4. Multidimensional Information Visualization II

Concepts for visualizing univariate to hypervariate data

Dr. Thorsten Büring, 15. November 2007, Vorlesung Wintersemester 2007/08

Ludwig
Maximilians-
Universität
München
LMU

Outline

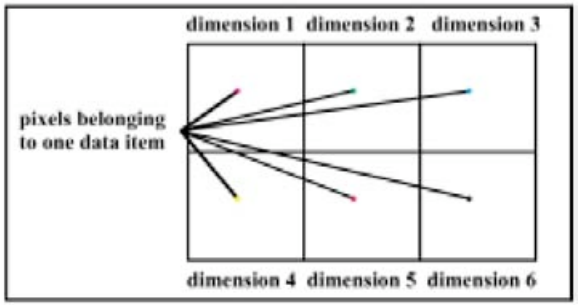
- Reference model and data terminology
- Visualizing data with < 4 variables
- Visualizing multivariable data
  - Geometric transformation
  - Glyphs
  - Pixel-based
  - Dimensional Stacking
  - Downscaling of dimensions
- Case studies: support for exploring multidimensional data
  - Rank-by-feature
  - Value & relation display
  - Dust & magnet
- Clutter reduction techniques

Topics of previous lecture: Multidimensional Information Visualization I

Pixel-Based Techniques

- Prof. Daniel Keim
- Basic idea
  - Map each data value to a single colored pixel
  - Present the data values belonging to one variable in separate windows (unlike color icons)
- Maximizes the amount of data that can be displayed
- Users can identify correlations, functional dependencies, and clusters between variables by relating corresponding regions in the different subwindows
- Arrangement of pixels must be the same for each window
- Query-independent
  - Space-filling curves
  - Recursive pattern
- Query-dependent
  - Spiral technique
  - Circle segment

Pixel-based techniques



Keim 2000

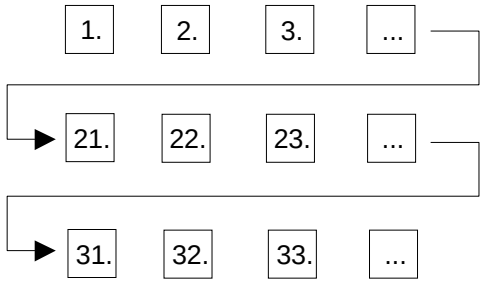
Pixel-Based Techniques

- Need for intuitive color-coding, which conveys distances in data values
- Problem
  - No implicit ordering of colors
  - Why not use gray scales then?
- Strategy: value ranges are mapped to a fixed color sequence
  - Full color (hue) scale,
  - Constant saturation
  - Monotonically decreasing brightness
- Keim 2000: color range from yellow over green, blue and red to almost black found most intuitive
- Still, users may also choose their own color-coding

Keim 2002

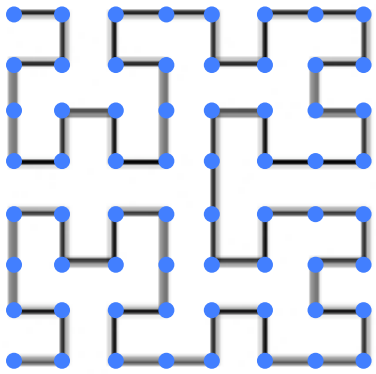
Space-filling curves

- Arrangement of pixels in each subwindow
- Key factor for expressiveness and effectiveness of the visualization
- **Optimization Goal (OG) 1**: arrangement of pixels in the subwindows should preserve the 1D ordering into the 2D plane as best as possible
- Case: naturally ordering of data cases based on one variable (e.g. time-based data)
- Naive approach:
  - Simple left-right or top-down arrangement
  - Provides usually no useful results on a pixel-level
  - No clustering of closely related data items
- Space-filling curves provide maximum of locality preservation

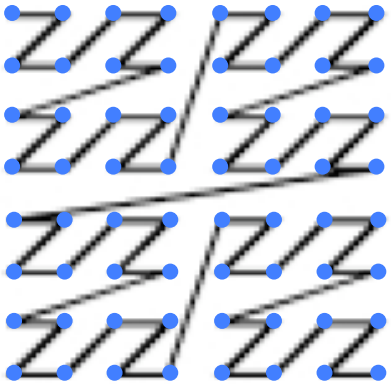


Space-filling curves

- Space filling curves provide a continuous curve, which passes through every point (in our case pixel) of a regular spatial region
- Based on such curves: data items close in 1D distribution are likely to be close in 2D



Paneo-Hilbert Curve

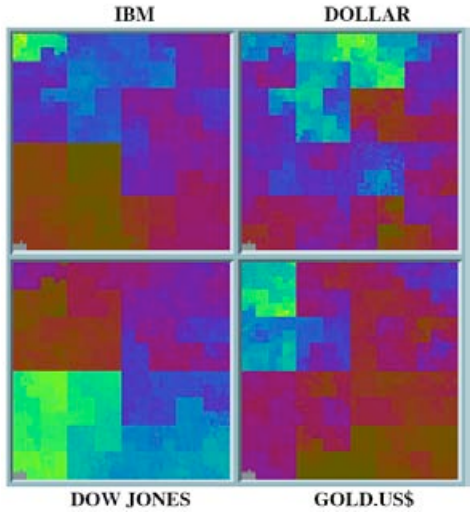


Morton-Curve

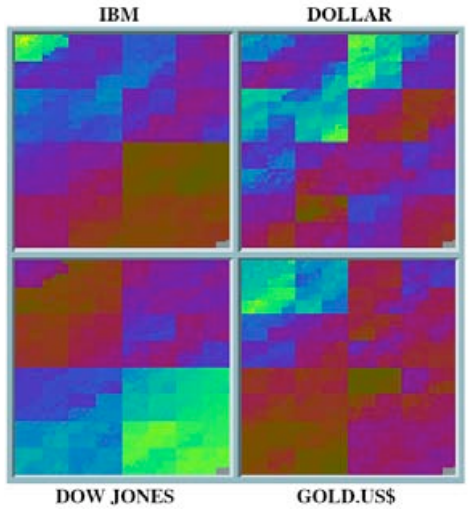
Space-filling curves

- Keim 1996
- Visualization shows 16,350 data values (January 1987- March 1993, 9 data values per day) of a stock exchange database based on Paeno-Hilbert and Morton curves
- Light colors map to high data values and dark colors to low data values
- P-H: Good clustering but difficult to follow and thus
- Morton: simpler regularity, but still not sufficiently intuitive; occasionally larger gaps between adjacent points
- Arrangement is not related to the semantics of the data
- Difficult to read and interpret the visualization

Paneo-Hilbert Curve



Morton-Curve



Space-filling curves

- Example of recursive algorithm for drawing the Morton curve (Keim 1996)

```
void Morton(int x, int y, int level)
{if (level>0)
   {Morton(x           ,y           ,level-1);
    Morton(x+pow(2,i) ,y           ,level-1);
    Morton(x           ,y+pow(2,i) ,level-1);
    Morton(x+pow(2,i) ,y+pow(2,i) ,level-1);
   }
 else {   // level==0
    SetPixel(x  ,y  ,color);
    SetPixel(x++,y  ,color);
    SetPixel(x  ,y++,color);
    SetPixel(x++,y++,color);
 }
}
called by  Morton(startX,startY,max_level);
```

Recursive Pattern

- Keim et al. 1995
- Retain clustering quality, but provide more intuitive / semantic arrangement via user input
- Recursive pattern visualization
  - Based on simple back and forth arrangement
  - Lower-level patterns used as building blocks for higher level patterns
  - User defines the level patterns
  - The more levels the more expressive the visualization becomes
  - E.g. for time-series: LP1 one day, LP2 one week, LP3 one month, LP4 one year ...

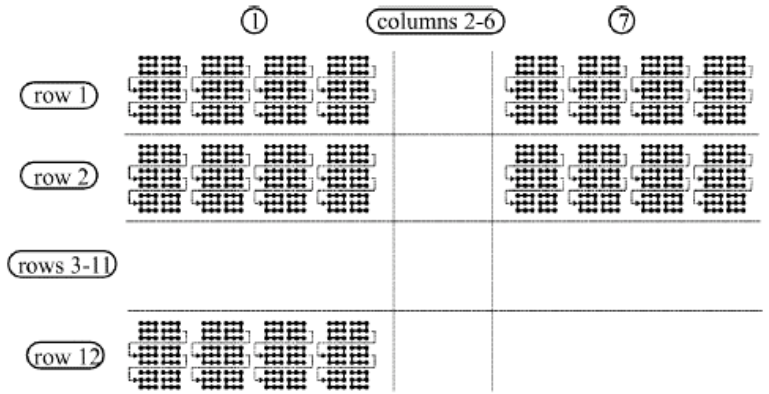


Fig. 9. Schematic representation of a highly structured arrangement $[(w_1, h_1) = (3, 3), \ (w_2, h_2) = (2, 3), \ (w_3, h_3) = (4, 1), \ (w_4, h_4) = (1, 12), \ (w_5, h_5) = (7, 1)]$. (Adapted from [39] ©IEEE.)

