

Random Sampling as a Clutter Reduction Technique to  
Facilitate Interactive Visualisation of Large Datasets

Geoffrey Philip Ellis

Submitted for the degree of Doctor of Philosophy

Computing Department  
Lancaster University  
UK

September 2008



*Ah Ha! Now that is interesting!*

Bob Spence



# Abstract

Within our physical world lies a digital world populated with an ever increasing number of sizeable data collections. Exploring these large datasets for patterns or trends is a difficult and complex task, especially when users do not always know what they are looking for. Information visualisation can facilitate this task through an interactive visual representation, thus making the data easier to interpret. However, we can soon reach a limit on the amount of data that can be plotted before the visual display becomes overcrowded or cluttered, hence potentially important information becomes hidden.

The main theme of this work is to investigate the use of dynamic random sampling for reducing display clutter. Although randomness has been successfully applied in many areas of computer science and sampling has been used in data processing, the use of random sampling as a dynamic clutter reduction technique is novel. In addition, random sampling is particularly suitable for exploratory tasks as it offers a way of reducing the amount of data without the user having to decide what data is important.

Sampling-based scatterplot and parallel coordinate visualisations are developed to experiment with various options and tools. These include simple, dynamic sampling controls with density feedback; a method of checking the reality of the representative sample; the option of global and/or localised clutter reduction using a variety of novel lenses and an auto-sampling option of automatically maintaining a reasonable view of the data within the lens. Furthermore, this work showed that sampling can be added to existing tools and used effectively in conjunction with other clutter reduction techniques.

Sampling is evaluated both analytically, using a taxonomy of clutter reduction developed for the purpose, and experimentally using large datasets. The analytic route was prompted by an exploratory analysis, which showed that evaluation of information visualisation based on user studies are problematic.

This thesis has contributed to several areas of research:

- ▶ the feasibility and flexibility of global or lens-based sampling as a clutter reduction technique are demonstrated through sampling-based scatterplot and parallel coordinate visualisations.
- ▶ the novel method of calculating the density for overlapping lines in parallel coordinate plots is both accurate and efficient and enables constant density within a sampling lens to be maintained without user intervention.
- ▶ the novel criteria-based taxonomy of clutter reduction for information visualisation provides designers with a method to critique existing visualisations and think about new ones.



# Acknowledgements

My thanks go to my supervisor Alan Dix, for his unrivalled enthusiasm, acute perception, mathematical wizardry and for generally letting me get on with things.

I am indebted to my wife Devina for supporting me in this venture and for taking on the daunting task of proof reading the manuscript.

I am grateful to the late Steve Pollitt, who gave me the opportunity to start my information retrieval research with the MenUSE and HiBROWSE projects in the mid 90's and who was an inspiration to me.

Special thanks to Enrico Bertini for helping to get the initial sampling-based visualisation off the ground.

I would also like to thank my examiners, Aaron Quigley and John Mariani for their valuable comments and suggestions.

I am grateful to Lancaster University for the 40th Anniversary Doctoral Fellowship award and to the Computing Department for their support.

I would also like to thank the following for their contributions towards the cost of attending conferences: The Royal Academy of Engineering, DELOS European Network of Excellence and at Lancaster University, The William Ritchie Travel Fund, The Faculty of Science and Technology and the Computing Department.

Finally I would like to thank the people in InfoLab21 for making it a pleasant work place, with special thanks to Kiel for taking time out to chat about the world of gaming and other more erudite matters.





# Table of Contents

<b>Abstract</b> .....	<b>i</b>
<b>Acknowledgements</b> .....	<b>ii</b>
<b>Table of Contents</b> .....	<b>iii</b>
<b>List of Figures</b> .....	<b>viii</b>
<b>List of Tables</b> .....	<b>xiii</b>
<b>Chapter 1 Introduction</b> .....	<b>1</b>
1.1 Visualising large datasets.....	3
1.1.1 Visual limits.....	3
1.1.2 Computational limits.....	7
1.2 Uses of randomness.....	9
1.2.1 Randomness in computing.....	9
1.2.2 Randomness for data processing.....	9
1.3 Random sampling for clutter reduction.....	11
1.4 Approach and Objectives.....	13
1.5 Novel characteristics of the work.....	19
1.6 Contribution to the research area.....	21
1.7 Structure of the thesis.....	23
<b>Chapter 2 Sampling as a clutter reduction technique</b> .....	<b>27</b>
2.1 The Astral Telescope Visualiser.....	29
2.1.1 Sampling issues.....	33
2.2 The z-index method.....	35
2.2.1 Display continuity.....	35
2.2.2 Reality Check.....	37
2.3 Randomness in visualisation.....	37
2.3.1 Sampling an infinite space.....	39
2.3.2 Non-uniform sampling.....	41
2.3.3 Sampling a network graph.....	43
2.4 Types of sampling.....	45
2.4.1 Constant density.....	47
2.4.2 Attribute dependent density.....	47
2.4.3 Sampling structured data.....	47
2.5 Sampling from databases.....	49
2.5.1 Issues and requirements.....	49
2.5.2 Database management system support.....	51
2.6 Summary and reflection.....	53
<b>Chapter 3 Clutter-reduction Taxonomy for Information Visualisation</b> .....	<b>57</b>
3.1 Information visualisation classification schemes.....	59
3.1.1 Ward's taxonomy of glyph placement strategies.....	63
3.1.2 Bertini's classification of clutter reduction techniques.....	65
3.2 Techniques for visual clutter reduction.....	65



3.2.1	Appearance.....	67
3.2.2	Spatial distortion.....	69
3.2.3	Temporal.....	71
3.3	Clutter reduction criteria.....	71
3.3.1	Avoids overlap.....	73
3.3.2	Keeps spatial information.....	73
3.3.3	Can be localised.....	73
3.3.4	Is scalable.....	73
3.3.5	Is adjustable.....	75
3.3.6	Can show point/line attribute.....	75
3.3.7	Can discriminate points/lines.....	75
3.3.8	Can see overlap density.....	75
3.3.9	Other criteria.....	75
3.4	Clutter-reduction Taxonomy.....	77
3.4.1	Discussion of clutter reduction technique.....	79
3.5	Evaluating the taxonomy.....	97
3.5.1	Validity.....	99
3.5.2	Utility.....	99
3.5.3	Comparison with Ward's taxonomy of glyph placement strategies.....	101
3.5.4	Comparison with Bertini's clutter reduction strategies.....	103
3.5.5	Criteria are important.....	109
3.6	Summary and reflection.....	111
<b>Chapter 4</b>	<b>Clutter reduction: random sampling and lenses.....</b>	<b>117</b>
4.1	Sampling-based scatterplot and parallel coordinates.....	119
4.1.1	Basic sampling.....	119
4.1.2	Reality Check.....	121
4.2	Comparing clutter reduction techniques using same dataset.....	125
4.2.1	Sampling.....	127
4.2.2	Opacity.....	127
4.2.3	Point size.....	129
4.2.4	Filtering.....	129
4.2.5	Filtering vs. Sampling.....	131
4.2.6	Sorting issue.....	133
4.3	Sampling Lens.....	133
4.3.1	Lens features.....	133
4.3.2	Generating the lens sample.....	135
4.3.3	Implementing the lens.....	137
4.3.4	Examples of the lens on scatterplots and parallel coordinates.....	139
4.4	Other lenses and techniques for parallel coordinates.....	139
4.4.1	Inter-axis.....	139
4.4.2	Axis (filter).....	141
4.4.3	RaDar.....	143
4.4.4	Fade and Twinkle – Reality Check transitions.....	147
4.5	Summary and reflection.....	149



<b>Chapter 5 The provision of auto-sampling.....</b>	<b>155</b>
5.1 Defining a clutter measure.....	157
5.1.1 Existing metrics for display clutter and density.....	157
5.1.2 Defining a measure for occlusion.....	159
5.2 The first attempt at auto-sampling.....	161
5.2.1 Defining further occlusion measures.....	165
5.3 Investigating occlusion measures for parallel coordinates.....	165
5.3.1 The experiments.....	165
5.3.2 Empirical results.....	167
5.3.3 Theoretical model.....	171
5.4 Methods for calculating occlusion.....	173
5.5 Comparing the occlusion algorithms.....	177
5.5.1 Accuracy : which is good enough?.....	177
5.5.2 Efficiency: which is fast enough?.....	181
5.5.3 The winner.....	187
5.6 Dealing with non-uniform density.....	187
5.6.1 Identifying the problem.....	189
5.6.2 The solution - using multiple bins.....	191
5.7 Summary and reflection.....	197
<b>Chapter 6 Evaluation of sampling.....</b>	<b>199</b>
6.1 User evaluation issues.....	201
6.1.1 User evaluation of information visualisations is problematic.....	201
6.1.2 Possibility for evaluating the Sampling Lens with users.....	205
6.1.3 Objectivity of criteria-based evaluation.....	209
6.2 Comparing sampling to other clutter reduction techniques.....	211
6.2.1 Criteria based evaluation of sampling.....	211
6.2.2 Comparison between sampling and clustering.....	219
6.2.3 Advantages and disadvantages of sampling.....	221
6.3 Further exploration of sampling-based scatterplots.....	223
6.3.1 Global sampling.....	223
6.3.2 Lens-based sampling.....	231
6.4 The Sampling Lens synthesis.....	233
6.4.1 Development of the Sampling Lens application.....	233
6.4.2 Functionality of the Sampling Lens visualisation.....	235
6.5 Can sampling be incorporated into other visualisations?.....	239
6.5.1 Hierarchical data structures.....	241
6.5.2 Visualisations that avoid overplotting.....	245
6.5.3 Visualisations that provide representative data.....	245
6.6 Astral Visualiser revisited.....	247
6.6.1 Thinking about the Astral Visualiser.....	249
6.7 Summary and reflection.....	251
<b>Chapter 7 Conclusion.....</b>	<b>255</b>
7.1 Main issues and outcomes.....	257
7.2 Summarising sampling as a clutter reduction technique.....	261
7.3 Meeting the objectives of this work.....	267



7.4	Future directions.....	271
7.4.1	Sampling structured and relational data.....	271
7.4.2	Visualising uncertainty.....	275
7.5	Final remarks.....	277
	<b>References.....</b>	<b>281</b>
	<b>Appendix A Examples of clutter reduction techniques.....</b>	<b>313</b>
A.1	Clutter-reduction Taxonomy techniques.....	313
A.1.1	Filtering.....	313
A.1.2	Change point size.....	315
A.1.3	Change opacity.....	317
A.1.4	Clustering.....	319
A.1.5	Displacement.....	321
A.1.6	Topological distortion.....	323
A.1.7	Space-filling.....	327
A.1.8	Pixel-plotting.....	327
A.1.9	Dimensional reordering.....	329
A.1.10	Animation.....	329
A.2	Other techniques.....	333
A.2.1	Summary statistics and aggregation.....	333
A.2.2	Dimensional reduction.....	335
A.2.3	Appearance other than point size and opacity.....	335
A.2.4	Anisotropic Volume Rendering.....	337
	<b>Appendix B Description of datasets used in this work.....</b>	<b>339</b>
B.1	Portland cars dataset (cars 5k and cars 1k).....	339
B.2	SIPP 2004 dataset.....	339
B.3	Parcels dataset.....	341
B.4	Synthetic clustering dataset.....	341
B.5	People dataset.....	343
B.6	Stockmarket dataset.....	343
B.7	Household Income dataset.....	343
	<b>Appendix C Details of experiments with parallel coordinate Sampling Lens....</b>	<b>345</b>
C.1	exp1 to exp21.....	345
C.2	exp22 to 29.....	347
C.3	exp30 to exp34.....	347
C.4	exp35 to exp41.....	349
C.5	exp42 to exp44.....	349
C.6	exp45 to exp55.....	351
C.7	exp56.....	353
C.8	exp57 to exp59.....	353
C.9	Data collected for each experiment.....	353
C.10	Example of data output from an experiment.....	357





<b>Appendix D Implementation Issues.....</b>	<b>359</b>
D.1 Instrumentation of the Sampling Lens.....	359
D.1.1 Density visualiser.....	359
D.1.2 Extended sampling controls.....	361
D.1.3 Empirical controls.....	361
D.2 Architectural issues.....	363
D.2.1 InfoVis Toolkit architecture.....	365
D.2.2 Implementing the sampling applications.....	365
D.3 OpenGL implementation.....	369
D.3.1 Java2D vs. OpenGL.....	369
D.3.2 OpenGL version of the Sampling Lens.....	371
<b>Appendix E Comparison of Natural Building Techniques.....</b>	<b>375</b>
<b>Appendix F An Explorative Analysis of User Evaluation Studies in InfoVis.....</b>	<b>379</b>



# List of Figures

Figure 1-1	A concept map showing a set of linked concepts surrounding the idea of information visualisation.....	0
Figure 1-2a	Examples of overplotted visualisations - NetMap movie database .....	2
Figure 1-2b	Examples of overplotted visualisations - Cars for sale.....	2
Figure 1-3	Relationship between the objectives of this research.....	12-18
Figure 2-1	Views of the <i>stars</i> through a zoomable telescope as the magnification increases.....	28
Figure 2-2	Model behind the Astral Telescope Visualiser.....	30
Figure 2-3	Example of zooming in to a scatterplot with the Astral Telescope Visualiser prototype, demonstrating automatic adjustment of the sampling rate.....	30
Figure 2-4	Generation of parameter-performance pair by random sampling the parameter space.....	40
Figure 2-5	Histogram view of the multi-dimensional parameter and performance spaces of the Influence Explorer.....	40
Figure 2-6	Examples of different types of sampling from Bertini and Santucci's sampling visualisation (a) original image, (b) best uniform sampling and (c) perceptual non-uniform sampling.....	42
Figure 2-7	Visualisation large network graph through sampling (Rafiei and Curial) (a) 0.1% sample size and (b) 0.2% sample size.....	42
Figure 2-8	Washing powder sales map: uniform sampling removes most of the data items in less populated regions .....	46
Figure 2-9	Washing powder sales map: non-uniform sampling provides a region wide comparison of brands.....	46
Figure 2-9	Washing powder and washing machine sales map: (a) identical non-uniform sampling for powder and machines (b) independent non-uniform sampling.....	84
Figure 3-1	Using sampling to reduce the number of overlapping lines.....	78
Figure 3-2	Dynamic query interface.....	78
Figure 3-3	Displacement reveals the data underneath and helps to disambiguate the edges.....	79
Figure 3-4	Examples of multi-attribute glyphs.....	80
Figure 3-5	The clustering used in Hierarchical parallel coordinates is scaleable to very large datasets, only limited by computational resources.....	80
Figure 3-6	PixelMap avoids overlap altogether by distorting the underlying map .....	81
Figure 3-7	Overlaid grid squares provide a reference for the user and helps to keep spatial information following a topological distortion.....	82
Figure 3-8	Example of an RSVP, Rapid Serial Visual Presentation.....	82
Figure 3-9	Space-filling algorithms avoid overplotting.....	84
Figure 3-10	An example of filtering with a lens.....	86
Figure 3-11	Constant density display.....	86
Figure 3-12	Liquid browsing displaces points locally based on the distance from the stylus position and the pressure exerted by the user.....	86
Figure 3-13	NodeTrix combines a node-link representation to give an overall view of a social network with adjacency matrices giving detailed analysis of local communities.....	88
Figure 3-14	Dimensional reordering in a parallel coordinate plot.....	88
Figure 3-15	Changing the visibility of structures within a parallel coordinate plot using transfer functions to map line density to opacity.....	90
Figure 3-16	Hierarchical clustering with Edge Bundles .....	90



Figure 3-17	Animated bubbles used in Cenimation avoid permanent overlap .....	90
Figure 3-18	Discrimination of lines in a parallel coordinate plot utilising the outlier-preserving technique of Novotny et al. and opacity.....	92
Figure 3-19	Curving the lines of the plot to help the user follow individual lines.....	94
Figure 3-20	Topological distortion along axes of a parallel coordinate plot which stretches region with many lines crossing to help disambiguate the paths of the lines .....	94
Figure 3-21	Reducing the opacity of lines can indicate the density of the overlapping lines.....	96
Figure 3-22	Information Mural utilises the display space by plotting at a pixel level.....	96
Figure 3-23	Fisheye Menu example.....	110
Figure 4-1	Sampling-based scatterplot visualisation showing age (horizontal axis) versus monthly income (vertical axis) for a sample of 9432 people.....	118
Figure 4-2	Scatterplot of the age-income-education data now reduced to 188 points (a 2% sampling rate).....	118
Figure 4-3	Basic z-index method to generate a sample and ensuring display continuity.....	120
Figure 4-4	Parallel coordinates visualisation at sampling rates from 100% to 5%.....	120
Figure 4-5	Scatterplots following successive Reality Checks showing that despite the 2% sampling rate, the distribution is fairly consistent in the more dense regions of the plot.....	120
Figure 4-6	Parallel coordinate plots following successive Reality Checks.....	122
Figure 4-7	The same section from three scatterplots following successive Reality Checks demonstrating an artefact of the sampling.....	122
Figure 4-8	Generating successive Reality Check samples with the z-index method.....	122
Figure 4-9	Reducing the overlap of points using random sampling.....	126
Figure 4-10	Reducing the opacity of the points gives a useful density map .....	126
Figure 4-11	A combination of sampling to reduce overlap and opacity to see the overlap.....	128
Figure 4-12	Effect of the size of the plotted points on the perceived density .....	128
Figure 4-13	Filtering on three vehicle types.....	130
Figure 4-14	The effect of sampling on showing the distribution of the three vehicle types.....	130
Figure 4-15	The effect of low sampling rates on showing the distribution of the three vehicle types.....	130
Figure 4-16	Inappropriate sorting of the data over emphasises the number of blue points.....	130
Figure 4-17	Parallel coordinates Sampling Lens with an early version of the sampling control panel...	132
Figure 4-18	Generating lens samples with the z-index method.....	136
Figure 4-19	Screen shots of the early version of the Sampling Lens.....	138
Figure 4-20	Use of the inter-axis lens on a parallel coordinate plot.....	140-141
Figure 4-21	Parallel coordinate plot of the Portland cars dataset showing the advantage of axis lens in reducing display clutter.....	142
Figure 4-22	Enhancing Figure 4-21d through the use of the RaDar technique.....	144
Figure 4-23	The use of an axis lens in conjunction with the rainbow colouring of RaDar.....	144-145
Figure 4-24	Parallel coordinate plots illustrate the use of an axis lens and RaDar colouring in revealing patterns in otherwise very overcrowded plots.....	146-147
Figure 4-25	Fade transitions for (a) parallel coordinate plot and (b) scatterplot.....	148
Figure 4-26	Twinkle transition on a parallel coordinate plot.....	148
Figure 4-27	Example of the use of a fisheye lens.....	152
Figure 5-1	Occlusion model: (a) scatterplot with overplotting occurring at two of the pixels. (b) two lines crossing at the centre point.....	160



Figure 5-2	Sampling control panel for the auto-sampling version of the Sampling Lens.....	162
Figure 5-3	(a) Example of lines meeting at a point on an attribute axis. (b) Setting a non-overlap zone near to an attribute axis so that lines meeting at a point on the axis are not counted as overlapping.....	162
Figure 5-4	Behaviour of the lines algorithm with and without zone clipping in the exceptional case where many lines meet on a vertical axis.....	164
Figure 5-5	Parallel coordinate plot using 1K car dataset (labels and lens positions for exp1,2 & 3 are superimposed).....	166
Figure 5-6a	Occlusion measures for exp1, exp2 and exp3 plotted against sampling rate .....	166
Figure 5-6b	Occlusion measures for exp1, exp2 and exp3 plotted against plotted points.....	169
Figure 5-6c	Occlusion measures for exp1, exp2 and exp3 plotted against behaviour of the measures at low densities.....	168
Figure 5-7	Overplotted% occlusion measures a wide range of line crossing patterns (experiments 1, 2, 3, 18, 20 and 21).....	168
Figure 5-8	Lenses at 10% sampling rate for experiments 1, 2, 3, 18, 20 and 21.....	169
Figure 5-9	Occlusion measures for experiments 1, 2 and 3 normalised against overplotted%.....	168
Figure 5-10	Model-based measures, overplotted%, overcrowded% and hidden%.....	170
Figure 5-11	(a) theoretical curves for measures based on random point placement and (b) comparing theoretical and empirical results.....	170
Figure 5-12	Line overlap proportion.....	174
Figure 5-13	Three different occlusion algorithms (exp1).....	176
Figure 5-14	Three different occlusion algorithms for (a) exp2 and (b) exp3.....	176
Figure 5-15	Raster values for the three experiments, plotted against the number of lines crossing the lens.....	178
Figure 5-16	The three occlusion measures, raster, lines and random for a dense region of a 10,000 record dataset (exp7).....	178
Figure 5-17	Lens position for exp7 showing the small low density region to the right.....	179
Figure 5-18	Exp1, 2 and 3 normalised against raster values.....	180
Figure 5-19	Modification of the original lines algorithm to deal with the special case of lines meeting at their end points.....	180
Figure 5-20	Calculation times for the three algorithms.....	182
Figure 5-21	Raster overplotted% for various cell widths plotted against sampling rate (exp1).....	184
Figure 5-22	When rasterising a line, the proportion of cells crossed increases with the cell size.....	184
Figure 5-23	Accuracy of different raster cell-widths (exp1).....	184
Figure 5-24	Reduction in the calculation times of the raster algorithm with increasing cell widths.....	186
Figure 5-25	Lens patterns used to investigate non-uniform density across the lens.....	188
Figure 5-26	Lines, raster and random overplotted% values at different sampling rates for lens positions exp30, exp32 and exp34.....	188
Figure 5-27	Lines, raster and random overplotted% values for lens positions exp30 and exp34 plotted against number of plotted points.....	190
Figure 5-28	Overlap density maps for exp30, exp32 and exp34 at a sampling rate of 20%.....	190
Figure 5-29	Example of dividing a 100 pixel wide lens area into bins 30 pixels wide.....	191
Figure 5-30	The positive effect of binning on random overplotted% in correcting for a partly covered lens (exp39).....	192
Figure 5-31	The number of plotted points is not exactly proportional to the sampling rate.....	193





