Kinesthetic modality components for the W3C Multimodal Architecture.

Florian Metze, Thomas Ziem, and Ingmar Kliche

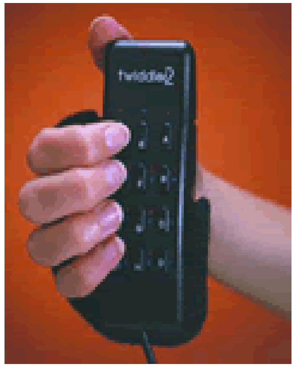
Workshop on W3C's Multimodal Architecture and Interfaces
November 16-17, 2007, Fujisawa, Japan.

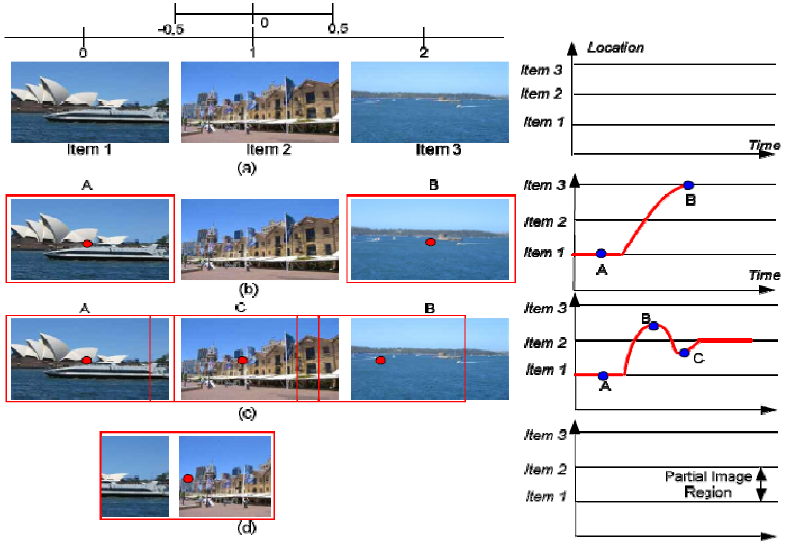
Agenda.

- **Introduction, Use cases, Motivation**
- Current implementations of kinesthetic input modality components
- Client vs. server side interaction management
- Integration into the W3C Multimodal Architecture
- A kinesthetic output component
- Conclusion

Kinesthetic modality components.
Introduction of kinesthetic input.

- Kinesthetic input for operating mobile devices, for example:
  - Twiddler (text input)
  - Nintendo Wii
  - Samsung, Android OS, others
- Two approaches
  - Accelerometers sensing physical movements (spatial translation)
  - Optical input sensors (cameras)
    - □ Rotation and translation
    - □ No extra hardware cost
- Main benefit: single-handed operation
  - Gaming
  - List navigation & selection
  - Mouse replacement for mobile devices
  - Short-cut gestures
  - Context recognition
- Computation needed: translate measurements into commands and feedback



