Information Visualization

Intro

Tamara Munzner
Department of Computer Science
University of British Columbia

3 September 2014

http://www.cs.ubc.ca/~tmm/courses/547-14

Audience

- no prerequisites
  - many areas helpful but not required
    - human-computer interaction, computer graphics, cognitive psychology, graphic design, algorithms, machine learning, statistics, ...
- open to non-CS people
  - if no programming background, can do analysis or survey project
- open to advanced undergrads
  - talk to me
- open to informal auditors
  - some or all days of readings/discussion, as you like
    - you'll get out of it what you put into it...

Class time

- week 1
  - 1 lecture
- weeks 2-3, 5-10
  - before class: you read chapter+paper, write questions/comments
  - during class: we discuss
- week 4
  - guest lectures (Ben Shneiderman, Michelle Borkin, Matt Brehmer)
- week 11
  - no class (annual VIS conference)
- weeks 12-13
  - before class: you each read paper on topic of your choice
  - during class: you present it to everybody else (~10 min)

Readings

- new textbook
  - Tamara Munzner. Visualization Analysis and Design. A K Peters Visualization Series. CRC Press. Oct 2014, to appear.
  - advance electronic copy through early October
    - password protected
  - can buy bundled ebook+hardcopy from CRC (less than Amazon hardcopy price)
    - http://www.cs.ubc.ca/~tmm/vadbook/
- papers
  - links posted on course page
  - if DL links, use library EZproxy from off campus
- readings posted by one week before class
- usually one chapter + one paper per class session

Participation

- written questions on reading in advance (18% of total mark)
  - due 12pm (30 min before class)
  - 2 on chapter, 1 on paper
  - bring printout or laptop with you, springboard for discussion
- discussion in class (12% of total mark)
- attendance expected
  - tell me in advance if you'll miss class (and why)
  - question credit still possible if submitted in advance
  - tell when you recover if you were ill

Questions

- questions or comments
- fine to be less formal than written report
  - correct grammar and spelling still expected
  - be concise: a few sentences is good, one paragraph max!
- should be thoughtful, show you've read and reflected
  - poor to ask something trivial to look up
  - ok to ask for clarification of genuinely confusing section
- examples on http://www.cs.ubc.ca/~tmm/courses/547-14/structure.html

Marking

- 50% Project
  - 1% Pitches
  - 10% Proposal
  - 4% Status Updates
  - 15% Final Presentation
  - 20% Final Report
  - 50% Content
- 20% Presentations
  - 75% Content: Summary 50%, Analysis 25%, Critique 25%
  - 25% Delivery: Presentation Style 50%, Slide Quality 50%
- 30% Participation
  - 60% Written Questions
  - 40% In-Class Discussion
- marking by buckets
  - great 100%
  - good 89%
  - ok 78%
  - poor 67%
  - zero 0%

Projects

- solo, or group of 2, or group of 3
  - amount of work commensurate with group size
- stages
  - pitches (in class), 5%: Oct 15
  - meetings (individual, outside class): Oct 20-30
  - proposals (written): Oct 31, 5pm
  - status updates (written): Nov 14, 5pm
  - final presentations (oral): Dec 12, noon-TBD
  - final reports (written): Dec 15, 5pm
- resources
  - software, data
  - project ideas
  - guest lecture: Brehmer on toolkits/resources

Projects

- programming
  - common case
  - I will only consider supervising students who do programming projects
  - three types
    - problem-driven design studies (target specific task/data)
    - technique-driven (explore design choice space for encoding or interaction idiom)
    - algorithm implementation (as described in previous paper)
- analysis
  - use existing tools on dataset
  - detailed domain survey
  - particularly suitable for non-CS students
- survey
  - very detailed domain survey
  - particularly suitable for non-CS students

Projects

- BYOD (Bring Your Own Data)
  - you have your own data to analyze
  - your thesis/research topic (very common case)
  - dovetail with another course (sometime possible but timing can be difficult)
- project possibilities will be posted on resource page soon
  - http://www.cs.ubc.ca/~tmm/courses/547-14/resources.html

