Visualization Analysis & Design

Tamara Munzner
Department of Computer Science
University of British Columbia

Microsoft Research
February 19 2015, Seattle WA

http://www.cs.ubc.ca/~tmm/talks.html#vad15seattle
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

Why represent all the data?

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

- summaries lose information, details matter
  – confirm expected and find unexpected patterns
  – assess validity of statistical model

**Anscombe’s Quartet**

<table>
<thead>
<tr>
<th>Identical statistics</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>x mean</td>
<td>9</td>
</tr>
<tr>
<td>x variance</td>
<td>10</td>
</tr>
<tr>
<td>y mean</td>
<td>8</td>
</tr>
<tr>
<td>y variance</td>
<td>4</td>
</tr>
<tr>
<td>x/y correlation</td>
<td>1</td>
</tr>
</tbody>
</table>
Why are there 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
Why focus on tasks and effectiveness?

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

• what counts as effective?
  – novel: enable entirely new kinds of analysis
  – faster: speed up existing workflows

• most possibilities ineffective
  – increase chance of finding good solutions by understanding full space of possibilities

• tasks serve as constraint on design (as does data)
  – representations do not serve all tasks equally!
Analysis framework: Four levels, three questions

• **domain** situation
  – who are the target users?

• **abstraction**
  – translate from specifics of domain to vocabulary of vis
  • **what** is shown? **data abstraction**
  • **why** is the user looking at it? **task abstraction**

• **idiom**
  • **how** is it shown?
    • **visual encoding idiom**: how to draw
    • **interaction idiom**: how to manipulate

• **algorithm**
  – efficient computation


Validation methods from different fields for each level

Anthonyology/ethnography

- 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 (*lab study*)
  - Observe target users after deployment (*field study*)
  - Measure adoption

- Design
- Computer science
- Cognitive psychology

- Anthropology/ethnography

- mismatch: cannot show idiom good with system timings
- mismatch: cannot show abstraction good with lab study
Why analyze?
• imposes a structure on huge design space
  – scaffold to help you think systematically about choices
  – analyzing existing as stepping stone to designing new


**Datasets**

- **Data Types**
  - Items
  - Attributes
  - Links
  - Positions
  - Grids

- **Data and Dataset Types**
  - **Tables**
    - Items
    - Attributes
  - **Networks & Trees**
    - Items (nodes)
    - Grids
    - Positions
    - Attributes
  - **Fields**
    - Items
    - Positions
  - **Geometry**
    - Items
    - Positions
  - **Clusters, Sets, Lists**
    - Items

**Attributes**

- **Attribute Types**
  - Categorical
    - +
    - •
    - □
    - ▲
  - Ordered
    - Ordinal
    - 
  - Quantitative

**Ordering Direction**

- Sequential
- Diverging
- Cyclic

**Dataset Availability**

- Static
- Dynamic

**Geometry (Spatial)**

**Why?**

**How?**
Dataset and data types

**Attributes (columns)**

- **Items (rows)**
- **Cell containing value**

**Dataset Types**

- **Tables**
- **Networks**

**Attribute Types**

- **Categorical**
- **Ordered**
  - **Ordinal**
  - **Quantitative**

**Spatial**

- **Fields (Continuous)**
- **Geometry (Spatial)**
  - **Grid of positions**
  - **Cell**
  - **Attributes (columns)**
  - **Value in cell**
  - **Position**
• {action, target} pairs
  – discover distribution
  – compare trends
  – locate outliers
  – browse topology
Actions I: Analyze

- **consume**
  - discover vs present
    - classic split
    - aka explore vs explain
  - enjoy
    - newcomer
    - aka casual, social

- **produce**
  - annotate, record
  - derive
    - crucial design choice
Actions II: Search

- what does user know?
  - target, location

Search

<table>
<thead>
<tr>
<th></th>
<th>Target known</th>
<th>Target unknown</th>
</tr>
</thead>
<tbody>
<tr>
<td>Location known</td>
<td>Lookup</td>
<td>Browse</td>
</tr>
<tr>
<td>Location unknown</td>
<td>Locate</td>
<td>Explore</td>
</tr>
</tbody>
</table>
Actions III: Query

• what does user know?
  – target, location

• how much of the data matters?
  – one, some, all

**Search**

<table>
<thead>
<tr>
<th>Location known</th>
<th>Target known</th>
<th>Target unknown</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Lookup</td>
<td>Browse</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Location unknown</th>
<th>Target known</th>
<th>Target unknown</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Locate</td>
<td>Explore</td>
</tr>
</tbody>
</table>

