Lecture Summary

Table of Contents

Info

Study Part

Lecture 1 – Hoelscher

The guiding questions for this lecture focus around the connection between psychology and IT (behavioral science), and models human cognitive performance for scientific insights (basic research) and the design of IT systems (applied research).

User-friendly Design – Usability

Psychology/cognitive science provides an inventory of methods for Human-Computer-interaction. Such methods include usability testing, expert analysis, design principles, models of human thinking as a tool for predicting behavior. A commonly used too is eye tracking.

Information processing and cognition

This includes classic media (books, newspapers, phone …) and information-based systems which focus on the end-user (e.g. e-commerce, advertising, e-learning, analysis of the user's needs etc.). A related field is media psychology e.g. joy of use, how sources are selected and the social context.

Cognition means "information processing" by means of a mental representation and an algorithmic process. Humans and computers are both information-processing systems and serve as model for one another.

Lecture 2 – Hoelscher

Cognitive Modeling

Cognitive modeling deals with computer programs (GMOS, ACT-R) simulating cognitive processes with the purpose of predicting behavior and the underlying condition while also proving guidelines for user-friendly system design. The computer serves as a metaphor for such cognitive processes and human abilities can be simulated (AI[1], machine learning, robotics …).

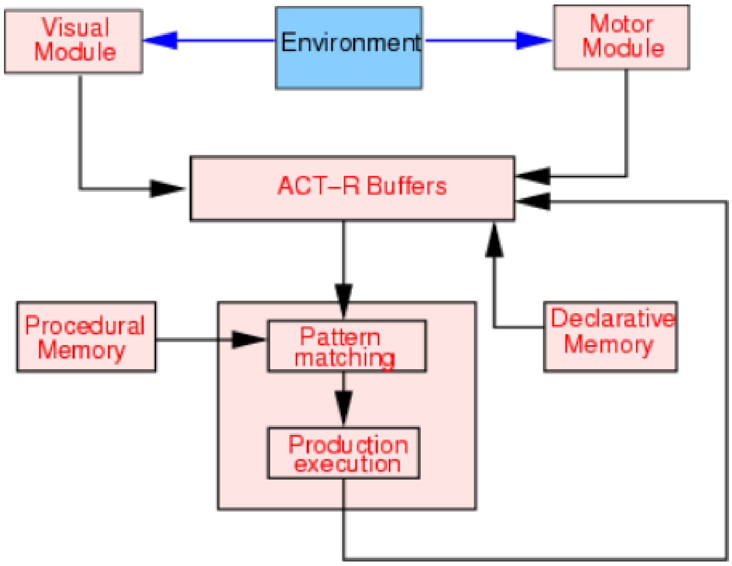
GOMS

GOMS, short for "Goals Operators Methods & Selection Rules", predicts human cognitive processes by representing task sequences in code. GOMS assumes the following, based on empirical studies:

- There are three mental processors: (visual) perception, motor processes, and central cognition.
- There is working memory and long-term memory

[1] Artificial Intelligence

ACT-R[2]



ACT-R is a cognitive architecture aiming to define the basic and irreducible cognitive and perceptual operations that enable the human mind. In theory, each task that humans can perform should consist of a series of these discrete operations.

Most of the ACT-R basic assumptions are also inspired by the progress of cognitive neuroscience, and ACT-R can be seen and described as a way of specifying how the brain itself is organized in a way that enables individual processing modules to produce cognition.

Information Foraging – Navigation in Information Spaces

This deals with the question whether a user stays on e.g. a websites or moves on. Users often follow the "scent of information" using proximal cues (icons, labels, names ...) to judge the degree of information and fit of the links.

User-adaptive interface

A prime example is Amazon where the user's preferences and context influences the content, layout etc. based on profiling, session history, location, social connections etc.

Examples & Applications

Key fields:

- Design of physical work places, communication and work-flows / organization
- User-friendly software and websites (e.g., information systems)
- User-centered, cognitively appropriate teaching/learning environments
- Health: prevention and rehabilitation

Examples include: computer-aided instruction, military research, rehabilitation and diagnostics, studies in the fields of politics, advertising and environment, AI, gesture-based interaction, user-centered software engineering.

Error-tolerant design

Often the user is blamed for an error that was caused by a behavior never anticipated by the developer. Such errors are often caused by inappropriate and inconsistent design. Such behavior can (almost) only be detected with empirical studies.

Software should be error-tolerant (e.g. undo button/action) and error messages should be written in a language tailored to the user's knowledge.