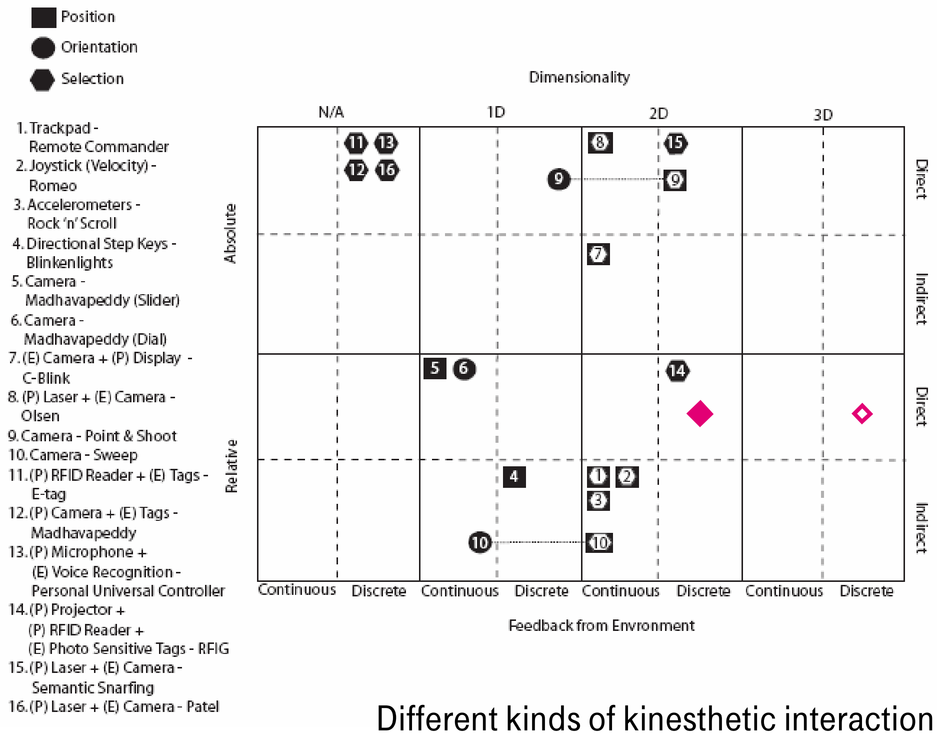


(from [Cho])

Kinesthetic modality components.
Use cases for kinesthetic input.

- Two main kinds of interaction/ feedback
  - Continuous (i.e. scrolling)
  - Discrete (i.e. list navigation)
- Push-to-activate?
- Relative or absolute positioning?

- Natural first candidate for integration in W3C framework/ standards:
  - Discrete interaction
  - Relative positioning
  - Indirect interaction
  - Selection in lists, menus
  - ➢ One-hand control of menu-based interfaces on mobile devices
  - ➢ Add shortcuts for frequently used functions



■ Position
● Orientation
⬢ Selection

1. Trackpad - Remote Commander
2. Joystick (Velocity) - Romeo
3. Accelerometers - Rock 'n' Scroll
4. Directional Step Keys - Blinkenlights
5. Camera - Madhavapeddy (Slider)
6. Camera - Madhavapeddy (Dial)
7. (E) Camera + (P) Display - C-Blink
8. (P) Laser + (E) Camera - Olsen
9. Camera - Point & Shoot
10. Camera - Sweep
11. (P) RFID Reader + (E) Tags - E-tag
12. (P) Camera + (E) Tags - Madhavapeddy
13. (P) Microphone + (E) Voice Recognition - Personal Universal Controller
14. (P) Projector + (P) RFID Reader + (E) Photo Sensitive Tags - RFIG
15. (P) Laser + (E) Camera - Semantic Snarfing
16. (P) Laser + (E) Camera - Patel

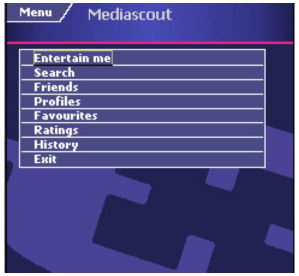
Different kinds of kinesthetic interaction
(from [Ballagas])

Kinesthetic modality components.
Application using kinesthetic input.

Example scenario 'Mediascout'

- While on the move, get recommendations from tonight's IPTV program.
- Multi-modal application for ease of use exploring different device capabilities.

Main menu

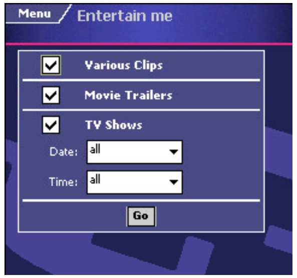


Menu / Mediascout

Entertain me
Search
Friends
Profiles
Favourites
Ratings
History
Exit

Entertain me

Settings

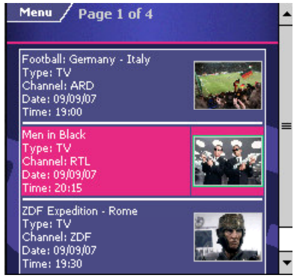


Menu / Entertain me

☑ Various Clips
☑ Movie Trailers
☑ TV Shows
Date: all
Time: all
Go

Select type

Recommended media



Menu / Page 1 of 4

Football: Germany - Italy
Type: TV
Channel: ARD
Date: 09/09/07
Time: 19:00

Men in Black
Type: TV
Channel: RTL
Date: 09/09/07
Time: 20:15

ZDF Expedition - Rome
Type: TV
Channel: ZDF
Date: 09/09/07
Time: 19:30

Play clip

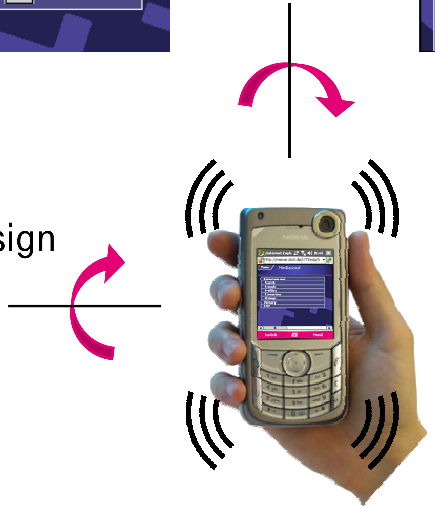
Rate content

Set reminder

...

