Topic Notes

Visualizing Big Data (Many Cases & Dimensions)

CS 7450 - Information Visualization

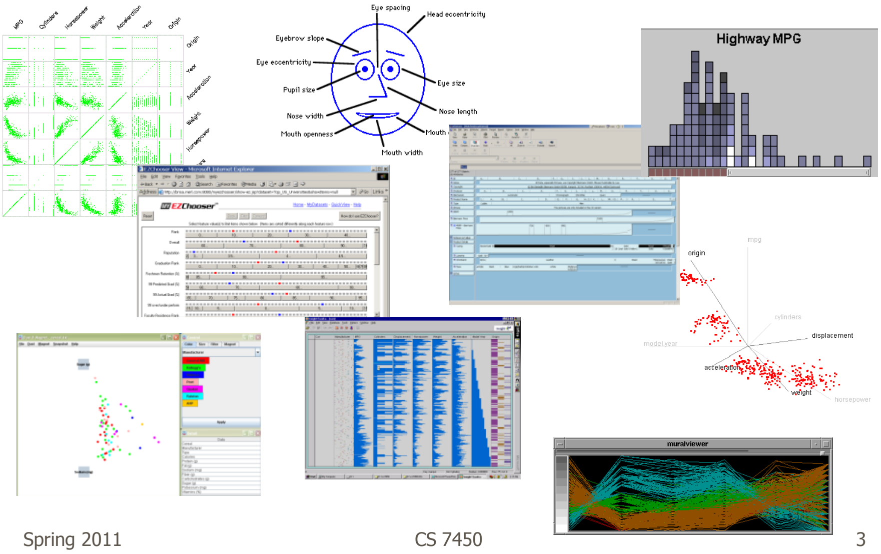
April 12, 2011

John Stasko

Previously

- We looked at a number of techniques for projecting >2 variables down onto the 2D plane
  - Parallel coordinates
  - Scatterplot matrix
  - Table lens
  - etc.

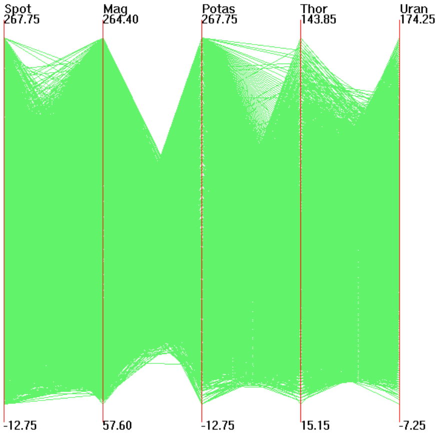
Varieties of Techniques



Potential Limitations

- What happens when you have lots and lots of data cases?

Parallel Coordinates

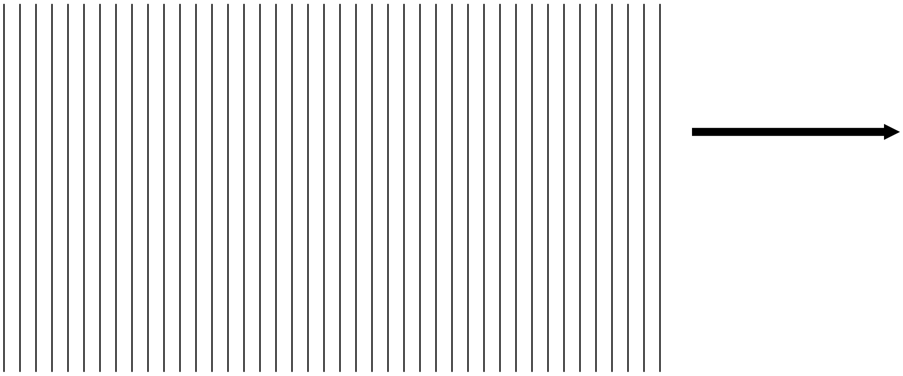


Out5d dataset (5 dimensions, 16384 data items)

(courtesy of J. Yang)

Potential Limitations

- Or, you may have many, many variables
  - Hundreds or even thousands



Strategies

- How are we going to deal with such big datasets with so many variables per case?
- Ideas?

General Notion

- Data that is similar in most dimensions ought to be drawn together
  - Cluster at high dimensions
- Need to project the data down into the plane and give it some ultra-simplified representation

- Or perhaps only look at certain aspects of the data at any one time

Mathematical Assistance 1

- There exist many techniques for clustering high-dimensional data with respect to all those dimensions
  - Affinity propagation
  - k-means
  - Expectation maximization
  - Hierarchical clustering

Mathematical Assistance 2

- There exist many techniques for projecting n-dimensions down to 2-D (dimensionality reduction)
  - Multi-dimensional scaling (MDS)
  - Principal component analysis
  - Linear discriminant analysis
  - Factor analysis

Comput Sci & Eng courses
Visual Analytics, Prof. Lebanon

Data mining
Knowledge discovery

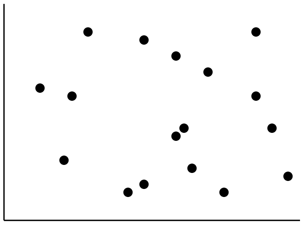
Other Techniques

- Other techniques exist to reduce data
  - Sampling – We only include every so many data cases or variables
  - Aggregation – We combine many data cases or variables

