Pixel-Adaptive Visual Comparison Between Many Phylogenetic Trees

Tamara Munzner
Department of Computer Science
University of British Columbia

Asilomar Microcomputer Workshop #50
25 Apr 2024

http://www.cs.ubc.ca/~tmm/talks.html#amw24
Hi again, 15 years later!

• still doing data visualization
  – yet more papers / projects / videos / software
  – I edit a book series

http://www.cs.ubc.ca/~tmm/talks.html#amw24
Visualization book series highlights

• Data Sketches, by Nadieh Bremer & Shirley Wu

• Making with Data, by multitudes

• Building Science Graphics, by Jen Christiansen

Hi again, 15 years later!

• still doing data visualization
  – yet more papers / projects / videos / software
  – I edit a book series
  – I wrote a book

http://www.cs.ubc.ca/~tmm/talks.html#amw24
Visualization Analysis & Design book

What?

Datasets

Attributes

Why?

Actions

Targets

How?

Encode

Manipulate

Facet

Reduce

→ Arrange
  → Express
  → Separate

→ Order
  → Align

→ Use

→ Map
  from categorical and ordered attributes
  → Color
    → Hue
    → Saturation
    → Luminance

→ Size, Angle, Curvature, ...

→ Navigate

→ Shape
  → Motion
    → Direction, Rate, Frequency, ...

→ Change

→ Juxtapose

→ Partition

→ Filter

→ Aggregate

→ Superimpose

→ Embed

domain
abstraction

idiom
algorithm
Nested model: Four levels of visualization concerns

Nested model: Four levels of visualization concerns

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

Nested model: Four levels of visualization concerns

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

• **abstraction**
  – translate from specifics of domain to vocabulary of vis
Nested model: Four levels of visualization concerns

- **domain** situation
  - **who** are the target users?

- **abstraction**
  - translate from specifics of domain to vocabulary of vis
  - **what** is shown? **data abstraction**


Nested model: Four levels of visualization concerns

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

• *abstraction*
  – translate from specifics of domain to vocabulary of vis
  – **what** is shown? **data abstraction**
    - often don’t just draw what you’re given: transform to new form

Nested model: Four levels of visualization concerns

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

• **abstraction**
  – translate from specifics of domain to vocabulary of vis
  – **what** is shown? **data abstraction**
    • often don’t just draw what you’re given: transform to new form
  – **why** is the user looking at it? **task abstraction**
Nested model: Four levels of visualization concerns

- **domain** situation
  - **who** are the target users?

- **abstraction**
  - translate from specifics of domain to vocabulary of vis
  - **what** is shown? **data abstraction**
    - often don’t just draw what you’re given: transform to new form
  - **why** is the user looking at it? **task abstraction**

- **idiom**
  - **how** is it shown?

---


Nested model: Four levels of visualization concerns

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

• **abstraction**
  – translate from specifics of domain to vocabulary of vis
  – **what** is shown? **data abstraction**
    • often don’t just draw what you’re given: transform to new form
  – **why** is the user looking at it? **task abstraction**

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

Nested model: Four levels of visualization concerns

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

• **abstraction**
  – translate from specifics of domain to vocabulary of vis
  – **what** is shown? **data abstraction**
    • often don’t just draw what you’re given: transform to new form
  – **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

Nested model: Four levels of visualization concerns

- **domain situation**
  - **who** are the target users?

- **abstraction**
  - translate from specifics of domain to vocabulary of vis
  - **what** is shown? **data abstraction**
    - often don’t just draw what you’re given: transform to new form
  - **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

---


Why is validation difficult?

• different threats to validity at each level

Why is validation difficult?

• different threats to validity at each level

[Domain situation
You misunderstood their needs

Why is validation difficult?

• different threats to validity at each level

[Domain situation] You misunderstood their needs

[Data/task abstraction] You're showing them the wrong thing

Why is validation difficult?

• different threats to validity at each level

Domain situation
You misunderstood their needs

Data/task abstraction
You're showing them the wrong thing

Visual encoding/interaction idiom
The way you show it doesn’t work

Why is validation difficult?