Figure 5-32	Random occlusion measure normalised against the raster standard for a partly covered lens (exp39).....	195
Figure 5-33	The advantage of using binning for lens with non-uniform density.....	194
Figure 5-34	The effect of binning on the calculated raster overplotted% values.....	196
Figure 6-1	FilmFinder application in Spotfire.....	212
Figure 6-2	Colour scale for USA household income scatterplots.....	223
Figure 6-3	Full 155K dataset showing the distribution of household income across the USA.....	222
Figure 6-4	(a) 5% and (b) 1% samples of the original USA household income 155K dataset.....	224-225
Figure 6-5a	Reducing the opacity of plotted points to 4% gives a good indication of the higher population density areas and an approximate average income through colour blending.....	226
Figure 6-5b	Reducing opacity to 1% highlights major population centres but other information is lost.	227
Figure 6-6a	Filtering highlights areas of high income.....	226
Figure 6-6b	Filtering highlights areas of low income.....	227
Figure 6-7	North-eastern states household income map (a) full dataset (b) reducing the sampling rate to 2%.....	228-229
Figure 6-8	Two successive Reality Checks following on from Figure 6-7b, demonstrate that different 2% samples present representative views.....	228
Figure 6-9a	Sampling lens over Philadelphia reduces the overplotting.....	230
Figure 6-9b	Four successive Reality Checks (top) and two other samples (bottom).....	230
Figure 6-10	Reality Check samples for the lens on a densely populated scatterplot.....	231
Figure 6-11	Combination of reduced opacity and a sampling lens on a cluttered scatterplot.....	232
Figure 6-12	Visualising large hierarchies with (a) Hyperbolic Browser and (b) Treemap.....	266
Figure 6-13	Changing the focus in a Hyperbolic Browser to expand lower nodes.....	241
Figure 6-14	Acyclic tree before sampling.....	242
Figure 6-15	Sampling the acyclic tree (50% sampling rate). (a) any node and (b) only leaf nodes.....	242
Figure 6-16	Acyclic tree with different path lengths.....	242
Figure 6-17	Zooming in on a scatterplot with automatic adjustment of sampling rate.....	244
Figure 6-18	Zooming in on a scatterplot with automatic adjustment of sampling rate.....	246,249
Figure 6-19	Zooming out on a scatterplot with automatic adjustment of sampling rate.....	248
Figure 7-1	Relationship between the objectives.....	266
Figure A-1	HomeFinder's dynamic query interface.....	312
Figure A-2	Attribute Explorer.....	314
Figure A-3	Enhanced dynamic query filters.....	314
Figure A-4	Examples of multi-attribute glyphs.....	314
Figure A-5	Constant density (a) original display (b) VIDA constant density display.....	316
Figure A-6	Reducing the opacity of lines in a parallel coordinate plot to produce a density map.....	316
Figure A-7	Changing the visibility of structures within a parallel coordinate plot using transfer functions to map line density to opacity.....	318
Figure A-8	Hierarchical parallel coordinates.....	318
Figure A-9	Hierarchical edge bundles with bundling strength $\beta$ increasing from left to right.....	321
Figure A-10	Resolving point occlusion a) no jitter b) random jitter and c) smart jitter.....	320
Figure A-11	EdgeLens displaces lines to reveal labels.....	322
Figure A-12	Curving the lines of a parallel coordinate plot to help the user follow individual lines.....	322



Figure A-13	Keim's PixelMap distorts the underlying spatial area to avoid overlapping items.....	324
Figure A-14	Carpendale's pliable surfaces indicates degree of distortion.....	324
Figure A-15	Topological distortion along axes of a parallel coordinate plot helps to disambiguate the paths of the lines.....	324
Figure A-16	Space-filling algorithms avoid overplotting: a) Treemaps and b) Sunburst.....	326
Figure A-17	Pixel-plotting: Keim's a) spirals and b) Pixel bar chart.....	326
Figure A-18	TableLens displays attribute values as pixel bars.....	329
Figure A-19	Information Mural utilises the display space by plotting at a pixel level.....	328
Figure A-20	Dimensional reordering in a parallel coordinate plot: a) before reordering, b) after.....	328
Figure A-21	Animation to avoid overlap: a) RSVP carousel b) Cenimation.....	330
Figure A-22	Feature animation imparts information on skewness and standard deviation to a cluster in a parallel coordinate plot.....	330
Figure A-23	Parallel coordinates and circular parallel coordinates.....	334
Figure A-24	Proximity-based colouring discriminates lines in the original parallel coordinate plot by automatically assigning different colours to clusters.....	334
Figure A-25	Blurriness can discriminate between points whilst still maintaining context.....	336
Figure B-1	A parallel coordinate plot of the Portland cars dataset showing the extent of the data.....	339
Figure B-2	Distribution of educational achievements for SIPP dataset.....	340
Figure B-3	Parcel dataset (German post office).....	340
Figure B-4	Distribution of median household income for USA census.....	342
Figure C-1	Lens positions for exp1, 2, 3, 18, 20 and 21.....	344
Figure C-2	Lens at 10% sampling rate for exp1, 2, 3, 18, 20 and 21.....	344
Figure C-3	Lens position for exp7 to 14 (People 10K dataset).....	344
Figure C-4	Lens positions for exp22, 23, 24, 25, 26, 27, 28, 29 (30% lens sampling rate).....	346
Figure C-5	Lens positions for exp30 to 34 (10% lens sampling rate).....	346
Figure C-6	Screen shots for synthetic data experiments 42, 43 and 44.....	348
Figure C-7a	Investigating binning with a synthetic dataset.....	350
Figure C-7b	As Figure C-7a but with random overplotted% normalised against raster overplotted%....	350
Figure C-8	Lens screen shots for exp45 (top left) to 55 (bottom right).....	351
Figure C-9	Lens screen shots for a range of occlusion values - exp56.....	352
Figure C-10	Example of the data output from an experiment and read into a spreadsheet.....	356-357
Figure D-1	Sampling Lens density visualiser.....	360
Figure D-2	Extended sampling controls for the Sampling Lens.....	362
Figure D-3	Part of a spreadsheet based on the output file produced via the empirical control panel...	364
Figure D-4	Control panel for conducting the experiments.....	364
Figure D-5	Internal structure of the InfoVis Toolkit.....	366
Figure D-6	Parallel coordinate lines in the lens sample are clipped to the lens outline and any attribute axes within the lens.....	368
Figure D-7	Comparison of the drawing time for Java2D and Jogj (OpenGL) version of a simple parallel coordinate applications.....	370
Figure D-8	Comparison of the drawing time for the Jogl (OpenGL) version of a simple parallel coordinate applications with and without the use of a display list.....	372



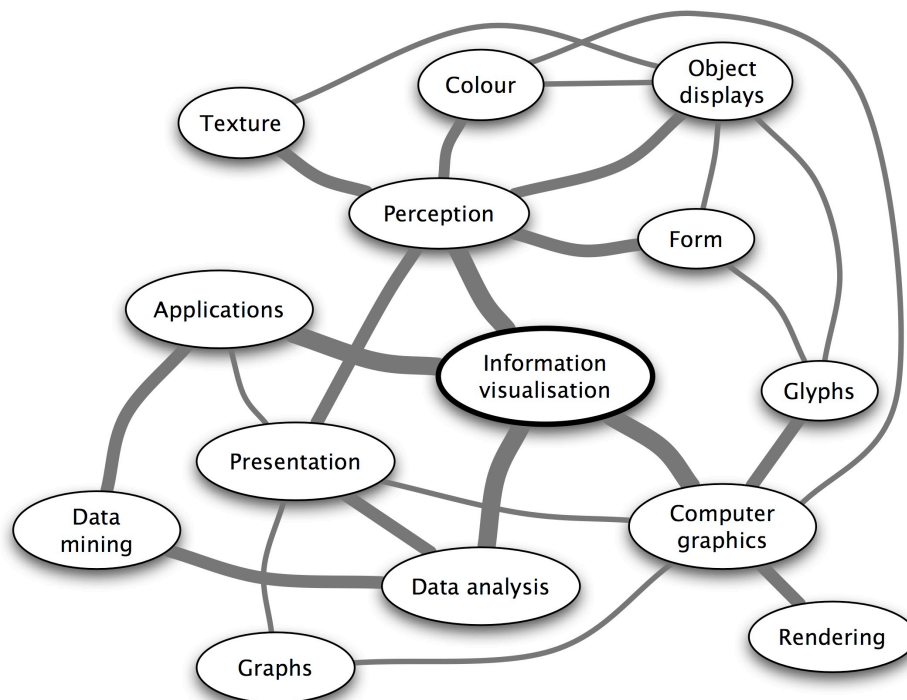
# List of Tables

Table 1-1	Contribution to the research area.....	20
Table 2-1	Data table with the addition of a z-index to facilitate random sampling.....	34
Table 2-2	Summary of the key benefits of a sampling approach to clutter reduction based on the proposed Astral Visualiser.....	54
Table 3-1	Ward's taxonomy of glyph placement strategies.....	62
Table 3-2	Bertini's design space characterisation.....	64
Table 3-3	Bertini's clutter reduction strategies.....	64
Table 3-4	Clutter reduction techniques used in the Clutter-reduction Taxonomy.....	66
Table 3-5	Example clutter reduction criteria search records.....	70
Table 3-6	Clutter-reduction Taxonomy for information visualisation.....	76
Table 3-7	Using the taxonomy to combine the strengths of several techniques to create new visualisations.....	98
Table 3-8	Sample points in a design space of glyph layout strategies.....	98
Table 3-9	Expanded version of Ward's taxonomy of glyph placement strategies.....	100
Table 3-10	Comparison of Bertini's clutter reduction methods with the clutter reduction techniques used in the Clutter-reduction Taxonomy.....	102
Table 3-11	Expanded version of Bertini's clutter reduction strategies.....	104
Table 3-12	Expanded version of Bertini's design space characterisation.....	106
Table 3-13	A comparison of natural building techniques for walls.....	108
Table 3-14	The clutter reduction taxonomy for the topological distortion technique.....	110
Table 4-1	Basic clutter reduction techniques.....	126
Table 4-2	Strengths and weaknesses of sampling and some other clutter reduction techniques.....	150
Table 5-1	Definition of the occlusion measures.....	164
Table 5-2	Lines within the lens at 10% lens sampling rate.....	166
Table 5-3	Raw data values for occlusion measures.....	174
Table 5-4	The average pixels per line of all the lines crossing the lens.....	178
Table 5-5	The major components of time taken to redraw the lens.....	182
Table 6-1	The strengths and weaknesses of a sampling approach to clutter reduction in relation to the criteria used in the Clutter-reduction Taxonomy and the objectivity in assessing each criterion.....	208
Table 6-2	Clutter-reduction Taxonomy (copy of Table 3-6).....	210
Table 6-3	A summary of the benefits of sampling for clutter reduction based on the taxonomy.....	220
Table 6-4	Comparing the performance of Java2D and OpenGL versions of the Sampling Lens with parallel coordinates datasets.....	232
Table 6-5	The Clutter-reduction Taxonomy for the sampling and topological distortion techniques...	250
Table 7-1	Main issues and outcomes arising from each chapter.....	256-260



Table 7-2	A summary of the benefits of a sampling approach to clutter reduction.....	260
Table 7-3	A summary of some disadvantages of a sampling approach to clutter reduction.....	264
Table 7-4	Main issues and outcomes for objective.....	266-270
Table C-1	Details for experiments 1 to 21.....	347
Table C-2	Details for experiments 22 to 29.....	347
Table C-3	Details for experiments 30 to 34.....	347
Table C-4	Details for experiments 35 to 41.....	349
Table C-5	Details for experiments 42 to 44.....	349
Table C-6	Details for experiments 45 to 55.....	351
Table C-7	Details for experiment 56.....	353
Table C-8	Details for experiments 57 to 59.....	353
Table D-1	Summary of the data produced for the parallel coordinate experiments.....	362
Table D-2	InfoVis Toolkit package structure together with the principal classes added to implement to Sampling Lens.....	368
Table D-3	Data sets used to compare the performance of Java2D and OpenGL versions of parallel coordinates.....	370
Table D-4	Comparison of OpenGL and Java2D for a variety of dataset.....	372

**Figure 1-1**



A concept map showing a set of linked concepts surrounding the idea of information visualisation. [After Ware 04, Figure 11.10]



# Chapter 1

## Introduction

Information visualisation is essentially about data, visual displays, people and their quest for understanding. The data is often very large, the visual displays are relatively small and hence we must explore ways, using the available computer hardware and software, to make this acquisition of knowledge as easy and enriching as possible.

We have 20 billion or so neurons of the brain devoted to analysing visual information for patterns. Combine this with an adaptive decision-making system that consults a vast mental library of experience and we can construct and utilise mental models to make sense of the physical world we live in.

In our physical world we have created a digital world. Inexpensive, powerful computers and mass storage devices, coupled with high speed communication systems have given rise to ubiquitous sensing [Essa 00] - whether this is recording our shopping habits, collecting environmental data in a forest or downloading emails, our digital world is data rich. Much of this data is abstract, in that it does not map easily onto a physical space. Although we could certainly produce a Google mashup showing where emails originated, it would be more interesting to get a clear view of our email communications in terms of who, what, where and perhaps, ultimately, why. This understanding or cognition is really about constructing a mental model and, as with the physical model of our world, a visual representation of the email data leverages our remarkable visual decision-making system.

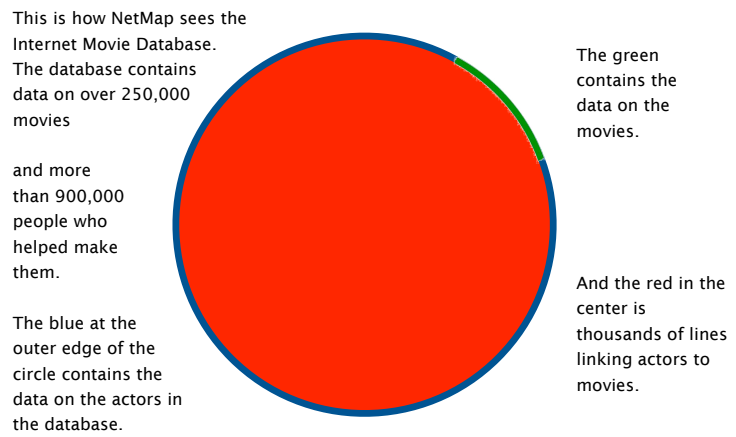
Now, let us add interactivity to permit exploration and we have *information visualisation*, succinctly defined by Card et al. in their book, subtitled "Using Vision to Think" [Card et al. 99]: "The use of computer-supported, interactive, visual representations of abstract data to amplify cognition"

However, what visual form do we use to represent the often complex, multi-dimensional abstract data so the user can make sense of it? And, how do we make this interactive so the user can explore the data, bearing in mind that the user often does not realise what information is hidden within?

These are some of the concerns of information visualisation designers who are faced with multifaceted tasks, as illustrated by the concept map in Figure 1-1.

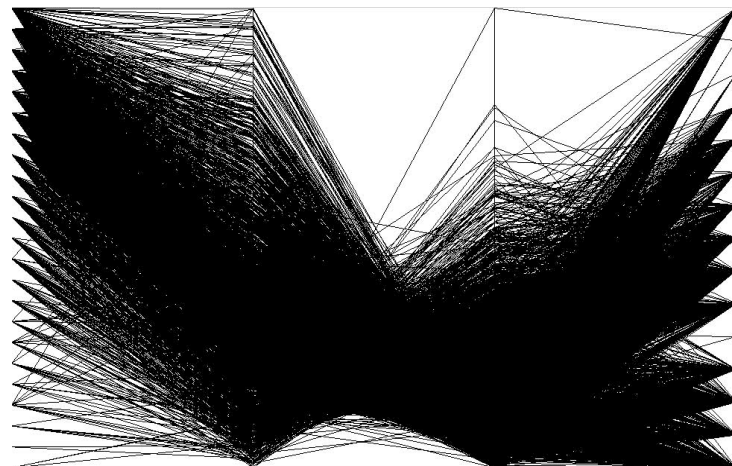
During the last two decades, a wide range of visualisations have been devised. We will meet some of the more enduring ones in Chapter 3.

**Figure 1-2a**



Example of overplotted visualisations - NetMap movie database [Netmap]

**Figure 1-2b**



Example of overplotted visualisations - Cars for sale

Desktop computers and their impressive graphics cards are increasingly powerful and so visual representations of large amounts of data can be created quickly enough to allow real-time interaction by the user, such as dynamic queries and zooming. However, with increasing computational power, large is rapidly becoming larger. A 10,000 record dataset could be considered as large by some visualisations, although in data mining, many millions of records are common. It is estimated that Wal-Mart's data warehouse is over 1000 Terabytes<sup>1</sup>. So how do we start to analyse the 800 billion IP packets per day transported on a section of the Internet backbone? [Keim 08].

The next section looks at the problems of visualising large datasets and some of the approaches that have been tried. Section 1.2 considers the use of randomness in computer science and data processing. Having exposed the main issues, the objectives of this work are stated in Section 1.3 followed by an outline of the approach taken, in Section 1.4. The novel characteristics of this work are then described, together with the contribution that have been made to the research area. Finally, the structure of the thesis is outlined in Section 1.7.

## 1.1. Visualising large datasets

Dealing with large datasets causes two main problems:

**visual limits** - the overwhelming number of items makes it difficult to comprehend the dataset due to perceptual or cognitive limitations of the user or hardware limitations of the display device.

**computational limits** - the amount of data is too great in terms of the necessary processing power, data storage or network traffic, especially when requiring interactive control of the visualisation.

Each of these problems will now be considered.

### 1.1.1. Visual limits

The visual limits of visualisations are apparent in many systems. For example, Figure 1-2a is from Netmap's Web site [Netmap], which illustrates the actor-film relationships in a large database of movies. On the circumference of the circle, 900,000 actors and 250,000 movies are drawn as points and red lines are drawn between each movie and the actors who appeared in it. The entire circle is solid red due to massive overplotting of lines, and of course some action is needed to reduce the number of actor-film relationships in order to identify any meaningful patterns. The second example, Figure 1-2b, is a parallel coordinate plot showing details of over 5000 cars advertised for sale on the Web. Although overplotting is not so extreme as the movie data

---

<sup>1</sup> <http://www.informationweek.com/news/storage/showArticle.jhtml?articleID=201203024> Aug 2007



example, much detail is hidden amongst the mass of lines.

Even with very high resolution displays, the human visual system can only resolve points at approximately 200dpi at normal viewing distance. However, taking into account the reduction in visual acuity due to low contrast and colour, we cannot normally reach this 200dpi level. So we have a limited number of pixels (approximately 2 million up to 10 million for advanced screens) to display the data and apart from space filling techniques (e.g. Keim's pixel spirals, TableLens, TreeMap) visualisations will result in overlapping data items as the number of items becomes large. Even with pixel plotting the number of data items is limited by the number of available pixels on the screen.

To make matters worse, the size of plotted points on say, a scatterplot, is usually much larger than 1 pixel, especially if using a glyph to represent one or more attributes, and each line of a parallel coordinate plot is drawn using many pixels. Not only does occlusion result in loss of data from view, but the adjacency of points, criss-crossing of lines, mixing of colours and shapes can also confuse the user and interfere with the process of building that all important mental model. This obstacle to understanding is often referred to as *visual clutter*.

So how do we reduce clutter? As with a cluttered desk, the simplest method is to remove some of the items. However, point and line visualisations rely on the human Gestalt visual system to extract trends and rules. We perceive wholes or patterns, rather than pieces or parts, but achieving the right density is critical - too few points and we see spurious connections or cease to see patterns just points, too many and the data becomes an amorphous blob. Of course, there is the question of which items to remove.

Other techniques include reducing the size of the data points to give each one more space, making the points semi-transparent so we become aware of any underlying data items, and clustering groups of similar points into a single point that effectively reduces the number of points. These all affect the appearance (or disappearance) of data items. Another possibility is to keep all the data items on the display but move them so they do not overlap or at least, overlap less. These are spatial distortion techniques. Alternatively, we can show the data items briefly in succession, so the user gets to see all the items but not all at once.

There have been few attempts to classify clutter reduction techniques, so part of this research work has been to devise such a classification. A criteria-based assessment of the techniques, resulting in the Clutter-reduction Taxonomy for information visualisation is presented in Chapter 3.



### 1.1.2. Computational limits

Along with the limits imposed by the resolution of display devices and human acuity, there are limits on the amount of data that can be dealt with by the computing hardware. The main players in this are the processing power of computational devices, the data storage and network bandwidth.

#### **processing power**

Some visualisation algorithms require substantial processing power. Plotting the points will take time proportional to the number,  $O(N)$  and likewise for simple processing such as filtering or calculating display attributes. However, if we want to do even moderately interesting things such as sort the data, this will take  $O(N\log N)$  time and more complex manipulations are likely to take times that rises quadratically,  $O(N^2)$ , or even exponentially. Whilst rapid interactive feedback may be obtained for small experimental datasets, scaling up to real datasets may take the application a prohibitive time to refresh, e.g. render a large number of lines to the screen.

As mentioned earlier, the sophistication and processing abilities of the graphics cards in current desktop and laptop computers are steadily increasing and, in raw processing power, often exceed the central processor in these machines. In addition to speeding up the rendering of the graphics from the basic machine, the graphics cards can be programmed directly from within an application using languages such as DirectX or OpenGL. This can enable interaction with far greater amounts of data items. Along with much faster drawing, inbuilt functions such as shading [Fekete and Plaisant 02], textures [Johansson et al. 06] and fog [Kosara et al. 02] have all been used effectively in clutter reduction and these are described in Chapter 2.

#### **data storage and network bandwidth**

For various reasons, such as the large amount of data or multi-user requirements, datasets are often held on remote database servers and hence the retrieval of the data is also dependent on the bandwidth and latency of the connecting network.

Sometimes it is possible to pre-compute meta-data and use this for visualisation, only retrieving detailed data on demand. However, even reduced meta-information may be too voluminous for very large datasets. Chalmers [Chalmers 99] points out that meta-data, such as index information for Web documents would be too great for normal storage systems, thus implying that meta-meta-data is required. Note that using a Web search engine effectively off-loads this storage problem in the same way that an SQL server does.





## 1.2. Uses of randomness

One possible way to address these visual and computing limits is random sampling<sup>2</sup>. We will first look at some examples of the effective use of randomness<sup>3</sup> in computer science and data processing, before questioning whether randomness can be used in clutter reduction.

### 1.2.1. Randomness in computing

Traditional algorithms are deterministic, attempting to find the unique or the best solution. In contrast, modern algorithmics (modern here really goes back at least 40 years), including neural networks, genetic algorithms and simulated annealing, makes heavy use of randomness. These algorithms are non-deterministic and find *a solution* rather than *the solution*, and *good* rather than *best*. Because of this more relaxed and inexact approach to solutions, these algorithms can tackle problems that are otherwise intractable, including NP-hard ones. Quality is traded for computation. In some cases, this is a simple cost-benefit trade-off, in others this is because the computation for the exact solution would be impossible. Further examples of the use of randomness include primality tests, spreadspectrum encoding techniques in wireless communications, telephone routing and parallel computing.

In some of the cases, randomness is used simply to reduce the computational effort – if one could do the calculation in full, it would be better but the random version is just *good enough*. However, in many cases, the randomness is essential otherwise the system would be worse in terms of performance. For example, RSA public key encryption requires the selection of two prime numbers, with larger numbers offering higher security. To check if a 128-bit number is a prime number by simple division would take approximately 3000 years (i.e.  $10^{20}$  divisions at 1 Gigafllops) whereas a simple probabilistic primality test such as the Fermat primality test provides a workable solution.

### 1.2.2. Randomness for data processing

Several commercial statistical and data mining applications refer to the use of random sampling. For example, Statistica<sup>4</sup> mentions using sampling to speed up processing

---

<sup>2</sup> “In statistics, a simple random sample is a subset of individuals (a sample) chosen from a larger set (a population). Each individual is chosen randomly and entirely by chance, such that each individual has the same probability of being chosen at any stage during the sampling process, and each subset of k individuals has the same probability of being chosen for the sample as any other subset of k individuals”. Yates, D.S., Moore, D.S., Starnes, D.S. The Practice of Statistics, 3rd Ed. Freeman. 2008

<sup>3</sup> “of or characterizing a process of selection in which each item of a set has an equal probability of being chosen”. Dictionary.com Unabridged (v 1.1). Random House, Inc.  
<http://dictionary.reference.com/browse/randomness> – accessed Oct 2008

<sup>4</sup> <http://www.statsoft.com>



and SAS Enterprise Miner<sup>5</sup> notes the use of sampling for predictive modeling.

There has been considerable research in the database literature, since the advent of data warehousing and related data mining application, on the use of sampling in connection with query optimisation. The cost of executing ad-hoc queries on very large databases is considerable, hence the ability to calculate an approximate answer based on a random sample from the database is desirable. Probabilistic counting techniques [Shah and Ramachandran 04] are good at estimating the size of multi-sets and techniques exist to help determine appropriate sample sizes to mine [Domingo et al. 02]. Another area of research is attempting to compensate for lost objects, size and structural distortions within a sample [Breunig et al. 01].

