Summary Visualization 2

Def.: The use of **computer-supported**, (most of the times) **interactive**, **visual representations** of (abstract) data to **amplify cognition**

Resource limitations: capacity of computers, of humans, and of displays

- Human in the decision making loop
- Representation generated by computer
- Human visual perception is channel of communication
- External representation is used
- Detailed structure of dataset important
- Intended task
- Operational definition of better
- Interactivity is on the table
- Resource-limits matter (humans, computers, displays)
- Visualization design space: huge, full of tradeoffs
- Most visualization designs are ineffective

Areas:

- Scientific Visualization
- Information Visualization
- Visual Analytics, Visual Data Science

Spaces:

- Known space (should be big)
- Consideration space (should be big)
- Proposal space
- Selected solution

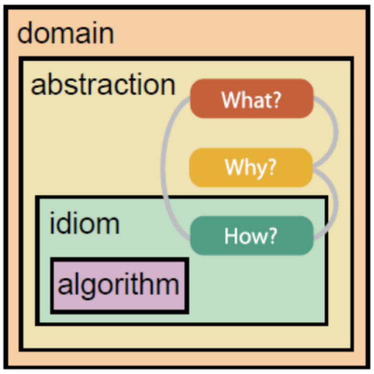
Nested model for visualization design

Nested levels of visualization design

- **DOMAIN**: Characterizing the problems of real-world users / target users
  Potential Problem: wrong problem!
  Pre-validation: Interview and observe target users:
  + RECORDED, SEMI-STRUCTURED INTERVIEW
  + Code the interview: Iterative characterization of qualitative data (open coding) to find CATEGORIES
  + multiple coders, using a code book (feature, description, example), if the coding agreement is above approx.. 80% the coding is done
  Post-validation: Observe adoption rates
  - **ABSTRACTION**: Abstracting into operations on data types
    Potential Problem: Showing the wrong thing!
    Post-validation: Field studies
    + Qualitative data acquisition: Users do their own thing

+ Observations (Field notes, Video or audio tapes, Field logs)
+ Interviews
+ Coding

- IDIOM: Designing encoding and interaction techniques
  Potential Problem: Showing it the wrong way!
  Pre-validation: Justification
  Post-validation: User studies
  - ALGORITHM: Creating algorithms to execute techniques



Example WHAT: **Datatypes** (e.g. Positions, Items, Attributes…) and **Datasettypes** (e.g. Tree, Tables…)
Example WHY: **Actions** (Analyzise, Search, Query) and **Targets** (All Data, Attributes, Network Data, Spatial Data)
Example HOW: **Encode** (Map to shape / color / motion etc.), **Manipulate** (Change, Select, Navigate), **Facet** (Juxtaposition › Nebeneinander, Partition, Superimpose › Überlagern) and **Reduce**

- No unjustified 3D
- Eyes over memory
- Resolution over immersion
- Function first, Form next
- Get it right in black and white

Spatio-Temporal Visualization

Spatial-Temporal Data:
Visualisation for data with a spatial and a temporal reference

DATA TYPES

- *Spatial statistical attributes* (related to spatial reference)
  - E.g. Income per country
- *Point-based data*
  - E.g. Growth ring maps (good, as country / area size doesn't influence perception)
- *Connections*
  - Origin-destination-relations (edge bundling / force-directed edge bundling)
- *Simulations*

- *Trajectory data:*
    - Movers = Objects that change spatial **position** over **time**
    - Movement = Change of spatial position of an object
    - Trajectories = Points sampled in space, interpolated curve, optional additional attributes like speed
    - Challenges:
        - **Noise**
    - Solutions:
        - Map-matching: align to road network (geometrically, topologically, probabilistic mapping)
        - Filtering (mean, median, Kalman (makes predictions for the next positions, if they are very off, those are thrown away), particle filter)
    - Stop detection: if threshold is passed for timespan between a couple of points

VISUALISATION

SPACE-TIME CUBE:

- X/Z: map data
- Y: time
- Problems: **Clutter**
- Solution: Clustering / Edge bundling

MAPS

- Choropleth Maps:
    - Questions:
        - Number of colors?
        - Spatial resolution (how many sections)
        - Diverging / sequential color maps
        - Absolute / relative values
- Cartograms: Problems with distortions, but interesting visualization
- Circle Maps
- Grid maps: Very useful - no perception influence due to area size