Presentations

- last two weeks of class
- present, analyze, and critique one paper
  - send me topic choices by Oct 24, I will assign papers accordingly
- expectations
  - slides required
  - summary/description important, but also your own thoughts
    - analysis according to book framework
    - critique of strengths and weaknesses
- timing
  - exact times TBD depending on enrollment
  - likely around 10 minutes each
- topics at http://www.cs.ubc.ca/~tmm/courses/547-14/presentations.html

Course Goals

- twofold goal
  - specific: teach you some infovis
  - generic: teach you how to be a better researcher
- feedback through detailed written comments on writing and presenting
  - both content and style
  - at level of paper review for your final project
  - goal: within a week or so
- fast marking for reading questions
  - great/good/ok/poor/zero
  - goal: turn around before next class
    - one week at most

Finding me

- email is the best way to reach me: tmm@cs.ubc.ca
- office hours Mon right after class (2-3pm)
  - or by appointment
- X661 (X-Wing of ICICS/CS bldg)
- course page is font of all information
  - don't forget to refresh, frequent updates
  - http://www.cs.ubc.ca/~tmm/courses/547-14

Chapters/Topics

- What's Vis and Why Do It?
- What: Data Abstractions
- Why: Task Abstractions
- Analysis: Four Levels for Validation
- Marks and Channels
- Rules of Thumb
- Arrange Tables
- Arrange Spatial Data
- Arrange Networks
- Map Color and Other Channels
- Manipulate View
- Facet Into Multiple Views
- Reduce Items and Attributes
- Analysis Case Studies

Topics Preview

Defining visualization (vis)

Computer-based visualization systems provide visual representations of datasets designed to help people carry out tasks more effectively.

Why?...

Why have a human in the loop?

Computer-based visualization systems provide visual representations of datasets designed to help people carry out tasks more effectively.

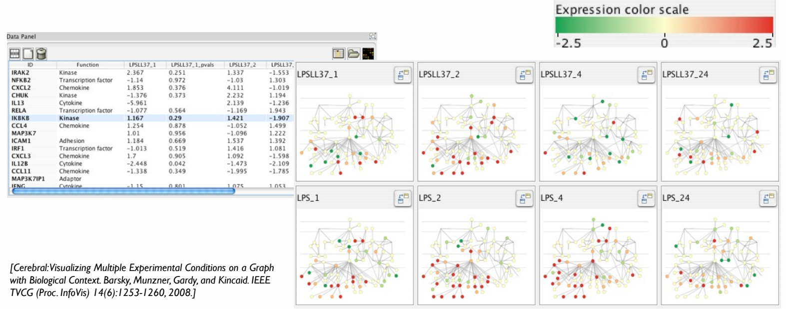
Visualization is suitable when there is a need to augment human capabilities rather than replace people with computational decision-making methods.

- don't need vis when fully automatic solution exists and is trusted
- many analysis problems ill-specified
  - don't know exactly what questions to ask in advance
- possibilities
  - long-term use for end users (e.g. exploratory analysis of scientific data)
  - presentation of known results
  - stepping stone to better understanding of requirements before developing models
  - help developers of automatic solution refine/debug, determine parameters
  - help end users of automatic solutions verify, build trust

Why use an external representation?

Computer-based visualization systems provide visual representations of datasets designed to help people carry out tasks more effectively.

- external representation: replace cognition with perception

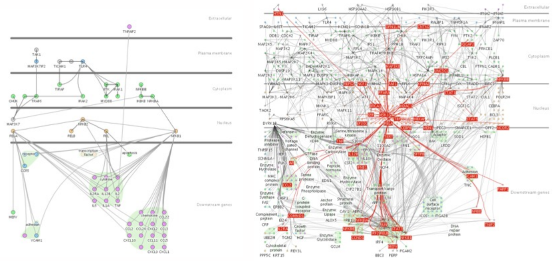


[Cerebral: Visualizing Multiple Experimental Conditions on a Graph with Biological Context. Barsky, Munzner, Gardy, and Kincaid. IEEE TVCG (Proc. InfoVis) 14(6):1253-1260, 2008.]

