Lecture 3: Visualization Design

Information Visualization
CPSC 533C, Fall 2011

Tamara Munzner

UBC Computer Science

Wed, 14 September 2011

Material Covered

Chapter 1: Visualization Design

LiveRAC - Interactive Visual Exploration of System Management Time-Series Data. Peter McLachlan, Tamara Munzner, Eleftherios Koutsofios, and Stephen North. Proc CHI 2008, pp 1483-1492.

Sizing the Horizon: The Effects of Chart Size and Layering on the Graphical Perception of Time Series Visualizations. Jeffrey Heer, Nicholas Kong, and Maneesh Agrawala. ACM CHI 2009, pages 1303 - 1312.

Nested Model

- separating design into four levels
- validate against the right threat based on level

problem: you misunderstood their needs
abstraction: you're showing them the wrong thing
encoding: the way you show it doesn't work
algorithm: your code is too slow

- you = visualization designer
- they = target user

Characterizing Domain Problem

problem
data/op abstraction
encoding/interaction
algorithm

- identify a problem amenable to vis
  - provide novel capabilities
  - speed up existing workflow
- validation
  - immediate: interview and observe target users
  - downstream: notice adoption rates

Abstracting Data/Tasks

problem
data/op abstraction
encoding/interaction
algorithm

- abstract from domain-specific to generic
- operations/tasks
  - sorting, filtering, browsing, comparing, finding trend/outlier, characterizing distributions, finding correlation
- data types
  - tables of numbers, relational networks, spatial data
  - transform into useful configuration: derived datan
  - more next time
- validation
  - deploy in the field and observe usage

Designing Encoding and Interaction

problem
data/op abstraction
encoding/interaction
algorithm

- visual encoding: drawings they are shown
- interaction: how they manipulate drawings
- validation
  - immediate: careful justification wrt known principles
  - downstream: qualitative or quantitative analysis of results
  - downstream: lab study measuring time/error on given task

Creating Algorithms

problem
data/op abstraction
encoding/interaction
algorithm

- carry out specification efficiently
- validation
  - immediate: complexity analysis
  - downstream: benchmarks for system time, memory

Upstream and Downstream Validation

- humans in the loop for outer three levels

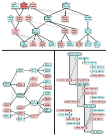
threat: wrong problem
validate: observe and interview target users
threat: bad data/operation abstraction
threat: ineffective encoding/interaction technique
validate: justify encoding/interaction design
threat: slow algorithm
validate: analyze computational complexity
implement system
validate: measure system time/memory
validate: qualitative/quantitative result image analysis
[informal usability study]
validate: lab study, measure human time/errors for operation
validate: field study, document human usage of deployed system
validate: collect anecdotes about tool utility from target users
validate: observe adoption rates

Validation Mismatch Danger

- cannot show encoding good with system timings
- cannot show abstraction good with lab study

problem validate: observe target users
encoding validate: justify design wrt alternatives
algorithm validate: measure system time
encoding validate: lab study, qualitative analysis
abstraction validate: observe real usage in field

Genealogical Graphs



[Fig 13. McGuffin and Balakrishnan. Interactive Visualization of Genealogical Graphs. Proc. InfoVis 2005, p. 17-24.]

Genealogical Graphs: Validation

justify encoding/interaction design

qualitative result image analysis
test on target users, collect anecdotal evidence of utility

MatrixExplorer

- domain: social network analysis
  - early: participatory design to generate requirements
  - later: qualitative observations of tool use by target users
- techniques
  - interactively map attributes to visual variables
    - user can change visual encoding on the fly (like Polaris)
  - filtering
  - selection
  - sorting by attribute

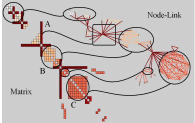
[MatrixExplorer: a Dual-Representation System to Explore Social Networks. Henry and Fekete. IEEE TVCG 12(5):677-684 (Proc InfoVis 2006)]

Requirements

- use multiple representations
- handle multiple connected components
- provide overviews
- display general dataset info
- use attributes to create multiple views
- display basic and derived attributes
- minimize parameter tuning
- allow manual finetuning of automatic layout
- provide visible reminders of filtered-out data
- support multiple clusterings, including manual
- support outlier discovery
- find where consensus between different clusterings
- aggregate, but provide full detail on demand