Clustering is used in data mining to help discover distributions and patterns in the underlying data. Techniques that use random sampling have been demonstrated to be efficient and accurate even for very large databases [Guha et al. 98, Kollios et al. 03, Palmer and Faloutsos 00]. Efficient strategies have been developed for single joins [Chaudhuri et al. 99] and also for some aggregate queries [Chaudhuri et al. 01].

### 1.3. Random sampling for clutter reduction

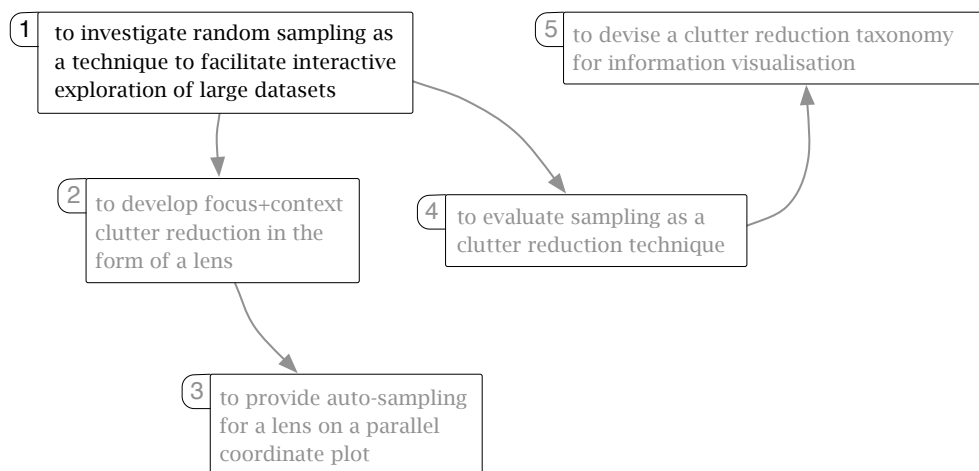
This thesis is about using random sampling to reduce the size of the dataset, either under user control or automatically that results in a reduction in the display clutter, thus providing users with a better understanding of the data. When the idea of using dynamic random sampling for clutter reduction was proposed [Dix and Ellis 02, Ellis and Dix 02] there were no visualisations at the time that utilised such a mechanism. The proposals sparked interest in this area. Since then, two applications, [Bertini and Santucci 06, Rafiei and Curial 05] have applied random sampling as part of the interaction process (see Section 2.3).

Given the success of randomness in other areas of computer science, there is little use of sampling within existing visualisation algorithms. There are examples of randomness being used in learning algorithms, sampling large or infinite data spaces and for pre-processing datasets to a manageable size, and these will be considered in Section 2.3. But little work has been published on its use in more interactive visualisations.

It is surprising that sampling has not been used more widely, despite very large datasets being regarded as problematic as already discussed in Section 1.1. For instance in other realms, if one wants to determine some aspect of real world data, it is normal to capture only a sample. For example, if we wish to discover if cats really do prefer Fishy Bytes to some other ordinary food, we do not have to tempt all the cat

---

<sup>5</sup> <http://www.sas.com/products/miner>



**Figure 1-3a** Relationship between objective 1 and the other objectives of this research

population in the UK with morsels of Fishy Bytes – a relatively very small sample of cats will do. So why is it so rare to find sampling used when visualising very large datasets?

The data being visualised is often itself sampled from the real world and so, re-sampling to reduce the dataset size to manageable proportions appears to be a natural extension of this external sampling. However, it seems that throwing away of information that is inherent in internal sampling from stored databases just seems unnatural to those trained to get the most out of limited data. Although the perceived reluctance of users to discard data was not investigated in the thesis, we can speculate that allowing users to control the size of the sample while they are engaged in data exploration might increase their confidence. In fact, as we will see later, the ability to dynamically alter the sample size helps in identifying structures within the data.

Randomness is an integral part of this work. Not only has dynamic random sampling been incorporated into a visualisation to **reduce clutter without prejudgment**, it has been used in the measurement of overlapping line density, as we will see in Chapter 5.

## 1.4. Approach and Objectives

The focus of this work is two-pronged:

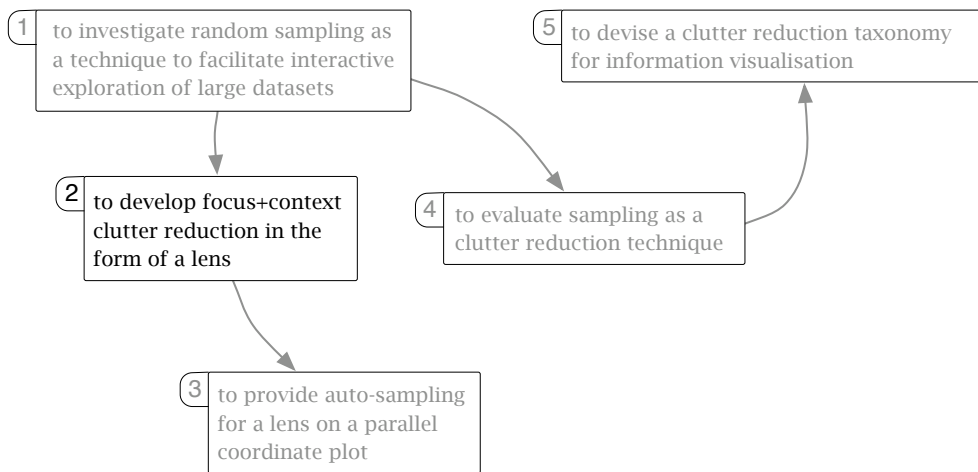
- ▶ to investigate random sampling as an interactive clutter reduction technique, and
- ▶ to devise a classification of clutter reduction technique that will be useful to designers of information visualisations.

The approach undertaken is based around exploratory prototyping backed up by algorithmic experimentation, which generated some of the objectives of this research. The discussion below highlights the methods employed in this thesis to achieve the objectives. Figures 1-3a to 1-3e illustrates how the different objectives interconnect.

The initial objective originated from the early work of the author and Alan Dix where the concept of using sampling as a clutter reduction technique was pioneered [Dix and Ellis 02, Ellis and Dix 02]. A novel 2D zooming interface was proposed, the Astral Telescope Visualiser, as well as a suggestion for a sampling database architecture. Various general usability and technical issues were raised. Since then several researchers, notably Derthick et al. [Derthick et al. 03], Bertini and Santucci [Bertini and Santucci 04], Rafiei and Curial [Rafiei and Curial 05] and Cui et al. [Cui et al. 06] have acknowledged the usefulness of sampling and applied this in their own work.

The first objective of this work was therefore:

- 1 to investigate random sampling as a technique to facilitate interactive exploration of large datasets



**Figure 1-3b** Relationship between objective 2 and the other objectives of this research

In order to show the application of sampling in clutter reduction, an existing Java-based visualisation toolkit was extended and a scatterplot visualisation was developed. One of the main requirements was to devise a method for generating the data samples, while maintaining data integrity and offering the ability to rapidly re-sample. This was met through the development of prototypes based on the z-index concept [Ellis and Dix 02] (as described in Chapter 2). A comparison was then undertaken between sampling and other basic clutter reduction techniques that were available in the chosen visualisation toolkit.

Parallel coordinates was selected as a suitable visualisation technique, to investigate if sampling could be usefully incorporated into different types of visualisations. This was due to various reasons, first, parallel coordinates is multi-dimensional and hence very different to a scatterplot; second, it is commonly used and tends to generate overcrowded displays, even with moderately sized datasets; and finally, its development involved minimal changes to the sampling code written for scatterplots. Visualisations such as Treemaps were also considered as discussed in Sections 2.4.3 and 6.5.

Through reflection on personal use of the initial proof of concept sampling application, it was noted that the majority of the plots, either scatterplot or parallel coordinates, exhibited a wide range of point or line densities. Non-uniform sampling was considered, however, a focus+context approach was thought to be more appropriate after reviewing existing published work on toolglasses and *magic lenses*. Thus, objective 2 was generated.

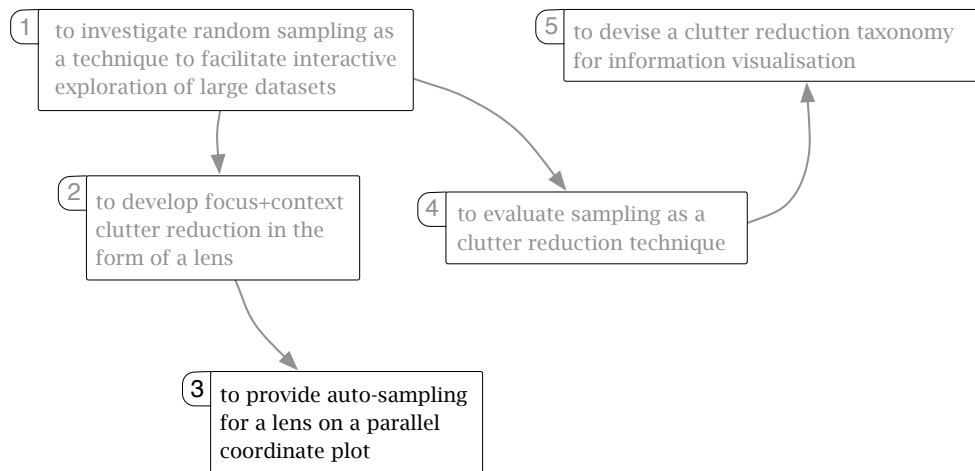
2 to develop focus+context clutter reduction in the form of a lens

In order to address this objective, a moveable lens with its own sampling control was designed and developed. The resulting application was called the Sampling Lens, and it will be referred to as such throughout this thesis, including any of the subsequent versions of the lens-based sampling visualisation.

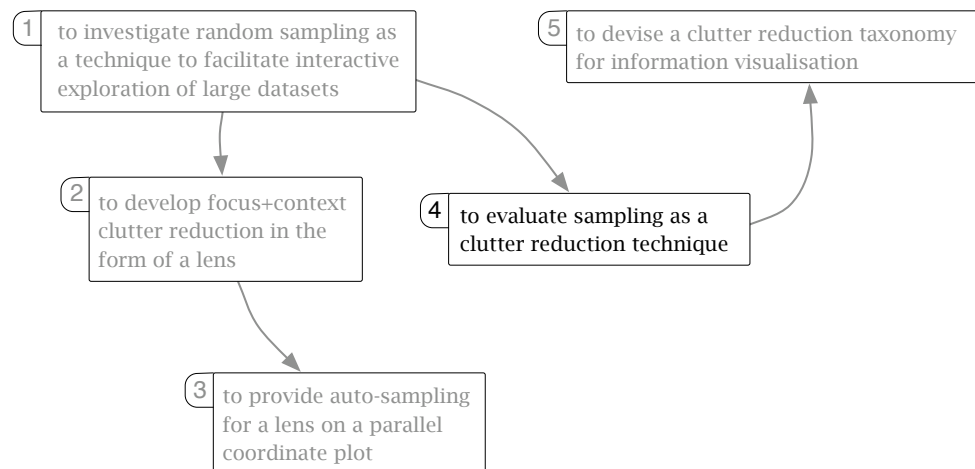
Manual adjustment of the sampling rate within the lens proved to be undesirable and hence the provision of auto-sampling was investigated, where the system adjusts the sampling to give a reasonable uncluttered view. This relies on an effective measure of occlusion within the lens area. Calculating the number of overlapping points on a scatterplot was reasonably straightforward and an occlusion measure was determined. However, an occlusion measure for a set of overlapping lines, as found in parallel coordinates, did not feature in the literature<sup>6</sup> and prompted an experimental

---

<sup>6</sup> Algorithms and metrics for edge crossings are utilised in graph drawing (e.g. [DiBattista et al. 98] [Purchase 02]) however, they focus on minimising edge crossing rather than display clutter.



**Figure 1-3c** Relationship between objective 3 and the other objectives of this research



**Figure 1-3d** Relationship between objective 4 and the other objectives of this research



investigation to determine an appropriate metric. Thus, objective 3 was generated.

3 to provide auto-sampling for a lens on a parallel coordinate plot

The pursuit of this objective involved undertaking a large number of experiments. In light of this, the Sampling Lens was instrumented to collect, process and output experimental data, and various software tools were created to set appropriate parameters, run the experiments, and visualise the density within the lens. Three different occlusion measures were compared and a theoretical model was developed to demonstrate the relationship between the seemingly diverse metrics.

An occlusion metric was selected and three very different methods for calculating the occlusion value were tested to determine the most appropriate, based on accuracy and efficiency. Due to past experience of using binomial approximations to obtain order of magnitude ideas of behaviour [Ellis and Dix 04a and 04b], probabilistic models were developed for two of the methods for calculating occlusion. A range of experiments were conducted to compare the occlusion methods and further prototypes were developed to speed up the occlusion calculation. Various problem cases were identified, in terms of patterns of lines across the lens, which were subsequently investigated further and a successful solution was devised.

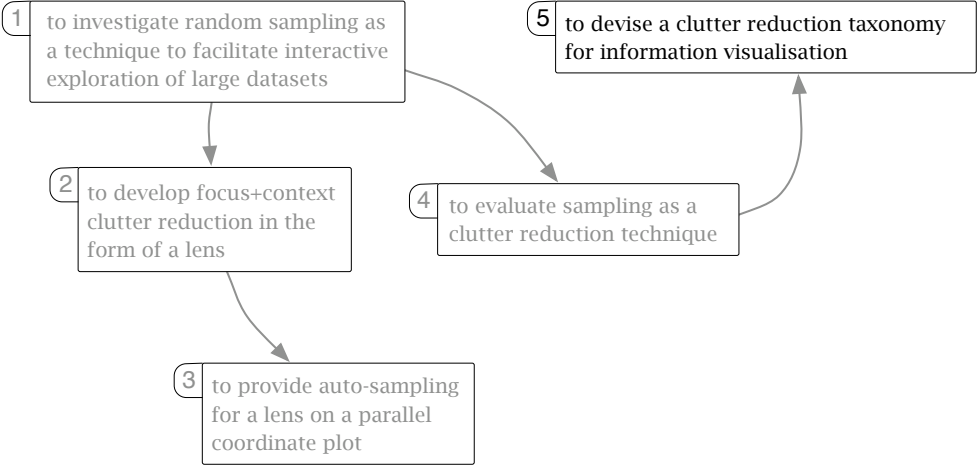
Although objective 3 was met through the creation of an auto-sampling lens, the interactive performance was disappointing, which capped the size of datasets and impeded the ability to try out novel features. A new version of the Sampling Lens, leveraging the advantages of OpenGL, permitted larger datasets to be explored and provided the opportunity to study a broader design space. Subsequently, a range of novel parallel coordinate lenses were implemented, some animated re-sampling transitions were investigated and the Astral Visualiser was realised.

The need to assess sampling as a clutter reduction techniques led to objective 4.

4 to evaluate sampling as a clutter reduction technique

A review of literature in the information visualisation domain was undertaken, which suggested that user-based evaluation was particularly difficult. However, little guidance was available on overcoming these problems. A literature survey of authors describing their information visualisations was carried out, with the expectation of learning from their experiences of user studies. However, this exercise did not produce the guidance sought and in fact the rationale behind of many of the user evaluations was questioned. Therefore, an alternative strategy for evaluating sampling was sought and objective 5 was generated.

5 to devise a clutter reduction taxonomy for information visualisation



**Figure 1-3e** Relationship between objective 5 and the other objectives of this research

This part of the work, involved an analytical criterion-based assessment of a wide range of clutter reduction techniques, supported by a detailed review of the literature. This enabled the previous objective to be met and sampling as a clutter reduction technique was then evaluated, albeit not with users.

## **1.5. Novel characteristics of the work**

Prior to the preliminary papers [Dix and Ellis 02, Ellis and Dix 02] there was no dynamic control for sampling within information visualisations, hence the application of random sampling to clutter reduction is innovative. This has led to a systematic exploration and analysis of the highly scalable sampling technique, in particular for scatterplot and parallel coordinate plots.

An effective approach to generating random samples, the z-index method (Sections 2.2 and 4.1), has been devised that not only ensures the display continuity necessary for interactive clutter reduction but provides a means for the user to check the reality of perceived artefacts.

Sampling has also been applied as a lens-based focus+context technique with automatic sampling rate adjustment. A noteworthy aspect of this work has been the creation of a metric and accompanying method for calculating occlusion in parallel coordinate plots (Chapter 5). Moreover, the metric has a strong theoretical underpinning as evidenced by a theoretical model developed by the author and supervisor.

Although the classification of clutter reduction techniques was initially devised as a method for evaluating the sampling-based approach, its scope broadened. The outcome is the Clutter-reduction Taxonomy (Section 3.4), a novel criteria-based taxonomy of clutter reduction for information visualisation, which visualisation designers can use to critique existing visualisations and inform new ones. The literature-based analytical approach used in the construction of the taxonomy is also a novel feature of this work.

	Publication	Objective	Chapter
1	Dix, A., Ellis, G.P. "by chance: enhancing interaction with large data sets through statistical sampling". <i>Proc. AVI'02</i> , L'Aquila, Italy, May 2002, ACM Press, pp. 167-176	1 (sampling)	2
2	Ellis, G.P., Dix, A. "Density control through random sampling : an architectural perspective". <i>Proc. IV'02</i> , London, July 2002, IEEE, pp.82-90	1 (sampling)	2
3	Ellis, G.P., Bertini, E., Dix, A. "The Sampling Lens: Making Sense of Saturated Visualisations". <i>Proc. CHI'05 Extended Abstracts</i> , Portland, USA, 2005, ACM Press, pp.1351-1354	2 (lens)	4
4	Ellis, G.P., Dix, A. "An Explorative Analysis of User Evaluation Studies in Information Visualisation". <i>Proc. BELIV'06, AVI Workshop, Beyond Time and Errors: Novel Evaluation Methods for Information Visualization</i> , Venice, Italy, May 2006, ACM Press, pp.1-7	4 (evaluation)	6 Appendix F
5	Ellis, G.P., Dix, A. "the plot, the clutter, the sampling and its lens: occlusion measures for automatic clutter reduction". <i>Proc. AVI'06</i> , Venice, Italy, May 2006, ACM Press, pp.266-269	3 (auto-sampling)	5
6	Ellis, G.P., Dix, A. "Enabling Automatic Clutter Reduction in Parallel Coordinate Plots". <i>Trans. Visualization and Computer Graphics</i> , 12(5), Sept 2006, pp.717-723	3 (auto-sampling)	5
7	Ellis, G.P., Dix, A. "A Taxonomy of Clutter Reduction for Information Visualisation". <i>Trans. Visualization and Computer Graphics</i> , 13(6), Nov 2007, pp.1216-1223	5 (taxonomy)	3

**Table 1-1**

Contribution to the research area

## 1.6. Contribution to the research area

The work presented in this thesis led to various conference and workshop presentations as well as publications in proceedings and journals, thus making an important contribution to the information visualisation research area. The list of publications is given in Table 1-1 and the reference numbers given below relate to the number in the first column. The major objective covered by each publication is given (see Section 1.4 for details) and the principal chapters of this thesis containing the published work are also stated. An overview of each chapter is given in the next section.

The proposal that random sampling can be used to make the visualisation of large datasets more computationally efficient and more perceptually effective was presented as a full paper at AVI'02 in L'Aquila, Italy [1].

Issues relating to the use of random sampling in display density reduction, including the provision of a sampling database were presented as a full paper at IV'02 in London [2].

The initial sampling-based visualisation, the Sampling Lens, described in Chapter 4, was published as a short paper and presented as a poster at CHI'04 in Portland, Oregon [3].

The investigation into finding an occlusion measure for lines in a parallel coordinate plot, necessary for auto-sampling was published as a short paper and presented as a poster at AVI'06 in Venice [5].

The ensuing work in determining an effective and efficient method of calculating the aforementioned occlusion metric was presented as a full paper at InfoVis'06 in Baltimore, Maryland and published in the IEEE Trans. on Visualization and Computer Graphics [6].

The taxonomy of clutter reduction for information visualisation was presented as a full paper at InfoVis'07 in Sacramento, CA and published in the IEEE Trans. on Visualization and Computer Graphics [7].

Finally, the explorative analysis of user evaluation methods in information visualisation was presented as a full paper at the BELIV'06 workshop in Venice and published by the ACM Press [4].



## 1.7. Structure of the thesis

**Chapter 2** investigates the use of random sampling to reduce clutter in overcrowded displays. Issues pertinent to sampling are raised by considering sampling-based star gazing and a solution to many of these issues is proposed through the z-index method. Three visualisations that use sampling in different ways are then discussed. Relevant statistical sampling methods are examined before considering current database support for sampling. The sampling-based visualisation proposed in this chapter leads to the design, implementation and experimental phase of this work that begins in Chapter 4.

**Chapter 3** presents the Clutter-reduction Taxonomy for information visualisation describing the novel method used in its construction. A review of classification schemes for information visualisation is presented, highlighting two schemes that relate specifically to clutter reduction. The clutter reduction techniques and criteria used in the taxonomy are described and the taxonomy table and accompanying discussion notes are presented. The utility of the taxonomy is demonstrated through several examples of its use to critique existing visualisations and propose new ones. The taxonomy is compared to the two clutter reduction classifications and the importance of criteria is illustrated.

**Chapter 4** documents the development of the first sampling-based scatterplot and parallel coordinate visualisations. The effectiveness of sampling in dynamic clutter reduction is demonstrated and a comparison is made to three other techniques - change opacity, change point size and filtering. A lens-based sampling visualisation is implemented to provide a focus+context solution to large overplotting density variations across a plot. The z-index method for generating lens samples is successfully adapted for a lens and also for re-sampling. The requirement for automatic adjustment of the lens sampling rate is identified and its pursuit and resolution is described in Chapter 5.

**Chapter 5** describes the work undertaken to facilitate an auto-sampling lens for parallel coordinates. An occlusion metric is first defined and justified through a series of practical experiments and a theoretical model. Three very different methods for calculating occlusion are then assessed for accuracy and efficiency by means of an extensive empirical study. The proposed solution is both very efficient and accurate, which is counterintuitive given its theoretical underpinnings.

**Chapter 6** presents an evaluation of sampling and consider the use sampling with different visualisations. We reflect on the difficulties of undertaking effective user studies of information visualisations and assess the objectivity of criteria-based evaluation of sampling. Sampling is compared to other clutter reduction techniques





using the Clutter-reduction Taxonomy (Chapter 3). Further exploration of scatterplot visualisations using sampling consolidate the use of global and lens-based sampling, and the constant-density interface proposed in Chapter 2. This chapter also reflects on the Sampling Lens application, the lessons learnt about sampling as a clutter reduction technique and explores the integration of sampling into visualisations other than the scatterplot and parallel coordinates.

**Chapter 7** reflects on the main issues raised by each chapter and their resolution or outcomes. It also summarises how the objectives of this work were met and their outcomes are noted. Finally, some future directions of sampling are considered with suggestions for further work.

**Appendix A** presents examples of the clutter reduction techniques used in the Clutter-reduction Taxonomy. The focus is on how each technique manipulates attributes such as position, visual, association and temporal to reduce display clutter.

**Appendix B** describes the datasets used in this work.

**Appendix C** presents details of the experiments in Chapter 5.

**Appendix D** examines various issues related to the implementation of the sampling-based visualisation, including the software instrumentation devised to carry out the empirical studies, an overview of the visualisation toolkit and the development of an OpenGL version of the Sampling Lens that improved the interactive performance considerably.

**Appendix E** reproduces the original data from a comparison of natural building techniques for walls by M.G.Smith as referenced in Section 3.5.5

**Appendix F** reproduces the BELIV'06 (Beyond time and errors: novel evaluation methods for Information Visualization) workshop paper.