Keim 2000

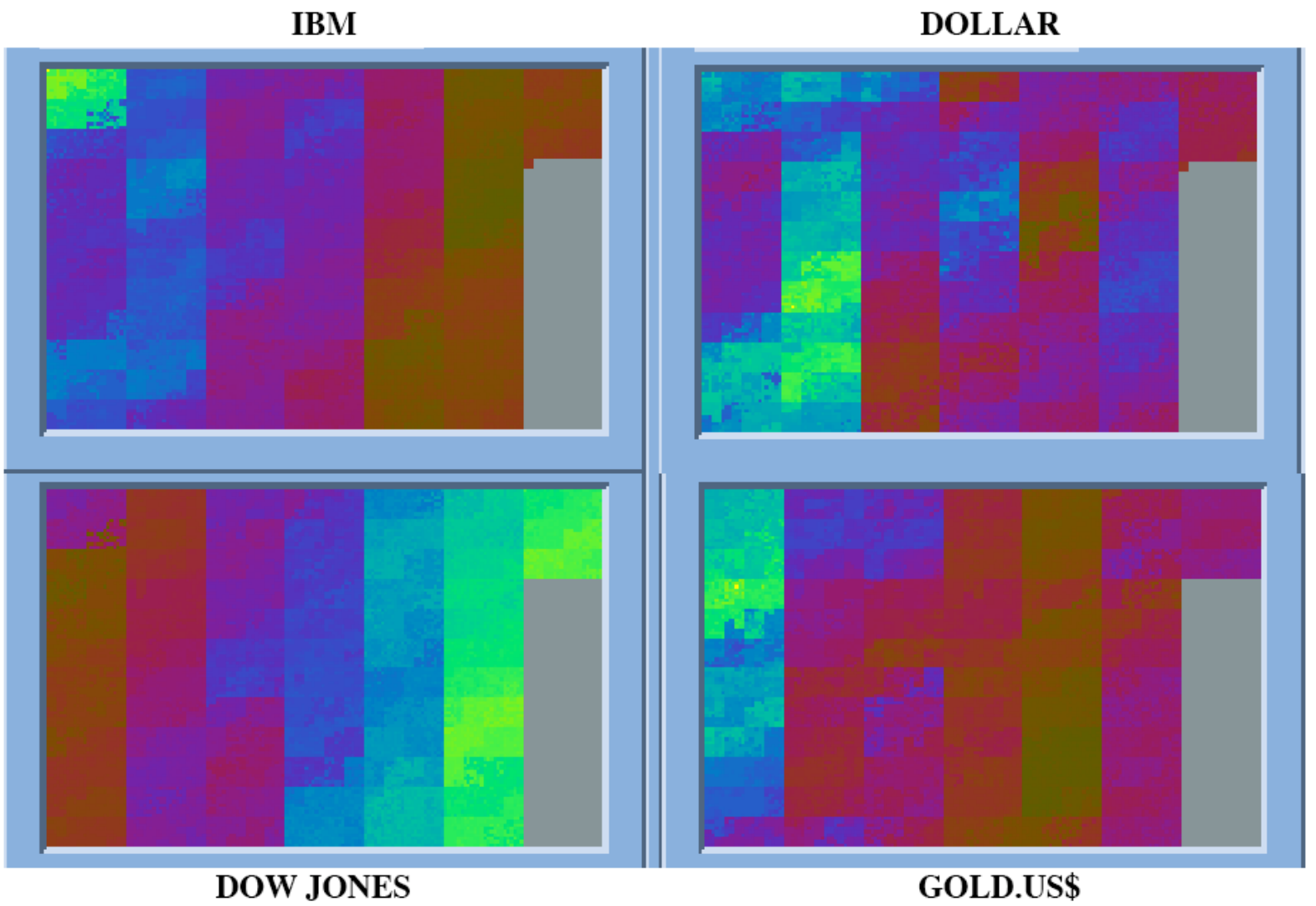


Figure 10: Highly Structured Arrangement

$[(w_1, h_1) = (3, 3), (w_2, h_2) = (2, 3), (w_3, h_3) = (4, 1), (w_4, h_4) = (1, 12), (w_5, h_5) = (7, 1)]$

Spiral vs. Generalized Spiral

- Ordering of data cases based on relevance to a given query
- Most relevant data case is placed in the center of the screen
- OG 2: for the pixel arrangement in each subwindow the distance to the center should correspond to the ordering of the data cases
- Simple spiral arrangement fulfills OG 2, but local clustering properties (OG 1) are weak, i.e. low probability that two pixels close on the screen are also close in the 1D ordered sequence of the query result set
- Generalized spiral technique: enhance the clustering qualities of the spiral technique by using screen-filling curves locally

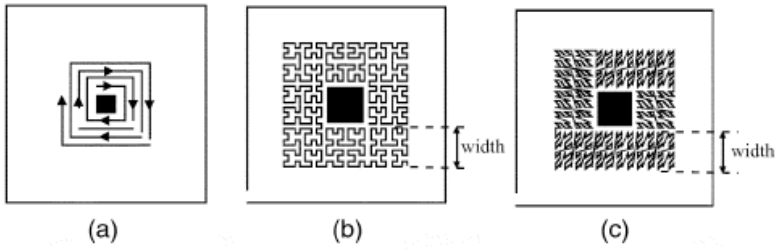
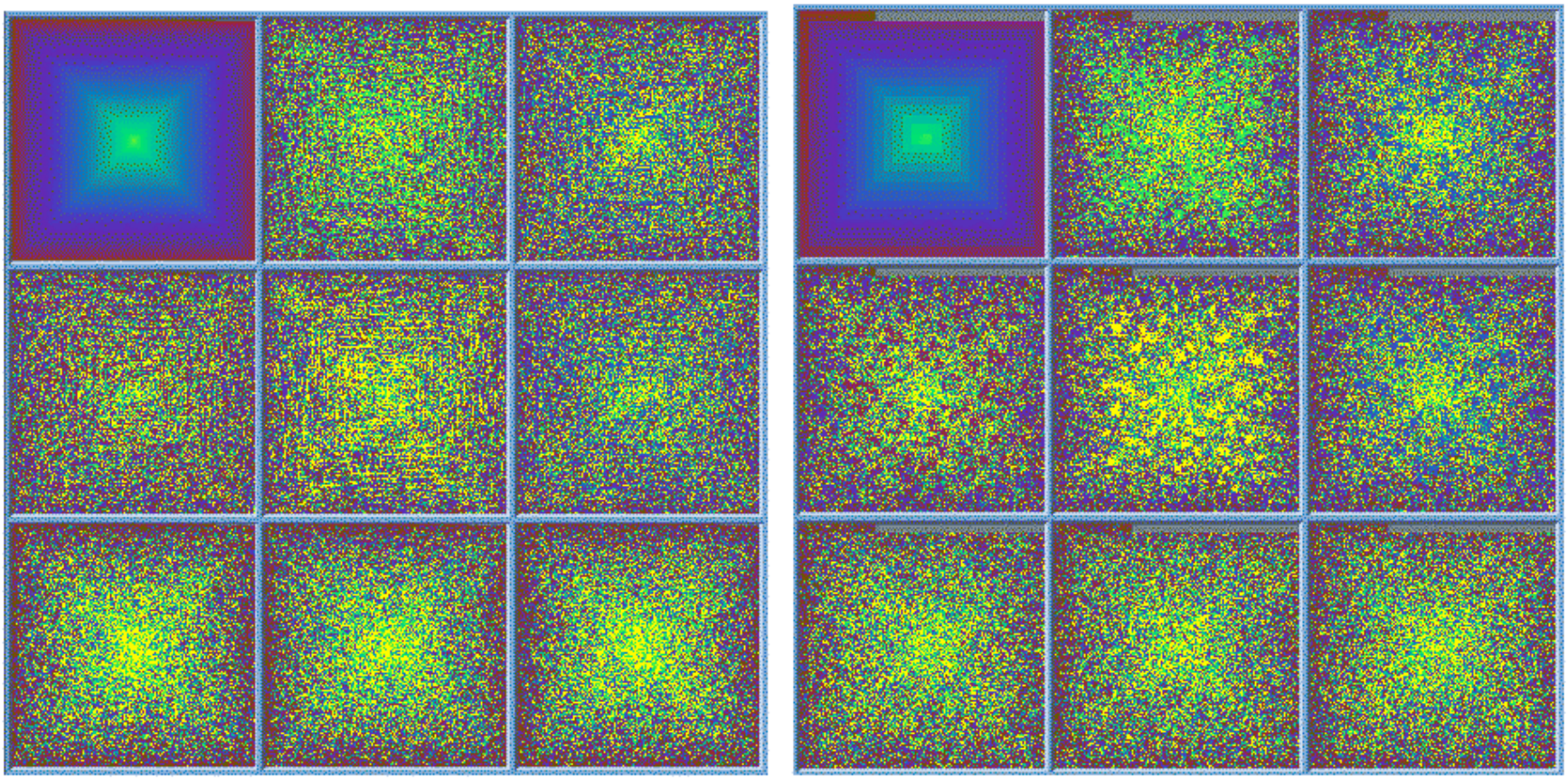


Fig. 11. Spiral and Generalized Spiral technique. (a) Spiral technique. (b) Peano-Hilbert spiral (width = 8). (c) Morton spiral (width = 8).

Keim 2000

Spiral vs. Generalized Spiral

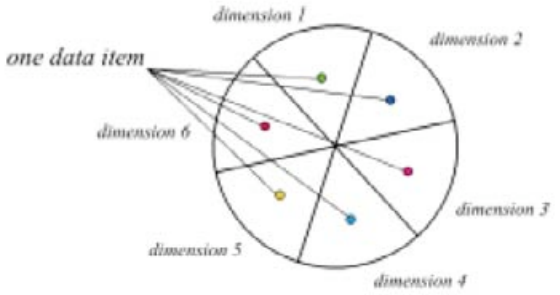
- 20,000 data case randomly defined; 4,000 data cases defined as clusters in 5D

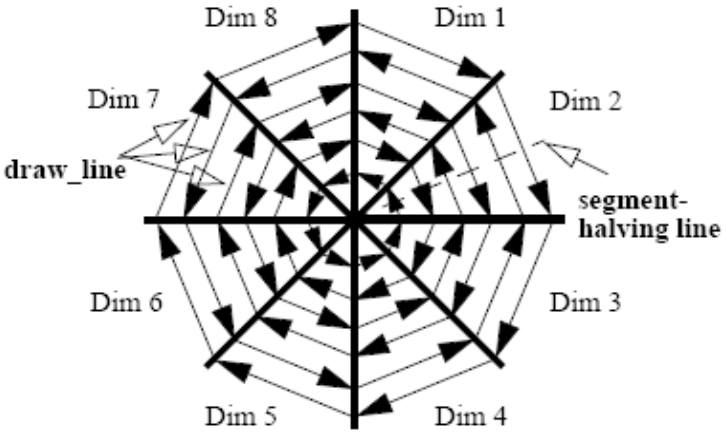


Keim1996

Circle Segment

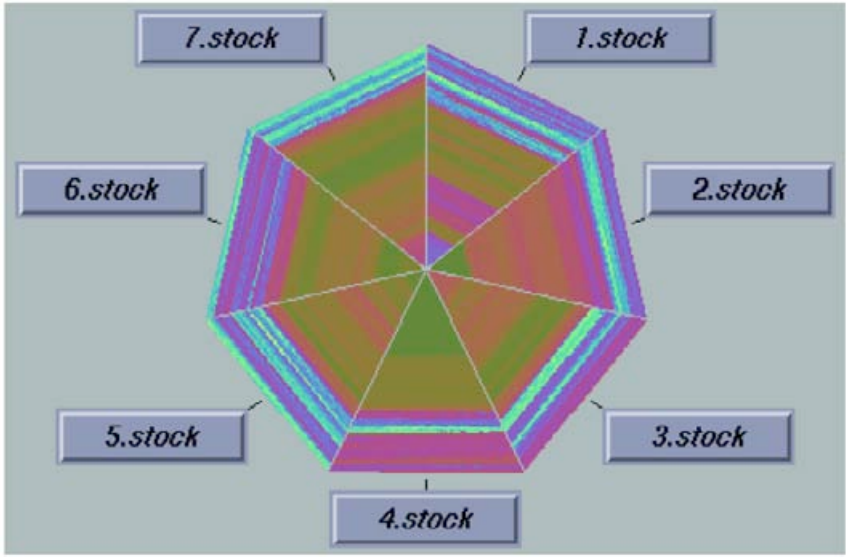
- Ankerst et al. 1996
- Shape of subwindow
- Rectangular shape of subwindows makes efficient use of the screen
- But: for data sets with many dimensions, the pixels of one data objects are rather far apart
- Makes it difficult to find patterns
- OP 3: minimize the average distance between the pixels (data values) belonging to one data case
- Circle segment
  - Each dimension corresponds to a segment of a circle
  - Values of one dimension are drawn in a back and forth manner from the center of the circle to the outside