[2] From http://en.wikipedia.org/wiki/ACT-R

Lecture 3 – Hoelscher

Usability

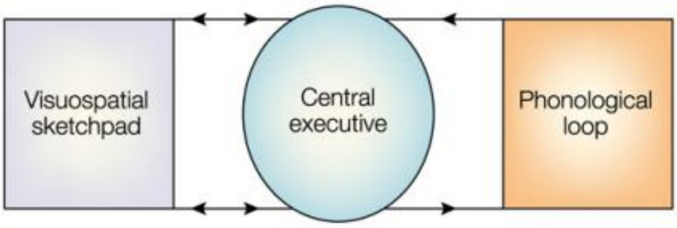
A nice example where bad wording of a dialog leads to catastrophic results: you have a text editor with file-related functions and the user wants to click "save" and instead clicks "delete". The confirmation window then asks "Do you really want to save/delete this file?"

While usability testing is time-consuming and expensive, it often helps finding unanticipated behaviors. The experience of the designer often isn't enough, even when they follow established guidelines. Studies, four to five participants tend to find about 80% of the serious errors, are also great for convincing decision makers.

A simple measure to improve handling is ergonomic design, e.g. a vacuum cleaner where the switch is integrated into the wheel to prevent trip-and-fall accidents. The same goes for the field of intuitive use where behavioral tendencies are considered in the design process. One such strategy is the S-R-computability (stimulus-response). A nice example is the design of doors which differs from country to country.

Targeted movements / pointing can be facilitated by respecting Fitts' Law where the time t (in milliseconds) is given as a function of moving the cursor a distance d to a target of size s: $t = 230 + \log_2(2d/s + 1)$.

Working memory[3]



Working memory, comparable to memory/RAM in a computer, is often a bottleneck since it has limited capacity. The capacity is around five to seven meaningful chunks of information. Working memory consists of activated units of long-term memory and with good encoding more long-term memory can be referenced and this the capacity of the working memory increased. Working memory quickly decays (in a matter of seconds) without active attention. The working memory model (pictured above) consists of three different parts. Obviously, any controlled processing required working memory capacity. Yet automatic processes/practice (provided there is an S-R coupling) doesn't require capacity.

A systematic search for the consequences of limited capacity of the working memory and the underlying strategies to unload it, are due to the required overhead often impossible. Strategies for unloading include relying on external representations, following signage instead of complete route planning, and writing down notes instead of memorizing.

Long-term memory is fairly stable over time and saves capacity by using automated routines (S-R schemas). Forgetting information is due to the non-rehearsal/usage over long times.

Additionally, there is the difference between novices and experts. For novices everything is difficult since declarative knowledge, knowledge which can be verbalized, is interpreted (e.g. shifting gears). Practice provide speed and security by compiling [declarative] knowledge into procedural

[3] There are a lot of similarities to an operating system

knowledge. Interestingly, while explicit instructions are useful for novices it is of no help to experts, who are characterized by deep understanding, they can, by understanding underlying processes and phenomena, flexibly react to problems. The problem however is, users are typically not experts.

Slips and errors

The preconditions for slips and errors are:

- There is an established behavioral schema (stimulus or cue-driven)
- There are identical to a similar context
- There is a loss of attentive/deliberative control

The result of such slips and errors is the intention not being carried out fully. There are two different types of slips: loss-of-activation slips (failure to perform the intended action) and caption slips (doing something else than primarily intended).

While mistakes are characterized by an error in planning of behavior slips happen when the action plan is correct originally yet during its execution parts of the sequence are omitted or erroneously adopted from other, similar plans (this is often the case with routine/automated schemata) and thus the behavior does not match the plan.

One, somewhat famous, example is the design of ATMs where people would often forget their bank cards since you would first retrieve the money (which is the goal) and then the card. By changing the design to first return the card, no one is able to leave the ATM without the card (provided they also want the cash).

In the category of slops, there is a special type called mode errors which are caused by insufficient differentiation of context variables. They can be prevented by using as few different modes as possible and clearly label/indicate them.[4]

Post Completion Error (PCE) & Working Memory[5]

Humans tend to implement goals as a stack with push and pop. When there are too many goals pushed onto the stack i.e. the working memory, the original goal gets popped (due to capacity restrictions). When a node of a sub-goal and the done-it-node are very similar, the main goal might be skipped. PCE is defined as follows: Once the main goal is achieved, the activation for a subsequent sub goal is weakened. If there is additional load on the WM system, then the probability increases for the sub goal to drop below an activation threshold.

