inheriting from base classes that interpret discrete points into gestures, gesture events can be simulated.
• Example: UIScroller class has all scrolling functionality built in, user can then intercept scrolling “events”
Sample Open Source iPhone Application

- PocketGuitar is a guitar synthesizer for iPhone/iPod Touch

Code can be obtained from GoogleCode:
http://code.google.com/p/pocketguitar/
Future Possibilities with Multitouch

- Multiuser collaboration on single device
Zooming User Interfaces (ZUI)

- Rather than containing overlapping windows, have one large work area, zooming in on details as needed.
- Pad++ --> Piccolo
Rich Gestures on the Desktop

- Apple patent on “Gesture Dictionary”
Conclusion

• What will be multitouch killer app?