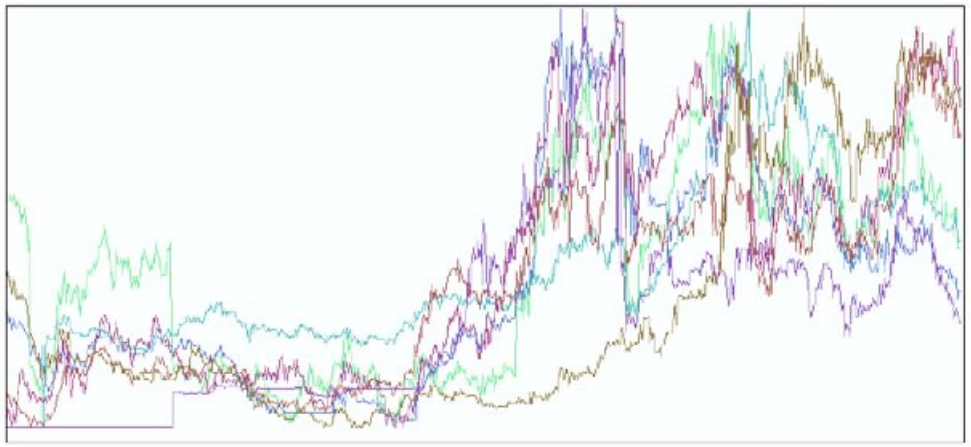




Circle Segment (CS) vs. Line Graphs

- 10 years of stock data for 7 stocks
- Line graph granularity is limited by the width of the screen
- CS: oldest data items in the middle of the circle, most recent ones are at the outside
- Easier to perceive patterns – no overlap of data





Ankerst et al. 1996

Pixel-Based Techniques

- Advantages:
  - Maximize the size of data sets which can be visualized on a single screen
  - Improved pattern detection due to non-overlap strategy
- Disadvantages
  - Rather difficult to understand and interpret
  - Mapping color to quantitative data
  - Users need to relate to different portions of the screen to perceive correlations

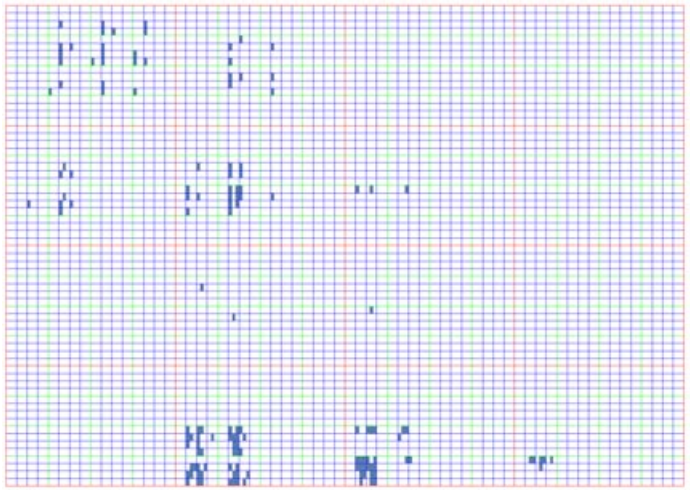
Outline

- Reference model and data terminology
- Visualizing data with < 4 variables
- Visualizing multivariable data
  - Geometric transformation
  - Glyphs
  - Pixel-based
  - Dimensional Stacking
  - Downscaling of dimensions
- Case studies: support for exploring multidimensional data
  - Rank-by-feature
  - Value & relation display
  - Dust & magnet
- Clutter reduction techniques

Topics of previous lecture: Multidimensional Information Visualization I

Dimensional Stacking

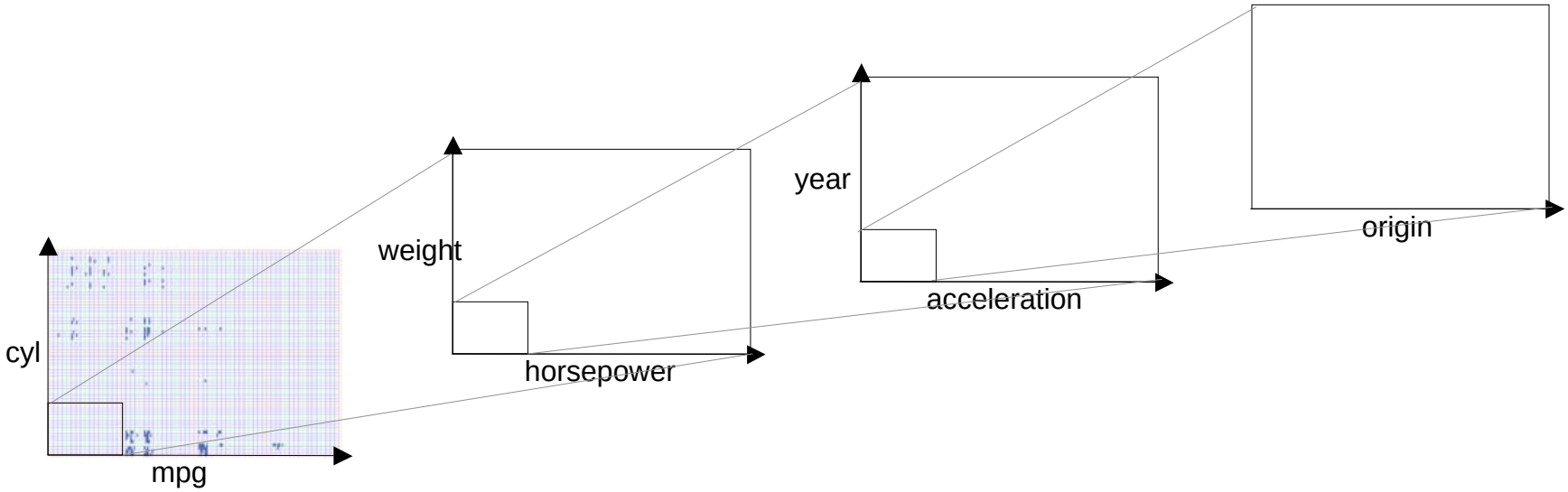
- LeBlanc et al. 1990
- Hierarchical technique like TreeMap (still to come)
- Projecting high-dimensional data by embedding dimensions within other dimensions
- Users must be able to dynamically modify the stacking order
- Most important variables should be chosen for the two out-most dimensions
- Stacking order visualized by grid lines of varying intensity
- Each data value maps to a unique position on the screen (called bucket)
- Particularly suitable for ordinal data with low cardinality
- No occlusion
- But: user easily gets lost, labeling is difficult
- Example shows again the car data set - **demo**



Xmdv-Tool

Dimensional Stacking

- How to
  - Assign number of buckets to each dimension (classing)
  - Assign orientation to each dimension (in 2D, horizontal or vertical)
  - Assign ordering of stack (in this case ordering of the input file)
- In his case the number of buckets for all dimensions except for origin is 4
- Number of buckets of inner-most dimension define the minimum size of the graphic



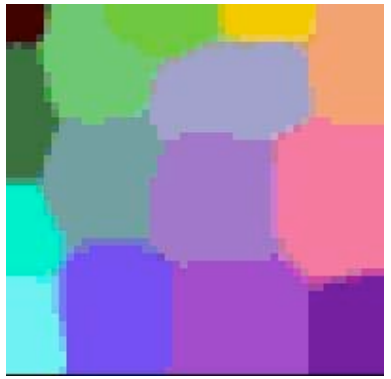
Outline

- Reference model and data terminology
- Visualizing data with < 4 variables
- Visualizing multivariable data
  - Geometric transformation
  - Glyphs
  - Pixel-based
  - Dimensional Stacking
  - Downscaling of dimensions
- Case studies: support for exploring multidimensional data
  - Rank-by-feature
  - Value & relation display
  - Dust & magnet
- Clutter reduction techniques

Topics of previous lecture: Multidimensional Information Visualization I

Downscaling of Dimensions

- Projecting n dimensions down to a lower dimensionality while retaining as much of the original information as possible
- Principal components analysis, Multidimensional scaling
  - Statistical approaches to reduce the number of dimensions by finding the data's main characteristics / patterns / similarities
  - Linear / non-linear combinations of dimensions for an axes, e.g. 3*(task-completion time)-2*(error-rate)+(galvanic skin response)
  - Difficult to understand the projection and thus to interpret a correlation found
  - See tutorial: http://csnet.otago.ac.nz/cosc453/student_tutorials/principal_components.pdf
- Self-organizing maps (SOM) aka Kohonen map
  - Reduce the dimensions of data through the use of self-organizing neural networks
  - Produces usually a 2D map which mirrors the similarity of cases (similar cases are grouped together)
  - See tutorial: http://davis.wpi.edu/~matt/courses/soms/
- Problems
  - Difficult to interpret, display coordinates have no semantic meaning
  - Iterative approach, which is computationally demanding



Example of a color SOM
Germano 1999

Outline

- Reference model and data terminology
- Visualizing data with < 4 variables
- Visualizing multivariable data
  - Geometric transformation
  - Glyphs
  - Pixel-based
  - Dimensional Stacking
  - Downscaling of dimensions
- Case studies: support for exploring multidimensional data
  - Rank-by-feature
  - Value & relation display
  - Dust & magnet
- Clutter reduction techniques