Web based Information Visualisation

Applications: Tableau, MS Power BI, MicroStrategy …

- More for representation than for analysis
- Not as flexible
- Tableau is quite flexible and easy to learn

Libraries: Python Plotly, D3, Python Matplotlib, Highcharts, React-Vis, Processing *

- For presentation and analysis
- Highly flexible, especially D3 and Processing BUT difficult to learn
- The wrapper libraries are less flexible

---

* Javascript Libraries => Web-based tools, those are more for presentation than for analysis

Integration of web-based visualisation

IMG

SVG

- Vector graphic, click-based interactions, might be slower than canvas
- Libraries using SVG: D3js, SVG.js

Canvas

- Bitmap format, Animations, interaction not based on objects, WebGL support
- Libraries using Canvas: ThreeJS

Visualisation Use Cases

- Exploration
  - Searching and analysis
  - No / little knowledge about data
  - Find potentially useful information
- Confirmation
  - Goal-oriented
  - Examination of a prior defined hypothesis
  - More knowledge of data
- Presentation
  - Efficient communication of data features / findings
  - Clear knowledge of data

Challenges

- Data size
- Security:
  - User might access data that is hidden in the visualization
  - Javascript is a client-side programming language

Solution

- Server-side visualization? – limited interaction possible
- You never see the data, just images

Tasks

- Know your users
- Identify your use-cases
- Carefully select your data
- Consider Open Source visualization

Non Destructive Testing

Analyzing techniques without causing damage

- e.g. x-ray, ultrasonic, thermography
- Mostly used for 3d data, represented as networks (not as trees normally)

'Rich' XCT (x-ray) data generation

- INPUT: Primary data (XCT data) and
- OUTPUT: Secondary data (important features, label data characteristics)

Components – can be complex, therefore getting insight to rich XCT data is difficult

- Data Volume
- Data Veracity (Certain / Uncertain)
- Data Velocity (Realtime / Static)
- Data Variety (Homo / Hetero)

Visualisation in NDT

1. Material Systems (6)
   a. Metals, non-metal inorganic materials, construction materials, biological materials …
2. Tasks
   a. *Material simulation tasks*: Exploration, Visual analytics of fluid dynamics, Analysis and Visualization
      i. *Analyze material characteristics and structural changes under external forces*
   b. *Material analysis tasks*: Feature extraction, **Stress tests**, **Damage analysis**, dimensional measurements, **Optimization**, Risk analysis, **Uncertainties**
      i. *Damage Analysis*: Escalation of stress and deformation (why does material fail when aging?)
      ii. *Uncertainties*: budget, measurement etc.
3. Testing Techniques & Data Types
4. Visual Representation
   a. For spatio-temporal data (Material anaylsis)
      i. 3D Renderings with Labels, Isosurfaces/lines, Networks
      ii. 3D animations, Juxtapositions, Maps
   b. For quantitative data (Material simulation)
      i. Tensor fields, Colorcoding
      ii. Parallel coordinate
5. Interaction Techniques
   a. Explore (translate, zoom etc.) and reconfigure (different perspectives changing the spatial arrangement)
   b. Linking & Brushing
   c. Focus + Context (Focus area shown with more detail)
   d. Filter (free text, drop down, sliders etc.)
   e. Interactive steering (realtime changes to parameters)

Challenges

- Integrated visual analysis (HOW to analyze?)
  + Quantitative data visualization (Comparison / differences?)
  - Encode (different representation, for example Blob visualization = finding closest contour around selected features)
  - Connect
  - Abstract / Elaborate (more / less detail)

    - Reconfigure (different arrangement, for example MObjects = mean objects / uncertainty cloud)
    - Filter (conditionally)
    - Select (mark)
    - Explore (show me sth else : scatter plot, parallel coordinates…)
- Debugger (Explore parameter space, Identify errors)
  + Interactive steering (Make predictions for production, Monitor trends…)
    - **Visual parameter space analysis (vPSA)**
        - Systematic variation of model input params
        - Generating outputs for each combination of params
        - Visual inspecting relations between inputs & outputs
        - Comparative analysis can then be applied
    - Similarity measures
        - Difference in characteristics
        - Difference in point distances
        - Differences in overlap
    - Abstract representation of volumes using line plots
    - Hilbert Space-filling curve generation
        - Very long line plots => nonlinear scaling of line plots