Lecture 5 – Hoelscher

Spatial Abilities in Navigation[6]

Reasons why some complex buildings are easier to navigate include:

- Environmental structure
- Way finding behavior
- Architectural design practice

[4] vi is such an example; also applications which ab-use common iconography/keyboard shortcuts
[5] This is the copying machine example
[6] This is the Seattle Public Library example

Attempts to facilitate building navigation are based on modeling user strategies and cognition, and, just like with software, how architects think vs. how users think.

Finding your way through a building is a decision making process building open perception, memory, learning, reasoning, emotion, expectation, heuristics, strategies, cognitive maps, local visual cues, inference from proximal information to distal targets, expectations of layout and semantics of destination, path choice strategies (routes, survey, simple, exhaustive), and spatial abilities. When conducting such research and design in this field, one option is to study way finding in a real building with pointing and sketching tasks. Another option is modeling environmental factors and perform a building analysis with space syntax. When re-designing the building, the dialogue with architects is important and virtual reality simulations can be of great help.

If a building has multiple floors, people assume the layout does not change. Consequently a change can disorient users, also because mental models across multiple floors are difficult to integrate. There are different path choices in a multi-level building:

- Central point strategy[7]	stick to well-known parts of the building
- Direction strategy	follow vertical direction first.
- Floor strategy	reach correct floor first
- Route is well-known	complete route available from memory

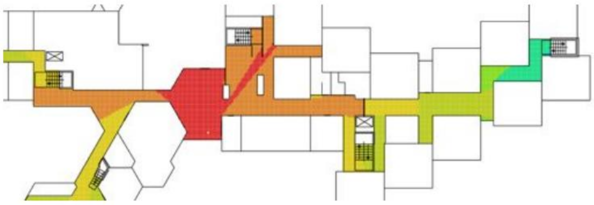
A typical experimental approach has the following properties:

- Between 12 and 50 people
  First-timers and experienced users
- Behavior observation with search or exploration task
  Where do people get lost, sources of errors?
- Verbal comments (“Thinking aloud”)
  Attentional patterns, reasoning & planning? Methodological triangulation: quantify frequencies of verbal data and compare with behavioral measures
- Video-Recording or Tracking with GPS, app
  Detailed reconstruction of movement trajectories and annotation of critical events (stops, re-orientation, consult sources of information)
- Pointing and Sketch Maps
  What is being learned? Development of a Cognitive Map?

Impact of Environmental Features

Four important environmental factors (pictured in order, from left to right) are visual access, architectural differentiation, signs and room numbers, and layout configuration and complexity.

[7] Preferred by novices



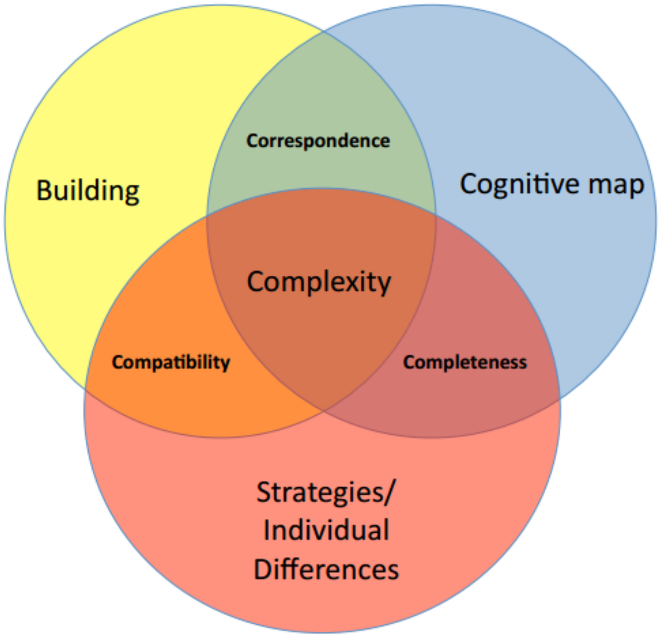
A visibility graph analysis (pictured) is a method of analyzing the inter-visibility connections.[8]

Space syntax translates geometric form into a network of sightlines and decision points, provides a quantitative prediction of streams of visitors or pedestrians, and provides a qualitative diagnosis of disorientation and getting lost.

Axial line measure gives insights into to connectivity (direct neighbors) of nodes and provides an institution of the centrality of a spatial unit.

Navigation & Usability

People rely on similarities across different floors, relationships, "main features" (e.g. main staircase), and also having the possibility to see the main (vertical) circulation easily (e.g. at the public entrance). Irreversible routes and escalators skipping floors tend to confuse people.