**Query**

- Identify
- Compare
- Summarize
Targets

- **All Data**
  - Trends
  - Outliers
  - Features

- **Attributes**
  - One
    - Distribution
    - Extremes
  - Many
    - Dependency
    - Correlation
    - Similarity

- **Network Data**
  - Topology
    - Paths

- **Spatial Data**
  - Shape
### Encode

<table>
<thead>
<tr>
<th>Arrangement</th>
<th>Express</th>
<th>Separate</th>
</tr>
</thead>
<tbody>
<tr>
<td>Order</td>
<td>Align</td>
<td></td>
</tr>
<tr>
<td>Use</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

#### Map
- from *categorical* and *ordered* attributes
  - Color
    - Hue
    - Saturation
    - Luminance
  - Size, Angle, Curvature, ...
  - Shape
    - + ● ■ △
  - Motion
    - Direction, Rate, Frequency, ...

### Manipulate

<table>
<thead>
<tr>
<th>Change</th>
</tr>
</thead>
<tbody>
<tr>
<td><img src="image" alt="Change" /></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Select</th>
</tr>
</thead>
<tbody>
<tr>
<td><img src="image" alt="Select" /></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Navigate</th>
</tr>
</thead>
<tbody>
<tr>
<td><img src="image" alt="Navigate" /></td>
</tr>
</tbody>
</table>

### Facet

<table>
<thead>
<tr>
<th>Juxtapose</th>
</tr>
</thead>
<tbody>
<tr>
<td><img src="image" alt="Juxtapose" /></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Partition</th>
</tr>
</thead>
<tbody>
<tr>
<td><img src="image" alt="Partition" /></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Superimpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><img src="image" alt="Superimpose" /></td>
</tr>
</tbody>
</table>

### Reduce

<table>
<thead>
<tr>
<th>Filter</th>
</tr>
</thead>
<tbody>
<tr>
<td><img src="image" alt="Filter" /></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Aggregate</th>
</tr>
</thead>
<tbody>
<tr>
<td><img src="image" alt="Aggregate" /></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Embed</th>
</tr>
</thead>
<tbody>
<tr>
<td><img src="image" alt="Embed" /></td>
</tr>
</tbody>
</table>

### Why?

What?

How?
How to encode: Arrange space, map channels

Encode

✅ **Arrange**
- Express
- Order
- Use

✅ **Map**
- from **categorical** and **ordered** attributes
  - Color
    - Hue
    - Saturation
    - Luminance
  - Size, Angle, Curvature, ...
  - Shape
    - Direction, Rate, Frequency, ...
  - Motion

✅ **Separate**

✅ **Align**
Encoding visually

• analyze idiom structure
Definitions: Marks and channels

• **marks**
  - geometric primitives

  ![Geometric primitives](image)

• **channels**
  - control appearance of marks

  ![Control appearance of marks](image)
Encoding visually with marks and channels

• analyze idiom structure
  – as combination of marks and channels

1: vertical position
mark: line

2: vertical position
horizontal position
mark: point

3: vertical position
horizontal position
color hue
mark: point

4: vertical position
horizontal position
color hue
size (area)
mark: point
Channels

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)

Spatial region
Color hue
Motion
Shape

Magnitude Channels: Ordered Attributes
Identity Channels: Categorical Attributes
Spatial region
Color hue
Motion
Shape
Channels: Matching Types

**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

• expressiveness principle
  – match channel and data characteristics
Channels: 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

- **expressiveness principle**
  - match channel and data characteristics
- **effectiveness principle**
  - encode most important attributes with highest ranked channels
### 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?
How to handle complexity: 3 more strategies

+ 1 previous

**Manipulate**
- Change
- Select
- Navigate

**Facet**
- Juxtapose
- Partition
- Superimpose

**Reduce**
- Filter
- Aggregate
- Embed

**Derive**
- change view over time
- facet across multiple views
- reduce items/attributes within single view
- derive new data to show within view
How to handle complexity: 3 more strategies

Manipulate

- Change
- Select
- Navigate

Facet

- Juxtapose
- Partition
- Superimpose

Reduce

- Filter
- Aggregate
- Embed

Derive

- change over time
  - most obvious & flexible
  of the 4 strategies
Idiom: **Animated transitions**

- smooth transition from one state to another
  - alternative to jump cuts
  - support for item tracking when amount of change is limited
- example: multilevel matrix views
  - scope of what is shown narrows down
    - middle block stretches to fill space, additional structure appears within
    - other blocks squish down to increasingly aggregated representations