Text anaylsis

Process of deriving information / meaning from unstructured text

Input: Documents / Corpus (collection of documents) / Paragraphs / Sentences / Words

Use cases

- CONTENT / COMPARISON / EVOLUTION
- Which topics occur often?
- Comparison between texts / speeches…
- Is a trained model biased?
- Public opinions in social media over time
- Analysing customer feedback

1. Structure input

- LEXICAL LEVEL
    - Tokenization: Aufteilung eines Textes in Unterteilungen, z.b. Buchstaben / Wörter / n-gram, Sätze, Paragraphen…
    - WordTree: Sätze werden in Baumstruktur zusammengesetzt
    - Arc Diagrams: visualisation of repetitions in sequences

- SYNTACTICAL LEVEL
    - Chunking: segment and label multi-token sequences
        - Find nouns / adjectives etc.
        - Tense

- SEMANTIC LEVEL (nltk)
    - Extract relationships
    - Coreference resolution (what is getting references by a word, e.g. he -> developer)

2. Transformation to spatial data

- Token importance
    - E.g. counting of word appearences without stop words (.,…)
    - Lowercasing to aggregate equivalent words
    - Weighting Schemes
        - Tf(t,d): term frequency
        - Df(t): document frequency
        - Idf (inverse document frequency) is a logarithmic function, that weights the term frequencies against the document frequencies. If it occurs just in one document => more important for that document

- E.g. WordCloud: Map token importance to font-size
    - Wordle algorithm (greedy algorithm):
        - First word in the middle, usually the most important word is the one in the middle
        - Position not important
        - Find free space for each new word on spiral
        - Orientation and color doesn't matter
- CompareCould
    - Compare Word Clouds with each other
- Semantic WordCould (more sophisticated than WordClound)
    - Spatial grouping of words that belong together
        - Create similarity graph
        - 2d graph using a force directed layout algorithm
        - Word cloud layout optimization by removing overlaps
    - Distributional Hypothesis to tell similarity: occurrence of parts relative to other parts
        - Terms co-occuring in a document are similar
        - Documents containing the same terms are similar
    - Bag-of-words representation: orderless representation of words / words-document relationship
        - Can be represented using vectors in a n-dimensional coordinatesystem, the documents being the axes, the words being the vectors => cosine similarity formula to find similarity
        - Create similarity matrix from vectors (symmetrical matric)
        - Visualization (Visual summary of corpus content): map similarities of words from matrix to distances of words in representation (non-linear dimensionality reduction)
            - Analysis Task: Corpus Content Overview
            - Items: Tokens or entities
            - Attributes: document dimensions

            - Similarity matrix and visualualization can as well be done for term-document matrices
                - Analysis Task: Corpus Content Overview
                - Items: documents
                - Attributes: term dimensions
    - Terms Co-Occurences: Machine learning technique (today more used than bag-of-words)
        - Word2Vec: Derive vectors for each word using a skip-gram model to predict the context words for each given word. The hidden layer is the vector, representing the word. Word vector arithmetic can then be applied to compare words (find biases)
        - Example: Google News, 300 dimensions
          => Words that are similar are not always very close together in the visualization, it is not possible…

Find biases in texts

- Custom projections:
    - select points in plot, that you know belong to a specific class
    - low-end and high-end of customer axes
    - connect those
    - project all the other points to that connection

Word embeddings

- **WordNet** is a lexical database of semantic relations (synsets): manually generated lookup table
- **BERT**: Context embeddings
    - Computes attention for series of input words
    - Result: feature vector for one word in its context
    - Model learns different contexts for a word, for example 'die'

Clustering: Topic modeling

- Factorization of term-document matrix (fuzzy clustering)
    - Non-negative matrix factorization :
        - words – topics – documents
        - Topics are defined by the highest ranked words that belong to its vector
        - Example: TopicLens, using binary topic hierarchy

Topic Evolution visualization

- For words:
    - Spark Counds: find out how tokens / entities change over time
    - TempoTaggram: e.g. old words lighter, current words darker…
- For topics:
    - ThemeRiver: Topics over time are more and less important
    - ThemeDelta: topics for timestamps and relations between words in those topics over time (How do topics change over time?)

Visualisation design and evaluation

Purpose of visualization: INSIGHTS for…

- Discovery
- Decision making
- Explanation

Questions:

- Which tools?
- Does the visualization effect (negative) the interpretation of the data?