Case Studies and Research Methods

Evidence-based design paradigm can be used to obtain performance measures of implemented designs and/or derive predictions of such measures for design options. Human factors, as component of building performance, are taken into account and perception, emotion/appraisal, well-being, and behavior is being recorded.

Way finding / movement studies as a key example use theoretical understanding and a lot of research on human spatial abilities, spatial memory and underlying neuroscience by using objective success criteria which allow for behavioral benchmarks.

High-Level aspects for Designing Navigable Buildings

- Straighter, more direct, routes are preferable to routes containing many changes of direction
- Ensure unimpeded lines of sight: connect entrance spaces and other key, central spaces such as atria to the means of vertical circulation: stairs, lifts and escalators
- Wherever possible ensure that differences in layout between floors are not too great. Building users will assume that each floor is laid out in an analogous manner to the preceding floors
- Atria can provide a "short-cut" to survey-knowledge, as they facilitate views to and hence knowledge of other floors that would otherwise be unavailable (view enhancing)

[8] Adapted from https://en.wikipedia.org/wiki/Visibility_graph_analysis

- Building users may become lost or disorientated in locations that bear strong visual similarity to other locations in the same building,

Lecture 7 – Hoelscher

What is "Usability"

Since (most) computer systems are error prone and difficult to use, there's a huge demand in usability, be it hardware or software. This is an interdisciplinary effort and there are ISO[9] norms in place among other means to evaluate usability.

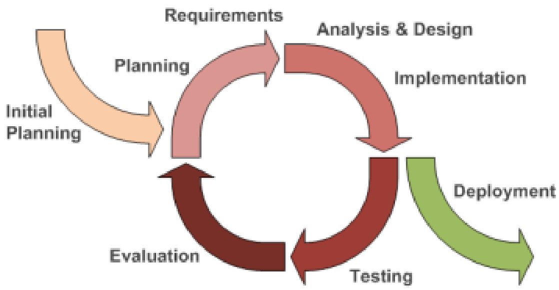
Methods of data collection and evaluation & Comparison of Techniques

Such means include the typical survey/questionnaire (which is general and retrospective; e.g. ISO-METRICS: detailed, standardized evaluation based on ISO 9241-10, very exhaustive), expert-based assessments (e.g. a cognitive walkthrough where HCI experts simulate users and precisely define an optimal task sequence while collecting "success" and "failure" stories. A storyboard and persona are required.), focus groups (measure preferences and acceptance; limited value for identifying concrete deficits), and behavior observation (e.g. novice and expert). Another method is heuristic evaluation, which while loosely defined is very quick and easy to perform in an iterative design process, where the interface is freely explored by experts, compared to guidelines while checking for common problems.

Behavior and cognition can be capture by behavior observation (time or event sampling), surveys, and verbal protocols (thinking aloud).

If the work environment/software is not available, the Wizard-of-Oz technique can be used where the environment is simulated (can also be real people instead of software).

Usability in the design process



Early phases:

- Define scenarios, manual before coding
- Use mockups, simulations, prototypes
- Thinking aloud in early, qualitative testing: provides information on possible interpretations, tasks, misconceptions etc.

Late-stage development:

- Field studies and usability tests based on previously defined performance criteria
- Technical soundness: invite testers to find technical flaws, crash the system, test security etc.

---

[9] ISO 9241, see https://en.wikipedia.org/wiki/ISO_9241; since these norms aren't available for free, an excerpt of keywords of ISO 9241, Part 10 (Dialogue Principles) is here:

- Suitability for the task
- Self-description
- Controllability
- Conformity with user expectations
- Error tolerance
- Suitability for individualization
- Suitability for learning

After deployment: follow-up studies (e.g. hotline-calls, usage log files etc.)

Usability[10] Testing: data collection in the lab

The approach is a qualitatively-exploratory one with the goal of identifying relevant design problems and not about individual design problems. Reasonable results are expected with only ten participants.

There are four different types of UX testing in the design process: exploratory (formative), assessment (summative), validation (verification), and comparison test.

The setting can be in a single room or in a classic testing lab complete with web cam, moderator, microphone, one-way mirror, observers etc. The test supervisor should have basic knowledge of UX, cognitive science and of the domain while having the necessary soft skills. Understandably, there are also some deficits since the supervisor might lead instead of enabling, should balance between observing and collecting data, should not help, and not jump to conclusions. Some, but not all, of these problems can be address with technology.