Design considerations:

- Screens using familiar interface/ design
- One-hand operation
- Voice and haptic feedback



Example Video!

Deutsche Telekom Laboratories

Kinesthetic modality components.

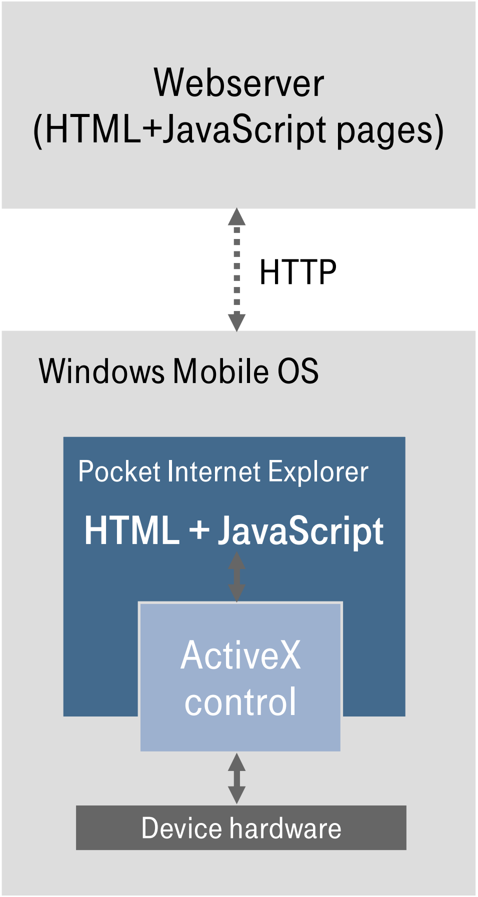
Kinesthetic input.

Current implementation approach

- HTML/JavaScript based GUI component
- ActiveX control accesses device hardware (inertial sensors or camera pictures)
- ActiveX control implements signal processing (either processing of inertial sensors signal or camera pictures), generates events for predefined motion patterns (up, down, left, right)
- ActiveX control integrated into Windows Mobile Pocket Internet Explorer
- JavaScript interacts with ActiveX control
- JavaScript based interaction management (client side)

Open issues

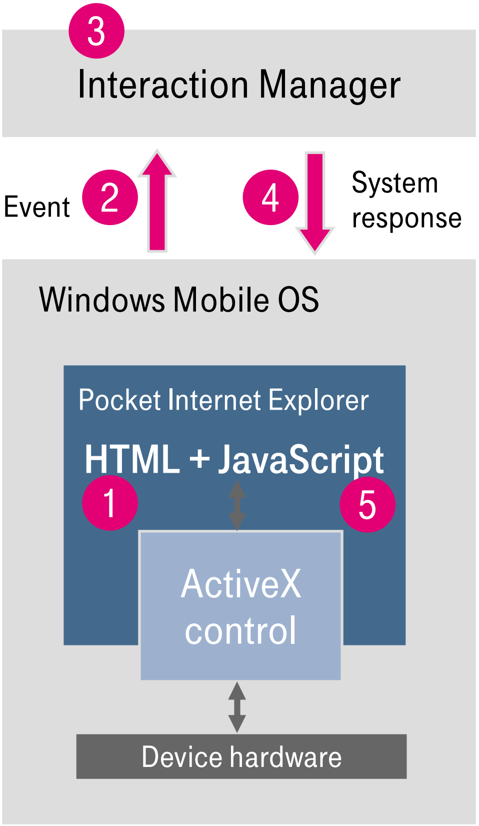
- Limitation to Microsoft Windows Mobile devices
- Motion patterns are currently burned into the ActiveX control, a pattern description language might be useful, comparably to SRGS for speech recognition
- Could InkML help?



Kinesthetic modality components.
Kinesthetic input.

The challenge: server side interaction management

- Server side interaction management provides more flexible and powerful functionalities (e.g. semantic interaction management)
- Drawback: Events need to be sent to the server side interaction management
- W3C Multimodal Framework defines a Modality component API (MMI lifecycle events)
- Open issues:
  - Latency of event transport introduces delay of system reactions.



1. Event at ActiveX control, handled within JavaScript
2. Event sent to server side interaction management, e.g. using AJAX
3. Calculation of system response
4. Transmission of system response to client
5. Execution of system response

Kinesthetic modality components.
Kinesthetic input.

