Human Information Processing

CS160: User Interfaces
John Canny
Review

• Children and Learning
  – Zone of Proximal Development
  – Metacognition

• Teaching Techniques
  – Connect with experience, make ideas concrete
  – Give students a say over the outcome
  – Developing skills for thinking about learning

• Learning Games
  – Historical strategy
  – Math, Music, Environmental Science
Model Human Processor

- Simplified, system-level view of the human brain
- Shows paths of information flow
- Types of “processor” and amount of parallelism
- Approximate timings
- Memory capacities (also approximate)
Human Info. Processor
- Perceptual
- Cognitive
- Motor (will discuss later)
- Working memory
- Long-term memory

Unified model
- Probably inaccurate
- Predicts perf. well
- Very influential

Key
- $\delta$ = half-life
- $\mu$ = capacity
- $\kappa$ = memory type
Perceptual Processor

Physical store from our senses: sight, sound, touch, …
- Code directly based on sense used
  - Visual, audio, haptic, … features

Selective
- Spatial
- Pre-attentive: color, direction…

Capacity
- Example: 17 letters for visual image store

Decay 200ms

Recoded for transfer to working memory
- Progressive: 10ms/letter
How many 3’s

[based on slide from Stasko]
How many 3’s

[based on slide from Stasko]
Visual Pop-Out: Color

http://www.csc.ncsu.edu/faculty/healey/PP/index.html
Visual Pop-Out: Shape

http://www.csc.ncsu.edu/faculty/healey/PP/index.html
Feature Conjunctions

http://www.csc.ncsu.edu/faculty/healey/PP/index.html
# Preattentive Features

<table>
<thead>
<tr>
<th>Orientation</th>
<th>Curved/straight</th>
<th>Shape</th>
</tr>
</thead>
<tbody>
<tr>
<td><img src="orientation.png" alt="Image" /></td>
<td><img src="curved-straight.png" alt="Image" /></td>
<td><img src="shape.png" alt="Image" /></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Size</th>
<th>Number</th>
</tr>
</thead>
<tbody>
<tr>
<td><img src="size.png" alt="Image" /></td>
<td><img src="number.png" alt="Image" /></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Gray/value</th>
<th>Enclosure</th>
<th>Convexity/concavity</th>
</tr>
</thead>
<tbody>
<tr>
<td><img src="gray-value.png" alt="Image" /></td>
<td><img src="enclosure.png" alt="Image" /></td>
<td><img src="convexity-concavity.png" alt="Image" /></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Addition</th>
<th>Juncture</th>
<th>Parallelism</th>
</tr>
</thead>
<tbody>
<tr>
<td><img src="addition.png" alt="Image" /></td>
<td><img src="juncture.png" alt="Image" /></td>
<td><img src="parallelism.png" alt="Image" /></td>
</tr>
</tbody>
</table>

[Information Visualization. Figure 5. 5 Ware 04]
Perceptual Processor

Cycle time
  - Quantum experience: 100ms
    • Percept fusion
Perceptual Processor

Cycle time
- Quantum experience: 100ms
  - Percept fusion
- Frame rate necessary for movies to look continuous?
  - time for 1 frame < Tp (100 msec) -> 10 frame/sec.
  - Note: flicker sensitivity much faster (60-70 f/s)
- Max. morse code rate can be similarly calculated

Perceptual causality
- Two distinct stimuli can fuse if the first event appears to cause the other
- Events must occur in the same cycle
Perception of Causality [Michotte 46]

Michotte demonstration 1. What do you see? Most observers report that the red ball hit the blue ball. The blue ball moved “because the red ball hit it.” Thus, the red ball is perceived to “cause” the red ball to move, even though the balls are nothing more than color disks on your screen that move according to a program.

http://cogweb.ucla.edu/Discourse/Narrative/Heider_45.html
Perception of Causality  [Michotte 46]

Michotte demonstration 1. What do you see? Most observers report that the red ball hit the blue ball. The blue ball moved “because the red ball hit it.” Thus, the red ball is perceived to “cause” the red ball to move, even though the balls are nothing more than color disks on your screen that move according to a program.