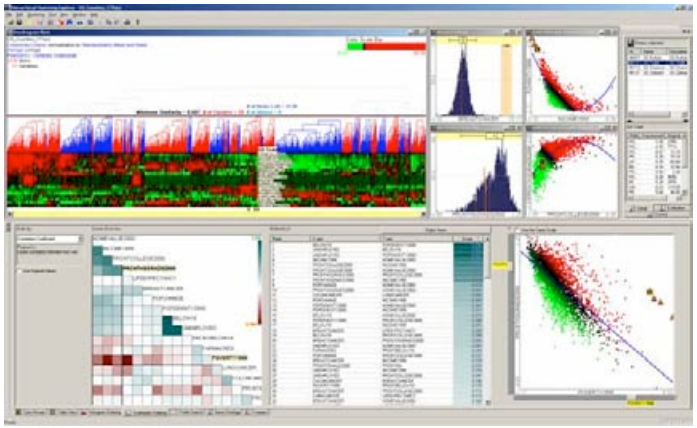
Topics of previous lecture: Multidimensional Information Visualization I

Exploration of Multivariate Data

- Systems to support the users in finding the most interesting low dimensional projections of a multidimensional data space
- Case studies
  - Rank-By-Feature framework
  - Value and Relation Display
  - Dust & Magnet

Rank-By-Feature

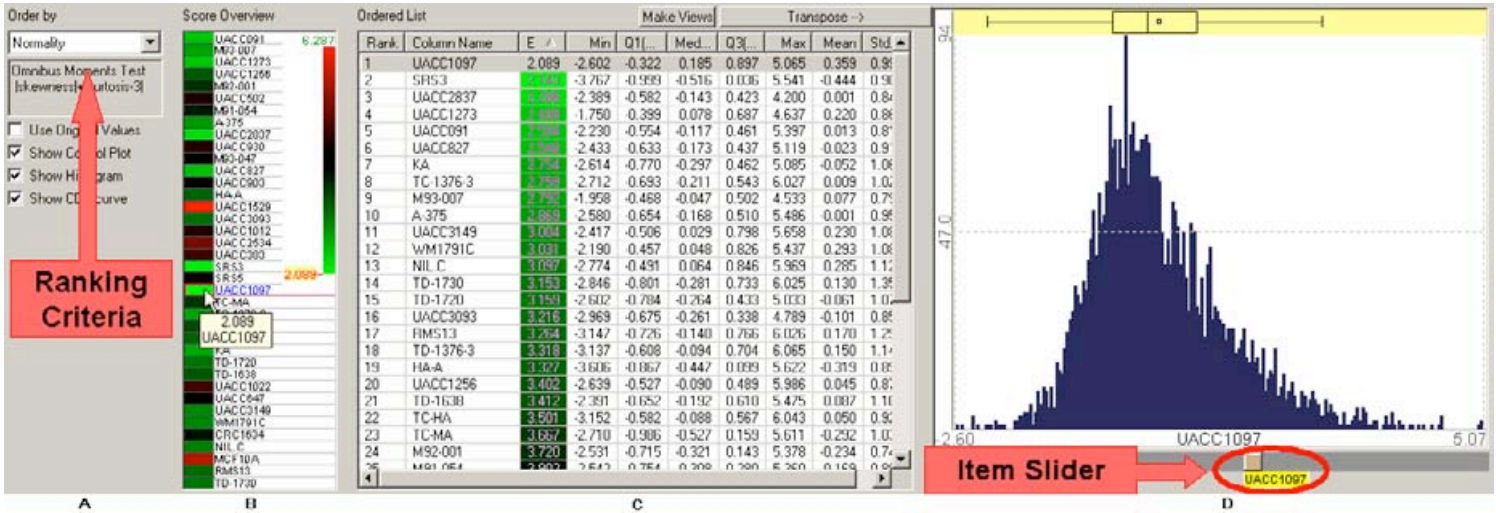
- Seo & Shneiderman 2004
- Part of the Hierarchical Clustering Explorer (HCE) 3.0 (http://www.cs.umd.edu/hcil/multi-cluster/hce3.html)
- Tabs: histogram and scatterplot ordering
- Implements systematic approach for data exploration
  - (1) study 1D, study 2D, then find features
  - (2) ranking guides insight, statistics confirm
- Tool provides low-dimensional projections as a histogram (1D) or scatterplot (2D)
- Users can select a feature detection criterion (e.g. test for normal distribution (1D), correlation coefficient (2D)) to rank projections
- The ranking facility is particularly helpful when the number of possible projections is too large to investigate: concentrate on the interesting ones



Rank-By-Feature

- Users start with 1D projections (histogram ordering)
- Four coordinated views
  - A: selection of ranking criterion
  - B: overview of scores for all dimension (color coding: the brighter the color, the higher the score)
  - C: numerical / statistical detail for each dimension (e.g. score, mean, standard deviation)
  - D: display of histogram + boxplot (minimum, first quartile, median, third quartile, maximum)
- Demo

Seo & Shneiderman 2004



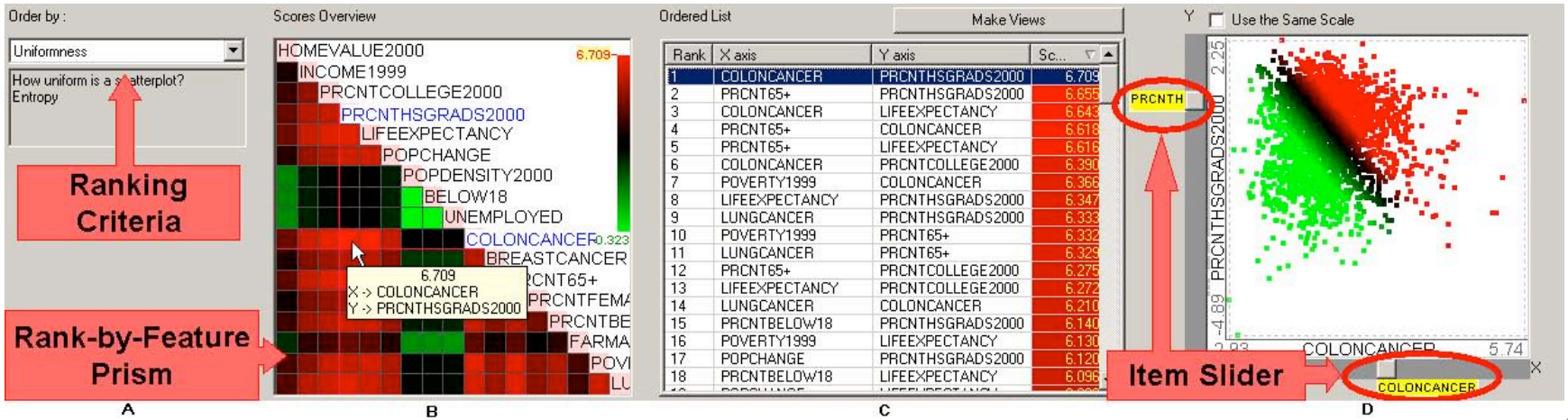
Rank-By-Feature

- Some basic statistical terms
  - Mean: Sum of all values divided by the number of values
  - Median: Middle value of a distribution of values when ranked in order of magnitude
  - Mode: Single most common value
  - Variance: average squared deviation between the mean and the values
  - Standard deviation: square root of the variance (translates the variance into the original units of measurement)
- Statistical tests supported by HCE for 1D ranking
  - Normality of the distribution: distribution of items forms a symmetric, bell-shaped curve
  - Uniformity of the distribution: all of the values of a random variable occur with equal probability (results in a flat histogram)
  - Number of potential outliers
  - Number of unique values

Rank-By-Feature

- Move on to 2D projections (scatterplot ordering)
- Identify pairwise relationships between dimensions
- B: prism provides overview of scores for dimension pairs; score is color coded
- D: scatterplot browser; multiple browsers are possible;
- Ranking creteria
  - Correlation coefficient: direction and strength of linear relationship
  - Least square error for simple linear / curvilinear regression: how well does the regression model fit
  - Number of items in a user-defined region of interest & uniformity of scatterplot
- Demo

Seo & Shneiderman 2004

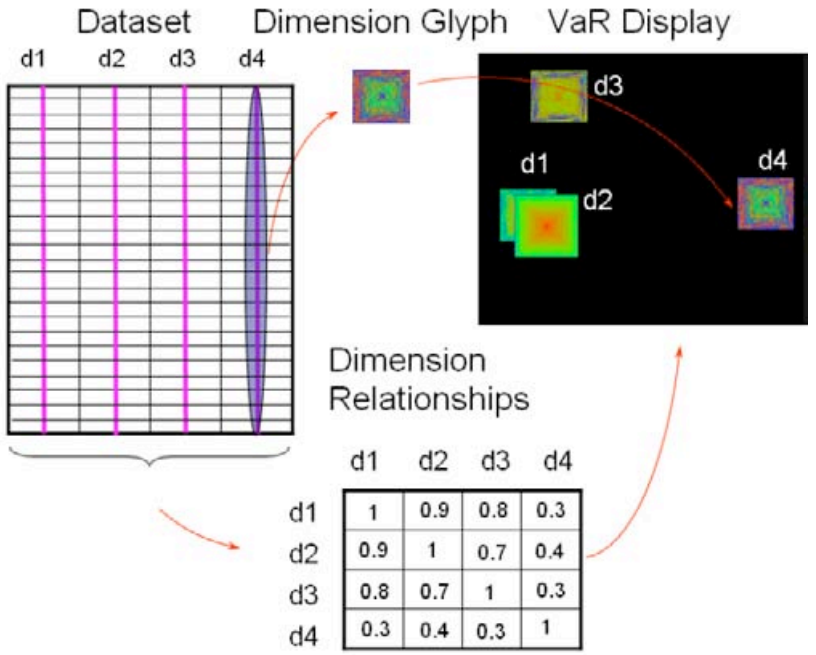


Value and Relation Display

- Yang et al. 2006
- Explore large data sets with hundreds of dimensions
- Various visualization and interaction components
- Basic principle
  - Create dimension glyphs: data values for each dimension are displayed as pixel- and scatterplot-based visualizations
  - Present glyphs in a layout, which conveys the relations (e.g. correlation) between dimensions