Our Focus

- Visual techniques

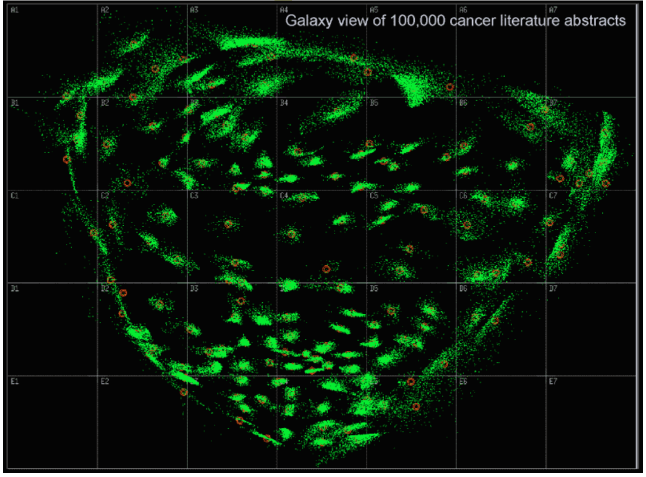
- Many are simply graphic transformations from N-D down to 2-D



Example

- Big document collection
- Accumulate all different words used throughout
- Each word becomes a dimension
- Value of that data case (document) in a dimension is the number of times the word appears in that document
- (May be thousands of dimensions)

PNNL's SPIRE



Each dot is a document

Similarity provokes nearby positioning

Will see more later in term on Text day

Wise et al
InfoVis '95

Pluses & Minuses

- Can have as many cases as there are pixels and unlimited number of dimensions
- Shows similarity of data cases

- Only a dot for each case
- Doesn't say much about dimensions or cases

Use?

- What kinds of questions/tasks would you want such a technique to address?
  - Clusters of similar data cases
  - Useless dimensions
  - Dimensions similar to each other
  - Outlier data cases
  - …
- Think back to our "cognitive tasks" discussion

Today

- We'll examine a number of other visual techniques intended for larger, high-dimensional data sets

Can We Make a Taxonomy?

- D. Keim proposes a taxonomy of techniques
  - Standard 2D/3D display

    Bar charts, scatterplots
  - Geometrically transformed display

    Parallel coordinates
  - Iconic display

    Needle icons, Chernoff faces
  - Dense pixel display

    What we're about to see…
  - Stacked display

    Treemaps, dimensional stacking

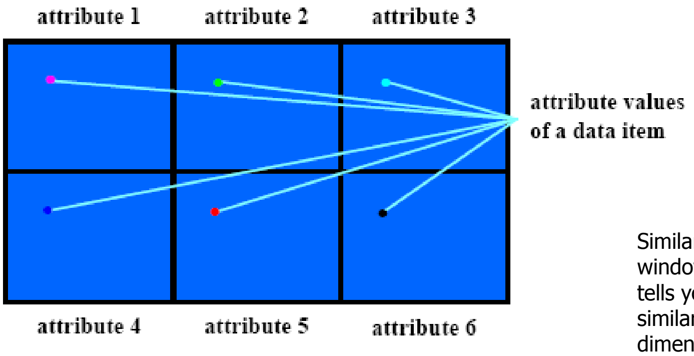
TVCG '02

Dense Pixel Display

- Represent data case or a variable as a pixel
- Million or more per display
- Seems to rely on use of color
- Can pack lots in

- Challenge: What's the layout?

One Representation



Each variable is in a window
Data cases in grid in each window

Similarity of window views tells you about similarity of dimensions

Uses color scale

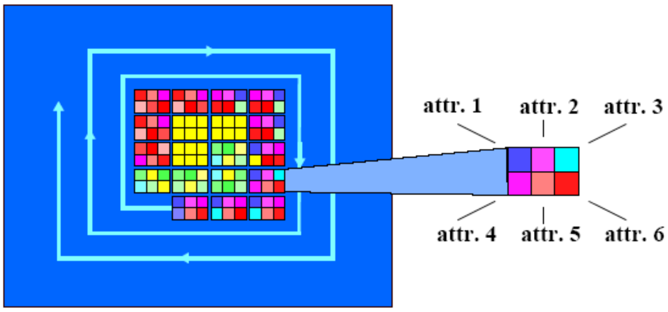
🟨🟩🟩🟦🟦🟪🟥🟥⬛

Alternative

- Grouping arrangement
- Doesn't use multiple windows
- Each data case has its own small rectangular icon
- Plot out variables for data point in that icon using a grid layout

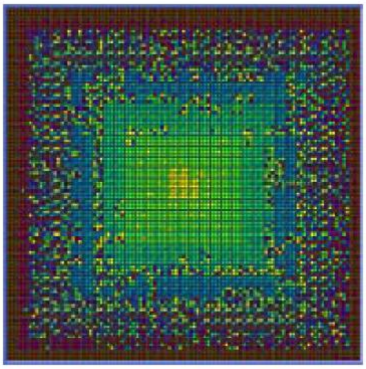
Another View

schematic representation of 6-dim. data



Levkowitz
Vis '91

Example Large View



DB Applications

- Database of data items, each of n dimensions
- Issue a query that specifies a target value of the dimensions
- Often get back no exact matches
- Want to find near matches

Keim & Kriegel
IEEE CG&A '94

Relevance Factor

- How close an item is to the query
  - Data items have some value that can be numerically quantified
  - Each dimension is some distance away from query item
  - Sum these up for total distance
  - Relevance is inverse of distance

Example

- 5 dimensions, integers 0->255
- Query: 6, 210, 73, 45, 92
- Data item: 8, 200, 73, 50, 91
- Distance: 2 + 10 + 0 + 5 + 1 = 18
- Relevance: 1275 - 18 = 1267

Issues

- What if dimensions are real numbers or text strings?
- What if they're the same type, but of different orders of magnitude?