## Chapter 2

# Sampling as a clutter reduction technique

The simplest way to reduce clutter in overcrowded displays is to remove some data items. Which data items should we remove? We can achieve this by filtering, but this presupposes that the user knows what particular data is unimportant and hence can be left out. On the other hand, taking a random sample of the data does not require such a judgement to be made. Furthermore, with a very large dataset, even if we know what we are looking for and filter accordingly, we may still have an overcrowded display (discussed further in Chapter 4). In adopting randomness, we are in some way, solving an impossible problem - the inability to see anything useful.

As suggested in Chapter 1, users may be reluctant to throw away data, but by engaging them in the exploration process and allowing them control over the size of the sample, it may be possible to reassure them. Incorporating dynamic random sampling in a visualisation to reduce clutter shows promise. Although it may seem a fairly obvious solution, when this work began, random sampling had only been applied as a pre-processing operation and did not form part of an interactive clutter reduction process.

This chapter will investigate the issues surrounding the use of sampling as a clutter reduction technique. Section 2.1 discusses the ideas behind a proposed sampling-based application, the Astral Visualiser, which not only allows the user to control the degree of sampling, but also introduces the idea of assisting the user by maintaining the chosen density automatically when zooming. This exercise raises various important points relevant to the design and implementation of a sampling-based visualisation and these are subsequently discussed. Section 2.2 describes the z-index method, a proposed solution to meet the requirements generated in the previous section, such as display continuity and Reality Check.

Section 2.3 describes some examples of the use of randomness in visualisation before reviewing three interactive visualisations which utilise random sampling in various ways. As well as demonstrating clutter reduction, some of these example visualisations highlight particular issues relating to the structure and distribution of data which are subsequently discussed in Section 2.4. An awareness of statistical sampling methods appropriate to various data distributions and the requirement of the visualisation are also addressed by this section.

Section 2.5 considers various issues relating to sampling from databases and reviews database management system support for sampling. Finally, we reflect on the likely

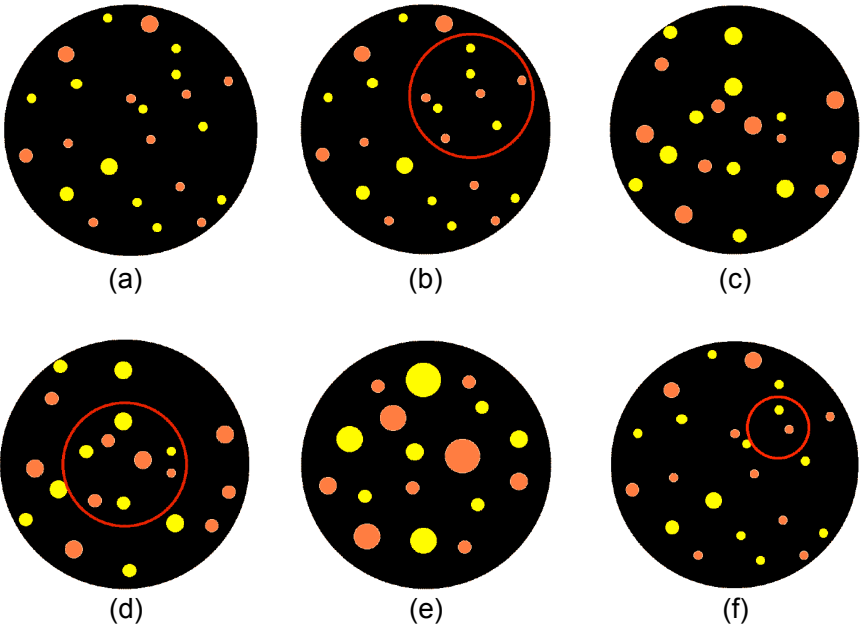


Figure 2-1

Views of the *stars* through a zoomable telescope as the magnification increases. Note that (a) and (f) are the same view.

benefits of sampling as a clutter reduction technique.

## 2.1. The Astral Telescope Visualiser

We will now start by considering a scatterplot with a sampling control to adjust the plot density. We will use the analogy of observing stars through a telescope with a zoom lens<sup>1</sup> to present a visualisation that shows how the sampling rate might be adjusted automatically when zooming in to a region of the plot.

In a scatterplot it is normal to be able to zoom into a portion of the data, but this changes the density of points. This may be an advantage in an overcrowded display as less points will be seen as the selected region is magnified, but if the overcrowding has been reduced already by sampling then the user will be missing out on the data which is present in the dataset but is not within the original sample. It is possible for the user to re-adjust the sampling rate manually to again produce an acceptable density, but given the relationship between magnification and density we can automatically compensate for the changing zoom factor by re-sampling the data and adding points.

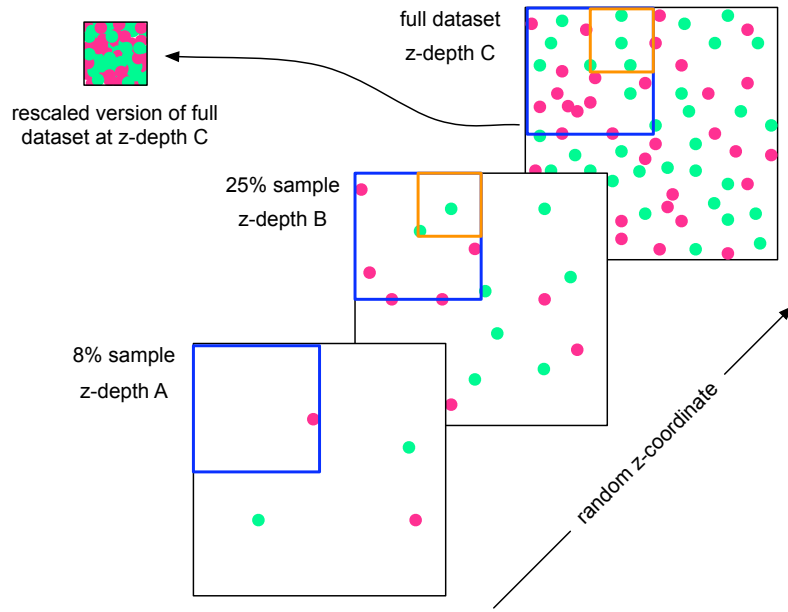
This form of dynamic re-sampling was realised in a visualisation technique based on the metaphor of a star-gazer's telescope. As you look at the night sky with your naked eye, you see a small number of stars. If you then take a telescope and look in the same area of the sky, you not only see the same constellation of stars brighter and bigger, but also more stars appear that were previously too dim to see. If you increase the magnification of the telescope, yet more stars appear as the field of view shrinks. This is illustrated in Figure 2-1. The initial view through the telescope is shown in (a) with (c) showing a x2 magnified view. The corresponding region of the sky is shown in (b) by a red circle. The next two diagrams (d) and (e) show a further x2 zoom. The last diagram (f) shows the region of the un-zoomed view, which is the same as view (a), with the addition of a red circle which represents the x4 magnification view to its left (e).

The Astral Telescope Visualiser works in a very similar fashion, except that the size of the plotted points are not magnified when zooming in. Assume that we have a large dataset which results in an overcrowded scatterplot. The dataset is sampled under the control of the user to give an acceptable display density. The user can select an area of interest. As Astral zooms in, it samples more records in that region of the display. The sample is chosen so that the density of sampling increases with the square of the zoom value, this means that the actual visible density remains approximately constant, provided that all the data items are not already on view.

---

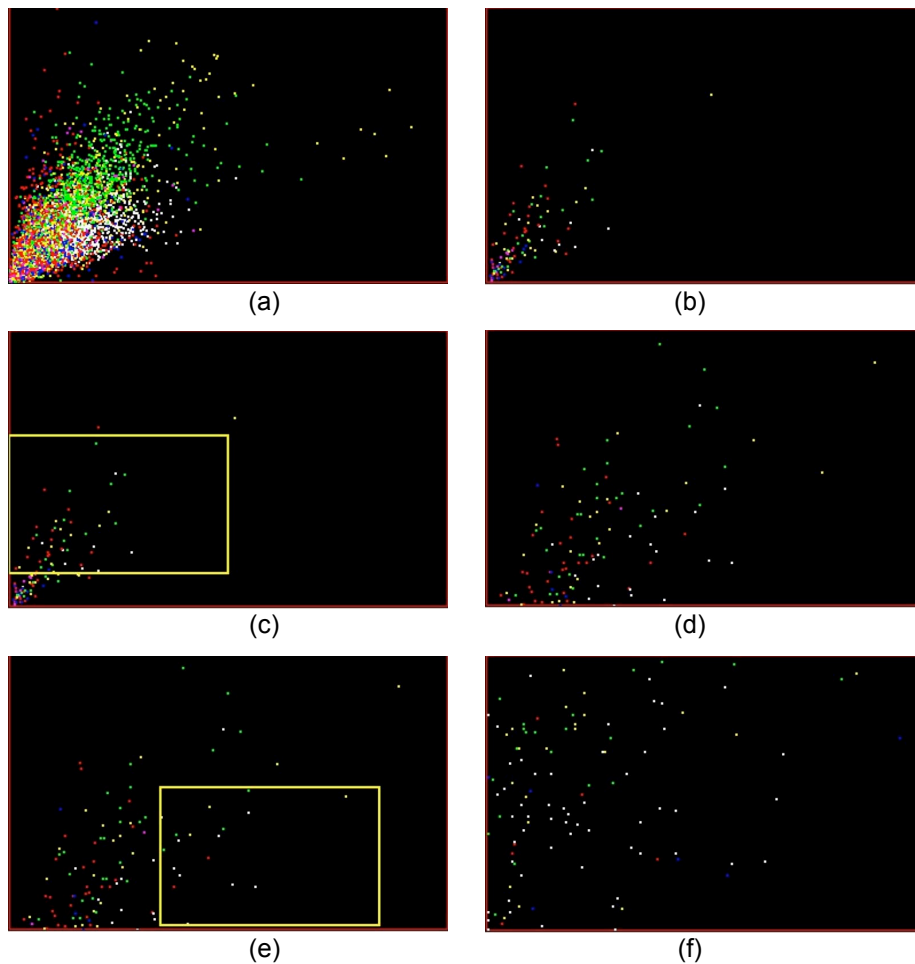
<sup>1</sup> Telescopes do not usually have adjustable magnification, apart from changing eyepieces, so this is more like a camera lens with optical zoom.

Figure 2-2



Model behind the Astral Telescope Visualiser

Figure 2-3



Example of zooming in to a scatterplot with the Astral Telescope Visualiser prototype, demonstrating automatic adjustment of the sampling rate. (a) original view, (b) 1.2% sample reduces the clutter, (c) user selected region of interest, (d) zoomed in view of selected region with extra points added to maintain density, (e) & (f) user zooms in again and sampling rate is adjusted to maintain density

One way of thinking about this, is that we determine the x and y coordinates from the attributes, but randomly allocate a z-coordinate and then determine how far away we see by the current zoom factor. Figure 2-2 shows this model. For illustration purposes there are a small number of points and only three discrete z-depths to which the user zooms. In the real system the z values are uniformly distributed throughout the depth and the user can zoom to any amount. Initially the user sees all the points resulting in



a very overcrowded display, thus



decides to apply sampling to reduce the clutter

in the model (Figure 2-2), the first z-plane is visible, z-depth A, nearest the viewer. The user then selects the top left hand corner, shown in blue. As the system zooms in, extra points are sampled from this region; consequently the points from the second layer, z-depth B, become visible (this includes the points on the first layer) and results



in . The overall sampling rate is now 25%. Finally the user selects and zooms into the top right, shown in orange. The points in this area are re-sampled and in this example, we are now at a sampling rate of 100% (i.e. the full dataset as z-depth C) and



the user sees . Note that if the user zoomed in further, no additional data points would appear.

A simple prototype application of the Astral Visualiser was built and screenshots illustrating the automatic adjustment of the sampling rate are shown in Figure 2-3. The first screenshot (a) shows the full 9309 data points and (b) shows the screen after the user has adjusted the sampling control (a 1.2% sample). To see more detail, the user zooms in to the region marked with a yellow box in (c). The visualiser algorithm automatically adjusts the sampling rate to give additional points as shown in (d), hence maintaining a constant overall density. With reference to the model in Figure 2-2, the z-coordinate has been moved back to include the extra points, although some of the extra points will be outside the new field of view. Note that all the points in the previous view are still there. The last two screenshots in Figure 2-3, (e) and (f), show the user selecting another region of the scatterplot and zooming in further, to reveal more detail. The user can continue to zoom in, although as noted above, when the z-coordinate is at its maximum the full dataset will be visible, but in a small region. Also, the user has full control on the density of the data points shown as the sampling rate can be altered at any time to suit their purpose.





### **2.1.1. Sampling issues**

The Astral Visualiser raises a number of issues related to the interaction between the user and the system, which will now be discussed. The random effects of sampling may potentially be confusing when interacting with a visualisation, hence proper attention should be given to the finer details of interaction and sampling.

#### **zoom in**

The metaphor of the Astral Visualiser implies that in order to zoom in, for example, by a factor of 2, we need to increase the sampling density by a factor of 4. Generating a given number of new points is not a problem and neither is the distribution, as they are randomly chosen. However, we must decide between producing a totally new sample of data points and adding points to the existing sample. Clearly, we need an appropriate sampling method but we must also consider the user's perception of the changes in the display. Research has shown [Vinson 99] that users require landmarks when navigating to develop their spatial knowledge as quickly as possible. So, this suggests that adding points would be the better solution as the existing sample points could act as landmarks.

#### **zoom out**

Zooming out also needs careful consideration on how to remove the extra re-sampled data from the previous zooming in operation. The previous data can be reinstated or a new sample can be generated. The former solution will have the overhead of remembering the data at each level whilst the latter solution will not maintain the continuity of the display.

#### **panning**

Maintaining display continuity is also important when panning a zoomed-in display, as in Figure 2-3d (similar to moving a telescope across the sky), so that moving back to a previously viewed location gives the same points.

#### **appearance of new points**

Another issue is whether the new points should be displayed differently during zooming in. In the telescope example (Figure 2-1), the old points are drawn bigger and although this has been exaggerated in the diagram, it is intended to add to the sense of zooming into a starfield. Instead of using size, other visual attributes can be altered such as brightness, colour, or animated effects such as sparkle or shimmer. It is known [Nakayama and Silverman 86] that the latter are good candidates for pre-attentive visual processing so would be easily seen by the user.

#### **density control**

Similar problems encountered with zooming need to be considered when the user changes the sampling rate. Presenting a new sample of the dataset every time the

**Table 2-1**

z-index	data
252	data Item 1
971	data Item 2
41	data Item 3
537	data Item 4
827	data Item 5
511	data Item 6
188	data Item 7
265	data Item 8
363	data Item 9
10	data Item 10
88	data Item 11
900	data Item 12
584	data Item 13
~	~
310	data Item 999
288	data Item 1000

z-index	data
1	data Item 420
2	data Item 626
3	data Item 910
4	data Item 319
5	data Item 514
6	data Item 543
7	data Item 723
8	data Item 850
9	data Item 905
10	data Item 10
11	data Item 283
12	data Item 212
13	data Item 631
~	~
999	data Item 857
1000	data Item 895

Example of a data table with the addition of a z-index to facilitate random sampling. (a) original data table (b) sorted by z-index

control is moved, resulting in a rapidly changing display would no doubt be most confusing to the user.

### **artefacts and coincidence**

It is easy to look at random data and see patterns in it that are simply artefacts of the sampling process or coincidental alignments such as lines or clusters. But how can the user know whether these patterns are in fact features or trends? A solution, which we will refer to as the Reality Check, is described in Section 2.2.2.

### **re-sampling**

Re-sampling may also take time, which implies either slowing down interaction or perhaps adding the new data as it is sampled. With the Astral Visualiser the stars could appear gradually over time as we zoom in. Such effects that expose the underlying sampling, may maintain user awareness of the statistical nature of the visualisation.

## **2.2. The z-index method**

To deal with the issues of sampling raised by the Astral Visualiser a possible solution that provides the required functionality and performance is the z-index method.

In the Astral Visualiser, data items are assigned a randomly chosen depth to control which data items appear when the user changes the telescope magnification. The z-index method is based on a random index column, which if populated with randomly chosen values between 1 and the number of data items, then only requires a simple query to select the data items to display. For example, taking a 1000 record dataset, a z-index query retrieving records 1 to 100 will give a random 10% sample of the data. To increase the sample by a factor of 4 (x2 zoom) only needs a query to retrieve records 1 to 400. Part of an example 1000 record table is shown in Table 2-1.

The use of a z-index should be set against the initial cost of pre-processing the z-index information, but experience with sampling from standard SQL databases shows that this is often necessary. It should be pointed out that the notion of a z-dimension is used in visualisation tools such as DataSplash [Woodruff et al. 98b], which employ semantic zoom techniques; however, in these instances, z refers to a layer with different visual attributes (e.g. a point becomes a picture of a tree when viewed at a lower level) rather than a sampling attribute, as is proposed here.

### **2.2.1. Display continuity**

Many of the issues discussed in Section 2.1.1, mentioned the maintenance of display continuity. For instance, when adding new points (zooming in on the Astral Visualiser), these should be added in the reverse order to which they were removed (when reducing the density). As noted, this requires a history of which points were removed,



however, the order of the points is in fact fixed by the z-index value and hence this functionality is provided for free. Therefore, it is easy to ensure continuity if zooming out to an unvisited magnification and when panning. Density control is again simply a matter of moving the top z-index query limit.

### 2.2.2. Reality Check

In Section 2.1.1 it was suggested that the user would gain confidence in the visualisation sample if they could easily gauge whether a pattern is real or just an artefact of the sampling process. The Reality Check achieves this by adding a function that re-samples the data, displaying a new set of points<sup>2</sup>. So, when the user sees a potential pattern in the data, they can initiate a Reality Check (for example by pressing a button on the interface) and see if the pattern persists. In fact, a standard statistical technique is to perform some sort of calculation on a sub-sample of data and then compare this with a similar calculation on another sub-sample. This gives an indication of the robustness of the statistic. Likewise, in machine learning and neural networks it is common to train on a sub-sample and then test the learnt rules on another sub-sample.

In addition to allowing the user to control the sample size, mentioned at the beginning of this chapter, the Reality Check is another way of reassuring the user that sampling is a viable method of clutter reduction.

In terms of the z-index method, a Reality Check corresponds to selecting a fresh z-index or querying a new section of the z-index (e.g. instead of selecting 1 to 100, select 101 to 201).

## 2.3. Randomness in visualisation

As mentioned in Chapter 1, randomness is helpful in resolving particular problems in the field of visualisation. So far, random sampling has only been considered as selecting, purely at random, a certain proportion of data items from the full dataset. However, the distribution of the data can have a marked effect on the validity of the sample and other selection algorithms might need to be considered. Examples of the use of sampling in visualisations are now described, some of which draw attention to particular distribution of data issues that will be discussed in Section 2.4.

We will first consider some examples where randomness has been used in learning algorithms, to sample a non-uniform space and to reduce the size of the dataset. We will then focus on three other visualisations that utilise random sampling in a more interactive manner, two of which are directly concerned with clutter reduction.

---

<sup>2</sup> A completely new set of data items will only be given if the size of the sample is less than or equal to half the size of the dataset.



**sampling a non-uniform space** - An interesting application of randomisation techniques has been applied to the analysis of spatial data from archaeological studies [Kvamme 96]. The problem there was to overcome the often non-random distribution of data to see statistically valid patterns. Inferential solutions using standard methods were intractable, yet random sampling gave useful approximations.

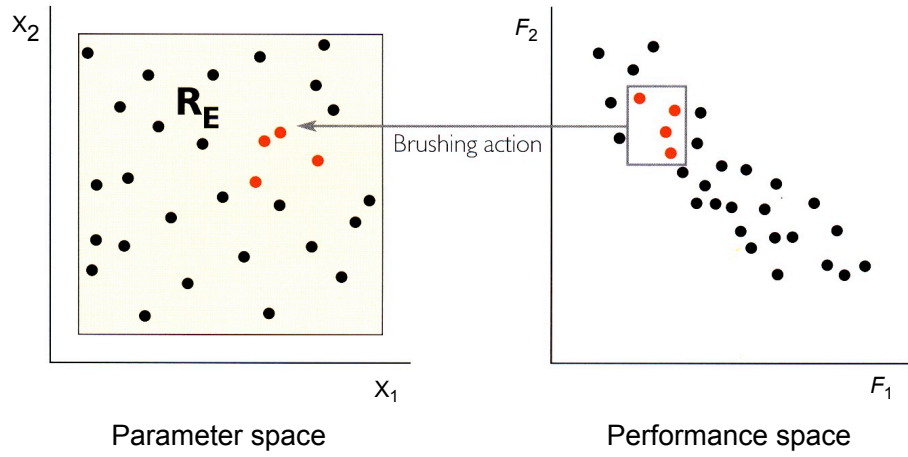
**reducing size of dataset** - In order to scale their multi-dimensional visualisation technique, Value and Relation, to large datasets, Yang et al. [Yang et al. 04] use random sampling to fetch a default maximum number of data items from the database server. They do this to either reduce response times and/or increase accuracy. The paper does not mention how this sampling is performed. Lee et al. [Lee et al. 01] have found that random sampling provides a plausible solution when visualising vast amounts of website usage data. The user may specify the size of the dataset to visualise, based on various attributes of web sessions and the system will also perform stratified sampling that is useful when comparing different attributes. Olken [Olken 93] points out that sampling is used in certain GIS to reduce dataset size prior to graphical display of the data, both with an aim to reduce computational effort and to match the resolution limits of the display. In a similar way, The Data Filter in visual framework VISTA [Chen and Liu 03], prepares the data for visualisation and amongst other functions can randomly sample the data to create a manageable dataset. To achieve fast multidimensional scaling Morrison et al. [Morrison et al. 03] make use of sampling to reduce complexity and run time and allow the visualisation of data sets of previously unfeasible size.

**learning algorithms** - There are various algorithms where random initial values or presentation orders are used. These include numerous force-directed visualisations for the Web and other information domains [Brodbeck et al. 97, Hendley et al. 95] and also neural network techniques such as Kohonen Maps [Kohonen 90, Lin 92 and 97]. Analytics in nVizn [Wilkinson et al. 01] currently include bootstrapping (random sampling with replacement) and simple random sampling.

