Recording in Progress

This class is being recorded

Please turn off your video and/or video if you do not wish to be recorded
CMSC436: Programming Handheld Systems
Multi-Touch & Gestures
Today’s Topics

MotionEvents
Touch Handling
Gestures
MotionEvent

Represents a movement in an input device reading

pen, trackball, mouse, finger
MotionEvent

Action Code

State change that occurred

Action Values

Position and movement properties, such as time, source, location, pressure, and more

This lesson focuses on touch events read from a touch screen
MultiTouch

MultiTouch screens emit one movement trace per touch source

Individual touch sources are called pointers
MultiTouch

Each pointer has a unique ID for as long as it is active

MotionEvents can refer to multiple pointers

Each pointer has an index within the event, but that index may not be stable over time
Some MotionEvent actions

ACTION_DOWN
ACTION_POINTER_DOWN
ACTION_POINTER_UP
ACTION_MOVE
ACTION_UP
ACTION_CANCEL
Consistency Objectives

For touch events, Android *tries to guarantee* that touches

- Go down one at a time
- Move as a group
- Come up one at a time or are cancelled

Applications should be tolerant to inconsistency
MotionEvent methods

getActionMasked()
getActionIndex()
getPointerId(int pointerIndex)
getPointerCount()
getX(int pointerIndex)
getY(int pointerIndex)
findPointerIndex (int pointerId)
Handling Touch Events on a View

The View being touched receives
View.onTouchEvent(MotionEvent event)
onTouchEvent() should return true if the
MotionEvent has been consumed; false otherwise
Handling Touch Events with a Listener

View.OnTouchListener defines touch event callback methods

```java
boolean onTouch(View v, MotionEvent event)
```

View.setOnTouchListener() registers listener for Touch callbacks
Handling Touch Events with a Listener

onTouch() called when a touch event, such as pressing, releasing or dragging, occurs

onTouch() called before the event is delivered to the touched View

Should return true if it has consumed the event; false otherwise
Handling Multiple Touch Events

Multiple touches can be combined to form a more complex gesture

Must identify & process the combinations of touches

For example, a double tap consists of:

ACTION_DOWN, ACTION_UP, ACTION_DOWN, ACTION_UP in quick succession
Multi-touch Handling
## Multi-touch Handling Example

<table>
<thead>
<tr>
<th>1\textsuperscript{st} touch</th>
<th>Action</th>
<th>IDs</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>ACTION_DOWN</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td>ACTION_MOVE ...</td>
<td>0</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>2\textsuperscript{nd} touch</th>
<th>Action</th>
<th>IDs</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>ACTION_POINTER_DOWN</td>
<td>1</td>
</tr>
<tr>
<td></td>
<td>ACTION_MOVE ...</td>
<td>0,1</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>1\textsuperscript{st} lift</th>
<th>Action</th>
<th>IDs</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>ACTION_POINTER_UP</td>
<td>0</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>2\textsuperscript{nd} lift</th>
<th>Action</th>
<th>IDs</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>ACTION_UP</td>
<td>1</td>
</tr>
</tbody>
</table>
## Multi-touch Handling Example

<table>
<thead>
<tr>
<th>1(^{st}) touch $\rightarrow$</th>
<th>Action</th>
<th>IDs</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>ACTION_DOWN</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td>ACTION_MOVE ...</td>
<td>0</td>
</tr>
<tr>
<td>2(^{nd}) touch $\rightarrow$</td>
<td>ACTION_POINTER_DOWN</td>
<td>1</td>
</tr>
<tr>
<td></td>
<td>ACTION_MOVE ...</td>
<td>0,1</td>
</tr>
<tr>
<td>2(^{nd}) lift $\rightarrow$</td>
<td>ACTION_POINTER_UP</td>
<td>1</td>
</tr>
<tr>
<td>1(^{st}) lift $\rightarrow$</td>
<td>ACTION_UP</td>
<td>0</td>
</tr>
</tbody>
</table>
## Multi-touch Handling Example

<table>
<thead>
<tr>
<th>Action</th>
<th>ID</th>
</tr>
</thead>
<tbody>
<tr>
<td>ACTION_DOWN</td>
<td>0</td>
</tr>
<tr>
<td>ACTION_POINTER_DOWN</td>
<td>1, 2</td>
</tr>
<tr>
<td>ACTION_POINTER_DOWN</td>
<td>2</td>
</tr>
<tr>
<td>ACTION_MOVE</td>
<td>0, 1, 2</td>
</tr>
<tr>
<td>ACTION_POINTER_UP</td>
<td>1</td>
</tr>
<tr>
<td>ACTION_POINTER_UP</td>
<td>0</td>
</tr>
<tr>
<td>ACTION_UP</td>
<td>2</td>
</tr>
</tbody>
</table>
TouchIndicateTouchLocation

Application draws a circle wherever the users touches the screen

  Circle’s color is randomly selected

Redraws circles as user drags across the screen
TouchIndicateTouchLocation

The size of the circles are proportional to the number of currently active touches
GestureDetector

A class that recognizes common touch gestures
Some built-in gestures include confirmed single tap, double tap, fling
GestureDetector

Activity creates a GestureDetector that implements GestureDetector.OnGestureListener interface.

Activity receives calls to onTouchEvent() when Activity is touched.

onTouchEvent should delegate call to GestureDetector.OnGestureListener.
TouchGestureViewFlipper

Shows a TextView displaying a number
If the user performs a right to left “fling” gesture,
The TextView will scroll off the screen
A new TextView will scroll in behind it
Creating Custom Gestures

GestureBuilder applications let you create & save custom gestures

Search in Google Plat Store
Creating Custom Gestures

GestureLibraries supports loading custom gestures & then recognizing them at runtime
Creating Custom Gestures

Include a GestureOverlayView in your layout.
The Overlay intercepts user gestures and invokes your application code to handle them.
GestureBuilder

Can find Gesture Builder tool in Google Play Store
Stores gestures to /mnt/sdcard/gestures
Copy this file to /res/raw directory
TouchGestures

Application displays a small View with a colored background

User can swipe left and right to cycle between different candidate background colors

Can make a “check” or “X-like gesture” to set or cancel the application’s current background color
Next Time

MultiMedia
Example Applications

TouchIndicateTouchLocation
TouchGestureViewFlipper
TouchGestures