Why have a computer in the loop?

Computer-based visualization systems provide visual representations of datasets designed to help people carry out tasks more effectively.

- beyond human patience: scale to large datasets, support interactivity
  - consider: what aspects of hand-drawn diagrams are important?



[Cerebral: a Cytoscape plugin for layout of and interaction with biological networks using subcellular localization annotation. Barsky, Gardy, Hancock, and Munzner. Bioinformatics 23(8):1040-1042, 2007.]

Why depend on vision?

Computer-based visualization systems provide visual representations of datasets designed to help people carry out tasks more effectively.

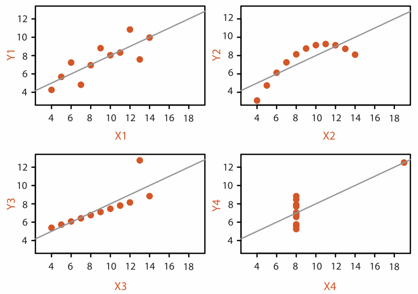
- human visual system is high-bandwidth channel to brain
  - overview possible due to background processing
    - subjective experience of seeing everything simultaneously
    - significant processing occurs in parallel and pre-attentively
- sound: lower bandwidth and different semantics
  - overview not supported
    - subjective experience of sequential stream
- touch/haptics: impoverished record/replay capacity
  - only very low-bandwidth communication thus far
- taste, smell: no viable record/replay devices

Why show the data in detail?

- summaries lose information
  - confirm expected and find unexpected patterns
  - assess validity of statistical model

Anscombe's Quartet

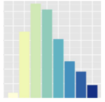
Identical statistics	
x mean	9
x variance	10
y mean	8
y variance	4
x/y correlation	1

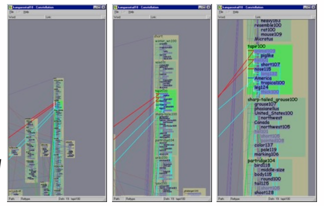


Idiom design space

The design space of possible vis idioms is huge, and includes the considerations of both how to create and how to interact with visual representations.

- **idiom**: distinct approach to creating or manipulating visual representation
  - how to draw it: **visual encoding** idiom
    - many possibilities for how to create
  - how to manipulate it: **interaction** idiom
    - even more possibilities
      - make single idiom dynamic
      - link multiple idioms together through interaction





[A layered grammar of graphics. Wickham. Journal of Computational and Graphical Statistics 19:1 (2010), 3–28.]

[Interactive Visualization of Large Graphs and Networks. Munzner. Ph.D. thesis, Stanford University Department of Computer Science, 2000.]

Why focus on tasks and effectiveness?

Computer-based visualization systems provide visual representations of datasets designed to help people carry out tasks more effectively.

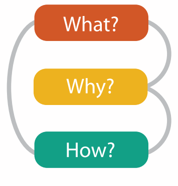
- tasks serve as constraint on design (as does data)
  - idioms do not serve all tasks equally!
  - challenge: recast tasks from domain-specific vocabulary to abstract forms
- most possibilities ineffective
  - validation is necessary, but tricky
  - increases chance of finding good solutions if you understand full space of possibilities
- what counts as effective?
  - novel: enable entirely new kinds of analysis
  - faster: speed up existing workflows