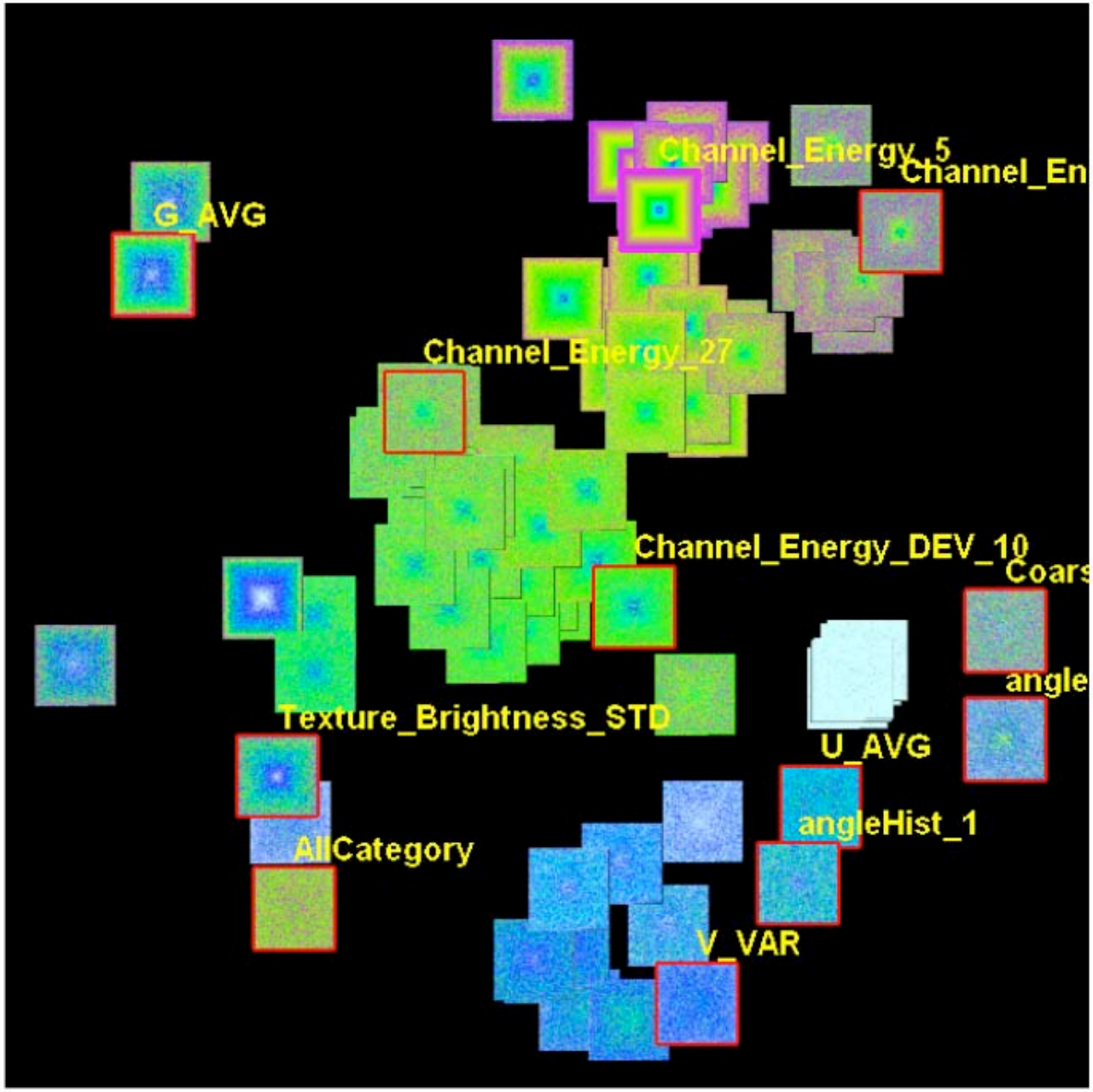
Value and Relation Display

- Pixel-based glyphs and MDS layout
- Glyph for each dimension
  - Each value is represented by a pixel whose color indicates a high or low score
  - Each value / pixel is placed at the same position across dimensions / glyphs
  - Spiral-layout of pixels inside the glyphs
- Layout of glyphs
  - Based on correlation matrix
  - Downscaling of N dimensions to 2D positions via multidimensional scaling
  - The closer a pair of glyphs the stronger the correlation of the corresponding pair of dimensions



	d1	d2	d3	d4
d1	1	0.9	0.8	0.3
d2	0.9	1	0.7	0.4
d3	0.8	0.7	1	0.3
d4	0.3	0.4	0.3	1

Yang et al. 2006

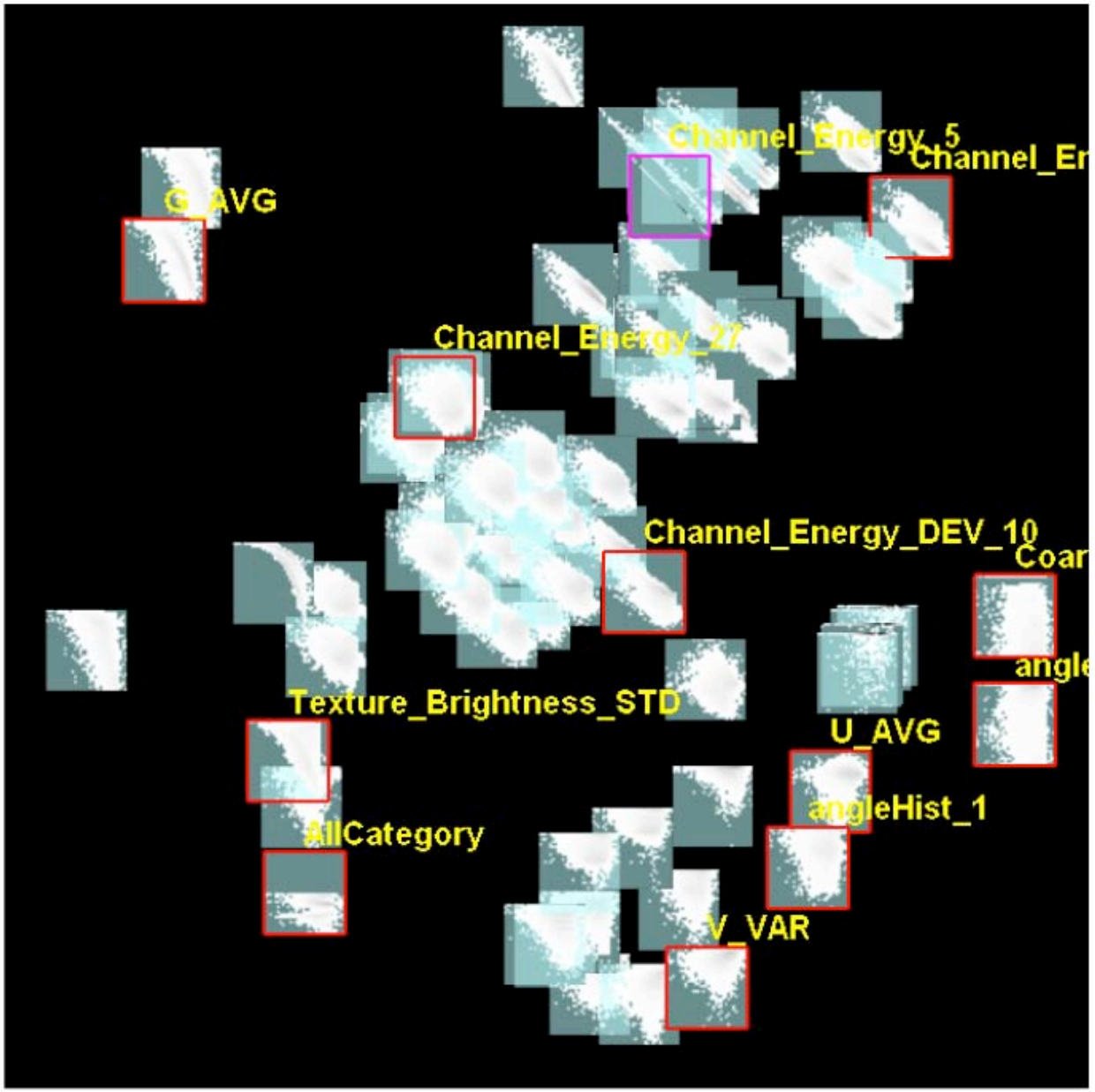


Yang et al. 2006
89 dimensions with in total
10,417 cases

Value and Relation Display

- Density-based scatterplot glyphs for each dimension
  - Color intensity of a pixel maps to value density
  - Different hue to make unoccupied areas more distinct
  - Unoccupied areas are also semi-transparent to reduce clutter effect of MDS layout
- All scatterplots share the same X dimension (can be set dynamically)