Approaches for distribution of interaction management

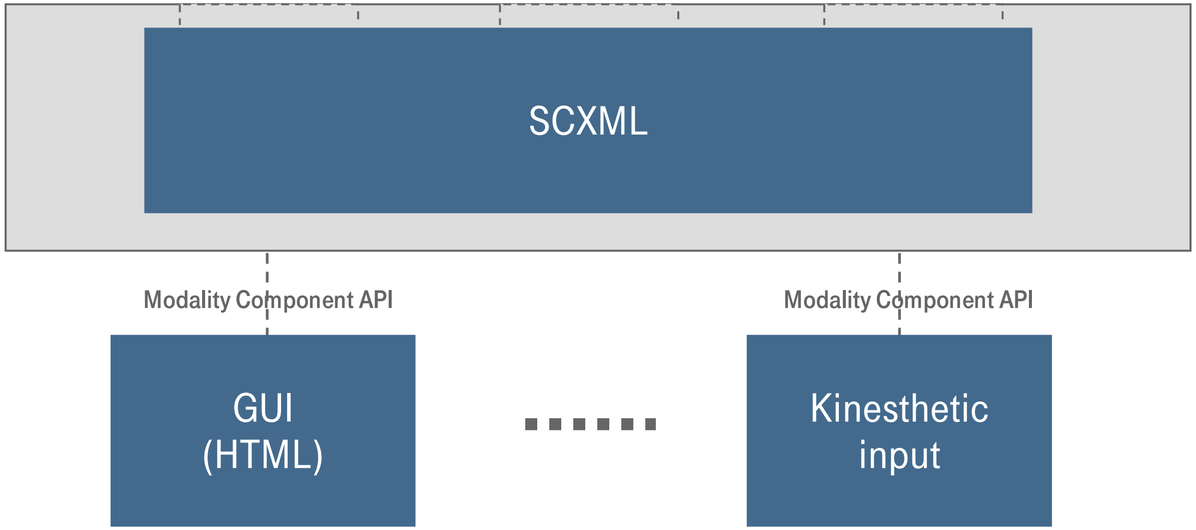
- Client side interaction management:
  - Advantage: fast
  - Disadvantage: allows less complexity, control logic platform dependent
- Distributed interaction management
  - Advantage: provides potentially more functionality for interaction management
  - Disadvantage: more effort and complexity for synchronization of server and client side interaction management
- Server side interaction management:
  - Advantage: powerful interaction management, centralized and client platform independent control logic
  - Disadvantage: Latency, Online connectivity required

Approaches for distribution of signal processing

- Current approach: client side signal processing
- distributed ?
- server side ?
- Drawback: signals need to be transferred to the server, much bandwidth required (e.g. for images).

Kinesthetic modality components.
Kinesthetic input.

W3C Multimodal Architecture

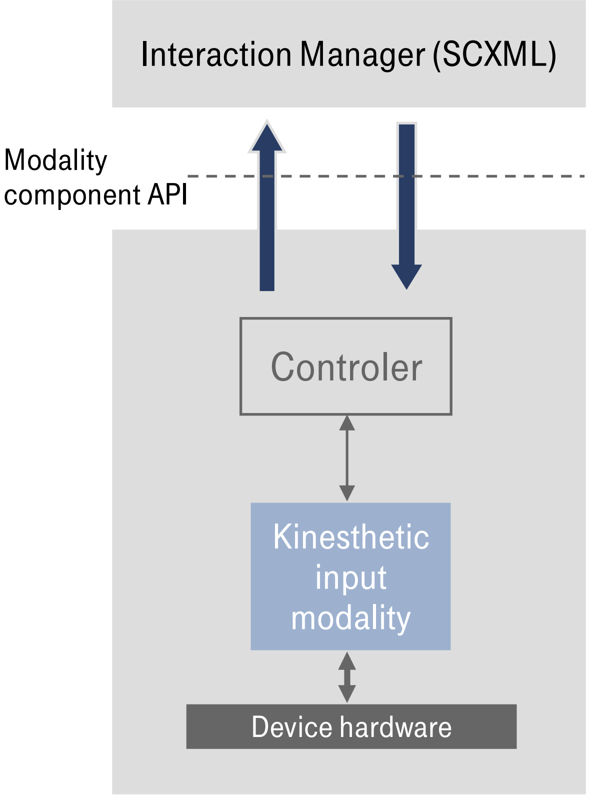


Kinesthetic modality components.
Kinesthetic input.