http://cogweb.ucla.edu/Discourse/Narrative/Heider_45.html
Perceptual Processor

Cycle time
- Quantum experience: 100ms
  - Causality
Working Memory

Access in chunks
- Task dependent construct
- 7 +/- 2 (Miller)

Decay
- Content dependant
  - 1 chunk 73 sec
  - 3 chunks 7 sec
- Attention span
  - Interruptions > decay time
Long Term Memory

Very large capacity
  – Semantic encoding

Associative access
  – Fast read: 70ms
  – Expensive write: 10s
    • Can also move from WM to LTM via rehearsal

Context at the time of acquisition key for retrieval
Cognitive Processor

Cycle time: 70ms
  - Can be modulated

Typical matching time
  - Digits: 33ms
  - Colors: 38ms
  - Geometry: 50ms…

Fundamentally serial
  - One locus of attention at a time
    • Eastern 401, December 1972
      - Crew focused on landing gear indicator bulb,
      - Aircraft is loosing altitude (horn, warning indicator…),
      - Aircraft crashed in the Everglades
      - see “The Human Interface” by Raskin, p25
    • But what about driving and talking?
Recognize-Act Cycle

Are $Q$ and $X$ letters?
Cognitive Processor

- Page 70 of Card Moran and Newell

- Clocks starts when 2\textsuperscript{nd} letter is flashed
- Move letter x into visual store WM $T_p$  
- Classify the codes in WM as letters + 2 $T_c$
- Match the letters + $T_c$
- Initiate motor response + $T_c$
- Process motor command + $T_m$
- Approx 450 (180-980) ms
Human Interaction Loops
(Newell)

- System loop: $10^8$ s (years)
- Design loop: $10^5$ (days)-$10^7$ s (months)
- Task loop: $10^2$-$10^4$ s (weeks)
- Unit task loop: 10 s
- Operator loop: 1 s
- Evaluation
- Motor loop: 0.1 s
- Execution

- Biological band: $10^{-4}$-$10^{-2}$ s
- Cognitive band
- Rational Band
- Social Band
Principles of Operation

Interface should respect limits of human performance
- Preattentive features pop-out
- Events within cycle time fuse together
- Causality

Recognize-Act Cycle of the cognitive processor
- On each cycle contents in WM initiate cognitive actions
- Cognitive actions modify the contents of WM

Discrimination Principle
- Retrieval is determined by candidates that exist in memory relative to retrieval cues
- Interference by strongly activated chunks
  - Two strong cues in working memory
  - Link to different chunks in long term memory
Memory
Simple Experiment

Volunteer

Start saying **colors** you see in list of words
  – When slide comes up
  – As fast as you can

Say “done” when finished

Everyone else time it…
Simple Experiment

Do it again

Say “done” when finished
Stage Theory

- Sensory Image Store: decay
- Working Memory: decay, displacement
- Long Term Memory: decay? interference?

Maintenance rehearsal

Chunking/Elaboration
Stage Theory

Working memory is small
  – Temporary storage
    • decay
    • displacement

Maintenance rehearsal
  – Rote repetition
  – Not enough to learn information well
LTM and Elaboration

Recodes information

Organize (chunking)

Relate new material to already learned material

Link to existing knowledge, categories

Attach meaning
  – Make a story
LTM Forgetting

Causes for not remembering an item?
   1) Never stored: encoding failure
   2) Gone from storage: storage failure
   3) Can’t get out of storage: retrieval failure

Interference model of forgetting
   – One item reduces ability to retrieve another
   – Proactive interference (3)
     • Earlier learning reduces ability to retrieve later info.
   – Retroactive interference (3 & 2)
     • Later learning reduces the ability to retrieve earlier info.
Recognition over Recall

Recall
- Info reproduced from memory

Recognition
- Presentation of info helps retrieve info (helps remember it was seen before)
- Easier because of cues to retrieval

We want to design UIs that rely on recognition!
Recall

Write names of the 7 dwarves in Snow White?
Recall

Write names of the 7 dwarves in Snow White?
Recognition

- Grouchy
- Sneezy
- Smiley
- Sleepy
- Pop
- Grumpy
- Cheerful
- Dopey
- Bashful
- Wheezy
- Doc
- Lazy
- Happy
- Nifty
Recognition

- Grouchy
- **Sneezy**
- Smiley
- **Sleepy**
- Pop
- **Grumpy**
- Cheerful
- **Dopey**
- Bashful
- Wheezy
- **Doc**
- Lazy
- **Happy**
- Nifty
Facilitating Retrieval: Cues

Any stimulus that improves retrieval

- Example: giving hints
- Other examples in software?
  - icons, labels, menu names, etc.