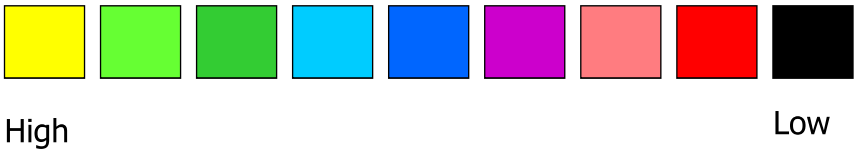
- Have to define some kind of distance, then a weight function to multiply by

Technique

- Calculate relevance of all data points
- Sort items based on relevance

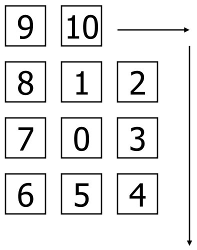
- Use spiral technique to order the values – Emanate out from center
- Color items based on relevance

Relevance Colors



Empirically established

Technique



9	10	
8	1	2
7	0	3
6	5	4

Spiral Method

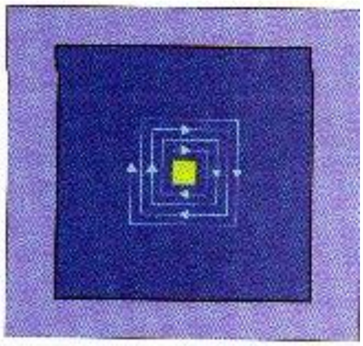


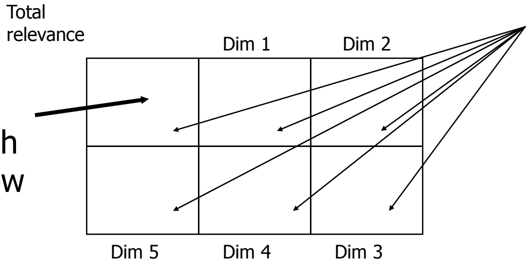
FIGURE 1

Spiral-shaped arrangement of one dimension.

Highest relevance value in center, decreasing values grow outward

Display Methodology

Example: five-dimensional data



Total relevance

Dim 1

Dim 2

Dim 5

Dim 4

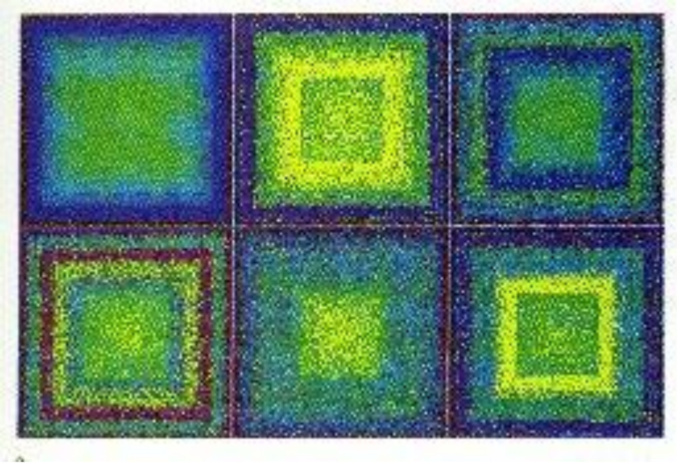
Dim 3

Spiral in each window

Same item appears in same place in each window

Items ordered by total relevance

Example Display



Alternative

- Grouping arrangement
- Doesn’t use multiple windows
- Create all relevance dimensional depictions for an item and group them
- Spiral out the different data items’ depictions

Grouping Arrangement

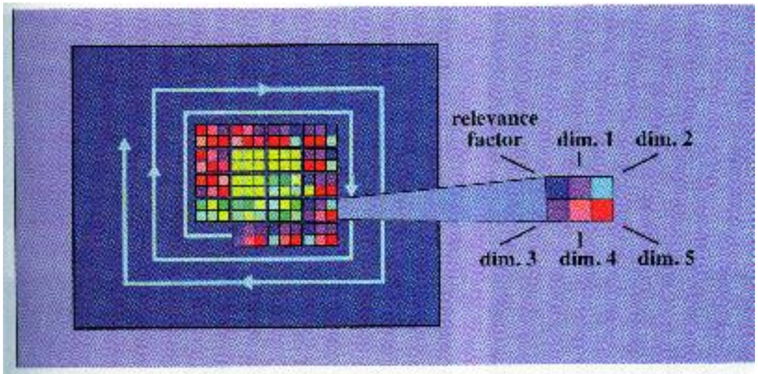


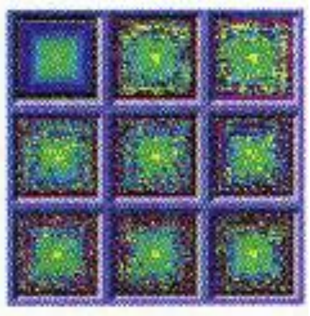
FIGURE 4

Grouping for five-dimensional data.

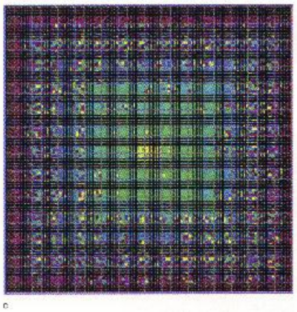
Example Display

8 dimensions

1000 items



Multi-window



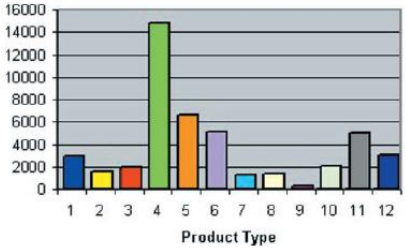
Grouping

Related Idea

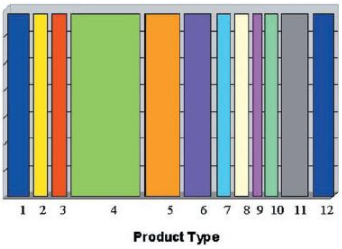
- Pixel Bar Chart
- Overload typical bar chart with more information about individual elements