Design Principles:

- Expressiveness: Die Fakten aus dem Datensatz werden repräsentiert – NUR diese
    - „Tell the truth and nothing but the truth – don't lie, don't lie by omission!"
- Effectiveness: Ist diese Visualisierung verständlicher als andere Visualisierungen?
    - „Use encodings that people decode better"
    - Validation by studies that were already made:
        - E.g.Magnitude Channels-Ordered Attributes: position > area > volume
        - E.g. Identity Channels: Categorical Attributes: Spatial > hue > shape

Psychophysical Experiments: Methods to measure human sensation triggered by physical stimuli

- Assuming no / low instructional bias (not a lot of difference in perception between people) => wenige teilnehmer an studien, dafür sehr ausführliche studien
- Methods for threshold measurements:
  + E.g. absolute threshold: when is flickering not perceived anymore
    - Method of adjustment
        - Adjust stimulus intensity (bsp mit den zwei linien, die gleich lang sind, aber nicht gleich lang wirken)
    - Staircase procedure
        - Gleiches bsp wie oben, aber adaptiv, damit der user nicht den threshold selbst finden muss
        - Threshold – average of opinion changes (stimulus always changes diection when user changes opinion)
    - Diese Diskrepanzen können dann gegebenenfalls visuell kompensiert werden

  + E.g. Difference threshold: Find a just noticeable difference
    - Webers Law: proportional difference : k = delta I / I
        - ⇨ Noticeable difference is not an absolute number!
    - Fechners Law: perceived stimulus intensity as logarithmic function
        - ⇨ We need to increase stimulus logarithmically to perceive linear changes
    - Stevens Power Law
        - ⇨ P = k*I^a
        - ⇨ Saturation: overestimated
        - ⇨ Length: perceived correct
        - ⇨ Brightness: underestimated
    - Proportional judgement:
        - Objects right next to each other: super

        - Objects with gap: hmmm
        - Objects with stuff in between: NAH
    - Ordering color channels:
        - By Luminance: super
        - By Saturation: Yes
        - By Hue: NAH

User studies:

- Controlled user study
    - Types:
        - Comparative lab study
        - Crowdsourcing study
        - Eye tracking stud
            - Fixations (areas of interest), Saccades, Scanpath
            - Attention plots to visualize eyetracking
    - Hypothesis: logical, precise, testable
    - Independent variables: Visualisation method, data etc.
    - Dependent variables: Measured user performance
    - Experimental design:
        - Within-subject design (user repeats test with different variations of independent variables)
        - Between-subject design
        - Mixed design
    - Analysis: How big is the difference? Statistically significant?
- Inspections performed by experts
    - Heuristic evaluation
    - Cognitive walk-throughs
- Qualitative result inspection

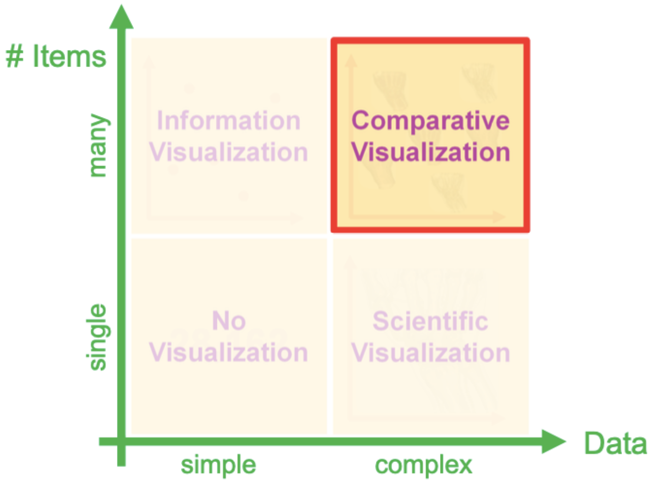
Heuristic Evaluation (instead of expensive User studies)

- 5 experts – they find > 75% of the problems

Qualitative result inspection (instead of expensive User studies)

- Look and judge

Comparative visualisation



Approaches:

- Juxtaposition
- Superposition
- Explicit encoding

Comparative slice view: Viewing multiple datasets on a single screen
Mean objects: MObjects

Differences visualized through:

- Caricaturistic visualization:
  - Extrapolate the differences between items
- Color
- Cut-outs

…