- different threats to validity at each level

Validation solution: Use methods from appropriate fields at each level

Validation solution: Use methods from appropriate fields at each level

- **Algorithm**
  Measure system time/memory
  Analyze computational complexity

computer science

Validation solution: Use methods from appropriate fields at each level

Algorithm
Measure system time/memory
Analyze computational complexity

computer science
technique-driven work

Validation solution: Use methods from appropriate fields at each level

- **Design**
  - Visual encoding/interaction idiom
    - Justify design with respect to alternatives

- **Computer science**
  - Algorithm
    - Measure system time/memory
    - Analyze computational complexity

- **Cognitive psychology**
  - Analyze results qualitatively
  - Measure human time with lab experiment (*lab study*)

Validation solution: Use methods from appropriate fields at each level

- **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

Validation solution: Use methods from appropriate fields at each level

- avoid mismatches between level and validation

<table>
<thead>
<tr>
<th>Domain situation</th>
<th>Data/task abstraction</th>
<th>Visual encoding/interaction idiom</th>
<th>Algorithm</th>
<th>Measure adoption</th>
</tr>
</thead>
<tbody>
<tr>
<td>Observe target users using existing tools</td>
<td></td>
<td>Justify design with respect to alternatives</td>
<td>Measure system time/memory Analyze computational complexity</td>
<td>Analyze results qualitatively Measure human time with lab experiment (<em>lab study</em>)</td>
</tr>
</tbody>
</table>

Validation solution: Use methods from appropriate fields at each level

- avoid mismatches between level 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 (*lab study*)

- **Observation after deployment**
  - Observe target users after deployment (*field study*)

- **Measure adoption**

- **Anthropology/ethnography**
  - problem-driven work

- **Design**
  - technique-driven work

- **Computer science**

- **Cognitive psychology**

- **Anthropology/ethnography**

Design Study Methodology

Reflections from the Trenches and from the Stacks

http://www.cs.ubc.ca/labs/imager/tr/2012/dsm/

Design Study Methodology: Reflections from the Trenches and from the Stacks.
Lessons learned from the trenches: 20+ between us

- Cerebral genomics
- MizBee genomics
- Pathline genomics
- MulteeSum genomics
- Vismon fisheries management
- QuestVis sustainability
- WiKeVis in-car networks

- MostVis in-car networks
- Car-X-Ray in-car networks
- ProgSpy2010 in-car networks
- RelEx in-car networks
- Cardiogram in-car networks
- AutobahnVis in-car networks
- VisTra in-car networks

- Constellation linguistics
- LibVis cultural heritage
- Caidants multicast
- SessionViewer web log analysis
- LiveRAC server hosting
- PowerSetViewer data mining
9-stage framework

PRECONDITION

CORE

ANALYSIS

PRECONDITION

CORE

ANALYSIS

RelEx

Visualization for Actively Changing Overlay Network Specifications

joint work with:
Michael Sedlmair, Annika Frank, Andreas Butz

http://www.cs.ubc.ca/labs/imager/tr/2012/relex/
Aggregated Dendrograms for Visual Comparison Between Many Phylogenetic Trees


Phylogenetic tree

Evolutionary relationships of organisms

Human
Chimpanzee
Macaque

Genetic information

ATGGACA
ATGGACA
ACGGACA

Computational workflow

Phylogenetic tree
Many phylogenetic trees

- Understand relationships between genes and species trees
- Explore trees generated with different methods and data

Human
ATGGACAG
Chimpanzee
ATGGACAG
Macaque
ACGGACAG

Genetic information

Computational workflow

Phylogenetic tree
Scalability of existing tree comparison systems

#Trees: how many trees to compare

Level of detail (LoD):
how much details are visible
Scalability of existing tree comparison systems

#Trees: how many trees to compare

Pairs

Few in full

Simplified structure  Full topology

Level of detail (LoD): how much details are visible

TreeJuxtaposer, Munzner, Guimbretière, Zhang, Zhou. SIGGRAPH 2003
Scalability of existing tree comparison systems

# Trees: how many trees to compare

- Thousands: Many as points
- Hundreds
- Dozens
- Pairs: Few in full

Level of detail (LoD): how much details are visible

Tree space.
Hillis, Health, John.
Systematic Biology 2005.
Scalability of existing tree comparison systems