Anything related to

- Item or situation where it was learned
Decision Making and Learning
Power Law of Practice

• Task time on the nth trial follows a power law

\[ T_n = T_1 n^{-a} + c \]

where \( a = .4 \), \( c = \) limiting constant
– You get faster the more times you do it!

Applies to skilled behavior (sensory & motor)

Does not apply to
– Knowledge acquisition
– Improving quality
Problem solving

(a)

Time to solution, sec

Number of problems

(b)

Log, sec

Log problems

Problem solving
Manual skills

Writing books
Stages of skill acquisition

Example: Using a manual transmission

Cognitive
- Verbal representation of knowledge

Associative
- Proceduralization
  - Form of chunking

Autonomous
- More and more automated
- Faster and faster
- No cognitive involvement
  - Difficult to describe what to do
Fitts’ Law
Motor Processor

Receive input from the cognitive processor
Execute motor programs
  – Pianist: up to 16 finger movements per second
  – Point of no-return for muscle action
Hand movement based on series of microcorrections

\[ X_i = \text{remaining distance after } i\text{th move} \]

relative movement accuracy remains constant \[ \frac{X_i}{X_{i-1}} = \varepsilon \]
Hand movement based on series of microcorrections

= remaining distance after ith move

relative movement accuracy remains constant \( \frac{X_i}{X_{i-1}} = \varepsilon \)

Each microstep reduces the log of the error by \( \log \varepsilon \)

The number of steps grows with \( \log \frac{D}{S} \)
Fitts’ Law

\[ T = a + b \log_2(D/S + 1) \]

- Models well-rehearsed selection task
- \( T \) increases as the \textit{distance} to the target increases
- \( T \) decreases as the \textit{size} of the target increases

\( a, b \) = constants (empirically derived)
\( D \) = distance
\( S \) = size

ID is Index of Difficulty = \( \log_2(D/S+1) \)
Considers Distance and Target Size

\[ T = a + b \log_2 (D/S + 1) \]

Target 1 \quad Target 2

Same ID → Same Difficulty
Considers Distance and Target Size

\[ T = a + b \log_2(D/S + 1) \]

Smaller ID $\rightarrow$ Easier
Considers Distance and Target Size

\[ T = a + b \log_2 (D/S + 1) \]

Larger ID → Harder

Target 1

Target 2
Experimental Data

TARGET WIDTHS
- ▲ 2 in.
- ◊ 1 in.
- ● 1/2 in.
- □ 1/4 in.

Time Per Movement (sec)

$log_2 (D/S + .5)$ Corrected for Errors
Microsoft Toolbars offer the user the option of displaying a label below each tool. Name at least one reason why labeled tools can be accessed faster. (Assume, for this, that the user knows the tool.)
1. The label becomes part of the target. The target is therefore bigger. Bigger targets, all else being equal, can always be accessed faster, by Fitt's Law.

2. When labels are not used, the tool icons crowd together.
Tool Matrix Example

You have a palette of tools in a graphics application that consists of a matrix of 16x16-pixel icons laid out as a 2x8 array that lies along the left-hand edge of the screen. Without moving the array from the left-hand side of the screen or changing the size of the icons, what steps can you take to decrease the time necessary to access the average tool?
Tool Matrix Example

1. Change the array to 1x16, so all the tools lie along the edge of the screen.

2. Ensure that the user can click on the very first row of pixels along the edge of the screen to select a tool. There should be no buffer zone.
Summary

• Model human processor: simplified view of cognitive processing for simple tasks
• Perceptual processor
  – Speed, sensitivity to change, causality, capacity, processing
• Memory
  – Computing time for recognition tasks
  – Long-term memory: discrimination
  – Recognition over recall
• Decision making and learning
• Fitt’s Law