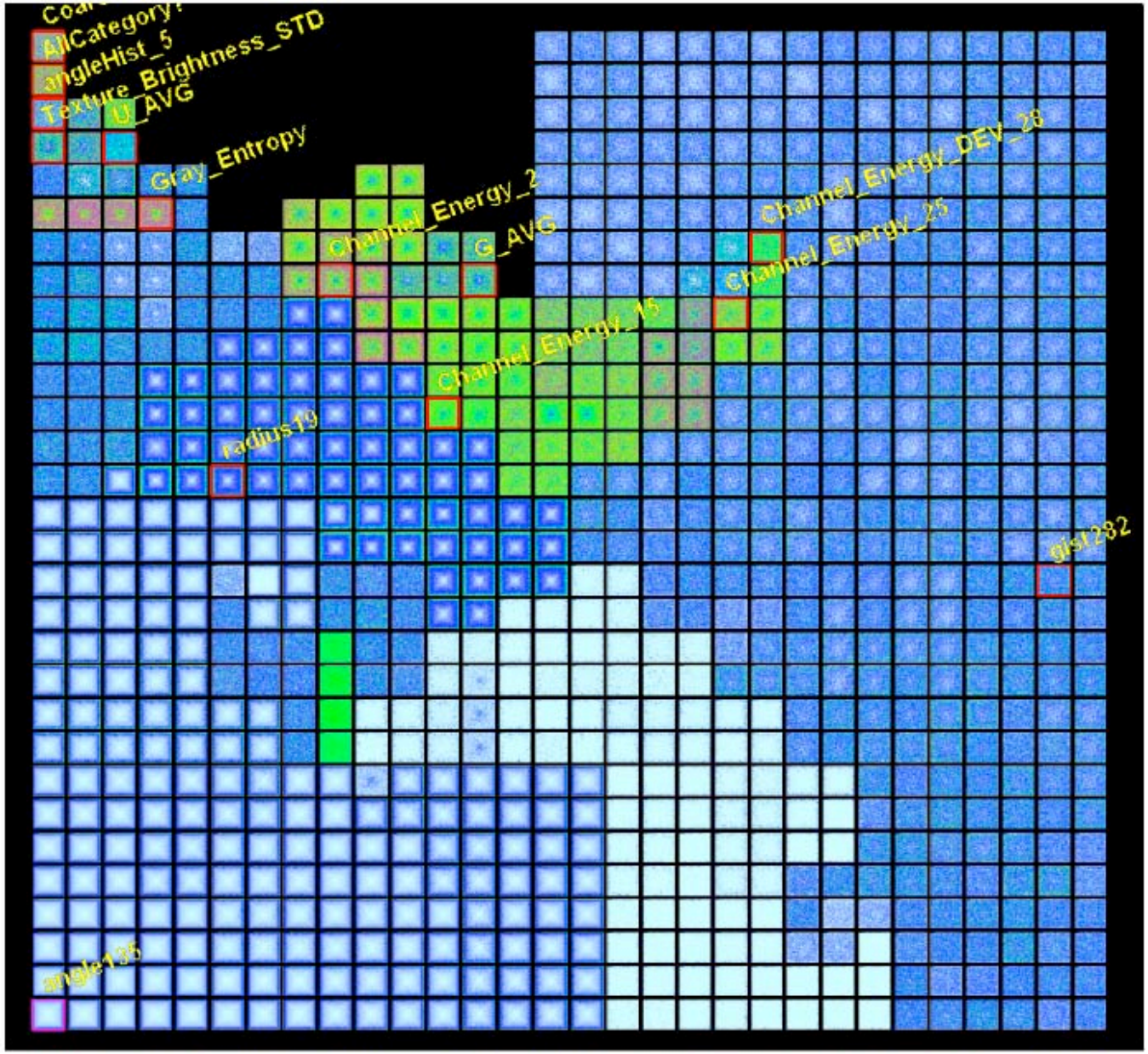
Yang et al. 2006



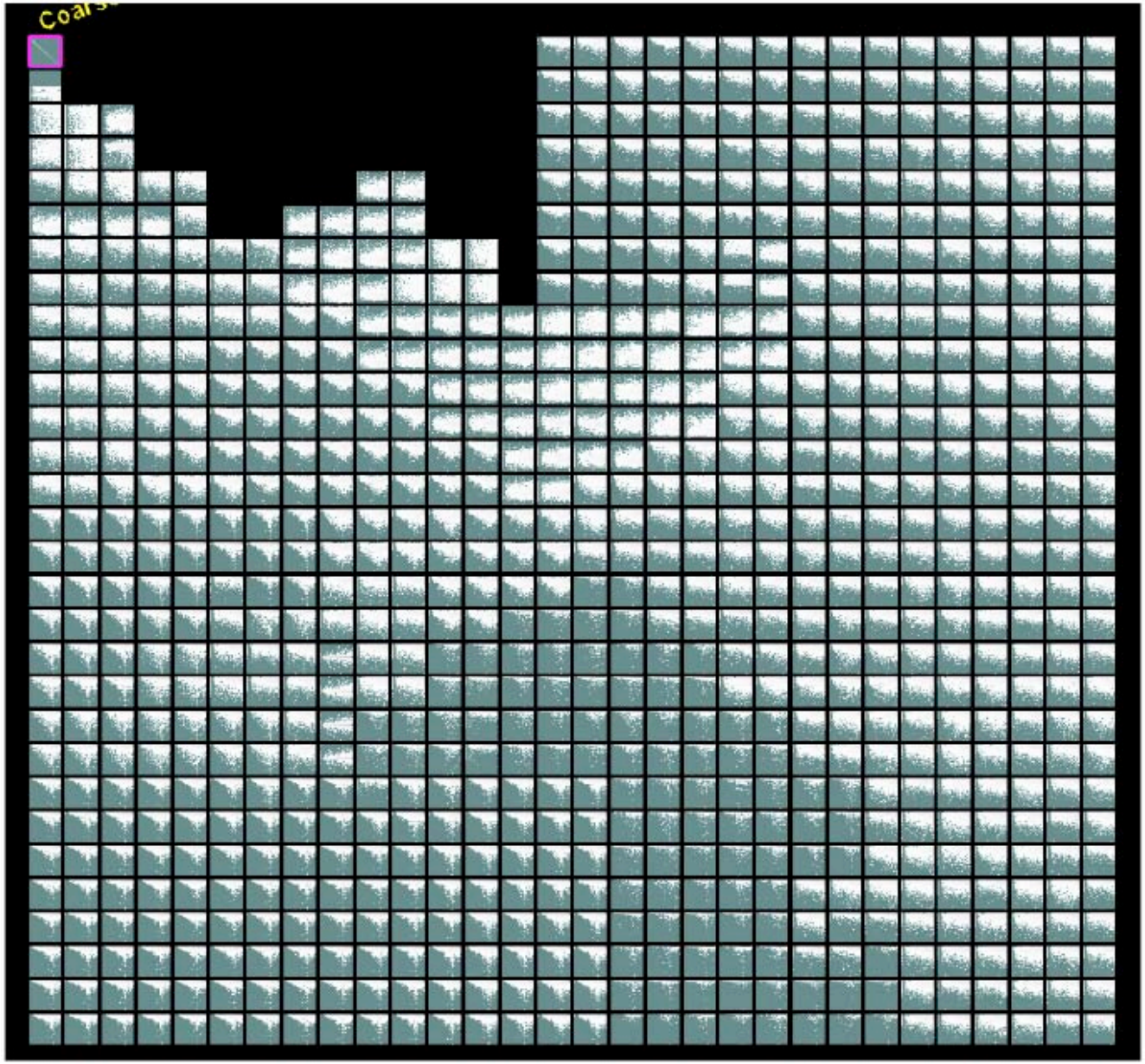
Yang et al. 2006

Value and Relation Display

- Jigsaw map to avoid clutter of glyphs in the MDS layout
  - Dimensions are grouped into a dimension hierarchy based on their closeness
  - Each leaf node is a dimension
  - A pair of more closely related dimensions has a smaller distance in the hierarchy
  - Dimensions are ordered in a 1D sequence via depth-first traversal of the hierarchy
  - Sequence is mapped to 2D display by using a space-filling curve (H-curve)
- No overlap of glyphs, still information about the relations between the dimensions can be conveyed
  - Similar dimensions are close to each other
  - Patterns of pixel-glyphs form boundaries



Yang et al. 2006

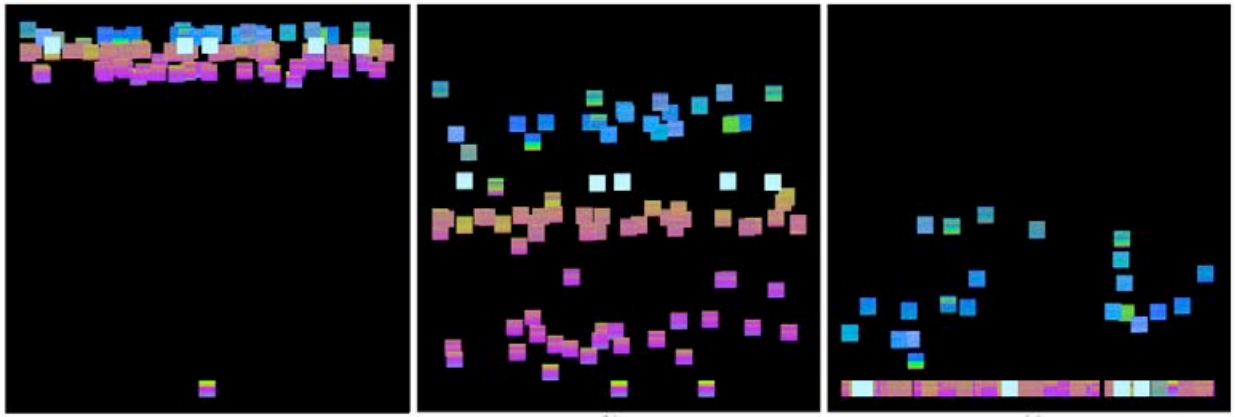


Yang et al. 2006

Yang et al. 2006

Value and Relation Display

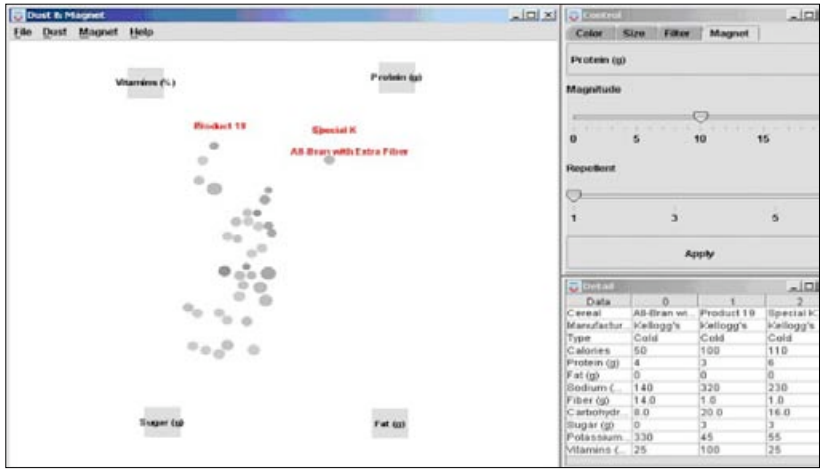
- Rainfall metaphor
  - Animation to convey relations between a single variable and the remaining dimensions
  - Select dimension (center bottom)
  - Remaining dimension fall from top (sky) to the bottom
  - Horizontal position is random
  - Acceleration of falling speed proportional to correlation with the selected dimension
- Playful exploration



Yang et al. 2006

Dust & Magnet

- Yi et al. 2005
- Data cases are represented as particles of iron dust
- Magnets represent the dimensions of the data set
- Users manipulate the magnets to move the dust
- Dust moves at different speed depending on its data values for the magnet dimensions
- **Movie**



Outline

- Reference model and data terminology
- Visualizing data with < 4 variables
- Visualizing multivariable data
  - Geometric transformation
  - Glyphs
  - Pixel-based
  - Dimensional Stacking
  - Downscaling of dimensions
- Case studies: support for exploring multidimensional data
  - Rank-by-feature
  - Value & relation display
  - Dust & magnet
- Clutter reduction techniques

