An Introduction to Information Visualization Techniques for Exploring Large Database

Jing Yang
Fall 2005

Visualization for Large Datasets

Class 8
Motivation

Out5d dataset (5 dimensions, 16384 data items)

Question

- What problems will be caused by such datasets?
### Question

- What problems will be caused by such datasets?
  - Cluttered display
  - Difficult user interactions

### Discussion 1

- How to overcome these problems?
Discussion 2

- Design a display to improve the parallel coordinates display.

Case study: Improve a Cluttered PC Display

- Different approaches:
  - Intensity (pixel space)
  - Histogram (data space)
  - Clustering (data space)
Intensity

Figure 2 – a) Visualization of Pollen data set; b) Visualization of the same data set, with the intensity of the grey levels set proportionally to the superimposition of the poly-lines over a black background [Wegman and Luo 1999].

Histogram

- Paper: Uncovering Clusters in Crowded Parallel Coordinates Visualizations
  Artero et. al. Infovis 04
- Key ideas
Paper: Uncovering Clusters in Crowded Parallel Coordinates Visualizations

Left to right: original, intensity, histogram approaches

Paper: Uncovering Clusters in Crowded Parallel Coordinates Visualizations

- Scale factor $S$

Figure 6 – $Sin^t$ data visualized with different values of $s$.

- Filtering: AND, OR thresholding
Paper: Uncovering Clusters in Crowded Parallel Coordinates Visualizations

Selection

Clustering

Hierarchical Parallel Coordinates for Exploration of Large Datasets  Fua et. al. Vis99

The following slides are from Fua’s presentation
Goal

- **Interactive** visualization of large multivariate data sets on parallel coordinates.
- Basis for development: XmdvTool
  - A visualization system that integrates multiple techniques for displaying and visually exploring multivariate data.
- Quantify "large" data set to contain hundreds of thousands of data elements or more.

Hierarchical Clustering

To structure and present data at different levels-of-abstraction.

Definitions:

Let $E$ be the a set of $k$ $N$-dimensional objects, i.e.,

$$E = \{e_1, e_2, e_3, ..., e_k\}$$

where $e_i$ is an $N$-vector:

$$e_i = \{x_{i1}, x_{i2}, x_{i3}, ..., x_{iN}\}.$$

An $m$-partition $P$ of $E$ breaks $E$ into $m$ subsets $\{P_1, P_2, ..., P_m\}$ satisfying the following two criteria:

- $P_i \cap P_j = \emptyset$ for all $1 \leq i, j \leq m$, $i \neq j$ , and
- $\bigcup_{i=1}^{m} P_i = E$
Overview of Hierarchical Clustering

Cluster Tree Construction

- To cluster a set of objects means to partition them based on some proximity measure.
- Conventional clustering methods may fail on large data sets due to storage and processing requirements.
- Recent clustering algorithms for large data sets:
  - Dysect (Andreae ‘90), Clarans (Ng ‘94), Birch (Zhang ‘96), Cure (Guha ‘98)
Cluster Summarization

- Number of data points enclosed
- Mean/Median/Standard deviation
- Extents/Quartile
- Measure of the cluster size, $\mathcal{V}$
  - If $T_i$ is an ancestor node of $T_j \Rightarrow \mathcal{V}_i > \mathcal{V}_j$

Display of Aggregate Information
Cluster Selection

- A cut across the tree is a partition on \( E \) if it intersects any given path exactly once.
- \( V_{\text{max}} = \max_{i \in T} V_i \) and \( V_{\text{min}} = \min_{i \in T} V_i \)
- \( w \in \{ V_{\text{max}}, V_{\text{min}} \} \)

\[
\text{Algorithm } \text{TraverseCutPartition}(T_i, w) \\
1. \quad \text{if } (v_i \leq w \text{ or } T_i.\text{num\_children} = 0) \text{ then} \\
2. \quad S \leftarrow S \cup T_i \\
3. \quad \text{return} \\
4. \quad \text{for } j \leftarrow 0 \text{ to } T_i.\text{num\_children} \text{ do} \\
5. \quad \text{TraverseCutPartition}(T_i.\text{child}[j], w)
\]
Monochromatic line drawings present an inherent difficulty in parallel-coordinates.

Use colors to discriminate such cases.
Assign colors through a similarity measure: cluster proximity.