For a successful test, some technical equipment is needed but the main components are concrete goals (concrete UX goals, measured before testing) and test manual which defines participants (number, demography, breadth of experts and novices, recruitment etc.), tasks and scenarios, is coordinated with the client, and is a precondition for delegating testing to others.

There are a lot of different benchmarks to capture (aside).

- Time required per task
- Percantage of tasks finished within defined time limit
- … per minute
- Time needed on error corrections
- Number of task steps (e.g., commands used)
- Number of (or percentage of) Errors
- Number of Repetitions of failed steps/commands
- Number of successful vs. failed rounds
- Number of 'Loss of control' episodes
- Number of times a user has to work around a problem (sub-optimal sequence)
- Number of (useful) commands not used
- Frequency of using Online-Help or manual
- Frequency of uttering satisfaction or frustration
- Number of positive and negative properties that users recall afterwards
- Number of user who prefer the system over alternatives

Data can be captured with screen and scene videos, thinking aloud protocols, log files, live coding of behavior etc.

When analyzing the data, it is challenging to rate the severity, compare it with the task analysis and information architecture, link behavioral data to design issues, perform a focused analysis of verbal data, and coordinate the analysis with the product manager. Video excerpts are a great way to illustrate a problem in a presentation.

Lecture 8 – Hoelscher

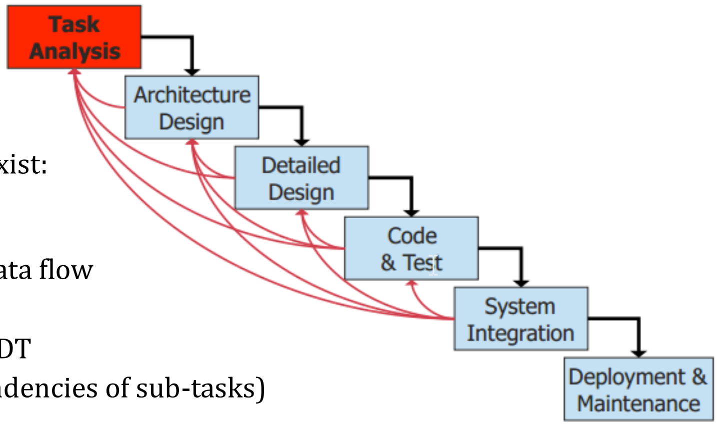
All complex tasks have an underlying/inherent structure and they often there is more than one solution / procedure is available. Finding and using the correct procedure requires knowledge. Knowledge Engineering is about reconstructing the knowledge required to solve tasks in a domain based on observation and interviewing of experts are the two dominant sources in psychological knowledge elicitation. Knowledge Engineering and task analysis thus typically go hand in hand.

---

[10] Short: UX

Task Analysis

Pictured: waterfall model, standard mode in software development.



In task analysis a number of methods exist:

- Simple Sequence Diagrams
  Flow diagrams for control flow & data flow
- Task Decomposition
  Hierarchical task decomposition, HDT
  Dependency analysis (logical dependencies of sub-tasks)
- Time-based analysis
  Timeline analysis, Gantt-Diagrams (determining the critical path in project planning)
- Cognitive Task Analysis (GOMS in several variants)

The goal of classic task analysis id to describe the structure of a task and not the requirements of the performing agents. Cognitive task analysis does both, it describes the structure of the task and models the cognitive requirements.

When performing hierarchical task decomposition, a hierarchy of sub goals[11] can be constructed, including timing requirements and preconditions. The time and sequence can be visualized in a Gantt diagram, with time on the x-axis, sub-tasks on the y-axis. Furthermore, a critical path analysis (CPA) can be performed.

GOMS

For concrete example, please refer to the corresponding lecture slides.

Goal – Operator – Method – Selection rule. The basis for GOMS is the human model processor:

- Screen + Keyboard/Mouse
- 3 “mental processors”: (visual) perception, motor processes, central cognition
- Working memory & long-term memory
- Key values of the model are derived from empirical studies
- Resolution of the operators: perceive icon, and press key / button

Different flavors of GOMS:

- CMN: goals, sub-goals, and methods; selection rules
- KLM (keystroke-level model): standard times per operator
- NGOMSL (natural GOMS language): formal modeling of task sequence
- CPM (cognitive-perceptual-motor): parallel methods; PERT (time/sequence) diagrams (CPM = critical path method)

[11] Remember making Spaghetti for dinner?

