Prototyping Physical User Interfaces

Short Course @ IT University of Göteborg

September 2003

Albrecht Schmidt
Media Informatics Group
University of Munich, Germany
http://www.medien.informatik.uni-muenchen.de/en/team/schmidt

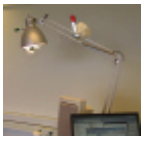
Brief Course Outline

- **Breaking Interface Conventions?**
- **Exercise – creating a cooperative multi user game**
- **Nature and Value of Physical Prototyping**
- **Break**
- **Smart-its basics**
- **Smart-its enhanced light**
- **Lunch break**
- **Students project (afternoon) Smart-its enhanced light**
- **Smart-its enhanced light – results**
- **Building Smart-its hardware**
- **Break**
- **Developing Smart-its Software**
- **Smart-its Examples**
- **Wrap-Up**

Smart-Its enhanced light

- **Task**
  Write a software that accesses the sensors attached to the lamp and transforms them into a higher level representation (textual or graphical).

State 1 | State 2 | State 3


Results

Development Steps

- Recording data in the situation
  - Data samples for all states
  - Data includes variations
- Analysing the data (manually or automatic), e.g.
  - "looking at the data"
  - Statistical analysis
  - Neural networks …
- Creating and Implementing a mapping from stimulus to state, e.g.
  - Define and coding rules
  - Training a neural network

Naïve approach – sample vectors for each state

- Sensor data at a time seen as vector
  V = (touch, pir, light1, light2, accx, accy, accz, accu)
- Record a number of vectors for each state and calculate a typical sample (e.g. average)
  - State 1: $V_{11}, V_{12}, \ldots, V_{1n}$ average resulting in V_1
  - State 2: $V_{21}, V_{22}, \ldots, V_{2n}$ average resulting in V_2
  - …
  - State m: $V_{m1}, V_{m2}, \ldots, V_{mn}$ average resulting in V_m
- A new stimulus V_x is compare to all sample vectors V_i and the state with the minimal distance $| V_x - V_i |$ is selected. The distance may be used as indicator of the certainty of the decision

Example

(just using acceleration values, set 1&2 as sample, 3 for test)

- State 1:
  (112,141,109,154,135,157,130,119)
  (107,143,108,154,130,159,131,123)
  (103,141,113,158,130,159,130,122)
- State 2
  (143,121,106,119,108,157,132,144)
  (142,122,107,119,108,157,132,144)
  (144,124,107,117,109,158,131,143)
- State 3
  (99,115,141,162,132,158,131,120)
  (101,115,141,161,133,159,131,120)
  (99,114,141,162,133,158,131,121)

Example – average vectors

(just using acceleration values, set 1&2 as sample, 3 for test)

- State 1:
  (112,141,109,154,135,157,130,119) sample 1
  (107,143,108,154,130,159,131,123) sample 2
  (110,142,119,154,133,158,131,121) avg-state1
- State 2
  (143,121,106,119,108,157,132,144) sample 1
  (142,122,107,119,108,157,132,144) sample 2
  (143,122,107,119,108,157,132,144) avg-state2
- State 3
  (99,115,141,162,132,158,131,120) sample 1
  (101,115,141,161,133,159,131,120) sample 2
  (100,115,141,162,133,159,131,120) avg-state3

Example - comparison

(just using acceleration values, set 1&2 as sample, 3 for test)

- (103,141,113,158,130,159,130,122) sample 1.3
  (110,142,119,154,133,158,131,121) avg-state1 → 24
  (143,122,107,119,108,157,132,144) avg-state2 → 152
  (100,115,141,162,133,159,131,120) avg-state3 → 67
- (144,124,107,117,109,158,131,143) sample 2.3
  (110,142,119,154,133,158,131,121) avg-state1 → 148
  (143,122,107,119,108,157,132,144) avg-state2 → 9
  (100,115,141,162,133,159,131,120) avg-state3 → 189
- (99,114,141,162,133,158,131,121) sample 3.3
  (110,142,119,154,133,158,131,121) avg-state1 → 69
  (143,122,107,119,108,157,132,144) avg-state2 → 179
  (100,115,141,162,133,159,131,120) avg-state3 → 4

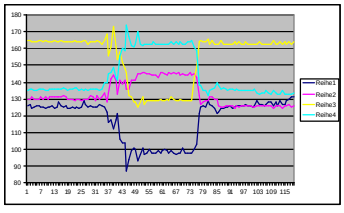
Disclaimer … calculations are done without a calculator – may include a small error :)

Problems with the approach and some simple solutions

- The range auf components is different, e.g. 0 or 1 for touch 0-191 for light
  → Normalization
- Some values do not contribute to the state or contribute less
  → weight/factor for components
- Sample vector may not be a good representation for the states
  → Keep several sample vectors for each state and compare to all of them. Use the sum of all difference as measure
- Further issues
  – Selection of the distance function
- Look into math, statistics, machine learning, and neural networks textbooks – there are many possibilities

Example using Excel

- Recording a text file
- Importing it into Excel
- Plotting it



Implicit Interaction and Sensor-based User Interfaces

- Implicit Interaction – user is observed by the system
- Example: outdoor light in front of the house that switches on when someone walks by
- Sensors:
  – light level
  – activity
- Actuator
  – light (on/off)
- Rules
  – `if (dark & movement) then light(on)`

Is it as simple as it looks?

More design decisions!
e.g. when switch it off?