With proximity-based coloring, relationships among clusters are highlighted.
Difficult to distinguish adjacent elements belonging to different clusters.

\[
C_0 = 0.5 \\
C_i = C_{\text{parent}(i)} + \frac{\pi(i)}{K^t + 1} \tag{1}
\]

\[
\pi(i) = \begin{cases} 
+1 & \text{if } i \text{ is odd} \\
-1 & \text{if } i \text{ is even} 
\end{cases} \tag{2}
\]
Color Assignment (2)

- Revised Equation:

\[
C_i = C_{parent(i)} + \pi(i) \left( b^i + \frac{1}{K^{l+1}} \right) \tag{3}
\]

Buffer, b<1

Interactive Data Exploration

- What tasks comprise a typical exploration process?
- How can they be made intuitive?

✓ Uncover patterns or anomalies not immediately obvious or comprehensible.
Structure-Based Brushing

- To give a general idea of the hierarchical structure.
- To localize and navigate a subspace in structure space.

![](image)

Drill-Down/Roll-Up

- View data at varying levels-of-detail via a smooth continuous level-of-detail control.
- *Selective* drill-down/roll-up.

![](image)
Dimension Zooming

Allow users to examine interesting data characteristics within a narrow brush strip.

Mini-map:
To maintain contextual information.

Dynamic Masking

- To interactively fade in/out the brushed nodes.
- To maintain context while reducing clutter.
Extent Scaling

- Allow thickness of bands to be varied dynamically.
- Isolate overlapping bands while maintaining relative sizes of extents.

Contributions (1)

- **Cluster-based hierarchy** that supports smooth multiresolutional display of high-dimensional data.
- **Proximity-based coloring scheme** assures data and clusters from similar parts of the hierarchical structure shown in similar colors.
- **Variable width opacity bands** visually encodes aggregation information at each cluster.
- **Structure-based brushing** facilitates efficient color-based localization of data.
Contributions (2)

- Smooth drilling operations using the cluster size measure as a level-of-detail controller.
- Other navigation operations:
  - selective drilling
  - dimension zooming, mini-map
  - extent scaling
  - dynamic masking

Demo – HPC and more
Discussion

- Compare the intensity, histogram, and clustering approaches

Paper: Dynamic Visualization of Transient Data Streams

Background
- Data streams: time-varying information that arrives continuously, unpredictably, and unboundedly without any persistent patterns
- Examples:
  - Newswires, internet click streams, network resource measurements, phone call records, and remote sensing imagery
- General requirements: fusing a large amount of previously analyzed information with a smaller amount of new information
- Challenges: time critical, influx rate exceeds processing rate
Paper: Dynamic Visualization of Transient Data Streams

Background
- Data streams: time-varying information that arrives continuously, unpredictably, and unboundedly without any persistent patterns
- Examples:
  - Newswires, internet click streams, network resource measurements, phone call records, and remote sensing imagery
- General requirements: fusing a large amount of previously analyzed information with a smaller amount of new information
- Challenges: time critical, influx rate exceeds processing rate

Figure 1: a) An illustration of the operation. b) A sketch of a hyperspectral image set with 169 spectral bands ranging from very short to very long bands. c) A color infrared (CIR) imagery of the semi-desert areas in Eastern Washington.

A remote sensing Imagery dataset
Paper: Dynamic Visualization of Transient Data Streams

- Adaptive visualization using stratification
  - Vector dimension reduction
  - Vector sampling

Figure 3: An adaptive ingest scheme for stream visualization.

Paper: Dynamic Visualization of Transient Data Streams

- Vector dimension reduction

Figure 4: a) A document vector with 200 terms. b) Result of the first wavelet decomposition with 100 terms. c) Result of the second decomposition with 50 terms.
Paper: Dynamic Visualization of Transient Data Streams

Vector dimension reduction

Figure 4: a) A document vector with 200 terms. b) Result of the first wavelet decomposition with 100 terms. c) Result of the second decomposition with 50 terms.

Figure 5: Scatterplots generated by MDS using document vectors with sizes equal to a) 200, b) 100, and c) 50 terms.
Paper: Dynamic Visualization of Transient Data Streams

Vector sampling

Figure 6: Scatterplots generated by MDS using a) 3298, b) 1649, and c) 824 document vectors.