Scale (sec)	Time Units	System	Analysis	Processes	World (theory)
10^5	days				Bounded rationality
10^4	hours	Task	Task analysis	Subtasks	
10^3	10 min				
10^2	min	Subtask	Unit task analysis	Unit tasks	
10^1	10 sec	Unit task	Cognitive task anal.	Activities	Cognitive band
10^0	sec	Activity	Embodied cognition	Microstrategies	
10^{-1}	100 ms	Microstrategy	Comput. models	Elements	
10^{-2}	10 ms	Elements	Architectural	Parameters	Biological band
10^{-3}	1 ms	Parameters			

GOMS-KLM procedures for modeling:

1. Choose one or more representative task scenarios.
2. Have the design specified to the point that keystroke-level actions can be listed for the specific task scenarios.
3. For each task scenario, figure out the best way to do the task, or the way that you assume users will do it.
4. List the keystroke-level actions and the corresponding physical operators involved in doing the task.
5. If necessary, include operators for when the user must wait for the system to respond
6. Insert mental operators for when user has to stop and think. Look up the standard execution time to each operator.
7. Add up the execution times for the operators.
8. Total of the operator times is the estimated time to complete task.

In addition to Fitts' Law where the time t (in milliseconds) is given as a function of moving the cursor a distance d to a target of size s: $t = 230 + \log_2(2d/s + 1)$ there is also Hick's Law which gives the increase in reaction time t (in milliseconds) relative to the number n of choices: $t = 173 + 138 \cdot \log_2(n + 1)$.

M-operators describe the mental act of routine thinking or perception and are used when making a decision, reclining information/search display, or when verifying that an action was successful.

CPM-GOMS:

- Critical Path Method
  - Parallel execution of operations / Concurrency (Nebenläufigkeit)
  - Analysis considers interdependencies and execution times
  - Critical path (PERT-Diagram) is constructed
    - Critical path: The path through a series of operations, taking into account interdependencies, in which the late completion of an operation will increase total execution time

        - Slack/float: operations that are not on the critical path and can be completed in parallel to the critical operations have some scheduling flexibility ("slack time“)
- Cognitive – Perceptual – Motor
    - Parallel processing on three different "tracks", corresponding to different resources :
        - Cognitive: mental processes, e.g., accessing memory
        - Perceptual: primarily visual processes, e.g., object localization
        - Motor: primarily keyboard & mouse operations

Application areas of GOMS: design of mouse-driven text editor, directory assistance workstation, space operations database system (for orbital objects), bank deposit reconciliation system, CAD system for mechanical design, television control system, nuclear power plant operator's associate, intelligent tutoring system, industrial scheduling system, CAD system for ergonomic design, telephone operator workstation.

Lecture Schinazi

Edward Tolman coined the term “cognitivism” (stimulus -> ??? -> response vs. behaviorism: stimulus -> response).

Cognitive maps

- Learning does not consist of a series of stimulus-response connections but ability to form a comprehensive and flexible map
- There are **place cells** (activated at a certain place; relies on eternal environment; stable over time; not visual cells), **grid cells** (multiple firing locations; define triangular array covering surface of foraged environment; provide a coordinate system), **border cells** (Firing rate increases when the animal approaches one of the walls (border) of the environment; irrespective of relationship with other borders; consistent across enclosures of different shape and size), **head direction cells** (fires in relation to the animal’s directional heading independent of location; not dependent on earth magnetic field but rely on landmarks and self-motion cues; fires whether animal is moving or still)
- Lesion to the hippocampus causes deficit in spatial memory

MRI

- The MRI scanner is essentially a large magnet
- Atoms (especially H+) in the subject’s brain react to MRI’s magnetic field
- Deoxygenated hemoglobin is paramagnetic
- MRI studies brain anatomy; functional MRI studies brain function (fMRI)

Parahippocampal Place Area (PPA)

- Recognizing scenes; PPA responds to background elements, not discrete objects
- When the parahippocampal cortex (PHC) is damaged, there is severe topographical disorientation

Retrosplenial Cortex (RSC)

- Damage to RSC leads to heading disorientation

Hippocampus

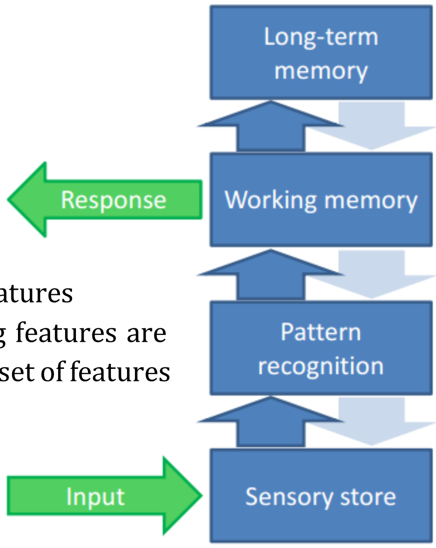
- Compared the volume of grey matter in the hippocampus
    - London taxi drivers vs. control (matched for IQ)