### 2.3.1. Sampling an infinite space

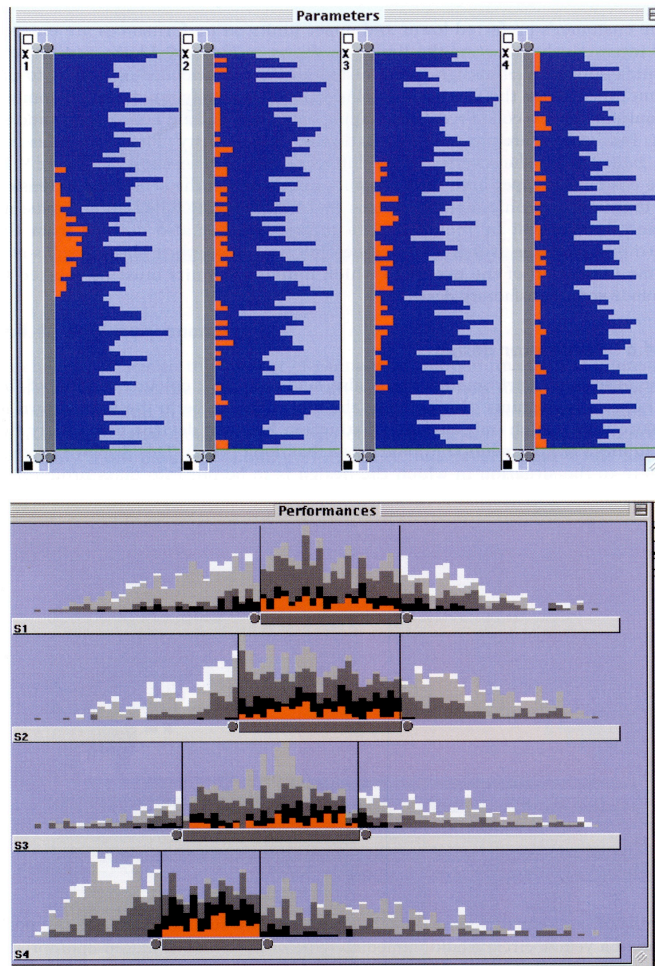
Tweedie, Spence and Dawkes [Tweedie et al. 95 and 96] Influence Explorer enlists the help of sampling to generate data points in an infinite parameter space that drive a simulation and allows the engineer to visualise the influence of parameters on a range of performance measures. The first edition of Spence's Information Visualisation book [Spence 01] illustrates the Influence Explorer with a simple example, reproduced here as Figure 2-4. The designer or engineer identifies a Region of Exploration ( $R_E$ ) defined by ranges of parameters  $X_1$  and  $X_2$  in which the solution to the problem is likely to be found. This 2-dimensional region in the parameter space is then randomly sampled

Figure 2-4



The generation of parameter-performance pair by random sampling the parameter space. Based on [Tweedie et al. 94]

Figure 2-5



Histogram view of the multi-dimensional parameter and performance spaces of the Influence Explorer. [Tweedie et al. 94]



and the selected  $X_1, X_2$  parameter pairs are fed into the equations of the simulation model and the resulting performance values can be plotted in the performance space. Brushing<sup>3</sup> a set of points in the performance space, reveals the corresponding parameter values. The choice of a random sample is important in order to avoid the possibility of phase errors inherent with a systematic sample (e.g. selecting points on a regular grid) [Mihalisin 91].

This type of design task is fairly common in engineering disciplines and according to Spence, is also to be found in financial decision making. The tasks are often more complex than the example above, with more parameters and performance measures. To analyse this multi-dimensional data, the Influence Explorer adopts a histogram visualisation similar to the Attribute Explorer [Tweedie et al. 94]. In the performances window given in Figure 2-5, the designer has specified acceptable limits for each of the performances S1 to S4. The corresponding points in the parameter space that match all the performance targets are coloured red, with black points representing ones that match all but one of the targets. The Parameters window (Figure 2-5) also indicates, in red, the range of parameters that match all four performance targets and those that do not, in blue. As this is modelling a manufacturing process, the yield can be calculated.

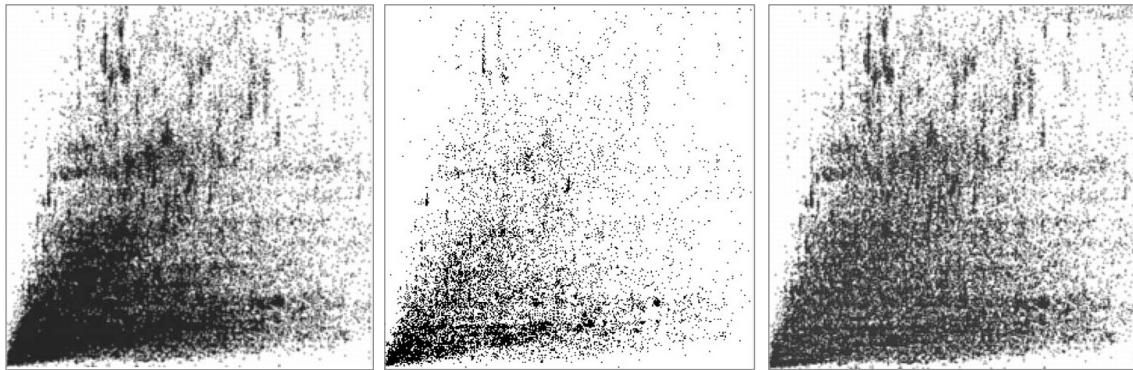
The Design Gallery for image exploration [Marks et al. 97] is another example of computer assisted selection of a set of input vectors that optimally disperses the resulting output values. An initial set of input vectors are chosen at random. Each vector is given a random displacement and the new value is substituted for the old one if the dispersion of the resulting output vectors is improved, as measured by the nearest neighbour distance.

### 2.3.2. Non-uniform sampling

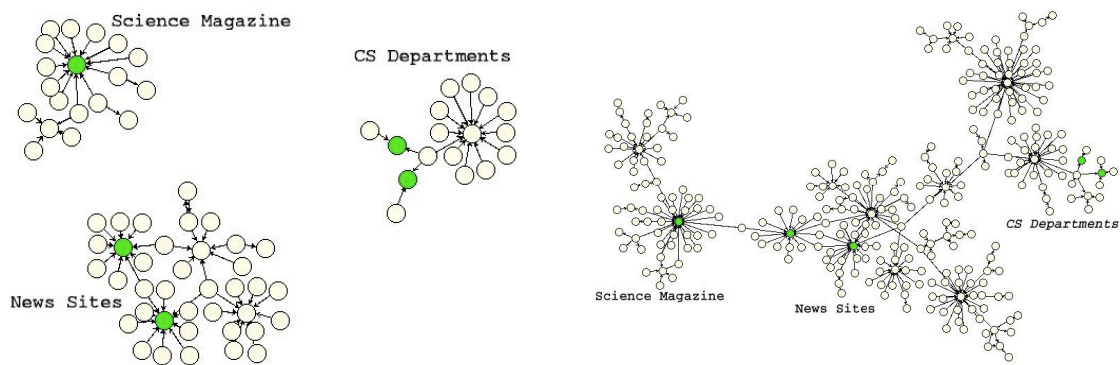
Bertini and Santucci [Bertini and Santucci 06] have pursued the use of sampling to reduce clutter on 2D scatterplots. Their strategy is to divide the display area into sample areas (e.g. 8x8 pixels) and count the number of plotted points in each, which gives a set of data densities. Often there is uneven distribution of densities across a scatterplot, which means that even after adjusting the density in each sample area, there may be little discernible difference between high-density areas. To maximise the number of data density differences across the scatterplot, the range of available densities (for an 8x8 pixel sample area this is 64) are allocated to particular sample areas based on the number of sample areas at each measured density. This in essence gives more contrast between adjacent sample areas.

---

<sup>3</sup> Interactive selection of one or more data items in one plot (either singly or multiply by identifying a region), which are then highlighted in all related plots.



**Figure 2-6** Mail parcel data [Bertini and Santucci 06] (a) original image, (b) best uniform sampling and (c) perceptual non-uniform sampling



**Figure 2-7** Network visualisation through sampling (a) 0.1% sample size and (b) 0.2% sample size. [Rafiei and Curial 05]

To compensate for a possible mismatch between the measure of density (number of plotted points) and how humans perceived density, an experiment was undertaken to see how many differences in density in a 8x8 sample area could be perceived by users. Users were shown a 10 by 10 grid of sample areas, with all but three areas containing a random arrangement of a given number of pixels. The three randomly positioned sample areas had extra pixels added. The number of extra pixels were increased until a significant number of users could pick out the higher density sample areas. They found that in the middle range of densities, 7 or 8 pixels needed to be added for the participants to perceive a higher density sample area. Consequently, they determined an actual density to perceptual density mapping function with 14 steps and adjusted the smoothed sample area densities to take account of this. Example images of their visualisation are shown in Figure 2-6, which are reproduced from the original paper. Note that (b) uses a constant sampling across the whole plot whilst (c) automatically adjusts the sampling rate using their quality metrics and a chosen quality threshold.

This work is noteworthy in that it utilises random sampling for automatic clutter reduction in both uniform and non-uniform ways. The estimation of display clutter is based on a formal model that also provides quality metrics from which to determine appropriate sampling rates. In addition, the user can dynamically adjust the sampling rate or set some metric threshold and switch back and forth between the un-sampled and sampled views. Visual consistency is maintained. The work is also valuable in that the human perception of density difference has been taken into account when in non-uniform sampling mode.

### **2.3.3. Sampling a network graph**

Rafiei and Curial [Rafiei and Curial 05] use sampling in their prototype system ALVIN to visualise very large networks. They show that even with relatively small samples, patterns that are present when viewing the full dataset tend to be preserved in the sample. They also note that sampling network data is more complex than sampling single table or flat data due to the relationships between the nodes and the edges. They consider three modes of sampling - nodes, edges or a combination of these. One of their strategies for initiating the display, is to select a set of seed nodes (either manually or by way of a search function) and then allow the user to interactively grow the network to any size they wish, as shown in Figure 2-7, reproduced from the paper. Three methods of growth are detailed - global, local and mixed. Note that due to the random sampling, each growth results in a different layout that may not be coherent with the old one. The visualisation has a feature they call "Rewiring" that removes all the edges and adds a new set and "may reveal properties that may not have been displayed by the original wiring". This is very similar to the Reality Check mentioned in



the previous section. Sampling distribution issues are discussed and they point out that weighted networks can bias sampling towards more important edges. Some examples are given that show that sampling can lead to a better understanding of the network graph.

## 2.4. Types of sampling

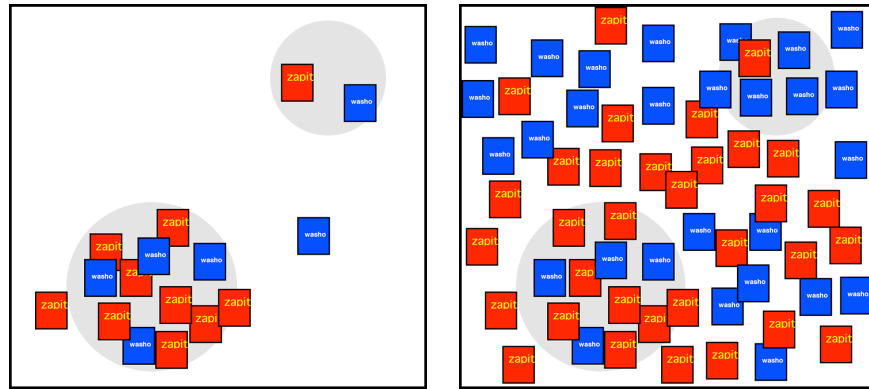
Sampling involves picking some items from the dataset. In some cases this is true but as we saw in some of the examples from Section 2.3, an awareness of the distribution of the underlying data is necessary to be confident in the results obtained. This section addresses some of these issues.

When we make a sample we make choices. If the sample is simply a uniform selection (such as 1 in 10 of the data set), then perhaps we do not need to worry, but most sampling regimes introduce some form of sampling distribution and bias. For example, a home telephone poll will disproportionately exclude those who do not have a phone or are away at work during the day if the poll is conducted during the day time.

Returning to the Influence Explorer described in Section 2.3.1, the calculation of the probable yield appears to be problematic due to the type of sampling used. Remember that this involves selecting bounds for input parameters and then seeing what proportion of the sample satisfies the output targets. Obviously, the sampling distribution will be uniform within the input parameter bounds as it is random, whereas in reality it is likely to have some more complex shape depending on the manufacturing process. For example if a parameter is the dimension of a part, it is likely to be normally distributed about the average value. So, a uniform distribution may be misleading during yield optimisation. Statisticians deal with this by using stratified samples (see Section 2.4.2) then weighting (based on knowledge of the real distribution) during later processing to fix sampling bias. Although the Astral Visualiser changes the level of sampling when zooming, this does not depend on the density of the data points but is more a matter of the overall view settings of the user interface. However, in some applications, sampling needs to be more tuned to the location and kind of data.

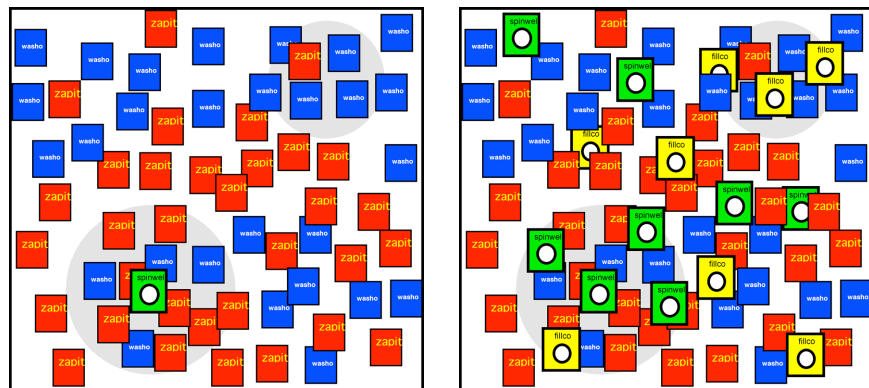
For example, in Bertini and Santucci's work described Section 2.3.2, they adjust the sampling rate in each 8x8 pixel sample area of their scatterplot so patterns in both dense and sparse regions can be perceived whilst still presenting a view that maintains an impression of the original density. Some more examples of non-uniform sampling will now be considered.

Figure 2-8



Washing powder sales map. (a) uniform sampling removes most of the data items in less populated regions. (b) non-uniform sampling provides a region wide comparison of brands

Figure 2-9



Washing powder and washing machine sales map. (a) identical non-uniform sampling for powder and machines (b) independent non-uniform sampling

### 2.4.1. Constant density

Constant density systems [Woodruff et al. 98b] are principally concerned with changing the size of a data point to give more information should there be room on the display. Although this is still possible with our sampling-based visualisation, we have an additional option of increasing the level of sampling in sparse areas to maximise the use of the display space. A brute force algorithm would be to repeatedly choose a new data point at random, excluding all points that are within radius  $r$  of existing plotted points until there are no suitable points remaining.

This would be the case if we wanted to visualise the sales of particular brands of washing powder across the UK. There are of course few supermarkets and few sales in less populated parts of the country, so a fair sample would mean it is impossible to see what was happening in the Highlands and Islands of Scotland (too few points) and London would just be an untidy pile. By altering the sampling density based on data density, we can get an even spread of data points over the whole country and thus easily spot trends of preferences (see Figure 2-8).

This technique distorts the actual density and so cannot be used if correlations are being sought in the  $x$  and  $y$  plane. Hence we cannot look for trends in the overall sales of washing powder, we can only look at the relative sales between brands.

### 2.4.2. Attribute dependent density

Assume we now want to see if there are patterns relating sales of washing machines and washing powder. There are of course many hundreds more sales of the latter and if we sampled both uniformly we would only see washing powder plotted on our map. Instead, we need to sample the two types of product separately to ensure approximately similar overall numbers of each so that relative differences in different areas become apparent (see Figure 2-9). This is a form of stratified sampling.

Again, this technique distorts density and cannot be used to compare actual sales of the two product types, but it can be used to look for correlations and trends between them. Hernández-Campos et al. [Hernández et al. 02a and 02b] describe similar issues with Internet HTTP traffic data where there are often “heavy tails” to distributions<sup>4</sup>, a small number of very large flows (sequence of packets with common source and destination) and very large numbers of small ones.

### 2.4.3. Sampling structured data

As we have seen from Rafiei and Curial’s work, more complex issues arise when sampling non-point data including hierarchies (trees and taxonomies), networks,

---

<sup>4</sup> An overview of power law distributions can be found at [http://en.wikipedia.org/wiki/Power\\_law](http://en.wikipedia.org/wiki/Power_law) - accessed Oct 2008





ontologies, web logs and relations. For example, in web log data one often looks for sessions as a series of accesses from the same person (based in IP address). We might choose (i) to divide the data into sessions first then sample, or (ii) to sample individual accesses and then track their associated sessions. Option (ii) would lead to more long sessions being sampled than option (i) as a long session includes more individual accesses. Which is *right* depends on the visualisation context. In the case of relations like the actor–film data set, we would get different distributions if we sampled actors first then films, films first then actors, or listed all the actor–film pairs and sampled that. Again, which is *right* is a matter of the way we intend to use the visualisation. It is important that the user is aware of these differences, but explaining subtle statistics is not straightforward. As with similar previous issues, these problems with sub-sampling for visualisation are simply reminders that the same issues occur whenever the raw data is itself incomplete.

## 2.5. Sampling from databases

Having considered the need for different types of sampling and highlighted the requirements for display continuity and re-sampling in Section 2.1.1, we will now discuss how we might generate the samples for large datasets. In smaller datasets the actual data only needs to be stored in memory and special data structures constructed to allow appropriate sampling. However, very large datasets may use a database approach and hence we have to consider the necessary requirements of a DBMS to meet the needs of random sampling.

### 2.5.1. Issues and requirements

The literature on random sampling from databases emphasises the importance of linking sampling and query operators. Olken [Olken 93] argues that sampling operators ought to be embedded within the query processing to reduce the amount of data retrieved and effectively exploit indices created by the database management system.

Olken identified the following five issues for database sampling:

1. the manner in which the sample size is determined
2. whether the sample is drawn with or without replacement
3. whether access pattern is random or sequential
4. whether or not the size of the population from which the sample is drawn is known
5. whether or not each record has a uniform inclusion probability

Some of these have clear answers in visualisation. The sample size (or sampling probability) will be defined as part of the interaction (zooming or density control). The



data points are to be plotted or otherwise visualised and hence should normally be drawn without replacement. The entire sample is likely to be used at once, but it may be refined later by the user. The population size is the size of the data set to be visualised, although filtering may mean that the size of the filtered dataset may not be known without querying the database. Finally, we may want to have non-uniform inclusion probability as discussed in the previous section when we considered constant density or attribute-based density.

In addition, the discussion of sampling issues in Section 2.1.1 raises some further requirements:

6. the ability to have repeatable samples
7. the ability to chose to re-sample and have labelled samples
8. the ability to have a randomised output that can be used as a z-index

Note that the repeatable samples (requirement 6) refers to the requirement that the new sample includes the original sample, when the size is increased. In terms of database operations, this means that the sampling operator should have nice algebraic properties. Therefore, if  $S_p$  is sampling at probability  $p$  and  $F_c$  is filtering on some condition  $c$ , we would expect properties such as:

$$\begin{aligned} \text{if } p \leq q \text{ then } S_p(\text{db}) &\subseteq S_q(\text{db}) \\ S_p(F_c(\text{db}) \cap F_{c'}(\text{db})) &= S_p(F_c(\text{db})) \cap S_p(F_{c'}(\text{db})) \\ \dots \end{aligned}$$

The inclusion of one or more generated random z-index fields is not only useful for visualisation itself, but also makes it easy to satisfy these quite stringent conditions. So, if the z-index is chosen in the range  $[0, Z_{\max}-1]$ , this will translate "with probability  $p$ " into "z less than  $p \cdot Z_{\max}$ " etc. Repeated sampling based on the same z-index will be identical (requirement 6). By shifting the window on the z-index different labelled samplings can be produced (requirement 7). Also, since the named z-index fields are regarded like other fields for selection purposes they can be included in the query result (requirement 8). Furthermore, the selection condition on the z-index can be varied depending on other field values, thus allowing non-uniform sampling. Wrapping when  $Z_{\max}$  is not an exact multiple of the window size adds to the complexity, but can be managed using extensions of the simple z-index method, as we will see in Chapter 4.

### 2.5.2. Database management system support

DBMS support for sampling from databases is still limited despite Olken's work in the 90's. Klein et al. [Klein et al. 06] state that apart from their prototype Derby/S system, there is no DBMS that natively supports the provision of sampling on a broader scale. Their extension to the SQL language include a CREATE SAMPLE statement that includes



the following options:

- ▶ specify a size as number or rows or percentage
- ▶ choose whether the sample is re-computed from scratch or uses the existing pre-computed samples
- ▶ specify that an approximate answer is acceptable
- ▶ specify confidence limits

In addition, samples are named and maintained by the DBMS as and when the underlying tables are updated. This extension to SQL is aimed at providing fast answers to queries on large datasets especially where aggregate functions are involved and as such, the requirements identified above for the z-index are clearly not met.

A far simpler sampling option is provided by SQL Server 2005. Its TABLESAMPLE<sup>5</sup> clause limits the results set to a sample, specified as a percentage or a number of rows. However, this can only be used for single table queries (not derived tables), the size of the results set is approximate and it is not guaranteed to be random as the selection is at a physical page level. An option to this clause is REPEATABLE(seed value) where specifying the same seed value will return the same results set, provided that the table has not been modified. For the issues and requirements stated above, only requirement 6 is partially satisfied.

## 2.6. Summary and reflection

The proposed Astral Visualiser sampling-based visualisation, based on the scenario of star gazing with a telescope demonstrated the advantage of sampling in reducing overplotting on a scatterplot. It also showed the feasibility of auto-sampling, where the sampling rate is adjusted by the system, based on the zoom factor, to maintain an approximate constant density. No measurement of *density* was made.

Various issues pertinent to sampling were discussed, such as adding and deleting points, panning and zooming, moreover the importance of display continuity was emphasised. As sampling might itself create artefacts or false patterns, Reality Check was proposed to allow the user to generate a new sample of the data to check whether the pattern persists.

Furthermore, the z-index method was proposed as a solution for generating a random sample that meets the display continuity concerns. A full implementation of the Astral Visualiser, with constant-density zooming, is illustrated in Section 6.6 with examples of zooming in and out of a map showing the distribution of household income data across the USA.

---

<sup>5</sup> ORACLE has a similar function called SAMPLE.

- ▶ reduces the amount of data, which generally leads to a reduction in clutter.
- ▶ is relatively easy to implement.
- ▶ is good for explorative visualisation, where we are not sure of the question.
- ▶ has a simple interactive user control.
- ▶ can be adapted to take account of sampling distributions.
- ▶ can provide a particular density mapping across the display.
- ▶ can offer automatic density control when zooming.
- ▶ is a natural operation – the majority of data is a sample anyway.

**Table 2-2** Summary of the key benefits of a sampling approach to clutter reduction based on the proposed Astral Visualiser.

Visualisations that employ sampling are few in number and we saw three sampling-based applications. The Influence Explorer demonstrates that random sampling is useful for generating a dataset from an infinite parameter space. Bertini and Santucci's scatterplot application is a very good illustration of non-uniform sampling and of the pertinent use of perceptual mapping based on a user study. The third application by Rafiei and Curial applied sampling to a network structure to reduce clutter. Although this work is not concerned with clutter reduction in graphs per se, Rafiei and Curial application is of interest in two ways - it incorporates a *Rewiring* feature similar to the Reality Check and it could be applied to relational data; we will return to this in Section 7.4.1.

The importance of sampling in terms of the distribution of the data and the requirements of the user should also be taken into account. Different scenarios were proposed, to illustrate non-uniform, constant density, stratified and relative sampling.

In order to meet the needs of random sampling based visualisations, a database management system should provide appropriate sampling features, many of which are currently not supported.

Finally, Table 2-2 summarises the key benefits of a sampling approach to clutter reduction based on the proposed Astral Visualiser.





## Chapter 3

# Clutter-reduction Taxonomy for information visualisation

Having proposed sampling as a novel technique for clutter reduction (Chapter 2) and demonstrated its use through the Sampling Lens application (Chapter 4), the need to establish the effectiveness of sampling was apparent. Evaluating visualisation techniques are problematic as has been highlighted by several researchers [Kosara et al. 03, Plaisant 04, Tory and Möller 05]. The main issue lies with the difficulty in finding suitable datasets, representative tasks and participants with the necessary domain knowledge<sup>1</sup>.

After carrying out an extensive literature survey to learn how the creators of 2D information visualisations conducted their user studies, it was surprising to discover that very few visualisation applications are evaluated with users (in fact only 12 out of the 65 that were examined). Moreover, an explorative analysis of the experimental details and results questioned the viability of evaluations in cases where the outcome is probably a foregone conclusion, or where inappropriate experiments are perhaps carried out, or even where the results are possibly unconvincing. The findings from this study led to a paper [Ellis and Dix 06a] presented at the BELIV'06 workshop<sup>2</sup>; the full paper can be found in Appendix F.