Integration into the W3C Multimodal Architecture

- SCXML based interaction manager
- Server side interaction management
- Kinesthetic separate modality component or combination of kinesthetic and GUI modality component ?
- MMI- Lifecycle events to be used:
  - Data ???
  - Notify ???

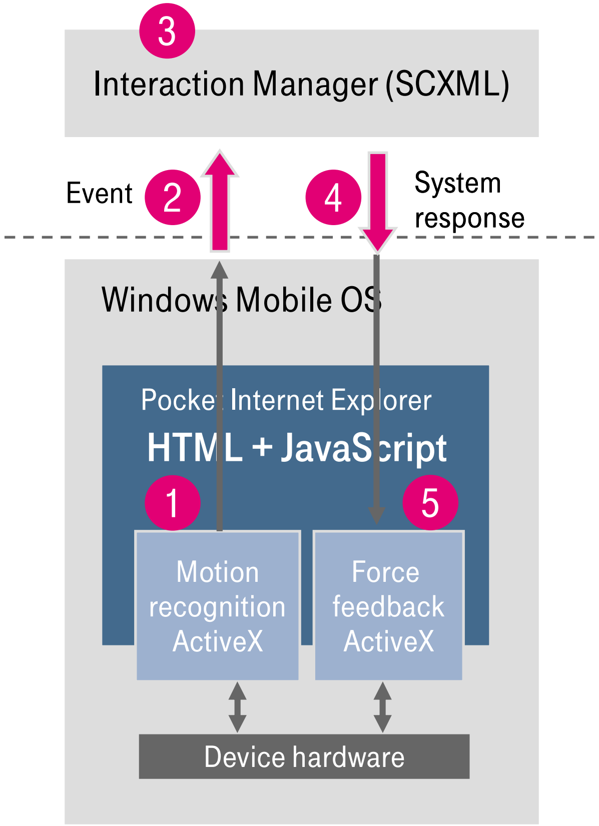
```
<mmi:Notify
  source="someURI"
  context="someURI" >
  <mmi:data>
    <emma:interpretation
      emma:confidence="0.53"
      emma:mode=“motion" >
      <pattern>up</pattern>
    </emma:interpretation>
  </mmi:data>
</mmi:Notify>
```



Kinesthetic modality components.
Force feedback.

Kinesthetic feedback

- Mobile phones provide vibrations as notification functionality
- A prototype of a kinesthetic output components has been implemented (using vibration functionality of mobile phones)
  - use case: feedback for recognition success on kinesthetic input
- Integration with W3C Multimodal Framework:
  - lifecycle event: Data or Notify



1 Event at ActiveX control, handled within JavaScript
2 Event sent to server side interaction management, e.g. using AJAX
3 Calculation of system response
4 Transmission of system response to client
5 Execution of system response

Deutsche Telekom Laboratories

Kinesthetic modality components.
Open issues.

- Latency and real-time requirements for kinesthetic control
  - Server-side interaction management
  - Feasibility of distributed multi-modal framework
- Description language for pattern matching rules
  - Describing motion input
  - Comparable to grammars for speech recognition
- Pattern and event space cardinality
  - Digital/ discrete
  - Analog/ continuous
- Dependency on type of sensor
- Integration of kinesthetic feedback
- Initialization/ Normalization
- Activation (push-to-move)

Kinesthetic modality components.
Conclusion.

- Presented kinesthetic input in multi-modal framework
  - Many isolated efforts exist
  - Tied to specific applications
  - Proprietary integration of client device functionalities
- Advantages of kinesthetic input are plenty
  - One-hand operation
  - Eyes-free
- Development of standards seems desirable
  - Need to identify a more complete set of needs
  - Identify synergies with existing frameworks

- But: How to ease implementation or integration of client functionalities into web browsers?
  - Embedded ASR, TTS
  - Kinesthetic sensors
  - others

Literature and References.
Used in this presentation.

- [Ballagas] *The smart phone: a ubiquitous input device;* Ballagas, R.   Borchers, J.   Rohs, M. Sheridan, J.G.; Pervasive Computing, IEEE; Vol. 5, Issue 1; Jan.-March 2006
- [Cho] *Dynamics of tilt-based browsing on mobile devices*; Sung-Jung Cho, Changkyu Choi, Younghoon Sung, Kwanghyeon Lee; Yeun-Bae Kim; Roderick Murray-Smith; CHI '07 extended abstracts on Human factors in computing systems; ACM, 2007; San Jose, CA, USA