Keim et al
Information Visualization ‘02

Idea 1



Height encodes quantity

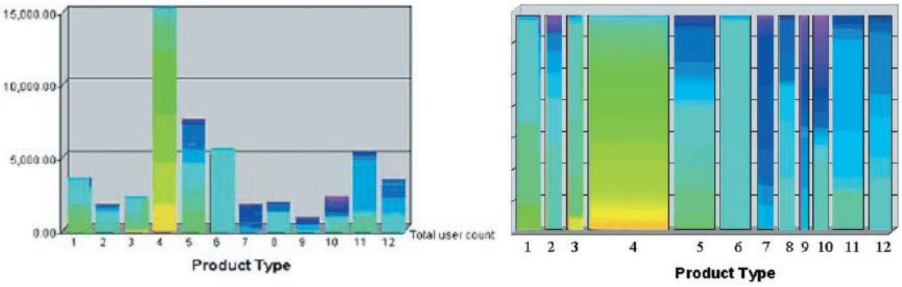


Width encodes quantity

Idea 2

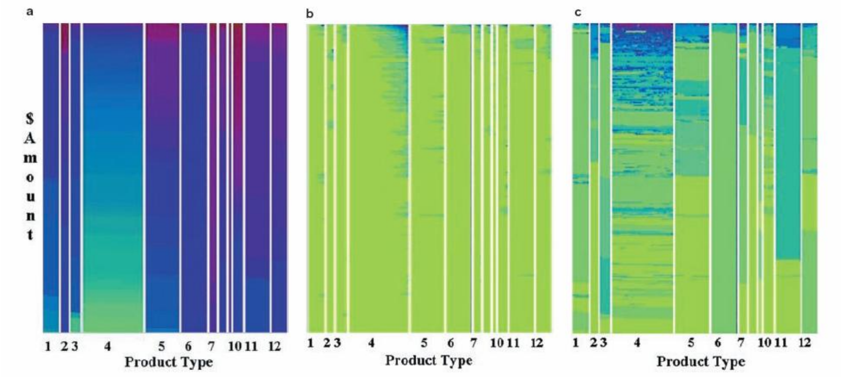
- Make each pixel within a bar correspond to a data point in that group represented by the bar
  - Can do millions that way
- Color the pixel to represent the value of one of the data point’s variables

Idea 3



Each pixel is a customer
Color encodes amount spent by that person
High-bright, Low-dark
Ordered by that color attribute too
Right one shows more customers

Idea 4



Product type is x-axis divider
Customers ordered by
y-axis: dollar amount
x-axis: number of visits
Color is (a) dollar amount spent, (b) number of visits, (c) sales quantity

Idea 5

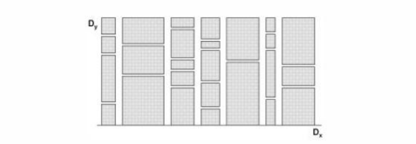


Figure 7 Dividing attributes on x- and y-axis (e.g., D_x=*Product Type*, D_y=*Region*).

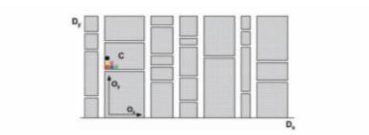


Figure 8 Ordering attributes on x- and y-axis (e.g., O_x=*Dollar Amount*, O_y=*Quantity*).

Can divide on two different attributes on x and y

Order items on both x and y

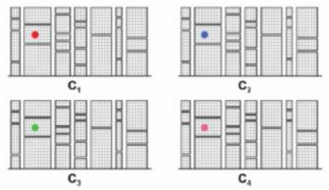


Figure 9 Multiple coloring attributes (e.g., C_1=*dollar amount*, C_2=*no. of visits*, C_3=*quantity*, C_4=*region*).

Color maps to some attribute
(Same item always at same x,y position)

Idea 6

Mapping specified by 5 tuple <D_x, D_y, O_x, O_y, C>

D_x – Attribute partitions x axis
D_y – Attribute partitions y axis
O_x – Attribute specifies x ordering
O_y – Attribute specifies y ordering
C – Attribute specifies color mapping

Example Application

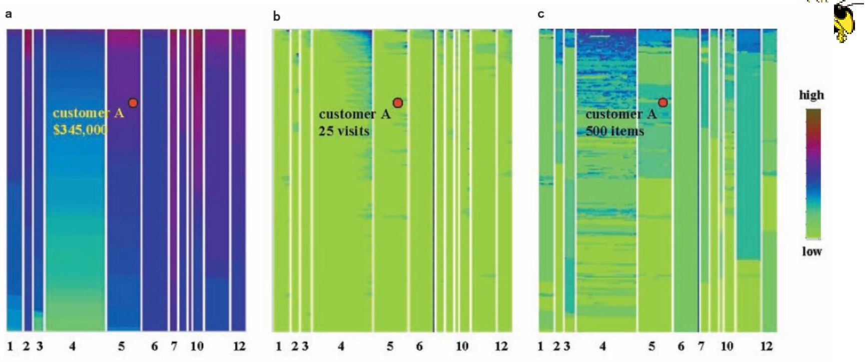


Figure 13 Multi-pixel bar chart for mining 405,000 sales transaction records. (D_x = *Product Type*, D_y =⊥, O_x = *no. of visits*, O_y = *dollar amount*, *C*). (a) Color: dollar amount. (b) Color: no. of visits. (c) Color: quantity.

1. Product type 7 and product type 10 have the top dollar amount customers (dark colors of bar 7 and 10 in Figure 13a)
2. The dollar amount spent and the number of visits are clearly correlated, especially for product type 4 (linear increase of dark colors at the top of bar 4 in Figure 13b)
3. Product types 4 and 11 have the highest quantities sold (dark colors of bar 4 and 11 in Figure 13c)
4. Clicking on pixel A shows details for that customer

Thoughts?