The reason behind so little user involvement in evaluations may be because information visualisations are very difficult to evaluate. Besides the difficulties of finding appropriate datasets, tasks and participants, the visualisation process is made up of a complex set of interactions and ideally, we should understand the mechanisms inherent in the process to assess the viability of an evaluation. In the BELIV'06 paper, we also discuss the importance of understanding the meaning of accuracy, precision and significance of the statistical data as well as the problem of finding a valid point of comparison between visualisations.

Given the complexity involved in evaluating a particular visualisation, a novel method to evaluate sampling as a clutter reduction technique is based on an analytical approach, and is the subject of this chapter. The Clutter-reduction Taxonomy for information visualisation compares a set of clutter reduction techniques with a set of

---

<sup>1</sup> The problems of evaluating information visualisations is discussed further in Section 6.1

<sup>2</sup> Beyond Time and Errors: Novel Evaluation Methods For information Visualization



criteria, based primarily on how researchers describe the benefits of their visualisation in terms of clutter reduction, following an analytical review of the literature.

Section 3.1 briefly reviews some published classification schemes for information visualisation, however most pay little attention to clutter reduction with the exception of Ward's taxonomy of glyph placement strategies [Ward 02] and Bertini's design space characteristics and clutter reduction strategies [Bertini 07].

Section 3.2 gives an overview of the clutter reduction techniques, which are compared in the taxonomy. Examples of these techniques are included where appropriate in the discussion notes on the taxonomy in Section 3.4.1. Further examples and extended descriptions of techniques are presented in Appendix A.

Section 3.3 defines eight criteria through which clutter reduction techniques can be assessed.

Section 3.4 presents the Clutter-reduction Taxonomy for information visualisation, which shows the possible benefits of each technique in terms of the criteria. The taxonomy reveals exceptions and not so clear cut cases and the discussion of these forms an important part of the taxonomy.

Section 3.5 reflects critically on the process of creating the taxonomy and gives practical examples of its use in combining techniques and promoting thinking about clutter reduction and visualisations. The two existing classifications described in Section 3.1, namely Ward's taxonomy of glyph placement strategies [Ward 02] and Bertini's clutter reduction strategies [Bertini 07] are then revisited and compared with the Clutter-reduction Taxonomy. This section concludes by discussing the importance of establishing criteria in building a classification/taxonomy.

Section 3.6 finally reflects on the salient issues raised in this chapter.

### **3.1. Information visualisation classification schemes**

Whilst classification schemes covering various aspects of information visualisation are found in the literature, few mention clutter reduction. This section provides a brief review of some existing classification schemes<sup>3</sup> presented in chronological order before considering in more detail, Ward's taxonomy of glyph placement strategies and Bertini's design space characteristics and clutter reduction strategies, which explicitly deal with clutter reduction.

In the early nineties **Leung and Apperley** [Leung and Apperley 94] presented a comprehensive review of distortion-oriented visualisations. Although it does not

---

<sup>3</sup> Note that this review does not include graph drawing, as there is already a large body of work which deals with the effective layout of graphs.



mention clutter directly, in the conclusion it states, “Other non-distortion techniques, such as information suppression, should be investigated further since they are potentially powerful”.

**Shneiderman's** type by task taxonomy [Shneiderman 96] illustrates how high level tasks (overview, zoom, filter, details-on-demand, relate, history and extract) can be applied to some basic data types (1, 2 & 3-dimensional data, temporal and multi-dimensional data, tree and network data). In terms of clutter reduction he points out that, zooming reduces the amount of data as does filtering, which “filters out uninteresting items”. Note that details-on-demand could be thought of as clutter reduction in that it hides data until requested.

**Wong and Bergeron** [Wong and Bergeron 94] reviews the developments in multidimensional multivariate visualisation. Although clutter is not raised when discussing various techniques, in the conclusion it states that scientists have to deal with data that is many thousand times bigger than the number of pixels on display. This observation made over a decade ago is even more pertinent today due to the increase in the size of datasets and the prevalence of mobile devices with small displays.

**Keim** [Keim 97] presents a classification of data visualisation techniques (geometric, icon-based, pixel-oriented, hierarchical, graph-based and hybrid), interaction techniques (mapping, projection, filtering, link&brush, zoom, detail on demand) and distortion techniques (simple, complex). This work mentions sampling, querying, segmentation and aggregation as data processing techniques, designed to reduce the amount of data displayed. In addition, he gives techniques for dimensional reduction including multidimensional scaling. However, this does not reduce the number of data points, but the number of values to be visualised.

**Card and Mackinlay** [Card and Mackinlay 97] present a data-centric approach to classifying visualisations based on succinct mappings between the data and presentational properties.

In a later paper, **Card et al.** [Card et al. 99] proposed the Reference model for Visualisation that introduces human interaction into the process of mapping the data to a visual form. In so doing, it categorises techniques such as zooming, focus+context, magic lens and dynamic queries, which are forms of clutter reduction.

**Chi's** taxonomy [Chi 00] takes a more detailed, processing-centric approach to classifying visualisation techniques going from value to view via analytical and visualisation abstractions. The transformations used to change state are also detailed for a large number of example visualisations. In doing this, Chi shows how techniques can be built using a modest number of operators. Note that dynamic value filtering

<i>data-driven</i>	<i>raw</i>	<i>original</i>
		<i>distorted</i>
	<i>derived</i>	<i>original</i>
		<i>distorted</i>
<i>structure-driven</i>	<i>ordered</i>	<i>overlapping</i>
		<i>space-filling</i>
		<i>separation</i>
	<i>hierarchical</i>	<i>overlapping</i>
		<i>space-filling</i>
		<i>separation</i>
	<i>network</i>	<i>overlapping</i>
		<i>space-filling</i>
		<i>separation</i>

**Table 3-1**

Ward's taxonomy of glyph placement strategies [Ward 02].

occurs at the analytical, visualisation stages as well as in the user view stage. Other clutter reduction operations such as zoom, view filtering, level-filtering and change distortion focus are included at the view stage.

A comprehensive review of data clustering algorithms is given by [Jain et al. 99] and [Murtagh 02], the latter focusing on massive datasets. These do not deal directly with display clutter but they provide a means of pre-processing the data into a more manageable set.

Two publications that discuss clutter reduction more fully are Ward's taxonomy of glyph placement strategies [Ward 02] and Bertini's design space characteristics and clutter reduction strategies [Bertini 07].

### **3.1.1. Ward's taxonomy of glyph placement strategies**

Ward's paper explicitly deals with glyph placement strategies, which can have a significant affect on display clutter. He considers four aspects of placement, (a) position calculated from the data or determined by the structured representation of the data, (b) degree of overlap allowed, (c) screen utilisation and (d) localised displacement to improve visibility. In addition, he discusses a variety of distortion techniques to reduce clutter and overlap and recommends smooth animation between the original and distorted views. Ward states that placement techniques are a trade-off between efficient display use, amount of occlusion and distortion and suggests that the user should be able to alter these dynamically. Furthermore, he proposes that the system could analyse congestion and adjust parameters automatically to reduce clutter, noting that this is a relatively unexplored area. Ward's taxonomy is presented in Table 3-1.

The taxonomy is focused on how the coordinates of the data points are derived subdividing the strategies into data-driven and structure-driven placement. Data-driven means that the data is used to compute or specify the location; structure-driven means using implicit or explicit connectivity or relationships between the data points. Hence, under structure-driven we have hierarchical trees, network graphs and sequential pixel-plotting arrangements. This classification does not relate directly to clutter reduction, however some of the distortion strategies mentioned within the text can be compared to clutter reduction techniques.

Table 3-2

<i>placement</i>	<i>strategy</i>	<i>partitioning</i>
		<i>overlapping marks</i>
	<i>degree of freedom</i>	<i>fixed</i>
		<i>constrained</i>
		<i>free</i>
<i>visual marks</i>	<i>pixel</i>	
	<i>line</i>	
	<i>area</i>	
	<i>text</i>	

Bertini’s design space characterisation [Bertini 07].

Table 3-3

<i>visual density reduction</i>	<i>suppression</i>
	<i>subsetting</i>
	<i>level of detail</i>
<i>spatial organisation</i>	<i>layout</i>
	<i>ordering</i>
<i>retinal properties</i>	<i>brightness</i>
	<i>colour mapping</i>
	<i>transparency</i>
	<i>shading</i>

Bertini’s clutter reduction strategies [Bertini 07].



### 3.1.2. Bertini's classification of clutter reduction techniques

In Chapter 3 of his thesis, Bertini [Bertini 07] presents a classification of design spaces and clutter reduction techniques with the twin aims of presenting a holistic view of the issues relating to display and of enabling existing and new visualisations to be criticised.

Table 3-2 presents Bertini's design space characterisation that considers the design space in terms of an objects placement and the visual marks that represent the object on the display. Placement strategies revolve around whether the points are allowed to overlap or not. In the case of no overlap, Bertini notes that visual noise may be caused by the graphic element used to partition data items (e.g. a border line). Degree of freedom discusses the amount of flexibility available to reposition points. For example, geographical information is fixed, the axes on parallel coordinate plots are constrained in the vertical axis but can be re-ordered horizontally, and *free* means that only relative spatial information is conveyed (e.g. order and distance). The visual marks classification is a modification of Card and Mackinlay's proposal [Card and Mackinlay 97] and considers the different ways to represent data items (i.e. pixel, line, area, text) and identifies the possibilities for clutter.

Table 3-3 shows Bertini's characterisation of clutter reduction techniques, with three high-level strategies, each one broken down into methods. The first strategy, visual density reduction, discusses three methods for reducing the number of data items. Suppression in this context is similar to Ward's - filtering and sampling. Spatial organisation looks at ways of reducing clutter by rearranging points on the display - this refers to the degrees of freedom in Table 3-2. The last strategy, retinal properties, considers four methods of altering a subset of the visual attributes to reduce clutter, i.e. brightness, colour mapping, transparency and shading.

In a separate section, Bertini, outlines how distortion can be applied to the appearance of objects (e.g. position, form, size and retinal properties) to reduce clutter. These include non-uniform magnification, point displacement and non-linear colour mapping.

## 3.2. Techniques for visual clutter reduction

The analysis presented here is based on an extensive review of the literature (around 80 publications) from which eleven of the most widely used clutter reduction techniques were selected. These have been restricted the choice to 2D visualisations because the benefits of 3D (or more often 2.5D), such as larger virtual space or the additional dimension, outweigh usability problems like navigation, perception and occlusion. This is backed up by work such as the evaluation of cone trees by Cockburn

**Table 3-4**

	<b>clutter reduction technique</b>
<b>appearance</b>	sampling
	filtering
	change point size
	change opacity
	clustering
<b>spatial distortion</b>	point/line displacement
	topological distortion
	space-filling
	pixel-plotting
	dimensional reordering
<b>temporal</b>	animation

Clutter reduction techniques used in the Clutter-reduction Taxonomy

and McKensie [Cockburn and McKensie 00]. In addition, many of the clutter reduction techniques can be applied to 3D visualisations anyway.

In addition, node-edge graphs have been excluded from the survey as there is already a wealth of research on the effective layout of these structures [e.g. DiBattista et al. 98, Munzner 00, Eades and Hong 04], whereas there is relatively little work on classifying methods for dealing with overcrowding in other display structures. However, sampling structured data (e.g. graphs, trees) is considered when looking at uses of sampling in visualisation (Section 2.3.3), types of sampling (Section 2.4), adding sampling to other visualisations (Section 6.5) and suggestions for further work (Section 7.4).

The survey of the literature revealed other techniques that have not been included in the Clutter-reduction Taxonomy as they were considered to be either more pre-processing operations, not suitable for interactive clutter reduction or too specialised. They include summary statistics and aggregation, dimensional reduction, appearance other than point size and opacity and anisotropic volume rendering. However, these techniques can be used for clutter reduction and hence are described in Appendix A.2.

The chosen techniques have been organised into three main groups, namely *appearance*, *spatial distortion* and *temporal* (as shown in Table 3-4), based on how each technique manipulate visual, position, or temporal attributes of the data items to reduce display clutter.

### 3.2.1. Appearance

The *appearance* group in Table 3-4 lists those techniques which tend to affect the look of the data item.

Filtering and sampling are included in this group as they have a dramatic affect on the appearance of data items – the items disappear. The method of choosing which items remain in view differs between filtering and sampling. With filtering, the user specifies constraints that may be one or a range of attribute or derived values. For example “all houses with a double garage and garden area greater than 400 sq.m” or “number of word co-occurrence links between documents greater than 50”. Whereas sampling<sup>4</sup> is the random selection of a set of data items and requires no user intervention. Filtering is often used in combination with other clutter reduction techniques and it is suggested in Section 4.2.5 that it may be advantageous to sample the results of a filtering operation.

Change point size implies reducing the size of the displayed items which will reduce or remove the overlap on previously overlapping, but non-coincident points. However,

---

<sup>4</sup> A definition of random sampling is given in Section 1.2 and different types of distribution-based sampling are discussed in Section 2.4



as we will see later, smaller points may be less discernible and suffer from colour blending. This problem also exists with the change opacity technique, where the opacity of data items are reduced (transparency increased) so that previously hidden points are revealed.

Clustering is where data items with some commonality are grouped together. This usually results in a different representation of the group of individual lines or points (e.g. single point or line) so is included in the *appearance* group.

Note that there are other appearance attributes such as colour, blurriness and texture, which have not been used as part of Clutter-reduction Taxonomy. Blurriness and texture are fairly specialised techniques and the use of colour was found to be very difficult to define. However, they are included in the discussion of other techniques in Appendix A.2.

### 3.2.2. Spatial distortion

*Spatial distortion* includes techniques that displace the data items in some way with the purpose of reducing or avoiding overlap.

The first technique in this group, point/line displacement adjusts the position of each data item. In the case of a line, this may include distorting the line along its length (e.g. curving a straight line). The displacement vector may be generated randomly, based on some user input or as a result of a calculation, such as a packing algorithm. If the data has spatial attributes then this will be taken into account.

The next technique, will be referred to as topological distortion. This results in the movement of points on the display to reduce or avoid overlap and may well look the same as point/line displacement techniques. However, the deformation model is different, which can influence the user's perception of the distortion and ultimately their understanding of the data. Instead of moving the points within their coordinate system, topological distortion stretches (and/or shrinks) the underlying 2D space (i.e. background) which means that points may appear to have moved when mapped back to the display space. The distortion can be either non-uniform (e.g. hyperbolic) or uniform (e.g. traditional zoom).

Space-filling is essentially a non-overlapping rearrangement of large points, often driven by a particular structure. For example, Treemaps [Shneiderman 92] visualise hierarchically structured data with the size and colour of each point (a rectangle) representing some attribute value.

Pixel-plotting is a technique where each data item is represented by one (or perhaps a few) screen pixels and hence there are no overlapping points. This means that more than one million data items can be represented on a modest display. The arrangement

<b>paper</b>	<b>criteria found</b>	<b>classified as</b>	<b>comments</b>
Fekete 2002 - Million items	closely packed points often merge so a good idea to be able to distinguish individual points in a crowded display	can discriminate points/lines	use smooth shaded rectangle (tilt + fog) implemented in graphics hardware
Keim 2004 - PixelMaps	can see all the points so as not to lose point attribute information	avoids overlap can show point/line attribute	densest regions get the space required to place all the data points close to each other
Stasko 00 - Sunburst	shows overview of hierarchy and allows user to examine small items in detail whilst in context	can be localised	radial, space filling visualisation. 3 different ways of zooming in to show detail (angular detail, detail inside, detail outside)

**Table 3-5** Example clutter reduction criteria search records

of points may vary between the Keim's recursive space-filling algorithms [Keim 00], the mini-bar charts of TableLens [Rao and Card 94] and the temporal and spatial distributions of Information Mural [Jerding and Stasko 98].

The last technique in this group is dimensional reordering [Peng et al. 04] in which the dimension order is changed to minimise clutter. For example, the axes of a parallel coordinate plots or the order of the scatterplots in a scatterplot matrix can be rearranged.

### **3.2.3. Temporal**

*Temporal* in this context refers to techniques that have a time dimension rather than temporal data type, as used by Shneiderman in his task by data type taxonomy [Shneiderman 96]. Hence, animation techniques that are mentioned in the literature in relation to clutter reduction fall in this group. An example is Rapid Serial Visual Presentation [Spence 02] where images are show to the user in quick succession (100-200ms), trading display space for time. Animation is also used to convey additional information to the user in a compact form and thus avoid cluttering a display unnecessarily.

## **3.3. Clutter reduction criteria**

To compare the selected clutter reduction techniques, a set of criteria have been identified. Some of these criteria are based on what the author considered desirable features while others have been added after carrying out a literature survey.

From a survey of approximately 50 papers, 68 benefits as stated or implied by their authors were extracted. Examples of criteria search records are presented in Table 3-5. Along with the four columns shown, any disadvantages mentioned by the authors were noted. Each visualisation was then classified according to the clutter reduction techniques employed.

This highly systematic methodology for establishing the criteria is essential to avoid simply reproducing the criteria that the author personally deem appropriate based on the author's research and concerns. Thinking explicitly about criteria is important but is not always easy. This was evident as many of the papers surveyed, were not explicit about the criteria that either drove the work or were used to determine success; in such cases the criteria would often be buried deep in the text.

This process led to eight high-level criteria. These are listed below together with an explanation of why these particular ones were chosen.





### 3.3.1. Avoids overlap

This is a major benefit cited by many researchers and perhaps an obvious one. These include *reduce clutter* [e.g. Stone et al. 94, Woodruff et al. 98b, Yang et al. 03b], *ability to see/identify patterns* [Ahlberg and Shneiderman 94, Ellis et al. 05], *gain a better understanding of the network graph* [Rafiei and Curial 05], *have less hidden data* [Dix and Ellis 02], *avoid the problem of losing information* [Keim et al. 04], *improve a messy display* [Zhang et al. 03], *provide efficient browsing on small displays* [deBruijn and Spence 00], *give more display space to points/nodes* [Kreuseler and Schumann 99, Ahlberg 96], *see more detail* [Fua et al. 99, Lamping and Rao 96] and *see the number of points* [Trutschl et al. 03].

### 3.3.2. Keeps spatial information

Obviously for geo-spatial data, the x-y position of a point is significant. However, the accuracy to which the user can measure the absolute position of a point is questionable and other factors such as landmarks (e.g. representations of physical or political boundaries) may have a greater influence on our spatial awareness. It could be argued that when searching for patterns within the data, only relative positions are important, namely those which essentially define the clusters of points or lines so absolute positions hold less importance. As a result, the criteria would also pertain to keeping relative spatial information. Similarly, for some scale-less visualisations (e.g. Treemaps, graphs, self-organising maps) it is the relative position of data items that conveys information to the user as the absolute position is not quantifiable. In the set of papers examined, only Keim et al. [Keim et al. 04] use a measure of the smallest average deviation from the original position to evaluate various placement algorithms, however the importance to the user of this measure is not discussed.

### 3.3.3. Can be localised

Localised in this context means a specific region or regions of the display. The most common manifestation of this is in focus and context techniques, although non-linear distortion or sampling would also meet this criterion. Examples of benefits of localisation include, *reducing the clutter in localised regions to reveal information underneath* [Wong et al. 03], *providing an overview and detail in a temporal dataset* [Plaisant et al. 96], *allowing the user to examine small items in detail whilst keeping in context* [Stasko and Zhang 00], *avoid losing information in low density areas when reducing overplotting in high density areas* [Bertini and Santucci 06].

### 3.3.4. Is scalable

A clutter reduction technique that can handle very large datasets would seem to be a desirable characteristic. Several papers simply refer to large datasets and few quantify this such as *this method is only limited by the number of available pixels* [Jerding and



Stasko 98].

### 3.3.5. Is adjustable

Most visualisations are interactive in that they allow the user to control some aspect of the visual display. For example, from the early days of dynamic queries [Ahlberg et al. 92], users could reduce the size of the results set by adjusting a slider control. This criteria is concerned with the ability to adjust some parameter of the system that influences the degree of display clutter. The benefits given in the literature include, the *ability to adjust the sampling rate to an appropriate level to see patterns* [Dix and Ellis 02], *interactive adjustment helps the user understand the cluster distribution* [Chen and Liu 03], *users can set visual characteristics and hence tune the clutter reduction* [Bertini and Santucci 04] and the *ability to highlight different aspects of data visualised with parallel coordinates* [Johannson et al. 06].

### 3.3.6. Can show point/line attribute

It is often useful, especially when displaying multivariate data, to map the value of one or more attributes to the colour, shape, opacity of the displayed points or lines. This criteria is not mentioned in the literature examined.

### 3.3.7. Can discriminate points/lines

It is desirable to distinguish between individual points or lines so they can be easily identified in what may be a crowded display. This is backed up by benefits such as *help to differentiate overlapping points* [Brodbeck et al. 97], *distinguish between individual points in a crowded display* [Fekete and Plaisant 02], *reduce apparent clutter by making each cluster of lines distinct* [Fua et al. 99] and *the user can quickly see a particular set of points within a crowded display* [Kosara et al. 02].

### 3.3.8. Can see overlap density

If overplotting is present in the display, then users ought to be made aware of this, otherwise they may not realise that data is hidden from their view. We may also want to see where the higher density regions are and gauge the amount of overplotting. Fekete mentions that *it would be useful to see the amount of overplotting and hence the distribution of data* [Fekete and Plaisant 02] and Wegman argues that *a density plot shows the degree of overplotting and helps to discriminate individual lines such as outliers* [Wegman and Luo 96].

### 3.3.9. Other criteria

A number of authors comment on the efficiency or speed of their algorithms or graphical techniques. Although this is important for improving the interactivity of the visualisation and/or the ability to handle large datasets, this is more of an

		1	2	3	4	5	6	7	8	9	10	11
		sampling	filtering	point size	opacity	clustering	point/line displacement	topological distortion	space-filling	pixel-plotting	dimensional reordering	animation
A	avoids overlap	possibly	possibly	possibly	partly	possibly	✓+	possibly	✓+	✓+	partly	✓+
B	keeps spatial information	✓	✓	✓	✓	partly	✗+	possibly	✓+	possibly	✓	✓
C	can be localised	✓	✓	✓	✓	✗+	✓	✓	✗	✗+	✗	✓+
D	is scalable	✓	✓	✗	✗+	✓	✗	✗	✗	✗	✗	✓+
E	is adjustable	✓	✓	✓	✓	✓	possibly	✓	✗+	✗+	✓	✓+
F	can show point/line attribute	✓	✓	✓	✗+	partly	✓	✓	✓	✓	✓	✓
G	can discriminate points/lines	✗	✗	possibly	✓+	✓+	possibly	possibly	✗	✗	✗	✗
H	can see overlap density	✗	✗	✗	✓+	possibly	✗	✗+	✗+	✗+	✗	✗+

Key: ✓ satisfies criterion;  
✗ does not satisfy criterion;  
+ some exception/special cases (discussed further)  
possibly/partly - more complex cases (see explanations)

**Table 3-6** Clutter-reduction Taxonomy for information visualisation

implementation issue. Some other benefits found in the literature such as *avoid overwhelming the whole view by only showing additional detail within the lens* [Stone et al. 94], *reduce distracting outliers which often confuse the user* [Peng et al. 04] and *automatic clutter reduction reduces the need for user interaction* [Bertini and Santucci 04, Ellis and Dix 06c] were specific to one or two applications and hence have not been included. Finally *provides a good mental model for the user* was considered as a useful criterion, but it was difficult to classify all the clutter reduction techniques as it is more an assessment of a visualisation system as a whole, so this has also not been included.

