Scalable Visualization with Accordion Drawing
Tamara Munzner
University of British Columbia
Department of Computer Science
Just work with James Slack, Kristian Hildebrand, Katherine St. John

Problem: Comparing Evolutionary Trees

Common Dataset Size Today

Future Goal: 10M Node Tree of Life

Sequence Juxtaposer

Paper Comparison: Multiple Trees

Common Dataset Size Today

Guaranteed Visibility Challenges

Tree Juxtaposer

What's Hard?

TJ Contributions

Best Corresponding Node

Marking Structural Differences

Guaranteed Visibility

Guaranteed Visibility Challenges

Guaranteed Visibility Challenges

Paper Comparison: Multiple Trees

Tree Juxtaposer

What's Hard?

TJ Contributions

Best Corresponding Node

Marking Structural Differences

Guaranteed Visibility

Guaranteed Visibility Challenges

Marking Structural Differences

Guaranteed Visibility

Guaranteed Visibility Challenges
Guaranteed Visibility Challenges
- hard with larger datasets
- reasons a mark could be invisible
  - outside the window
    - AD solution: constrained navigation
  - underneath other marks
    - AD solution: avoid 3D
- undersized features
  - AD solution: smart culling

Lots More Information
- download software: http://olduvai.sf.net
- TreeJuxtaposer, SequenceJuxtaposer
- many papers, talks, videos: http://www.cs.ubc.ca/~tmm

Guaranteed Visibility Rationale
- relief from exhaustive exploration
- missed marks lead to false conclusions
- hard to navigate small datasets
- controversy: does distortion help or hurt?
- strong rationale for comparison
- infrastructure needed for efficient computation

Rending Complexity
- reduce drawing complexity with sneaky culling
- For each frame: draw representative visible subset, not entire dataset
- (Total number of drawn objects per frame) << (Total dataset items)
- In tree dataset with 600,000 leaves, draw only 1000 leaves
- In sequence datasets, aggregate dense regions in software

Conflicting Rationale
- relief from exhaustive exploration
- missed marks lead to false conclusions
- hard to navigate small datasets
- controversy: does distortion help or hurt?
- strong rationale for comparison
- infrastructure needed for efficient computation

PRISAD Architecture
- world-space discretization
- preprocessing
  - initializing data structures
  - placing geometry
- screen-space rendering
  - frame updating
  - analyzing navigation state
  - drawing geometry

Stretch and Squish Navigation
- user selects any region to grow or shrink
- everything else shrinks or grows, accordingly
- goal: handle millions of items, landmarks always stay visible

Navigation Algorithm
- flow of our navigation algorithm

Navigation Algorithm Complexity
- logarithmic complexity: $|Q| \approx |K| \log |N| < |N|
- $Q$: Lines needing ratio updates
- $K$: Lines to move
- $N$: All lines
- many positions change, but few ratios require updates
- moving 2 grid lines only requires changing ratios for 8 split lines
- sublines not affected will conserve their internal ratios
- speed: under 1 millisecond for $|N| = 2,000,000$ lines

Lots More Information
- download software: http://olduvai.sf.net
- TreeJuxtaposer, SequenceJuxtaposer
- many papers, talks, videos: http://www.cs.ubc.ca/~tmm

Successive Navigations Preserve Visual History
- simple to use
- underlying infrastructure is complex to implement
- standard graphics pipeline has a single, monolithic transformation
- fast 4x4 matrix multiplication
- stretch and squish cannot be implemented using this pipeline

Implementing Stretch and Squish Navigation
- simple to use
- underlying infrastructure is complex to implement
- standard graphics pipeline has a single, monolithic transformation
- fast 4x4 matrix multiplication
- stretch and squish cannot be implemented using this pipeline

Lots More Information
- download software: http://olduvai.sf.net
- TreeJuxtaposer, SequenceJuxtaposer
- many papers, talks, videos: http://www.cs.ubc.ca/~tmm

Navigation Algorithm Complexity
- logarithmic complexity: $|Q| \approx |K| \log |N| < |N|
- $Q$: Lines needing ratio updates
- $K$: Lines to move
- $N$: All lines
- many positions change, but few ratios require updates
- moving 2 grid lines only requires changing ratios for 8 split lines
- sublines not affected will conserve their internal ratios
- speed: under 1 millisecond for $|N| = 2,000,000$ lines

Lots More Information
- download software: http://olduvai.sf.net
- TreeJuxtaposer, SequenceJuxtaposer
- many papers, talks, videos: http://www.cs.ubc.ca/~tmm