- Do you think that would be a helpful exploratory tool?

High Dimensions

- Those techniques could show lots of data, but not so many dimensions at once
  - Have to pick and choose

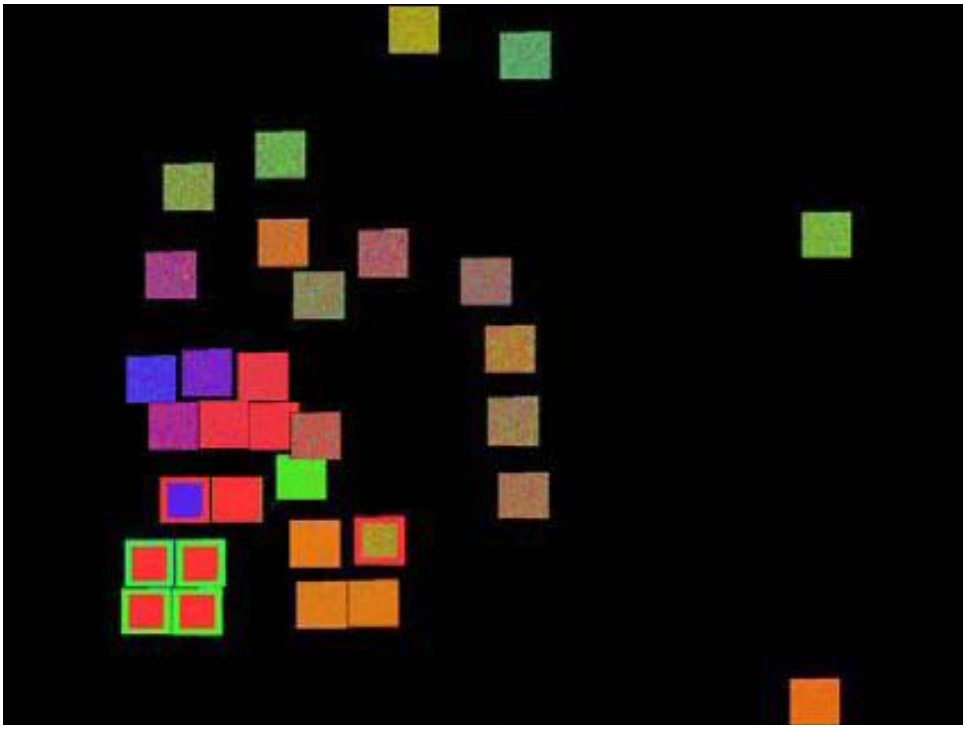
Another Idea

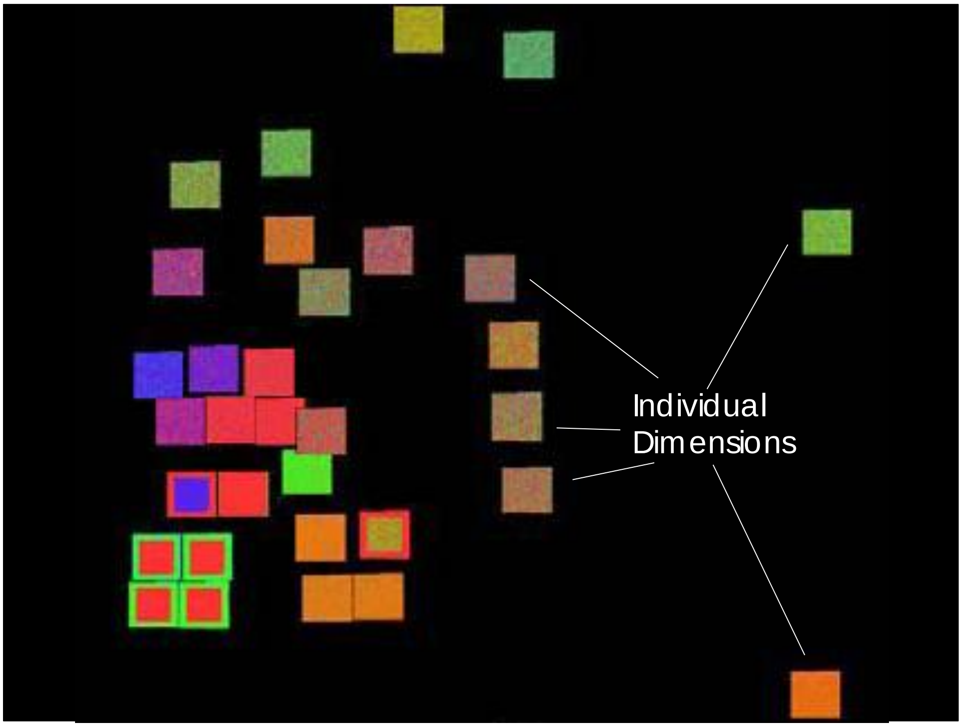
- Use the dense pixel display for showing data and dimensions, but then project into 2D plane to encode more information
- VaR – Value and relation display