### 3.4. Clutter-reduction Taxonomy

In this section the set of clutter reduction techniques (Table 3-4) are classified in terms of the criteria defined in the previous section. The resulting taxonomy is given in Table 3-6. A ✓ has been placed against those techniques that meet the given criteria and a ✗ against those that do not meet the criteria. However, some technique-benefit combinations have limitations, are special cases or warrant a mention and these are marked with a + or some text and are discussed below<sup>5</sup>. Note that *possibly* indicates that the criteria is met in some situation but not in others; whereas *partly* indicates that the criteria is only partly met in some situations. Some of the cases discussed are illustrated by examples of visualisations.

The assignment of ticks and crosses on Table 3-6 is based on the author's own assessment of existing systems and using the self-assessment of authors within the literature where present. As with any assessment, another person might rate things differently. To mitigate this, any cases that are felt to be problematic are discussed. Because it is based on what systems actually do, it does not distinguished whether, for example, the presence of a cross means that it is fundamentally impossible for the given technique, or just not found. It should be pointed out that the purpose of this table is not necessarily to assess the techniques against each other, still less to justify sampling, but to act more as a guide to match techniques to problems where different criteria may have different importance, and more importantly a means to critique and hence develop existing and new techniques.

Each criterion will now be discussed. References to a particular cell in the taxonomy table, representing a criteria and clutter reduction technique, are given as A1, G5, etc. Note that the main points of interest are marked with a +.

---

<sup>5</sup> For completeness, other combinations are discussed briefly but the main points of interest are those marked with a + or with possibly/partly.

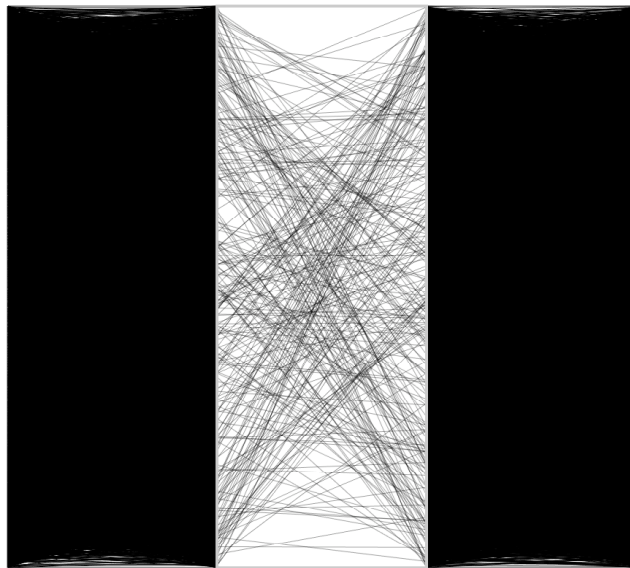


Figure 3-1

Using sampling to reduce the number of overlapping lines. Example shows inter-axis sampling lens [Ellis and Dix 06c] on a parallel coordinate plot with 30,000 records. Sampling rate is 1%.

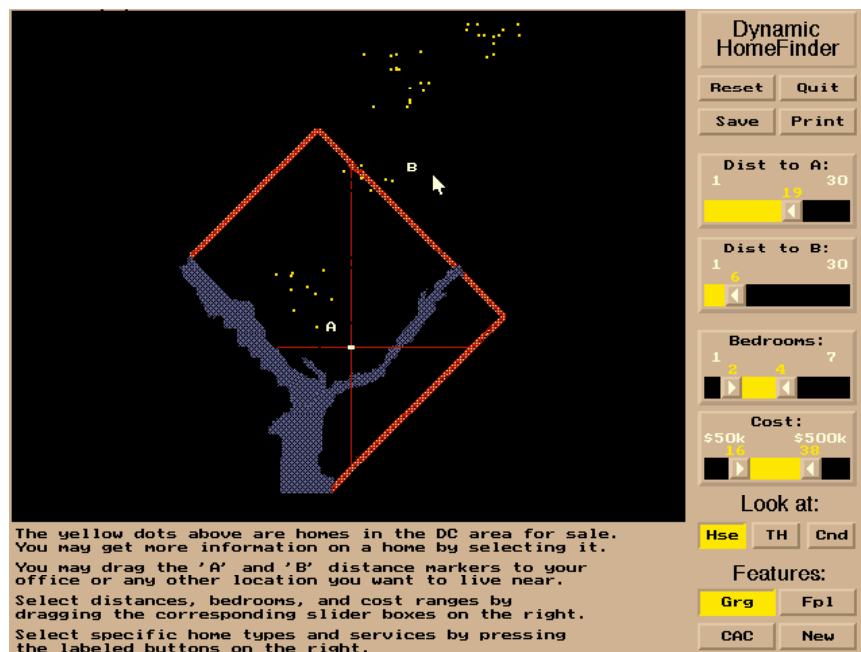


Figure 3-2

HomeFinder's dynamic query interface [Williamson and Shneiderman 92]

### 3.4.1. Discussion of clutter reduction technique

#### avoids overlap

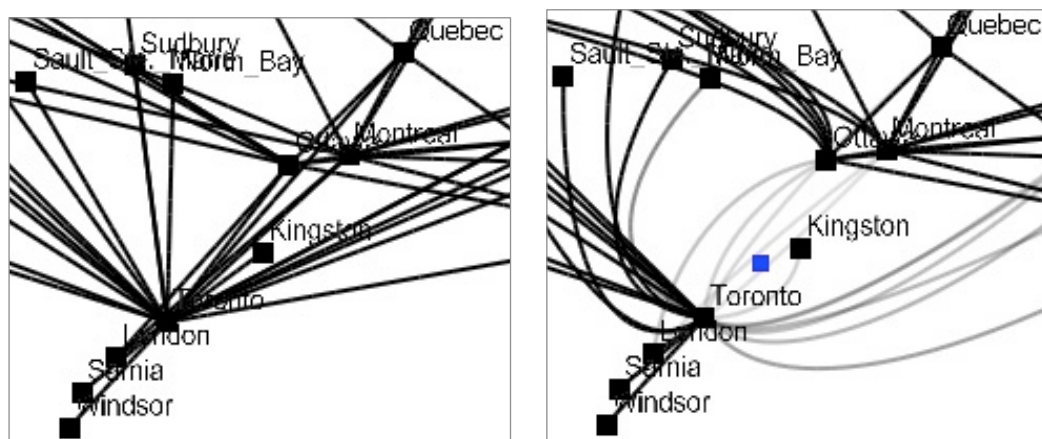
		1	2	3	4	5	6	7	8	9	10	11
		sampling	filtering	point size	opacity	clustering	point/line displacement	topological distortion	space-filling	pixel-plotting	dimensional reordering	animation
A	<b>avoids overlap</b>	possibly	possibly	possibly	partly	possibly	✓ <sup>+</sup>	possibly	✓ <sup>+</sup>	✓ <sup>+</sup>	partly	✓ <sup>+</sup>

Not all clutter reduction techniques avoid overlap.

**A1<sup>+</sup>** Sampling cannot avoid overlap altogether as data items are not displaced, but it can be used successfully to reveal hidden patterns. In the author's experience and as suggested by [Ward 02], an acceptable amount of overlap can be tolerated and allowing the user to adjust the sampling rate (or overlap amount with auto-sampling [Ellis and Dix 06c]) is an optimum solution. (Figure 3-1).

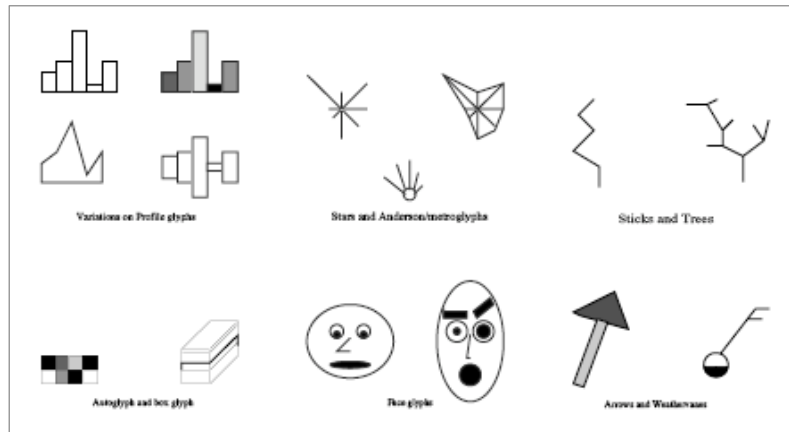
**A2<sup>+</sup>** Like sampling, filtering cannot necessarily avoid overplotting altogether, yet it can reduce the results set sufficiently to reveal the desired relationships for the chosen data range. One of the earliest examples of the use of dynamic filtering is HomeFinder [Williamson and Shneiderman 92] where the home buyer or estate agent can set the number of the bedrooms, say 2 to 4, the cost range and other parameters such as the type of dwelling (Figure 3-2). The locations of houses matching these conditions are shown on the accompanying map. Unlike sampling, the user has to decide what data to include or exclude and presupposes that the user knows or at least has some idea of this. Consequently, for exploratory activity, filtering is not so appropriate.

**A3<sup>+</sup>** Large points may conceal or partially conceal any points underneath (plotted earlier), so reducing their size may be beneficial. However, there are trade-offs between overlap and points size. If the point colour represents some attribute value, then too



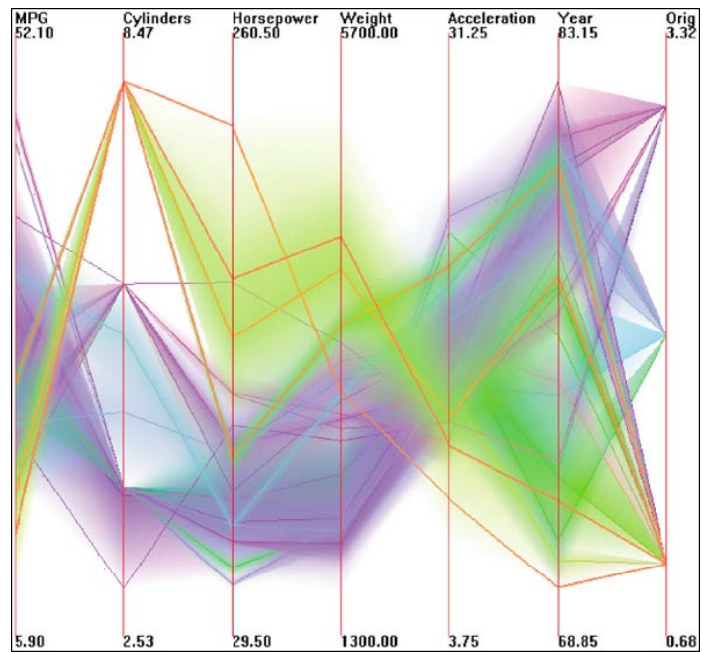
**Figure 3-3** EdgeLens [Wong et al. 03] displaces lines to both reveal the data underneath and helps to disambiguate the edges and nodes.

Figure 3-4



Examples of multi-attribute glyphs from [Ward 2002]

Figure 3-5



The clustering used in Hierarchical parallel coordinates [Yang et al. 03a] is scalable to very large datasets, only limited by computational resources.



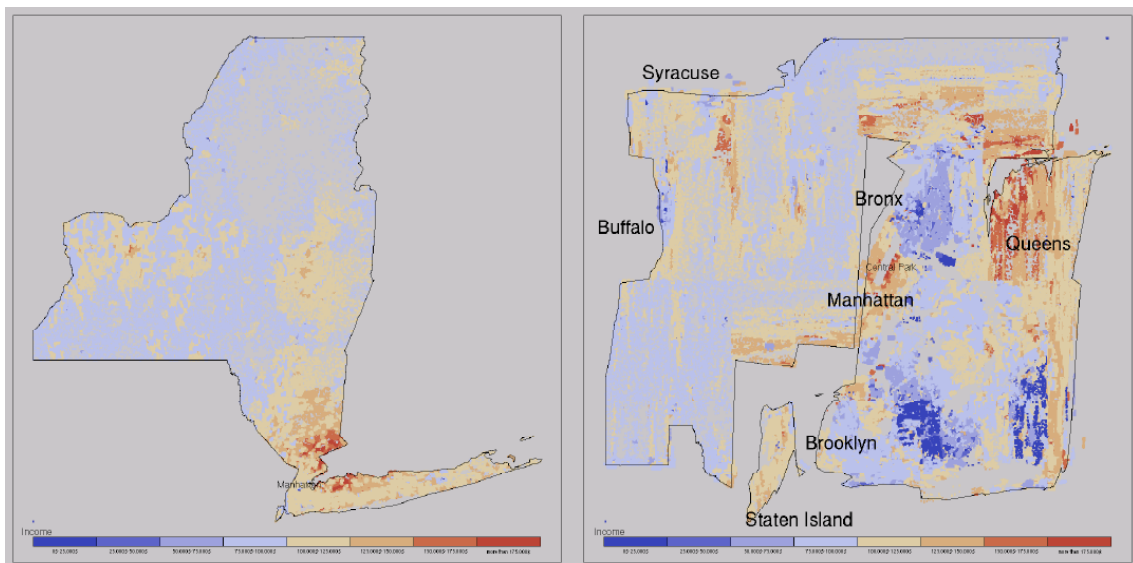
small a point makes the colour difficult to discern [Bertin 83, Ware 04]. Also, a glyph representing multiple attributes, such as those presented in Figure 3-4, may need simplifying when reduced in size, resulting in a loss of data. Furthermore, images need to be large enough to see the required detail [Derthick et al. 03].

**A4<sup>+</sup>** Although a change in opacity cannot avoid overlap, it can reveal a small number of underlying or partially overlapping points. The parallel coordinate plot shown in Figure 3-5 utilises reduced opacity to show otherwise partly hidden lines.

**A5<sup>+</sup>** Clustering here means reducing the number of data items to simplify the plot. So it can be used to avoid overplotting by either representing a group of points by a single point (the size of which represents the number of original points [Woodruff and Olston 98]) or a group of lines by a single line or band (as in some parallel coordinate clutter visualisations [e.g. Johannson et al. 06, Yang et al. 03a] – Figure 3-5). Zhang et al. [Zhang et al. 03] instead use *zip zooming* to combine several adjacent axes on parallel coordinate plots to reduce overlap.

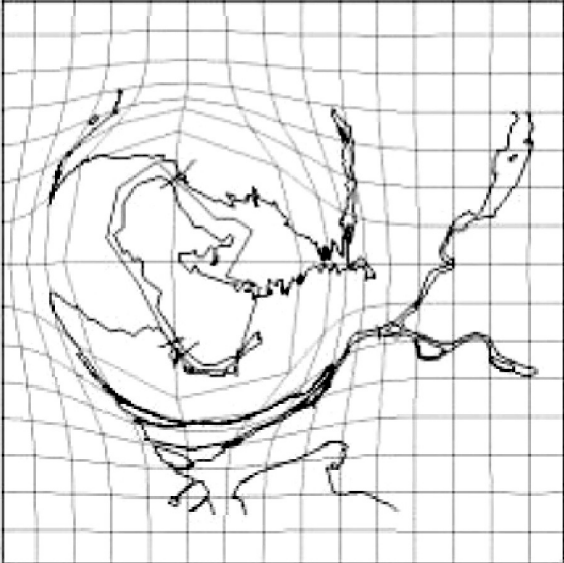
**A6<sup>+</sup>** Some visualisations that employ point displacement (e.g. smart jitter [Trutschl et al. 03], GridFit [Keim and Herrmann 98]) are specifically designed to avoid overlap. Other point/line displacement methods (e.g. random jitter [Ahlberg 96], Mobile 2D scatter space [Waldeck and Balfanz 04], EdgeLens [Wong et al. 03]) displace points or lines locally to reduce but not necessarily avoid overlap. (Figures 3-3 and 3-12).

**A7<sup>+</sup>** Topological distortion involves stretching the virtual drawing surface either uniformly (zooming) or non-uniformly to give extra display space to the data point. Some visualisations avoid overlap by virtue of their design (e.g. PixelMaps [Keim et al. 04]) while others such as pliable surface/rubber sheet interfaces [Carpendale et al. 95, Sakar et al. 93] do not. The size of the points do not change apart from Fisheye lens



**Figure 3-6** PixelMap avoids overlap altogether by distorting the underlying map [Keim et al. 04]. (Map on the left is undistorted)

Figure 3-7



The plot has undergone a topological distortion, however the overlaid grid squares do provide a reference for the user and helps to keep spatial information [Carpendale et al. 95].

Figure 3-8



In this example of an RSVP carousel, the data items obviously retain their image attribute [Cooper 06].

type distortion, if used as a magnifier. (Figures 3-6 and 3-7).

**A8<sup>+</sup>, A9<sup>+</sup>** Visualisations that use space-filling (e.g. Treemaps [Shneiderman 92]) and pixel plotting (e.g. Keim spirals [Keim 00]) are specifically designed to avoid overlap.

**A10<sup>+</sup>** Dimensional reordering as applied by Peng et al. [Peng et al. 04] arranges the scatter matrix dimensions or parallel coordinate order with a view to minimising their clutter measure.

**A11<sup>+</sup>** Animation used in Rapid Serial Visual Presentation [Spence 02] avoids overlap by showing a stack of images to the user in quick succession and in Cenimation [Engle et al. 06] overlapping data bubbles appear to float to the surface in quick succession, hence no data item is ever hidden completely. See Figures 3-8 and 3-17 for examples of RSVP and Cenimation.

### keeps spatial information

		1	2	3	4	5	6	7	8	9	10	11
		sampling	filtering	point size	opacity	clustering	point/line displacement	topological distortion	space-filling	pixel-plotting	dimensional reordering	animation
B	<b>keeps spatial information</b>	✓	✓	✓	✓	partly	* <sup>+</sup>	possibly	✓ <sup>+</sup>	possibly	✓	✓

**B1, B2, B3, B4** Sampling, filtering, changes to opacity and point size all preserve spatial information.

**B5<sup>+</sup>** Clustering, by default, loses individual spatial information but as a group (or cluster) it can show aggregate values, typically through colour/shading/opacity. Good examples can be found in [Johansson et al. 06].

**B6<sup>+</sup>** The actual point displacement depends on density of the data. The higher the number of overlapping points, the greater the spatial distortion to accommodate the points without overlap. However, as discussed in the previous section the amount of information actually lost is not necessarily directly dependent on the displacement as relative positions are perhaps a more determining factor.

**B7<sup>+</sup>** Topological distortion is similar to **B6** in that the amount of distortion from the normal x, y position is increased with the degree of overlap (density) but because the x-y space is being stretched or squashed it could be argued that the viewer does not lose spatial information, as long as sufficient landmarks are available. However, this perception of little or no spatial distortion must rely on providing appropriate spatial cues to the user. For example, Carpendale et al. [Carpendale et al. 95] attempt to inform the user of the amount of distortion by either superimposed grid or by shading (Figure 3-7).

a) Treemap

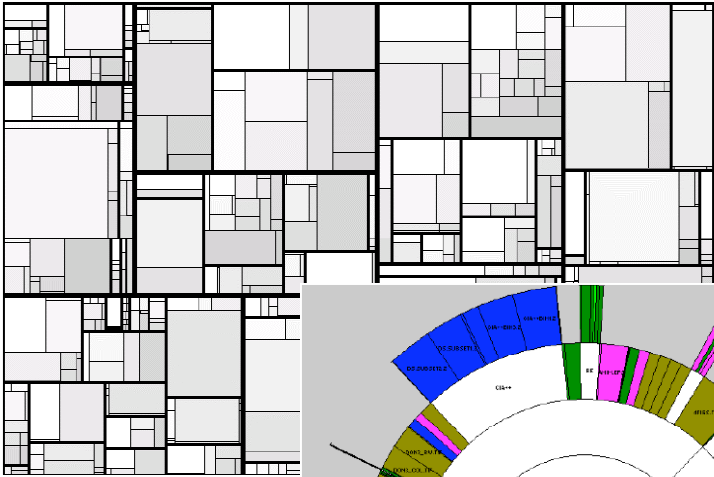
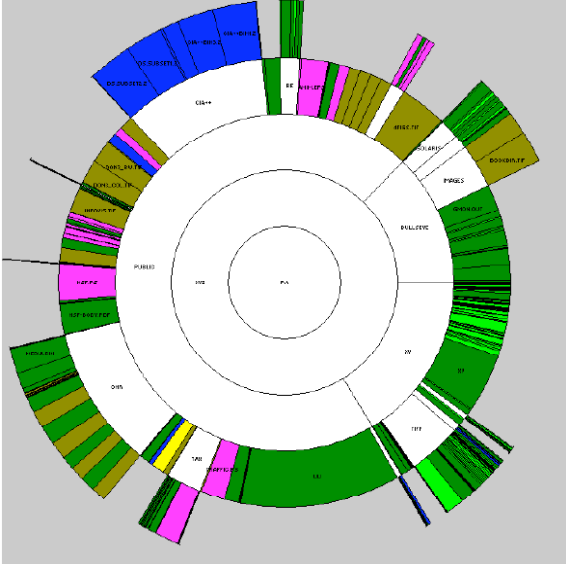


Figure 3-9

b) Sunburst



Space-filling algorithms avoid overplotting

**B8<sup>+</sup>** Space-filling techniques such as Treemaps [Shneiderman 92] and Sunburst [Stasko and Zhang 00] plot hierarchical data that is not geo-spatial in the first instance. So it is questionable whether this criterion can be applied. The level in the hierarchy is significant and as space-filling does keep this information, one could say that this criteria is met. Early space-filling algorithms suffered from often extensive reorganisation of the display with minimal changes in the data, however more recent algorithms (e.g. [Bederson et al. 02]) generate more stable ordered layouts. Figure 3-9 shows examples of Treemap and Sunburst diagrams.

**B9<sup>+</sup>** Pixel-oriented techniques such as Information Mural [Jerding and Stasko 98], TableLens [Rao and Card 94] and Pixel bar charts [Keim et al. 01] do retain the spatial information of the original data. However, pixel-spirals [Keim 00] plots data based on one of a set of packing algorithms and is not based on any attribute of the data apart from order. As with **B8**, it is questionable whether this criteria can be applied.

**B10** With parallel coordinate plots, the order of the axes is arbitrary and hence it would be difficult to argue that the position of an axis was spatially relevant in the first place. The adjacency of axes is more important in being able to show relationships between two attributes. On the other hand, the y-position of the lines crossing an axis are not affected so in this respect spatial information is kept.

**B11** Animation does not inherently displace data items. In some cases, such as RSVP with images presented on top of each other, there is no spatial dimension. Other cases of RSVP, where for instance the images are moved along a curve (as shown in Figure 3-8), the space is being utilised more for temporal ordering. This also applies to other photo viewing applications, some of which use pseudo 3D effects in intriguing ways [Porta 06]. Other animation effects are used to provide additional information, such as *jitter discs* [Brodbeck et al. 97] and *feature animation* [Johansson et al. 06]<sup>6</sup>.