Resource limitations

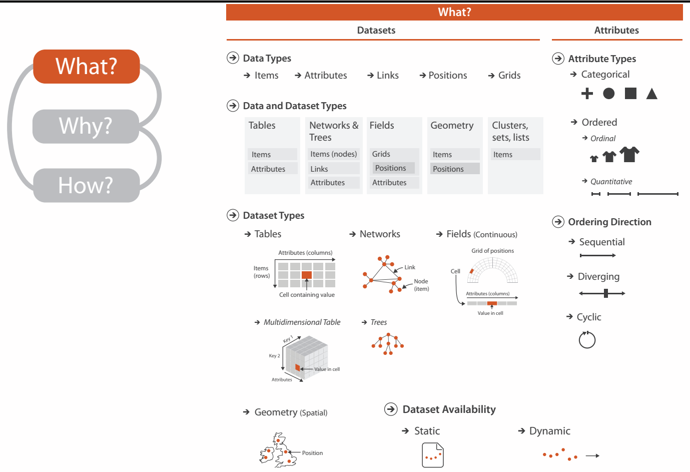
Vis designers must take into account three very different kinds of resource limitations: those of computers, of humans, and of displays.

- computational limits
  - processing time
  - system memory
- human limits
  - human attention and memory
- display limits
  - pixels are precious resource, the most constrained resource
  - **information density**: ratio of space used to encode info vs unused whitespace
    - tradeoff between clutter and wasting space, find sweet spot between dense and sparse

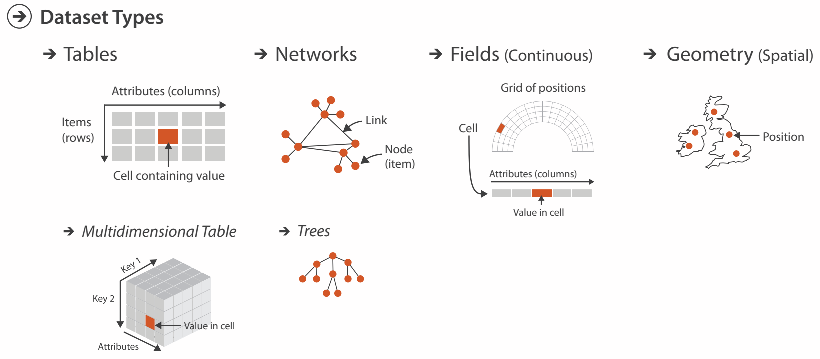
Analysis: What, why, and how



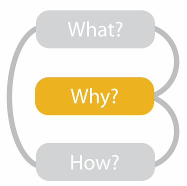
- what is shown?
  - **data** abstraction
- why is the user looking at it?
  - **task** abstraction
- how is it shown?
  - **idiom**: visual encoding and interaction
- abstract vocabulary avoids domain-specific terms
  - translation process iterative, tricky
- what-why-how analysis framework as scaffold to think systematically about design space