Combined Effect

Table 1: Execution times measured in wall clock seconds.

<table>
<thead>
<tr>
<th>Document Size</th>
<th>Vector Dimension</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>300</td>
</tr>
<tr>
<td>All (2268)</td>
<td>34.59s</td>
</tr>
<tr>
<td>1/2 (1649)</td>
<td>14.83s</td>
</tr>
<tr>
<td>1/4 (824)</td>
<td>8.83s</td>
</tr>
</tbody>
</table>
Paper: Dynamic Visualization of Transient Data Streams

- Robust Eigenvectors

Figure 12: Scatterplot a) (top right) is generated from the demo imagery (top left). Scatterplots b) to d) (bottom left) are generated from the corresponding cropped areas. Scatterplots a) to g) are generated by extracting the scatter points from a) that are found in the corresponding cropping windows which generate scatterplots b) to d).

Paper: Dynamic Visualization of Transient Data Streams

- Sliding windows

Figure 13: An illustration of our multiple sliding window design in visualizing data streams.
Paper: Dynamic Visualization of Transient Data Streams

Combined solution
1. When influx rate < processing rate, reprocess entire dataset using MDS
2. When influx rate > processing rate, halt the MDS process
3. Using sliding window to update the display, repeat step 3 for a pre-defined number of updates
4. Use stratification approach for a quick overview, and use it to evaluate the current display
5. If error threshold is reached, go to step 1, otherwise go to step 3.

Hardware-Based Techniques

- Interactive Information Visualization of a Million Items (Fekete & Plaisant, Infovis02)
  - Using stencil buffer to count overlaps in scatterplot display
  - Using texture indices to avoid color calculation
  - Making use of an accelerated graphics card, such as stereovision to replace line boundaries

* Many useful references in this paper!
Incremental dynamic Queries

  - Using auxiliary data structures
  - Different response time for different operations
  - Catching

**Incremental dynamic Queries**

[Image of a chart showing data visualization]
Incremental dynamic Queries

- Query preview
- Active set
- Maximum hit set

Prefetching

- Idea: make use of system idle time
Prefetching

- Idea: make use of system idle time

The following slides are from Punit's presentation. Used with permission of Matt.

Overview

Integration between visualization and database tools:
1. Cache the results of the database requests in main memory and make operations efficient
2. Improve performance by prefetching highly probable next used data
Background on Prefetching

- **Pure prefetching:**
  - There is sufficient time between user requests
    - Amount of data prefetched is limited only by cache size

- **Non-pure prefetching:**
  - Prefetch requests are often interrupted by user requests
    - Less data being prefetched
Prefetching

- Properties
  - Depends on type of caching used
  - Speculative (no specific hints)
    - navigation remains local
    - both user and the dataset influence exploration
  - Adaptive (strategy changes over time)
    - once more knowledge becomes available
  - Non-pure (interruptible prefetching)
    - leave buffer in consistent state

Approach to Prefetching

Characteristics of the interactive and visual environments :-
- No wrong queries because of the GUI interface
- Delays between user operations
- User queries are contiguous rather than ad-hoc since GUI is used
- Users are more predictable when they explore the data
Approach to Build Prefetcher

Three level adaptive strategy for prefetching

Prefetching Strategies

- No information
  - eg., Random
- Information about data and user
  - eg., Focus
- Information about user
  - eg., direction, Mean, exponential weight average

Prefetching strategies

Random Strategy :-

Working Principle -

Application -

- When predictor cannot extract prefetching hints or provides hints with low confidence measure
- When user is looking at unknown dataset and is trying to get rough idea about the clusters in data
Prefetching strategies

Direction Strategy :-
Working Principle -

Application -
- The predictor is able to predict direction of next user operation depending on user’s past explorations
- By intuition, user continues to use same manipulation tool for a while before changing to other

Prefetching strategies

Focus Strategy :-
Working Principle -

Application -
- Predictor is able to predict direction of next user operation depending on user’s past explorations and it knows patterns of data
Prefetching strategies

Mean Strategy :-
Working Principle -

Application -
✓ Predictor has sufficient knowledge about the user

Exponential Weight Average Strategy :-
Working Principle -

Application -
✓ Predictor has sufficient knowledge about the user
Dynamic Labeling

- Key idea: remove sth from the display while interaction