Techniques: Dual Views

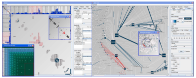
- show both matrix and node-link representations



[Fig 3. Henry and Fekete. MatrixExplorer: a Dual-Representation System to Explore Social Networks. IEEE TVCG 12(5):677-684 (Proc InfoVis 2006)

MatrixExplorer Views

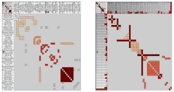
- overviews: matrix, node-link, connected components
- details: matrix, node-link
- controls



[Fig 1. Henry and Fekete. MatrixExplorer: a Dual-Representation System to Explore Social Networks. IEEE TVCG 12(5):677-684 (Proc InfoVis 2006)
www.aviz.fr/ nhenry/docs/Henry-InfoVis2006.pdf]

Automatic Clustering/Reordering

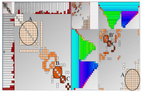
- automatic clustering as good starting point
- then manually refine



[Fig 6. Henry and Fekete. MatrixExplorer: a Dual-Representation System to Explore Social Networks. IEEE TVCG 12(5):677-684 (Proc InfoVis 2006)]

Comparing Clusters

- relayout, check if clusters conserved



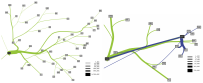
- encode clusters with different visual variables
- colorcode common elements between clusters

[Fig 11. Henry and Fekete. MatrixExplorer: a Dual-Representation System to Explore Social Networks. IEEE TVCG 12(5):677-684 (Proc InfoVis 2006).]

MatrixExplorer: Validation

observe and interview target users
justify encoding/interaction design
measure system time/memory
qualitative result image analysis

Flow Maps



- algorithm goals
  - move nodes to make room, but maintain relative positions
  - minimize edge crossings

[Fig 1c, 10. Phan, Yeh, Hanrahan, Winograd. Flow Map Layout. Proc InfoVis 2005, p 219-224.]

Flow Maps: Validation

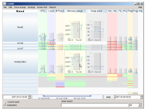
justify encoding/interaction design
computational complexity analysis
measure system time/memory
qualitative result image analysis

LiveRAC

- domain: large-scale sysadmin
- data: time series of system status from devices
  (10 Aug 2007 9:52:47, CPU, 95%)
- tasks
  - interpret network environment status
  - capacity planning
  - event investigation (forensics)
  - coordinate: customers, engineering, operations

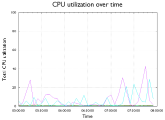
[McLachlan et al. LiveRAC - Interactive Visual Exploration of System Management Time-Series Data. Proc CHI 2008, pp 1483-1492.]

LiveRAC

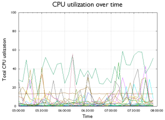


- techniques
  - semantic zooming
  - stretch and squish navigation

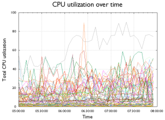
Time-Series Challenges



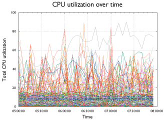
Time-Series Challenges



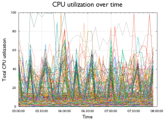
Time-Series Challenges



Time-Series Challenges



Time-Series Challenges



Design Approach

- time series challenges
  - not safe to just cluster/aggregate
  - need overview and details
- design principles
  - spatial position is strongest perceptual cue
  - side by side comparison easier than remembering previous views
  - multiple views should be explicitly linked
  - show several scales at once for high information density in context
  - preserve familiar representations when appropriate
  - overview first, zoom and filter, details on demand
  - avoid abrupt visual change
  - provide immediate feedback for user actions

Phased Design

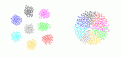
- target users hard to access: high-level corporate approval
- phase 1
  - external experts
  - simulated data
  - result: visenc/interaction proof of concept
- phase 2
  - internal engineers, managers
  - real data
  - result: hi-fi prototype
- phase 3
  - 4 internal technical directors
  - result: deployment-ready robust prototype
- phase 4
  - field test: 4 directors, 7 network engineers
  - prototype deployed for 4 months