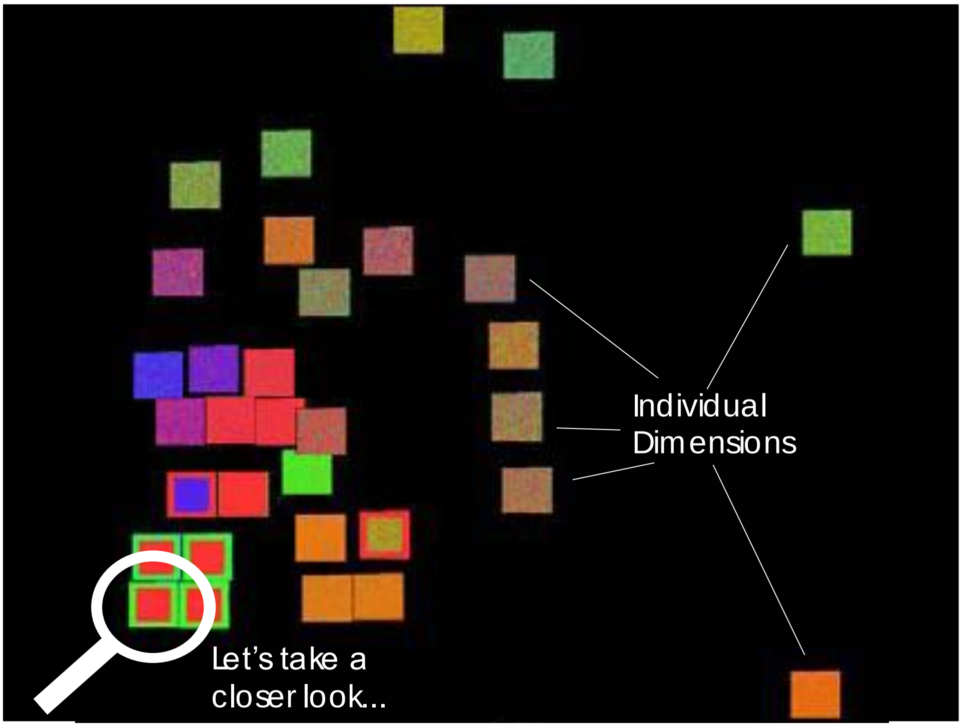
Yang et al
InfoVis '04

Algorithm

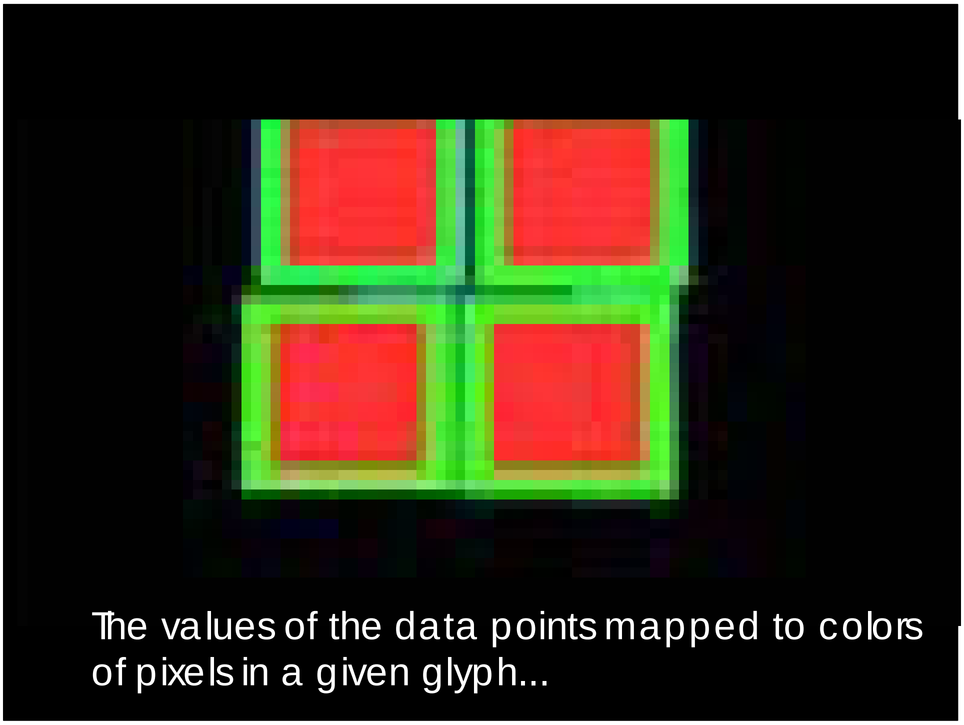
- Find a correlation function for comparing dimensions
- Calculate distances between dimensions (similarities)
- Make each dimension into a dense pixel glyph
- Assign position for each glyph in 2D plane using multi-dimensional scaling