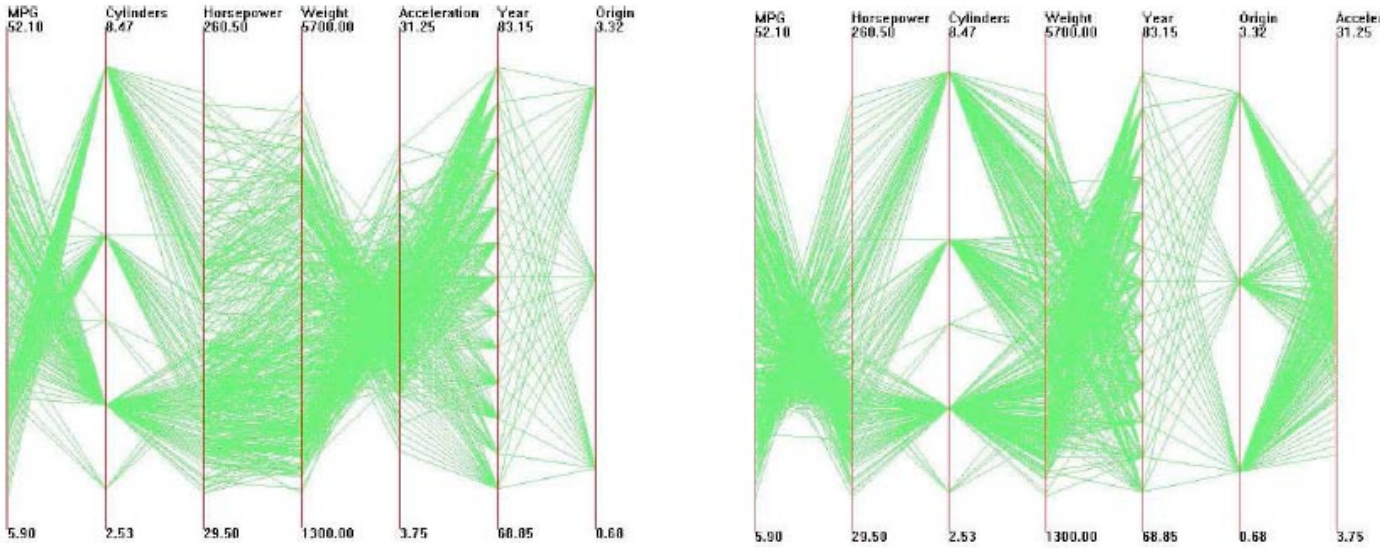
Topics of previous lecture: Multidimensional Information Visualization I

Clutter Reduction Techniques

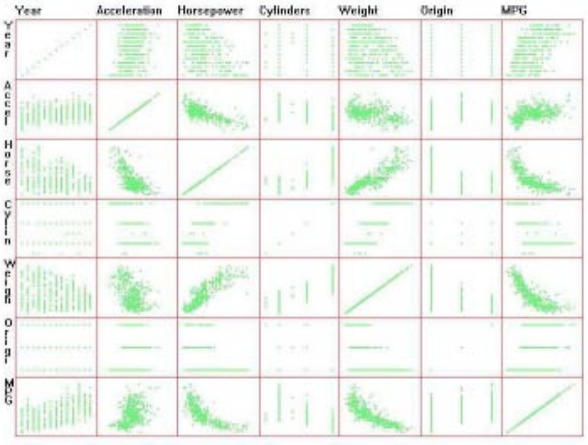
- Clutter: crowded and disordered visual entities that obscure the structure in visual displays
- Avoiding clutter is one of the main challenges in Information Visualization
- Common technique: aggregation / clustering (one mark represents more than one case, e.g. group day sales into months)
- Alternative techniques to reduce clutter
  - Dimension reordering
  - Sampling
  - Point displacement
  - Filtering (Dynamic queries, zooming - topic of lecture to come)

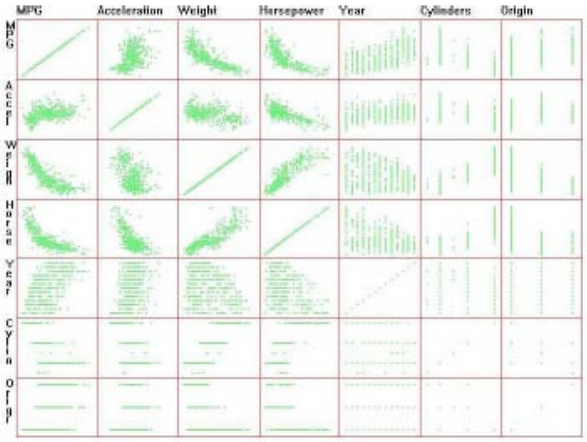
Dimension Reordering

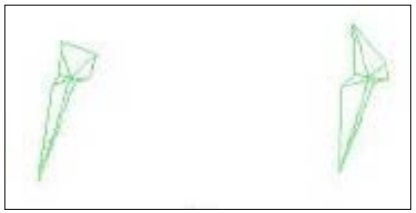
- Peng et al. 2004
- Automatically identify the views with the least amount of visual clutter
- Clutter definition and algorithms for different visualization techniques

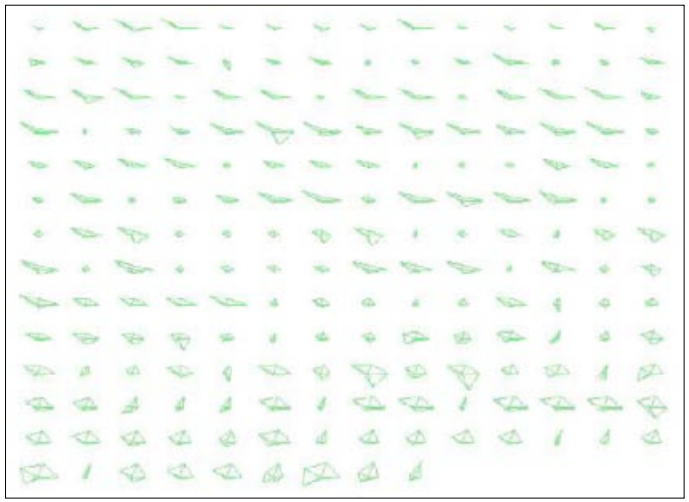


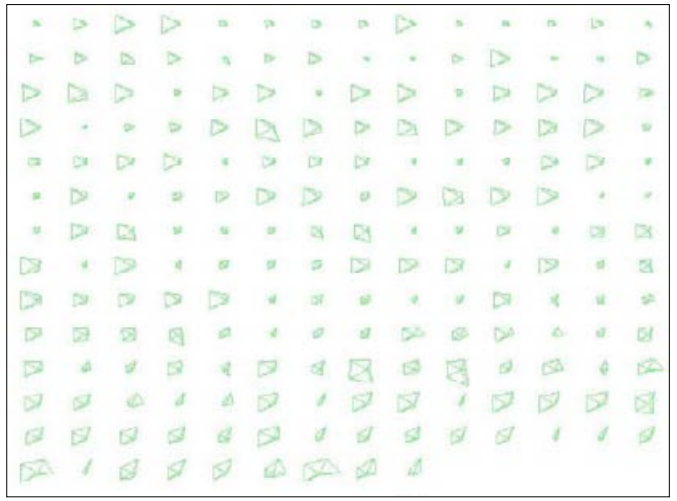
Dimension Reordering







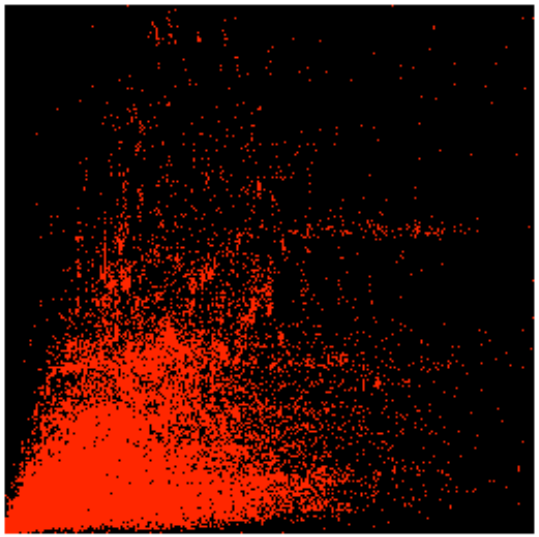




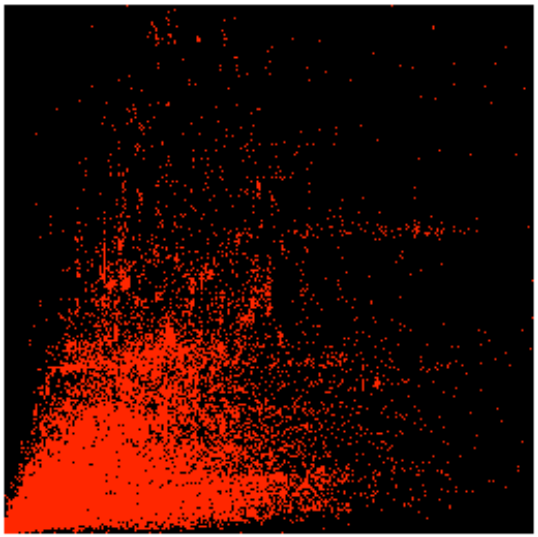
Peng et al. 2004

Sampling

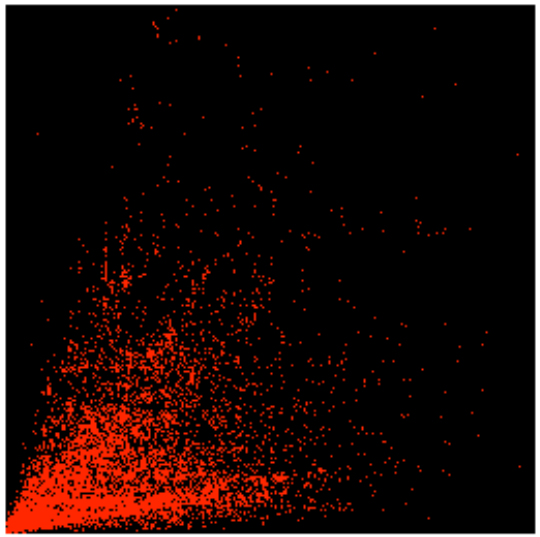
- Reduce the density of visual representation by displaying a random subset of the data
- Random sampling preserves the distribution of data
- Overall trends (e.g. correlation) can still be detected at a reduced density
- Uniform sampling (Ellis & Dix 2004)
  - Applying the same (manually or automatically defined) sampling factor to the entire data space
  - Problem: areas with low density may become empty
- Non-uniform sampling (Bertini & Santucci 2004)
  - Preserving relative density
  - Model to compute where, how, and how much to sample to preserve image characteristics