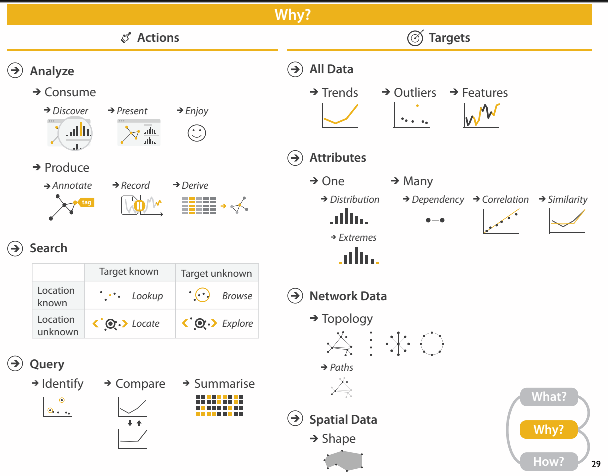


Dataset types



Attribute types

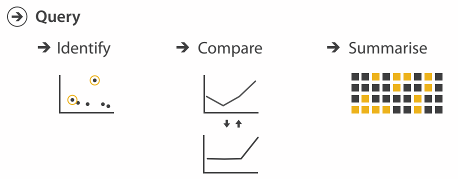




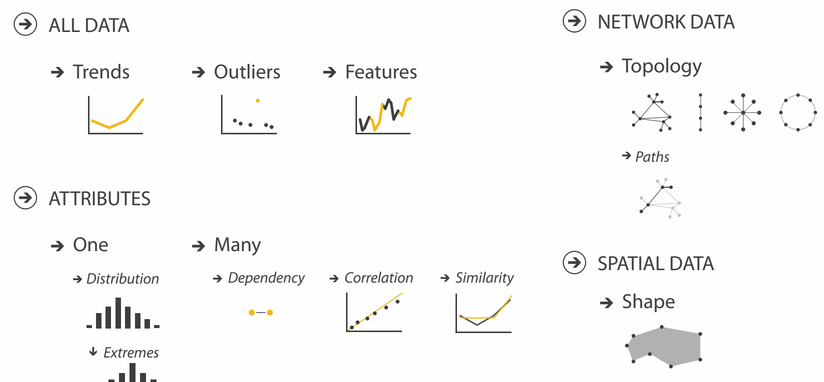
- {action, target} pairs
  - *discover distribution*
  - *compare trends*
  - *locate outliers*
  - *browse topology*

Actions: low-level query

- how much of the data matters?
  - one, some, all

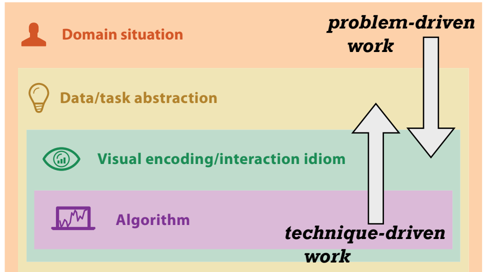


Why: Targets



Four Levels of Design

- two more levels to consider
  - domain problem: all aspects of user context
  - algorithm: efficient implementation of idioms

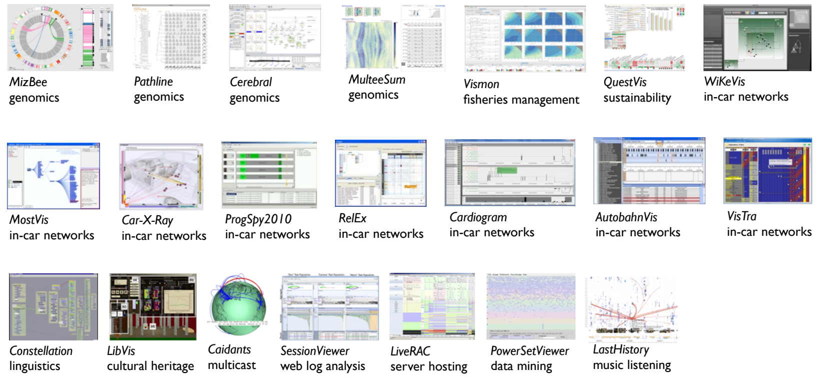


Nested Levels of Design and Validation

- Domain situation
  - Observe target users using existing tools
- Data/task abstraction
- Visual encoding/interaction idiom
  - Justify design with respect to alternatives
- Algorithm
  - Measure system time/memory
  - Analyze computational complexity
- Analyze results qualitatively
- Measure human time with lab experiment (*user study*)
- Observe target users after deployment (*field study*)
- Measure adoption

- mismatch: cannot show idiom good with system timings
- mismatch: cannot show abstraction good with lab study

Design Studies: Lessons learned after 21 of them



Marks and channels

- marks
  - geometric primitives
- channels
  - control appearance of marks

Points · Lines · Areas

Position: Horizontal, Vertical, Both · Color

Shape · Tilt

Size: Length, Area, Volume

Channels: Expressiveness types and effectiveness rankings

Magnitude Channels: Ordered Attributes

- Position on common scale
- Position on unaligned scale
- Length (1D size)
- Tilt/angle
- Area (2D size)
- Depth (3D position)
- Color luminance
- Color saturation
- Curvature
- Volume (3D size)

Identity Channels: Categorical Attributes

- Spatial region
- Color hue
- Motion
- Shape

Best · Effectiveness · Least · Same · Same

Encode

Encode

- Arrange
  - Express
  - Separate
  - Order
  - Align
  - Use
- Map from categorical and ordered attributes
  - Color
    - Hue
    - Saturation
    - Luminance
  - Size, Angle, Curvature, ...
  - Shape
  - Motion
    - Direction, Rate, Frequency, ...