#Trees: how many trees to compare

- Thousands: Many as points
- Hundreds: Dozens at multi-scale
- Dozens: Few in full
- Pairs: Single point, Simplified structure, Full topology

Level of detail (LoD): how much details are visible

Interactive visual comparison of multiple trees.
Comparing many phylogenetic trees

#Trees: how many trees to compare

- Thousands: Many as points
- Hundreds: Dozens at multi-scale
- Dozens: Few in full
- Pairs: Single point, Simplified structure, Full topology

Level of detail (LoD): how much details are visible

Hundreds / thousands at multi-scale?
Contributions at abstraction, idiom, & algorithm levels

- data and task **abstractions** for comparison of phylogenetic trees
- new visual encoding **idiom**: Aggregated Dendrogram
  - compact tree representation that focuses on selected subtrees
  - **algorithm** that adapts to available screen space
- interactive multi-view tool: ADView
  - covers multiple levels of details for tree comparison
Data abstraction: Trees

Reference tree vs. Tree collection
Task abstraction
Task abstraction

**Topological** relationships & distance between subtrees / leaf nodes

![Diagram showing topological relationships and distances between subtrees and leaf nodes](Image)
Task abstraction

**Topological** relationships & distance between subtrees / leaf nodes

**Leaf** node memberships compared to reference tree

Separated          Nested          Distance

Separated     Nested     Distance

Exact match     Partial match

Reference       Tree1       Tree2

\[ \begin{array}{ccc} 
S1 & S1 & S1 \\
S2 & S2 & S2 \\
S3 & S3 & S3 \\
S4 & S4 & S4 \\
S5 & S5 & S5 \\
\end{array} \]
Aggregated Dendrogram: Intuition

Use glyphs to compress a tree according to user selections
Visual design: focus + context

• focus
  – selected subtrees

(List task)
Visual design: focus + context

- focus
  - selected subtrees

Proportion of matching leaves

# leaf nodes

(Leaf task)
Visual design: focus + context

• focus
  – selected subtrees
  – topological relationships between them

(Topology task)
Visual design: focus + context

• focus
  – selected subtrees
  – topological relationships between them
Visual design: focus + context

• focus
  – selected subtrees
  – topological relationships between them

• context
  – neighboring subtrees
Visual design: focus + context

• focus
  – selected subtrees
  – topological relationships between them

• context
  – neighboring subtrees
  – upstream topology and root
Visual design: focus + context

• focus
  – selected subtrees
  – topological relationships between them

• context
  – neighboring subtrees
  – upstream topology and root
  – missing leaf nodes
Visual design: algorithm adapts to space

- show more info when space permitted
  - labels
  - # leaf nodes
  - neighboring blocks
ADView interface: Multi-level structure across views
Interface walkthrough: reference tree

Individual tree
subtree
branch and leaf
Interface walkthrough: individual & cluster ADs

Tree collection
Subset of trees

Individual tree
Subtree
Interface walkthrough: treespace
Validation with many biologists

- worked closely with a biology PhD student (second author)
Validation with many biologists

• worked closely with a biology PhD student (second author)
• demos, interviews and discussions
  – 10 biologists at different times throughout project
Validation with many biologists

• worked closely with a biology PhD student (second author)
• demos, interviews and discussions
  – 10 biologists at different times throughout project
• user study sessions
  – 5 biologists, using their own datasets
Validation with many biologists

- worked closely with a biology PhD student (second author)
- demos, interviews and discussions
  - 10 biologists at different times throughout project
- user study sessions
  - 5 biologists, using their own datasets
- biologists confirmed
  - validity of data and task abstractions
  - utility of ADView
https://www.youtube.com/watch?v=2SLcz7KNLJw
Problem-driven visualization with design study methodology

• work through all four levels of nested model
  – investigate domain

  – identify abstractions
    • crucial -- & difficult -- iterative process

  – select or create appropriate idioms

  – develop new algorithms
    • if need be
More information

• this talk
  http://www.cs.ubc.ca/~tmm/talks.html#amw24

• book
  http://www.cs.ubc.ca/~tmm/vadbook
  (hardcopy on demo/stuff table)

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