(a) - no sampling

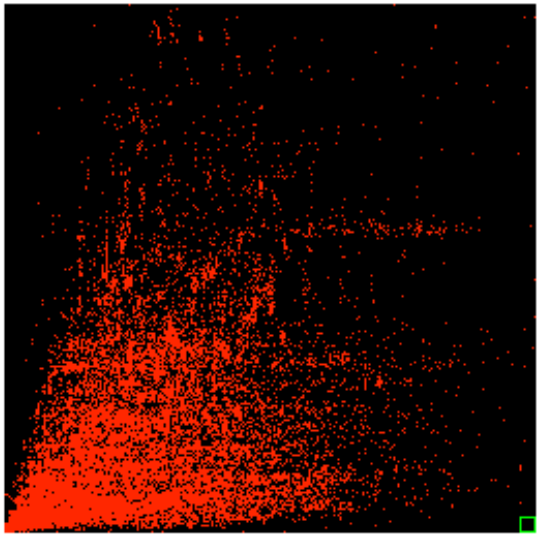


(b) uniform sampling 80%



(c) best uniform sampling 20%

Bertini & Santucci 2004

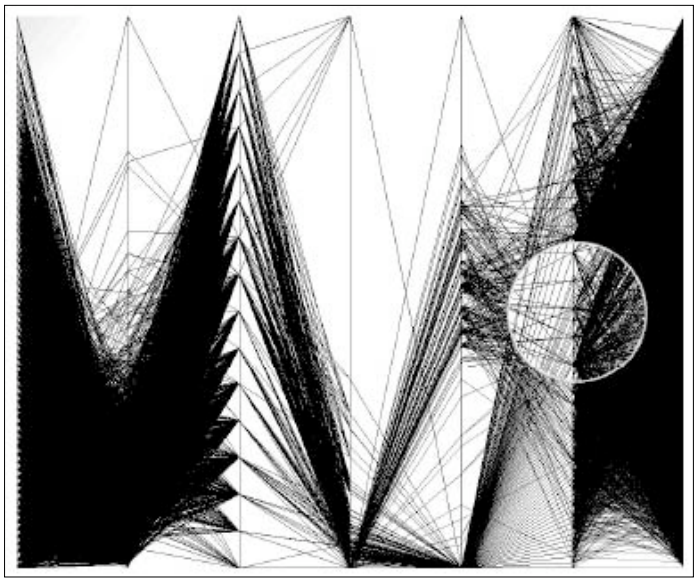


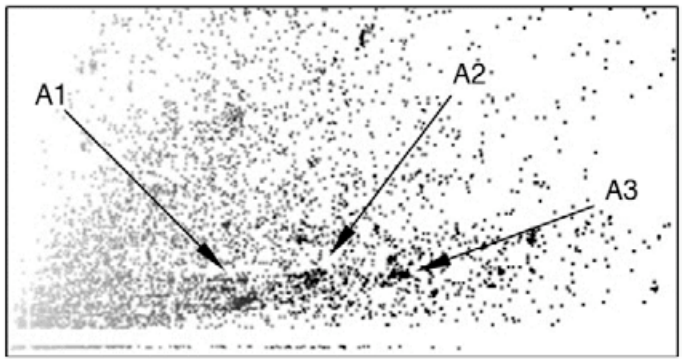
(d) non-uniform sampling

(e) most dense areas

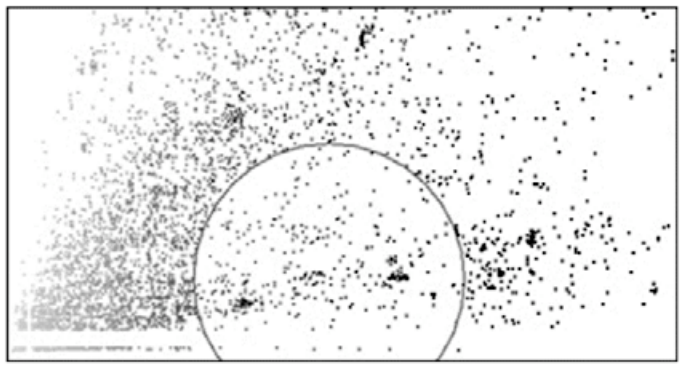
Sampling Lens

- Ellis et al. 2005
- Magic Lens approach to apply random sampling to user-defined regions while maintaining the original data density in the context





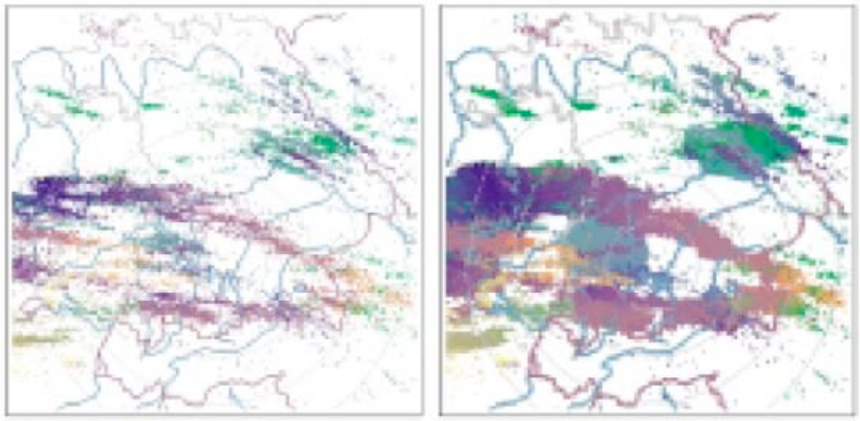
(a) without lens



(b) lens over clusters

Point Displacement

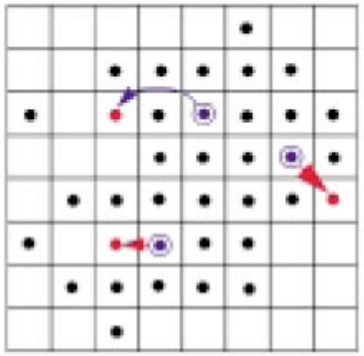
- Data points that would overlap other data points are moved to adjacent free positions
- Position of the data points and their distance should be preserved as much as possible
- Three algorithms for displacing pixels (e.g. adding abstract data to geographical maps) (Keim & Hermann 1998)
  - Nearest-Neighbor
  - Curve-based
  - Gridfit
- First two algorithms share the same procedure
  - All data points which have a unique position are placed on the display
  - New positions are determined for the remaining points

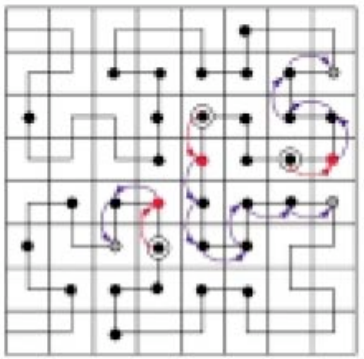


Keim 2000: Lightning strike data with and without overlap

Point Displacement

- Nearest-Neighbor algorithm
  - For all points which have not been set in the first step, place them on the nearest unoccupied position
  - Fast to compute, but limited effectiveness for very dense displays (pixels may be placed very far from their original positions)
- Curve-based algorithm
  - For all points which have not been set in the first step
    - Compute the nearest unoccupied position on a given screen-filling curve
    - Shift all point between the occupied pixel and the unoccupied position along the screen-filling curve
    - Place the point on the newly available unoccupied pixel

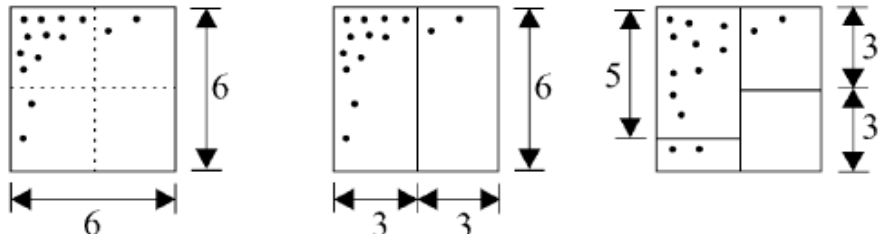


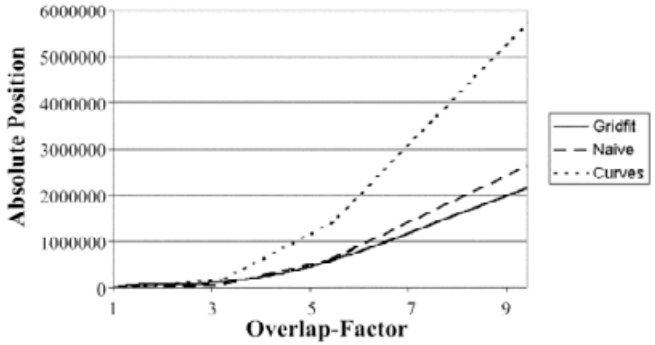


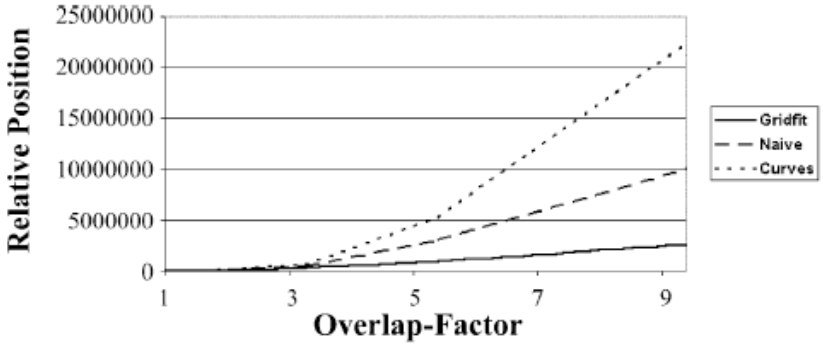
Keim & Hermann 1998

Point Displacement

- Gridfit Algorithm - recursive subdivision of the screen space
  - Recursively partition the data space into four subsets of data points belonging to four equally-sized screen subregions
  - If the data points do not fit into the screen subregions, determine a new extend of the four subregions such that the data points of each of the subsets can be visualized in the corresponding subregion
- Gridfit seems to provide the best preservation of item positions in relative and absolute terms (Keim & Hermann 1998)







Miscellaneous

- Update: Summaries of research articles: 100 words min, 500 words max
- Obligatory literature:

  D. Keim, M. C. Hao, J. Ladisch, M. Hsu: ”Pixel Bar Charts: A New Technique for Visualizing Large Multi-Attribute Data Sets without Aggregation”, 2001
  http://www.hpl.hp.com/techreports/2001/HPL-2001-92.pdf
- Send me an email for trying out the commercial visualization tool “Tableau Desktop” with test data sets + tutorial movies