Facet

- **Juxtapose**

- **Partition**

- **Superimpose**

---

**Coordinate Multiple Side By Side Views**

- **Share Encoding: Same/Different**
  - *Linked Highlighting*

- **Share Data: All/Subset/None**

- **Share Navigation**
How to handle complexity: 3 more strategies

- Manipulate
  - Change
  - Select
  - Navigate

- Facet
  - Juxtapose
  - Partition
  - Superimpose

- Reduce
  - Filter
  - Aggregate
  - Embed

- Derive

• facet data across multiple views

+ 1 previous
Idiom: **Linked highlighting**

- see how regions contiguous in one view are distributed within another
  - powerful and pervasive interaction idiom

- encoding: different
  - *multiform*

- data: all shared

Idiom: **bird’s-eye maps**

- encoding: same
- data: subset shared
- navigation: shared
  - bidirectional linking

- differences
  - viewpoint
  - (size)

- **overview-detail**

---

Idiom: **Small multiples**

- encoding: same
- data: none shared
  - different attributes for node colors
  - (same network layout)
- navigation: shared

---

## Coordinate views: Design choice interaction

<table>
<thead>
<tr>
<th>Encoding</th>
<th>Data</th>
</tr>
</thead>
<tbody>
<tr>
<td>Same</td>
<td>All: Redundant</td>
</tr>
<tr>
<td></td>
<td>Subset: Overview/Detail</td>
</tr>
<tr>
<td></td>
<td>None: Small Multiples</td>
</tr>
<tr>
<td>Different</td>
<td>All: Multiform</td>
</tr>
<tr>
<td></td>
<td>Subset: Multiform, Overview/Detail</td>
</tr>
<tr>
<td></td>
<td>None: No Linkage</td>
</tr>
</tbody>
</table>

### why juxtapose views?

- **benefits: eyes vs memory**
  - lower cognitive load to move eyes between 2 views than remembering previous state with single changing view
- **costs: display area, 2 views side by side each have only half the area of one view**
Partition into views

• how to divide data between views
  – encodes association between items using spatial proximity
  – major implications for what patterns are visible
  – split according to attributes

• design choices
  – how many splits
    • all the way down: one mark per region?
    • stop earlier, for more complex structure within region?
  – order in which attribs used to split
  – how many views
Partitioning: List alignment

• single bar chart with grouped bars
  – split by state into regions
    • complex glyph within each region showing all ages
  – compare: easy within state, hard across ages

• small-multiple bar charts
  – split by age into regions
    • one chart per region
  – compare: easy within age, harder across states
Partitioning: Recursive subdivision

• split by type
• then by neighborhood
• then time
  – years as rows
  – months as columns

System: HIVE

Partitioning: Recursive subdivision

- switch order of splits
  - neighborhood then type
- very different patterns

Partitioning: Recursive subdivision

- different encoding for second-level regions
  - choropleth maps

How to handle complexity: 3 more strategies

- **Manipulate**
  - Change
  - Select
  - Navigate

- **Facet**
  - Juxtapose
  - Partition
  - Superimpose

- **Reduce**
  - Filter
  - Aggregate
  - Embed

- **Derive**

  • reduce what is shown within single view
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, facet, change, derive

Reducing Items and Attributes

- Filter
  - Items
  - Attributes

- Aggregate
  - Items
  - Attributes
Idiom: **boxplot**

- static item aggregation
- task: find distribution
- data: table
- derived data
  - 5 quant attribs
    - median: central line
    - lower and upper quartile: boxes
    - lower upper fences: whiskers
      - values beyond which items are outliers
  - outliers beyond fence cutoffs explicitly shown

[40 years of boxplots. Wickham and Stryjewski. 2012. had.co.nz]
Idiom: Dimensionality reduction for documents

- attribute aggregation
  - derive low-dimensional target space from high-dimensional measured space

**Task 1**

**What?**
- In High-dimensional data
- Out 2D data

**Why?**
- Produce
- Derive

**Task 2**

**What?**
- In 2D data
- Out Scatterplot
- Out Clusters & points

**Why?**
- Discover
- Explore
- Identify

**How?**
- Encode
- Navigate
- Select

**Task 3**

**What?**
- In Scatterplot
- In Clusters & points
- Out Labels for clusters

**Why?**
- Produce
- Annotate
More Information

• this talk
  http://www.cs.ubc.ca/~tmm/talks.html#vad15seattle

• book page (including tutorial lecture slides)
  http://www.cs.ubc.ca/~tmm/vadbook
  – 20% promo code for book+ebook combo: HVN17

  – illustrations: Eamonn Maguire

• papers, videos, software, talks, full courses
  http://www.cs.ubc.ca/group/infovis
  http://www.cs.ubc.ca/~tmm