- Volume of posterior RHip greater for taxi drivers
    - Amount of time spent working correlates with grey matter volume
    - No apparent deficit in cab drivers for smaller anterior hippocampus

Lecture Thrash

Traditional approach

- Bottom-up versus top-down processes (pictured)
- Sensory receptors
- The sensory store
- Pattern recognition
    - Template theories – an unanalyzed pattern is matched to various possibilities
    - Feature theories – a pattern is deconstructed into features and compared to other sets of features
    - Structural theories – the relationships among features are compared to the relationships among another set of features



Visual perception

- Inverse projection problem: how do we interpret 2D retinal images as representing a 3D world?
- Indeterminacies
    - Shape and orientation: a 2D retinal image (by itself) cannot specify both 3D shape information and the orientation of the object
    - Surface: a 2D retinal image cannot specify the light source, reflectance of the object being seen, and that object's shadow all at the same time
        - Based on assumptions: surfaces are uniformly colored, fuzzy borders (low contrast) indicate shadows, objects tend to be lit from above, local contrast better indicates the shade of an object than absolute brightness
    - Size and distance: ocular (eye; accommodation, convergence and divergence, stereopsis) vs optical (light; familiar size, relative height, occlusion, texture gradient, linear perspective, atmospheric perspective) information

Computer vision

Challenges:

- Bottom-up and top-down processes are tightly interwoven
- Perceptual similarity is not the same as mathematical similarity
- Over-reliance on proof by example

Successful applications:

- Image-guided brain surgery
- Wild fire detection
- Fingerprint identification
- Food analysis

Gestalt theory

- "The whole is different from the sum of its parts"
- Perceptual constancy
- Principles of perceptual organization: proximity, continuity, similarity, closure, simplicity, figure/ground

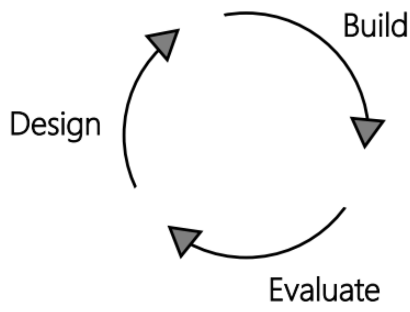
Information displays

- More graphics and less numbers
- High contrast graphics
- Icon size/shape matching function
- Values relative to some meaningful standard instead of absolute values

Lecture Hilliges

Definition of HCI: "Human-computer interaction is a discipline concerned with the design, evaluation and implementation of interactive computing systems for human use and with the study of major phenomena surrounding them." UX has a number of contributors: cognitive psychology & social sciences computer science, (electrical) engineering, and (industrial) design.

User Centered Design



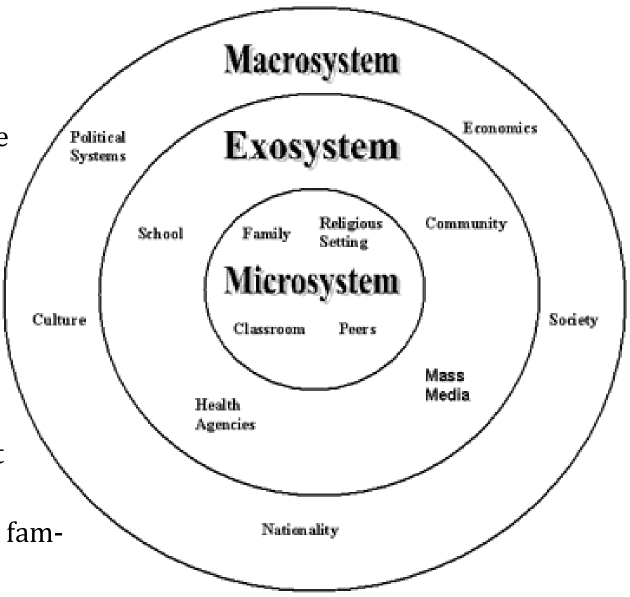
- Understanding the user, their practices, build models to understand work process, and create scenarios for actual use
- Create prototypes

Improvements to/gripes with the User Centered Design process

- All steps in the process rely heavily on human intuition
- Does the process fosters "good design"?
- Or do you just need a good designer (with good intuition)?