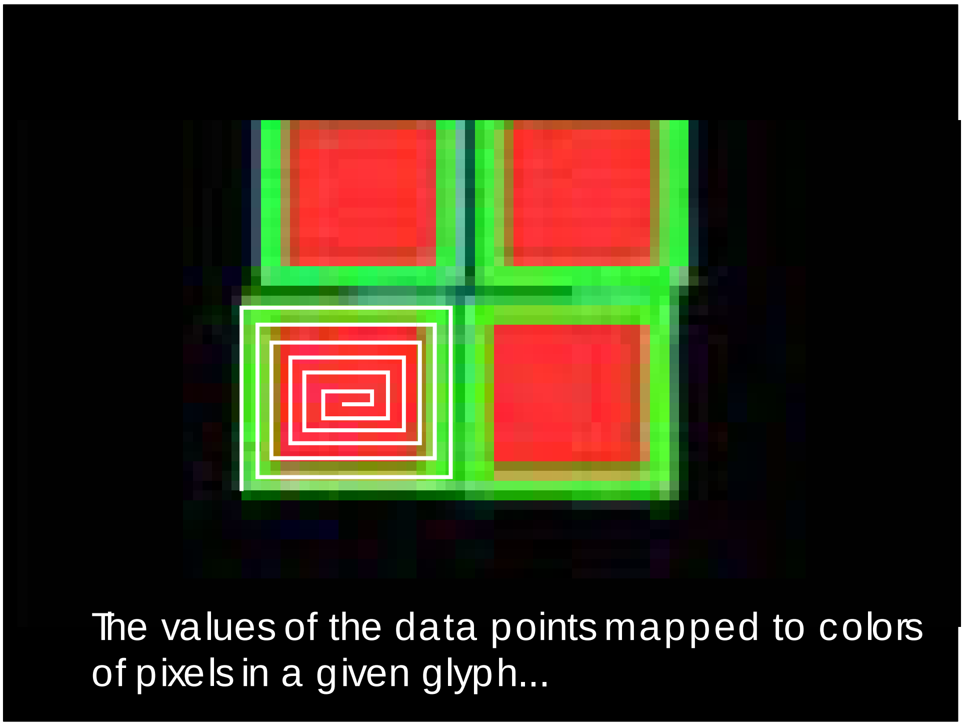

Individual
Dimensions



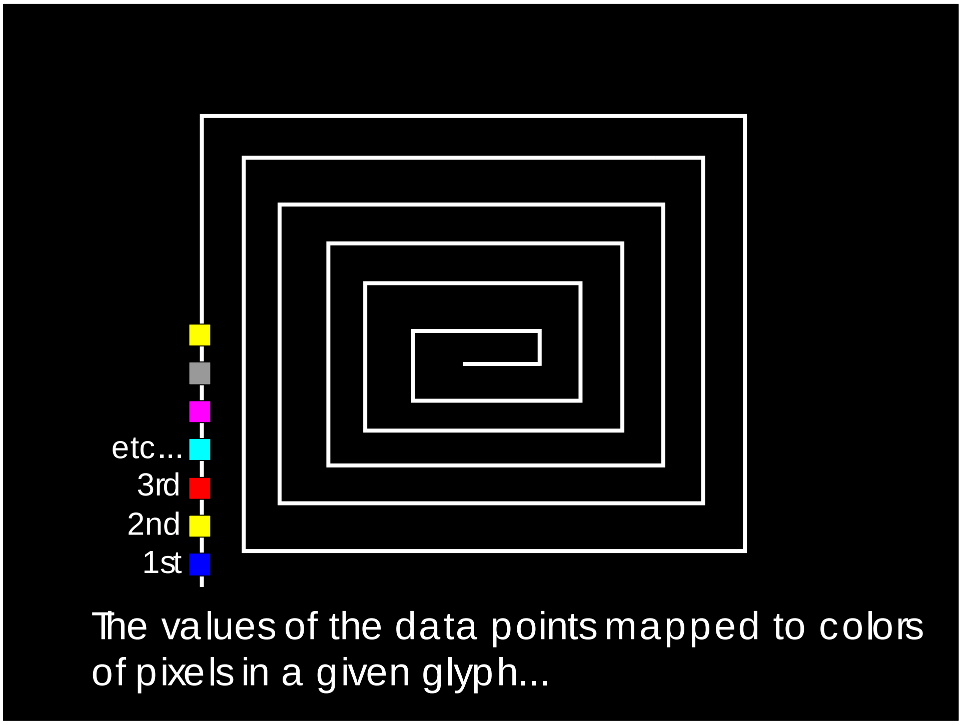
Let's take a
closer look...



The values of the data points mapped to colors of pixels in a given glyph...



The values of the data points mapped to colors of pixels in a given glyph...



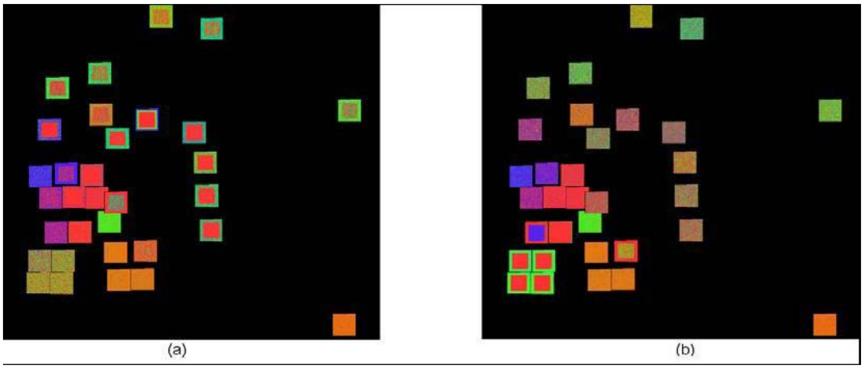
The values of the data points mapped to colors of pixels in a given glyph...

Questions

- What order are the data cases in each dimension-glyph?
  - Maybe there is a predefined order
  - Choose one dimension as "important" then order data cases by their values in that dimension
    - "Important" one may be the one in which many cases are similar

Reordering Data

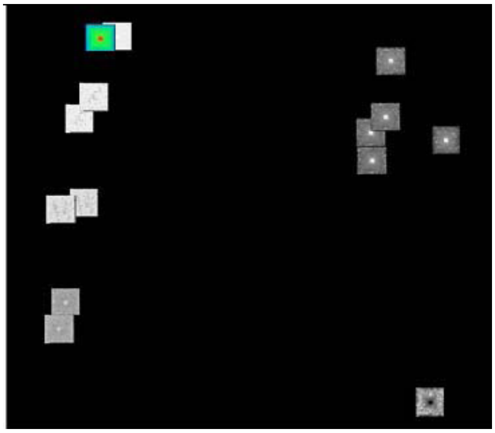
- Two different orderings of cases shown below (a- dimension near top is prototype, b- dimension near bottom is prototype)