What? Why? How?

Rules of Thumb

- No unjustified 3D
- Eyes beat memory
- Resolution over immersion
- Overview first, zoom and filter, details on demand
- Function first, form next
- ...

How?

Encode

- Arrange: Express, Separate, Order, Align, Use
- Map from categorical and ordered attributes
  - Color: Hue, Saturation, Luminance
  - Size, Angle, Curvature, ...
  - Shape
  - Motion: Direction, Rate, Frequency, ...

Manipulate: Change, Select, Navigate

Facet: Juxtapose, Partition, Superimpose

Reduce: Filter, Aggregate, Embed

What? Why? How?

Arrange space

Encode

- Arrange
  - Express
  - Separate
  - Order
  - Align
  - Use

Arrange tables

- Express Values
- Separate, Order, Align Regions
  - Separate
  - Order
  - Align
- 1 Key *List* · 2 Keys *Matrix* · 3 Keys *Volume* · Many Keys *Recursive Subdivision*
- Axis Orientation
  - Rectilinear
  - Parallel
  - Radial
- Layout Density
  - Dense
  - Space-Filling

Arrange spatial data

- Use Given
  - Geometry
    - *Geographic*
    - *Other Derived*
  - Spatial Fields
    - *Scalar Fields (one value per cell)*
      - *Isocontours*
      - *Direct Volume Rendering*
    - *Vector and Tensor Fields (many values per cell)*
      - *Flow Glyphs (local)*
      - *Geometric (sparse seeds)*
      - *Textures (dense seeds)*
      - *Features (globally derived)*

Arrange networks and trees

- **Node-link Diagrams** — Connections and Marks — ✔ NETWORKS ✔ TREES
- **Adjacency Matrix** — Derived Table — ✔ NETWORKS ✔ TREES
- **Enclosure** — Containment Marks — ✘ NETWORKS ✔ TREES

Color: Luminance, saturation, hue

- 3 channels
  - identity for categorical
    - hue
  - magnitude for ordered
    - luminance
    - saturation
- other common color spaces
  - RGB: poor choice for visual encoding
  - HSL: better, but beware
    - lightness ≠ luminance

Luminance · Saturation · Hue

Corners of the RGB color cube · L from HLS All the same · Luminance values

Manipulate

- Change View Over Time
- Select
- Navigate
  - Item Reduction
    - *Zoom* — *Geometric or Semantic*
    - *Pan/Translate*
    - *Constrained*
  - Attribute Reduction
    - *Slice*
    - *Cut*
    - *Project*

Facet

- Juxtapose
- Partition
- Superimpose
- Share Encoding: Same/Different
  - *Linked Highlighting*
- Share Data: All/Subset/None
- Share Navigation

Juxtapose and coordinate views

- Share Encoding: Same/Different
  - *Linked Highlighting*
- Share Data: All/Subset/None
- Share Navigation

Reduce items and attributes

- reduce/increase: inverses
- filter
  - pro: straightforward and intuitive
    - to understand and compute
  - con: out of sight, out of mind
- aggregation
  - pro: inform about whole set
  - con: difficult to avoid losing signal
- not mutually exclusive
  - combine filter, aggregate
  - combine reduce, change, facet

Reducing Items and Attributes

- Filter
  - Items
  - Attributes
- Aggregate
  - Items
  - Attributes

Reduce: Filter, Aggregate, Embed

Embed: Focus+Context

- combine information within single view
- elide
  - selectively filter and aggregate
- superimpose layer
  - local lens
- distortion design choices
  - region shape: radial, rectilinear, complex
  - how many regions: one, many
  - region extent: local, global
  - interaction metaphor