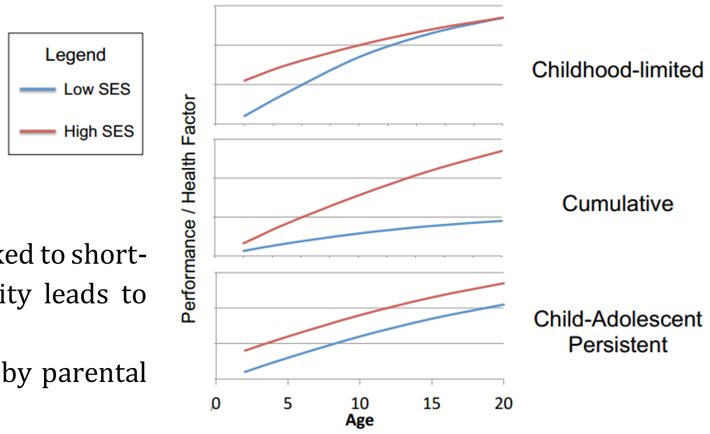
Lecture Hackman

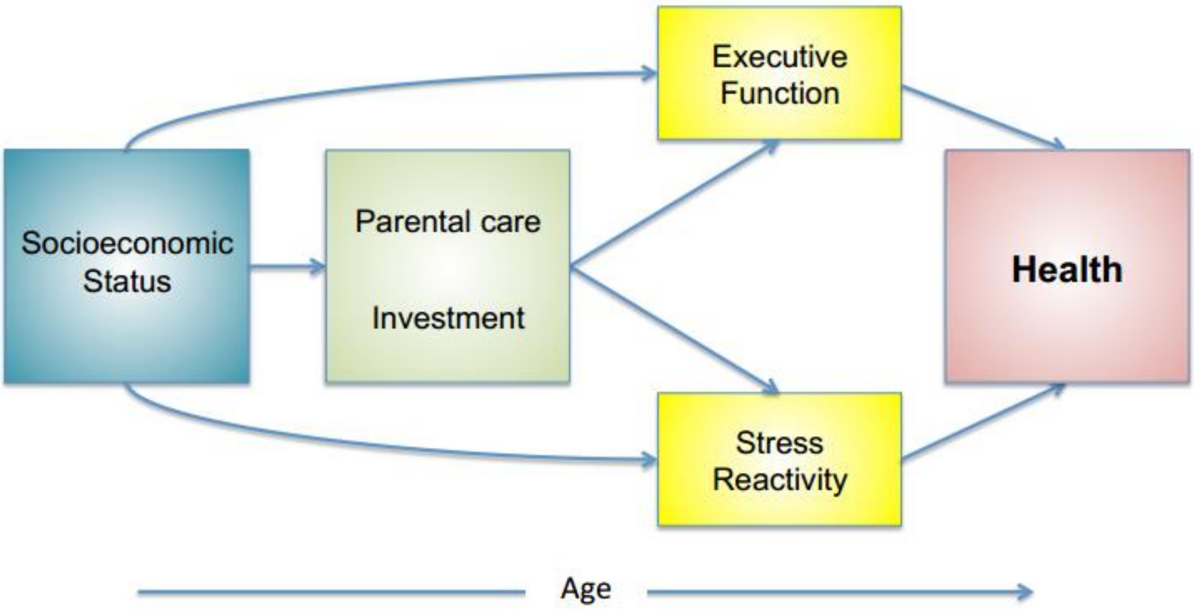
Notes:



- Increasing inequality in terms of income
- Relative risk pf premature death can be linked to family income
- Life expectancy can be linked to education level
- Socioeconomic status: "An aggregate concept that includes both resource-based and prestige-based measures, as linked to both childhood and adult social class position"
    - Multidimensional: individual, family, neighborhood, subjective
    - Measures: income, education, occupation
    - Dynamic across of the lifespan
    - Access to: social and material resources, life opportunities
    - Correlations: material (pollution, toxins, crowding, green space, housing quality, neighborhood quality, nutrition environment) and psychological (violence, family disruption, parenting, social networks and support, cognitive/language stimulation & enrichment, stress)

- Family problems/issues/pressures/emotional behavior/bad parenting etc. leads to the child having emotional & behavioral problems and impaired competence
- Parental responsivity can be linked to short-term[12] stress; better responsivity leads to higher short-term stress
- Working memory is influenced by parental education (actually, whole SES)





Sources

Unless otherwise stated: lecture slides.

[12] E.g. experienced when giving a presentation