Comparing dimensions

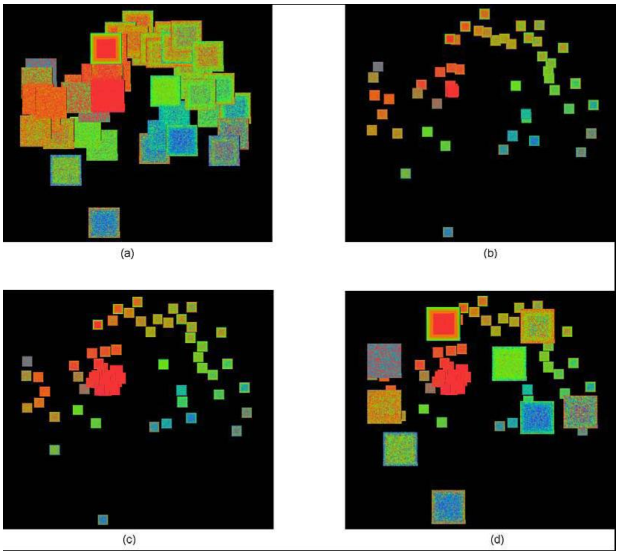
One dimension chosen as focus

Others shaded for how similar they are
- dark – big difference
- light – small difference



Interaction

a – lots overlaps
b – shrink size
c – jitter position
d – enlarge some



Contributions

- Highly scalable way to view dimensional relationships
- Computationally efficient
- Uses MDS for dimensions, not just data cases

Limitations

- Those glyph overlaps are a problem
- Similar dimensions are positioned near each other with lots of overlap

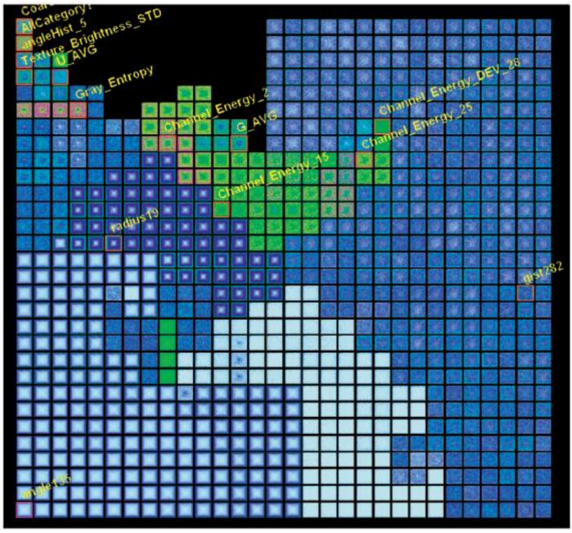
Follow-on Work

- Use alternate positioning strategies other than MDS
- Use Jigsaw map idea (Wattenberg, InfoVis '05) to lay out the dimensions into a grid
  - Removes overlap
  - Limits number that can be plotted

Yang et al
TVCG '07

New Layout

Plot the glyphs into the grid positions



HCE

- Hierarchical Clustering Explorer
- Implements "rank by feature" framework
- Help guide user to choose 1D distributions and 2D scatterplots from various dimensions of a data set
- Combine statistical analysis with user-directed exploration

Seo & Shneiderman
Information Visualization '05

Idea

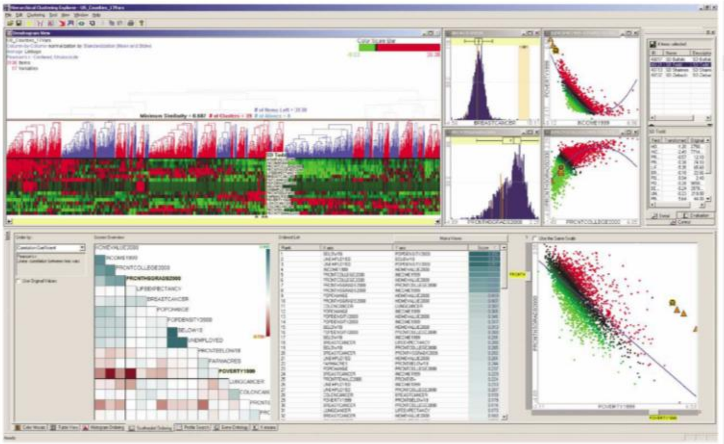
- Choose a feature detection criterion to rank 1D and 2D projections of a data set
- Use person's perceptual abilities to pick out interesting items from view

HCE UI

Some chosen distributions and scatterplots

Cases in columns, variables in rows

Group similar cases

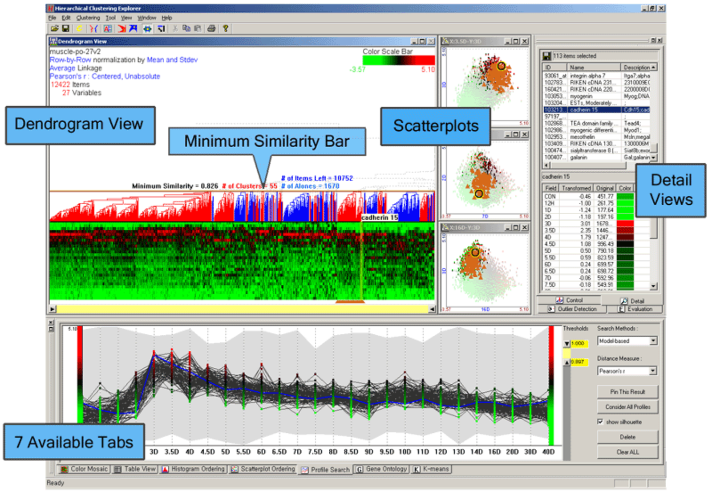


Seven tabs at bottom to choose from

Operation

- When you choose the histogram ordering or scatterplot ordering tabs at the bottom left, these give results based on various statistical measures
- You can then choose some of them to visualize

Demo



HW 8 Return

- Solution

Upcoming

- Evaluation
  - Reading:

    PLaisant
- Casual InfoVis
  - Reading:

    Pousman et al