Features, Regions, Gestures:

Components of a Generic Gesture Recognition Engine

Florian Echtler
<echtler@in.tum.de>

Andreas Butz
Gudrun Klinker

June 20th, 2010
Generic Gesture Recognition
Architectural Overview

- difficult question: *what is a gesture?*

- answer within this context: *any motion(s) which a person executes in order to trigger a certain response*

- directly leads to the next question: *how can these motions be described?*

- current focus: planar input and output
- extension to 3D entirely possible
Generic Gesture Recognition
Abstract Description of Gestures

- **Features**
  - simple geometric/numerical properties of motion data
  - e.g.: motion vector, relative rotation, travelled path ...
  - further specification through filters/boundary values
  - can take history of input blob(s) into account
- **Regions:**
  - closed polygons, corners defined in screen coordinates
  - extension of „traditional“ WIMP-UI window objects
- **Gestures**
  - sequences of features
  - pre-defined by capability description or customized by application
  - sent to application as asynchronous events
Generic Gesture Recognition

Abstract Description of Gestures

Three core elements: Regions, Gestures, Features
Generic Gesture Recognition
List of Features

- Motion – average motion vector
- Rotation – rotation around center of mass
- Scale – scaling w.r.t. center of mass
- BlobCount – number of blobs inside region
- BlobDimensions – describes equivalent ellipse
- BlobOrientation – orientation according to main axis
- BlobPosition – absolute position of blob
- Path – recognize „shape-based“ gestures („$1 recognizer“)
- BlobDelay – e.g. for „tap-and-hold“ gestures
- BlobID – match specific object or user
- BlobParent – match „parent“ object
Generic Gesture Recognition
Simple Example: Rotation

- simple example: „rotate“ gesture
- contains one feature: relative rotation

- result value generated through...
  - multi-finger rotation or
  - object rotation or
  - mouse wheel or ...

- mapping is hardware-dependent
- part of capability description
  - application doesn't have to care
Generic Gesture Recognition
Examples (continued)

- example: match 2-finger path gesture for specific user
- Gesture HandRotate
  - Feature BlobParent: \(\text{parent} == 42\) \&\& \(\text{type} == \text{user}\)
  - Feature Path: \(\text{path} == (x,y)\ldots\) \&\& \(\text{type} == \text{finger}\)
  - Feature BlobCount: \(\text{count} == 0\) \&\& \(\text{type} == \text{finger}\)
Generic Gesture Recognition

Summary

- generic „motion matching“ engine
  - hardware-independent
  - gestures can be reconfigured (on-the-fly!)

- approach towards standardization of...
  - ... hardware
  - ... interaction metaphors
  - ... libraries

- speed up development/prototyping process
  - allow developers to focus on the interface itself

References:
- libTISCH - http://tisch.sf.net/
- A Multitouch Software Architecture.
Thank you for your attention.

Questions & comments are welcome!