User Interaction

CS 465 lecture 21

Bad Computer Interfaces
(more on this later)


User Interaction

• Input devices
• User-centered design
• GUIs and GUI design
• Interaction with 2D and 3D scenes

Input devices

• Discrete events
  – Keyboard
  – Function keys
  – Mouse buttons
  – Game controller buttons
    • Including multi-way controllers (pseudo-joysticks)

• Valuators: generate continuous values
  – Rotary knobs (relative or absolute)
    • Recentering or free
  – Joysticks (two valuators in one)
Input devices

• Locators: give a continuous 2D position
  – Mechanical mouse (trackball is the same)
    • Two axes with optical encoders
      – Integrate rate of pulses on each axis
      – Result = position
  – Optical mouse
    • Image sensor looking out the bottom
      – Shift and correlate to estimate motion per frame
      – Integrate motion to get position
    – Mouse velocity scaling

Input devices

• Locators, cont.
  – Pen tablet
    • Directly senses absolute stylus position
    • Often used directly over a display
      – PDA
      – Tablet PC
    – Absolute vs. relative
    – Direct vs. indirect

Input devices

• Multidimensional controllers
  – More exotic devices
  – Spaceball
  – Data glove
  – 3D tracker
    • Magnetic
    • Acoustic
    • Optical

User-centered design

• Usability is an often undervalued objective in the design of devices
• Computers and computerized devices are some of the worst offenders!
• This section based heavily on the classic book by Donald Norman, *The Design of Everyday Things*
User-centered design

• We are often frustrated by the artifacts we build to work for us
  – door handles, water faucets, …
  – stereos, microwaves, …
  – airplanes, industrial equipment, …
  – computers (always!)
• …and we blame ourselves
  – I’m so dumb, I always push the pull door
  – I would need an engineering degree to figure this thing out!
  – The accident was caused by pilot error
• If we work at it, we can avoid many of these problems!

Psychopathology of everyday things

• (Norman’s phrase)
• Sometimes the interaction between a device and human behavior defeats the device’s purpose
• Best explored by examples

Doors: push or pull?
Ovens

- Two devices with the same basic function
  - one computer controlled, one not
  - one universally hated UI, one perfectly functional

Concepts of user-centered design

- Affordances
  - objects indicate by their appearance how they can be used
- Mappings
  - when several controls, directions, etc. exist, which is which?
- Conceptual models
  - don’t mislead the user about what is inside
- Visibility and feedback
  - let the user see what is going on
- Knowledge in the head vs. in the world
  - well-chosen cues help the user remember what to do
- Conventions
  - when all else fails, make the user memorize once

Cameras

Affordances

- Which turn, which slide, which push?
Affordances

• The big jog/shuttle knob has some hidden meanings...

Mappings

• Which control is for which burner?

Mappings

• Lighting controls: which switch controls what light(s)?
  – what lights are even being controlled? Which operate independently?
Mappings

- Which way do you turn?

Conceptual models

- Norman’s example: refrigerator adjustments
  - appearance: separate controls for fridge and freezer
  - reality: one cooling system, one thermostat (second control is for cold air distribution)
- My example: Microsoft word 1989 vs 2006
  - version 4: paragraph attributes associated with "paragraph mark"
    - this did lead to some surprising behavior
  - version 2004: same underlying model, but layers of "helpful" behaviors prevent users from discovering it

Visibility and feedback

- Buttons that light up when they are on can reveal state
  - and with a nice built-in mapping back to the control
- Faucet handles again...

Knowledge in the world

- Brief, well-designed markings
Standardization

- The QWERTY keyboard
  - cost to transition to a mildly better system is high
- Complex designs that are not new are not as hard

Graphical User Interfaces (GUIs)

- Using visual display coupled with pointing to present complex choices to the user
- The dominant mode of user interface today
- Many flavors exist, but all present similar widgets
  - Icons (objects)
  - Buttons (actions)
  - Menus (collections of choices/actions)
  - Lists
  - Trees
- All the same principles apply as for physical UIs
  - only you have to create everything: affordances, visible state, etc. don’t happen naturally.

Affordances

- Pressing
  - often suggested by raised appearance

- Sliding, adjustment
  - often suggested by track

- Active vs. inactive

Lack of affordance

- Web links (often)
  - how do I know what I can click on?

- Window controls
  - I can grab this window anywhere to move it. How do I know this?
Mapping

- Scroll bars: horizontal and vertical
- Software often has very arbitrary mappings

Other features

- Feedback
  - during control operation
  - revealing control state
- Conceptual models
  - a higher level question…
- Knowledge in the world
  - drop-down menus are a nice example
  - contrast to keyboard commands that you have to just know

Basic interaction tasks

- Positioning
- Selection
  - From large/continuous set (part of image)
    - Rectangle, lasso
  - From unorganized discrete set (icons on desktop)
    - Click and modifier-click, or drag area
  - From linearly organized set
    - Selection from list box
  - From hierarchically organized set
    - Drop-down menus, trees, columnar lists, …

3 Hierarchy UIs