LiveRAC: Validation

observe and interview target users
justify encoding/interaction design
qualitative result image analysis
field study, document usage of deployed system

LinLog

- energy model to show cluster structure
  - reject metric of uniform edge length
  - refine: two sets for length, within vs between clusters
- validation: proofs of optimality
- level is visual encoding not algorithm
  - energy model vs. algorithm using model for force-directed placement



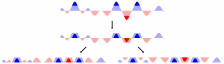
[Fig 1. Noack. An Energy Model for Visual Graph Clustering. Proc. Graph Drawing 2003, Springer LNCS 2912, 2004, p 425-436.]

LinLog: Validation

qualitative/quantitative result image analysis

Sizing the Horizon

- high data density displays
  - horizon charts, offset graphs



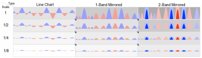
[Fig 2. Heer, Kong, and Agrawala. Sizing the Horizon: The Effects of Chart Size and Layering on the Graphical Perception of Time Series Visualizations. CHI 2009, p 1303-1312.]

Experiment 1

- how many bands? mirrored or offset?
- design: within-subjects
  - 2 chart types: mirrored, offset
  - 3 band counts: 2, 3, 4
  - 16 trials per condition
  - 96 trials per subject
- results
  - surprise: offset no better than mirrored
  - more bands is harder (time, errors)
    - stick with just 2 bands

Experiment 2

- mirror/layer vs line charts? effect of size?
- design: within-subjects
  - 3 charts: line charts, mirror no banding, mirror 2 bands
  - 4 sizes
  - 10 trials per condition
  - 120 trials per subject

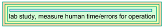


[Fig 7. Heer, Kong, and Agrawala. Sizing the Horizon: The Effects of Chart Size and Layering on the Graphical Perception of Time Series Visualizations. CHI 2009, p 1303-1312.]

Results

- found crossover point where 2-band better: 24 pixels
  - virtual resolution: unmirrored unlayered height
  - line: 1x, 1band: 2x, 2band: 4x
- guidelines
  - mirroring is safe
  - layering (position) better than color alone
  - 24 pixels good for line charts, 1band mirrors
  - 12 or 16 pixels good for 2band

Sizing the Horizon: Characterization



Key Ideas

- characterize methods using lab studies
  - more useful than A/B system comparison lab studies
  - finding thresholds
  - uncovering hidden variables
- controlled experiments
  - experimental design and statistical power

Critique

Critique

- strengths
  - very well executed study
    - best paper award
  - finding crossover points is very useful
- weaknesses

InfoVis Scope

- a human in the loop

InfoVis Scope

- a human in the loop
- visual perception

InfoVis Scope

- a human in the loop
- visual perception
- external representation

InfoVis Scope

- a human in the loop
- visual perception
- external representation
- a computer in the loop

InfoVis Scope

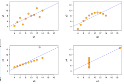
- a human in the loop
- visual perception
- external representation
- a computer in the loop
- show the data in detail

Identical statistics	
x mean	9.0
x variance	10.0
y mean	7.50
y variance	3.75
x/y correlation	0.816

InfoVis Scope

- a human in the loop
- visual perception
- external representation
- a computer in the loop
- show the data in detail

Identical statistics	
x mean	9.0
x variance	10.0
y mean	7.50
y variance	3.75
x/y correlation	0.816



[http://upload.wikimedia.org/wikipedia/commons/b/b6/Anscombe.svg]

InfoVis Scope

- a human in the loop
- visual perception
- external representation
- a computer in the loop
- show the data in detail
- driving task

InfoVis Scope

- a human in the loop
- visual perception
- external representation
- a computer in the loop
- show the data in detail
- driving task
- the meaning of better

Resource Limitations

- computational capacity
  - CPU time
  - computer memory: size, cache hierarchy
- human capacity
  - human memory: working, longterm recall
  - human attention: search, vigilance
- display capacity
  - information density
    - information encoded / total space used
    - show lots: minimize navigation/exploration
    - show less: minimize visual clutter