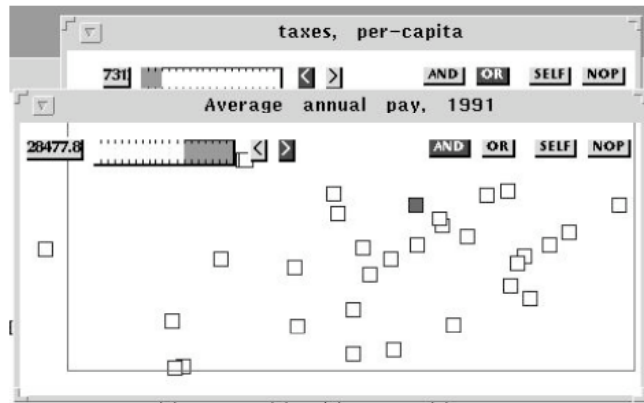
### can be localised

		1	2	3	4	5	6	7	8	9	10	11
		sampling	filtering	point size	opacity	clustering	point/line displacement	topological distortion	space-filling	pixel-plotting	dimensional reordering	animation
C	<b>can be localised</b>	✓	✓	✓	✓	x <sup>+</sup>	✓	✓	x	x	x	✓ <sup>+</sup>

Sampling can be localised to a particular region of the display by using a lens metaphor [Ellis et al. 05] and through non-uniform sampling across a scatterplot [Bertini and Santucci 06]. Likewise EdgeLens restricts line displacement to a lens [Wong

<sup>6</sup> A description of these animation effects are presented in Appendix A.11

Figure 3-10



An example of filtering with a Magic Lens [Stone et al. 94]. The spatial information is retained, although one could argue that the points that are no longer visible have been displaced significantly

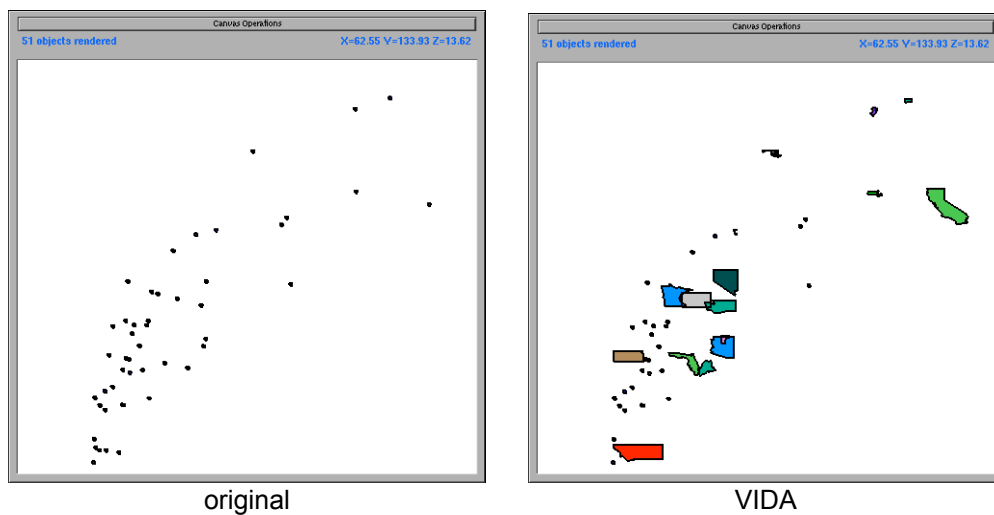


Figure 3-11 Visual Information Density Adjuster. Where there is sufficient space, the data items (which represent the states in the USA) have been shown in more detail. [Woodruff et al. 99]

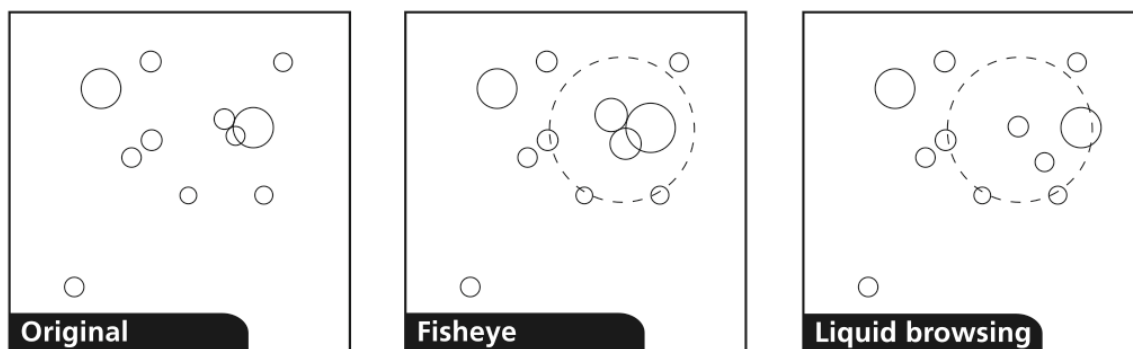


Figure 3-12 Liquid browsing displaces points locally based on the distance from the stylus position and the pressure exerted by the user (region of displacement is shown by the dotted circle). The magnifying effect of a fisheye lens is shown for comparison. [Waldeck and Balfanz 04]

et al. 03] to disambiguate node and edge relationships. By adopting a Magic Lens approach [Stone et al. 94], filtering could also be localised as well as changes to point size, opacity and animation effects. Topological distortion is localised with Fisheye lenses and with the pliable surface interfaces. However, it is difficult to see how clustering, pixel-oriented techniques or dimensional reordering could be restricted to a particular region of the display. It should be noted that localisation does not mean showing a different view of the data within a particular region of the display. For example, create an off-screen display after applying some function to the data set and then superimpose a particular section of this over the original display by means of a moveable lens. It means restricting the spatial coordinates of the data by means of a moveable region and then applying the particular function to that dataset. (Figures 3-10 and 3-3).

**C1** By means of a lens. For example the Sampling Lens [Ellis et al. 05] (See Chapter 4)

**C2** Moveable filters [Stone et al. 94, Fishkin and Stone 95] give the user the ability to apply dynamic filters to regions of the display (Figure 3-10).

**C3** By means of a lens or alternatively as a constant density display [Woodruff et al. 99] where data items in less dense regions are drawn larger (with greater detail) and vice versa. This is illustrated by Figure 3-11 where in the dense regions, the USA states are displayed as dots and in the sparse regions, they are displayed as polygonal outlines.

**C4** Possible by means of a lens.

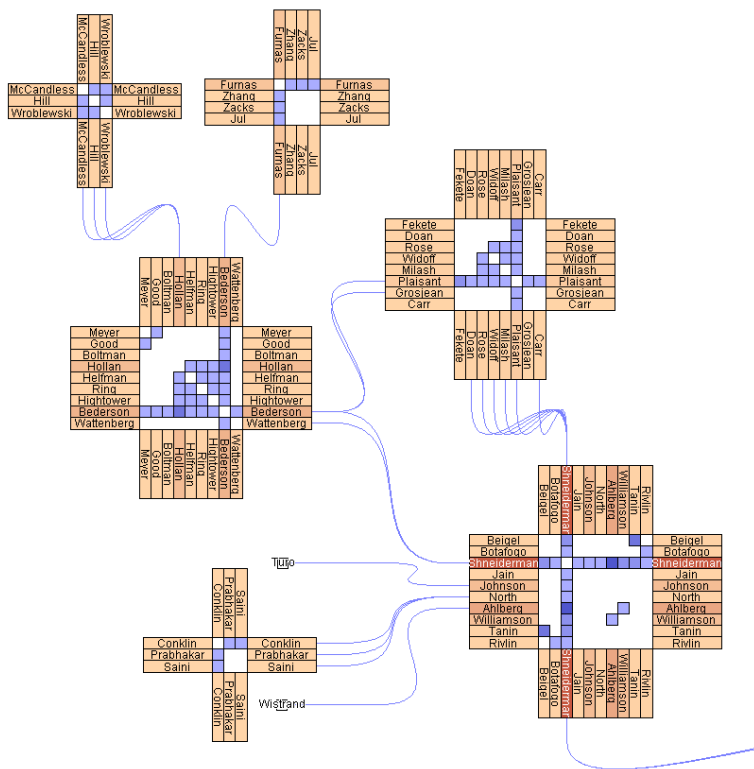
**C5<sup>+</sup>** Clustering often makes use of multidimensional data and hence restricting the spatial component to a particular region of the display does not appear to be commensurate. Clustering by its nature groups a set of data and one could anticipate problems occurring when a spatial lens was half way across a potential cluster.

**C6** EdgeLens [Wong et al. 03] displaces lines around a particular point of interest (Figure 3-3) and Mobile Liquid 2D scatter space [Waldeck and Balfanz 04] uses a distance manipulation-based expansion lens centred about the stylus position on the pen-based display (Figure 3-12).

**C7** Fisheye views may use a lens and a hyperbolic browser localises distortion to particular region - Kreuseler and Schumann [Kreuseler and Schumann 99] use the name Magic Eye View for this. Also Carpendale et al. [Carpendale et al. 95] localise distortion to particular regions of interest (Figure 3-7).

**C8** Space-filling itself cannot be localised, although one could imagine changing the level of detail within a particular space of Treemap (e.g. a high level in the hierarchy) to show more data.

Figure 3-13



NodeTrix combines a node-link representation to give an overall view of a social network with adjacency matrices giving detailed analysis of local communities. From [Henry et al. 07]

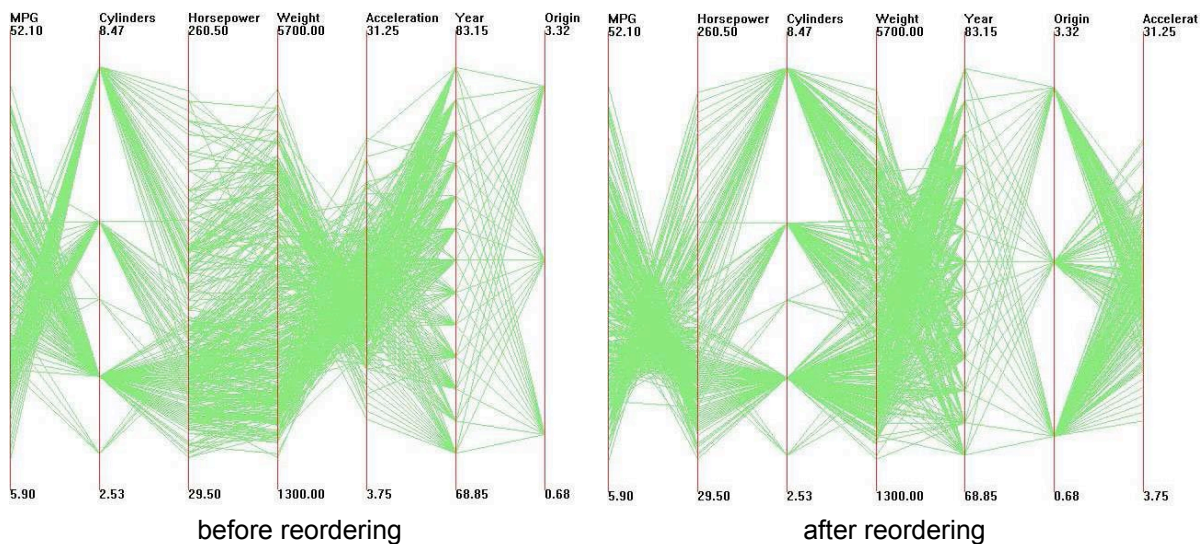


Figure 3-14

Dimensional reordering in a parallel coordinate plot. [Peng et al. 04]



**C9** Pixel plotting – the idea is to pack as much data into a small screen area as possible and therefore it is difficult to imagine how this could be localised. However, it is worth noting that NodeTrix [Henry et al. 07] uses adjacency matrices (which could presumably be plotted at near pixel scale) to show detail on demand in a node-link diagram (Figure 3-13).

**C10** The dimensional reordering algorithm employed by Peng et al. [Peng et al. 04] considers the whole plot so localisation is not possible (Figure 3-14).

**C11<sup>+</sup>** Animation can be used to show examples of the data in an overplotted area or in the case of RSVP [Spence 02], cycling through the images on a particular stack on the display (Figure 3-8). Properties of the data or derived data in the case of clustering could be animated within a localised region but this is only clutter reduction in the sense that the whole display is not cluttered.

### is scalable

		1	2	3	4	5	6	7	8	9	10	11
		sampling	filtering	point size	opacity	clustering	point/line displacement	topological distortion	space-filling	pixel-plotting	dimensional reordering	animation
D	<b>is scalable</b>	✓	✓	✗	✗ <sup>+</sup>	✓	✗	✗	✗	✗	✗	✓ <sup>+</sup>

Sampling, filtering and clustering can all be scaled up to deal with very large datasets as they all inherently reduce the number of plotted points. The limiting factor is the computational resource available. All the other technique, apart from animation, are ultimately limited by the number of pixels available on the display.

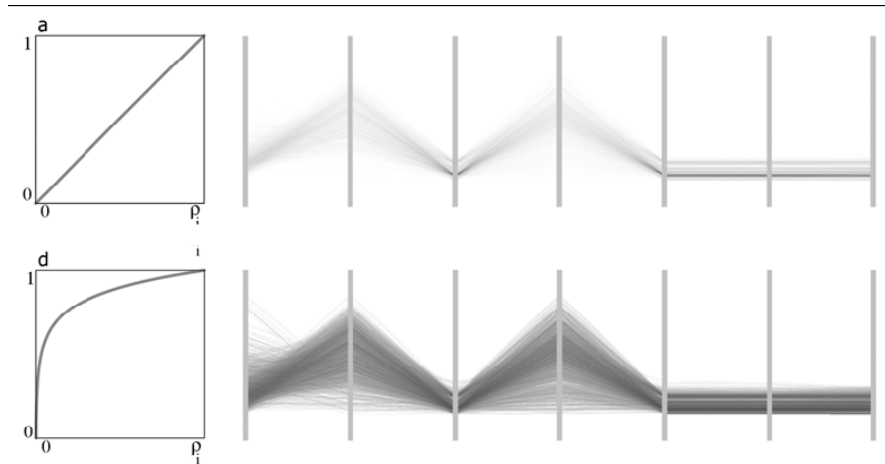
**D3** Reducing the point size is limited by the resolution of the display and visual acuity.

**D4<sup>+</sup>** Healey et al. [Healey et al. 95] suggests that opacity is only useful when up to five items overlap and hence has limited scalability.

**D6, D7, D8, D9** Limited by the number of pixels on the display and human visual acuity. With topological distortion, a very large number of data items within one area would lead to a significant spread of points (or stretching of the underlying topology), hence unmanageable distortion for the viewer (Figure 3-6).

**D11<sup>+</sup>** Animation techniques such as RSVP [Spence 02] and Cenimation [Engle et al. 06] which show data items in sequence can deal with very large datasets, however the time to show all the data would need to be taken into account (Figures 3-8 and 3-17).

Figure 3-15



Changing the visibility of structures within a parallel coordinate plot using transfer functions to map line density to opacity [Johansson et al. 06]

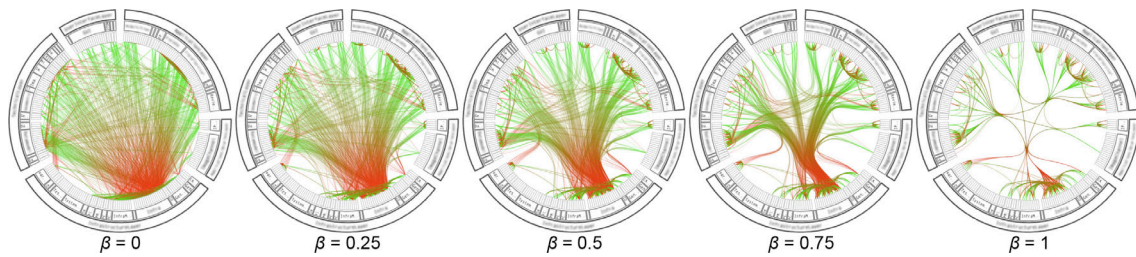
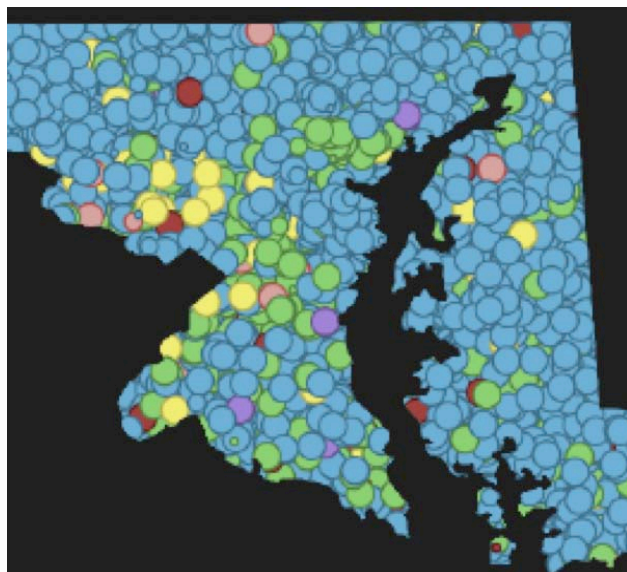


Figure 3-16

Hierarchical edge bundles [from Holten 06] with bundling strength  $\beta$  increasing from left to right. Low values provide node to node connectivity information whilst higher values reduce clutter as well as providing higher level information. Colour is used to denote direction and alpha shading helps to differentiate the lines.

Figure 3-17



Animated *bubbles* used in Ceneration to avoid permanent overlap [Engle et al. 06]

**is adjustable**

		1	2	3	4	5	6	7	8	9	10	11
		sampling	filtering	point size	opacity	clustering	point/line displacement	topological distortion	space-filling	pixel-plotting	dimensional reordering	animation
E	<b>is adjustable</b>	✓	✓	✓	✓	✓	possibly	✓	x <sup>+</sup>	x <sup>+</sup>	✓	✓ <sup>+</sup>

This criterion is looking at whether the amount or degree of clutter reduction can be adjusted interactively. Sampling rate, dynamic query range, point size, opacity and to some extent cluster size can all be adjusted. The magnification factor of localised topological distortion techniques (e.g. Fisheye lens, pliable surfaces) can be changed dynamically and in some of the dimensional reordering visualisations implemented by Peng et al. [Peng et al. 04], the user can adjust the cluster width threshold. (Figures 3-1 and 3-10).

**E4** Fekete and Plaisant [Fekete and Plaisant 02] concluded that opacity is only useful when it can be varied interactively to reveal overlaps and that it is of most use for transient inspections. Johansson et al. [Johansson et al. 06] map density values to opacity in parallel coordinate data. To achieve rapid interaction they produce high-precision textures of clusters within the plots to represent structures within the dataset, which can be rapidly manipulated with transfer functions as illustrated in Figure 3-15.

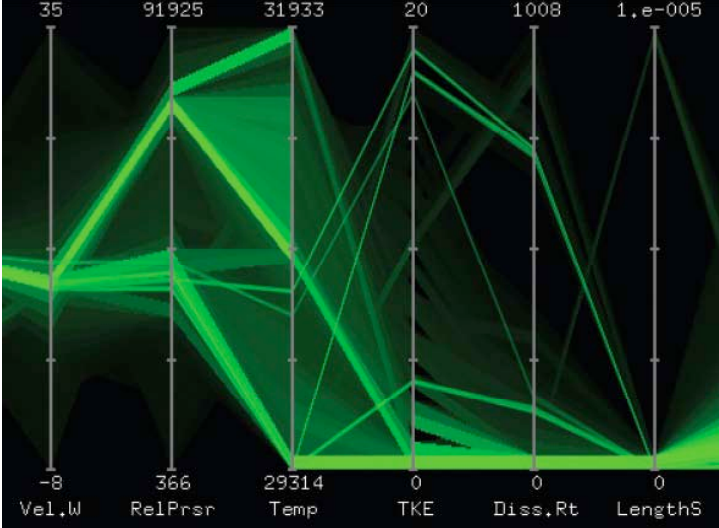
**E5** Hierarchical edge bundling [Holten 06] offers the user interactive control of the *bundling strength* to identify high level structures, as illustrated in Figure 3-16.

**E6<sup>+</sup>** Few applications allow the user to control the amount of displacement. One example is Mobile 2D scatter space [Waldeck and Balfanz 04]) which utilises a pressure sensitive pen (Figure 3-12). In applications where the technique avoids overlap (e.g. GridFit [Keim and Herrmann 98]) the amount of displacement will depend on the data density of the data items. Although they claim that points are drawn as close to their original position as possible, large displacements may cause a problem for the user.

**E8<sup>+</sup>, E9<sup>+</sup>** Although users can navigate through some space-filling visualisations (e.g. Sunburst [Stasko and Kraemer 00]), they cannot necessarily control the amount of clutter reduction. This also seems to apply for pixel-plotting visualisations.

**E11<sup>+</sup>** The animation rate can easily be adjusted for temporal techniques and this affects how the user perceives the data, but may have little effect on the clutter reduction.

Figure 3-18



Discrimination of lines in a parallel coordinate plot utilising the outlier-preserving technique of Novotny and Hauser 06 [Novotny and Hauser 06]

**can show point/line attribute**

		1	2	3	4	5	6	7	8	9	10	11
		sampling	filtering	point size	opacity	clustering	point/line displacement	topological distortion	space-filling	pixel-plotting	dimensional reordering	animation
F	<b>can show point/line attribute</b>	✓	✓	✓	x <sup>+</sup>	partly	✓	✓	✓	✓	✓	✓

All techniques apart from opacity and clustering do not affect the use of the physical attribute of the point/line (e.g. colour, shape) to represent another attribute, however one should be aware of the perception problems associated with packing pixels tightly. As mentioned earlier (A3), reducing the point size can affect the perception of colour as well. It should be noted that if there is overplotting, the point/line attributes of the top data item will be on view and hence the display is dependent on the order in which the items are plotted. Placing the items in a random order should also be considered [Keim et al. 04].

F3 Yes, but should be aware of perception problems associated with very small points [Ware 04].

F4<sup>+</sup> Reducing the opacity will diminish the significance of the data point, especially if colour is used to indicate an attribute value. In addition, the blending of colours from overlapping semi-transparent points can generate a range of unexpected colours and disable pre-attentive processing [Healey et al. 95].

F5<sup>+</sup> Clustering generally shows aggregate values rather than attributes of the raw data.

F6, F9 Yes, but should be aware of perception problems due to packing points tightly.

**can discriminate points/lines**

		1	2	3	4	5	6	7	8	9	10	11
		sampling	filtering	point size	opacity	clustering	point/line displacement	topological distortion	space-filling	pixel-plotting	dimensional reordering	animation
G	<b>can discriminate points/lines</b>	x	x	possibly	✓ <sup>+</sup>	✓ <sup>+</sup>	possibly	possibly	x	x	x	x

It seems desirable to distinguish between individual points or lines so they can easily be identified in a crowded display.

G1, G2 Does not help, apart from reducing crowding which may in turn assist in identifying individual data items. For instance, previously hidden points will be revealed if overlapping points are removed.

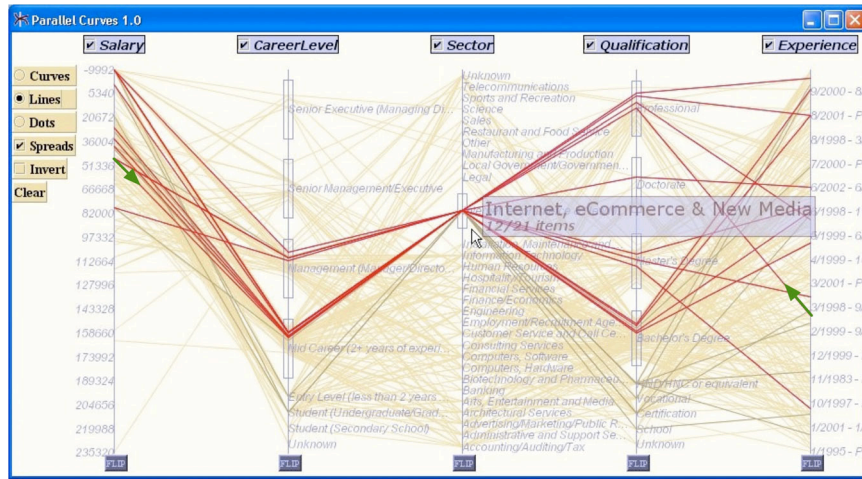
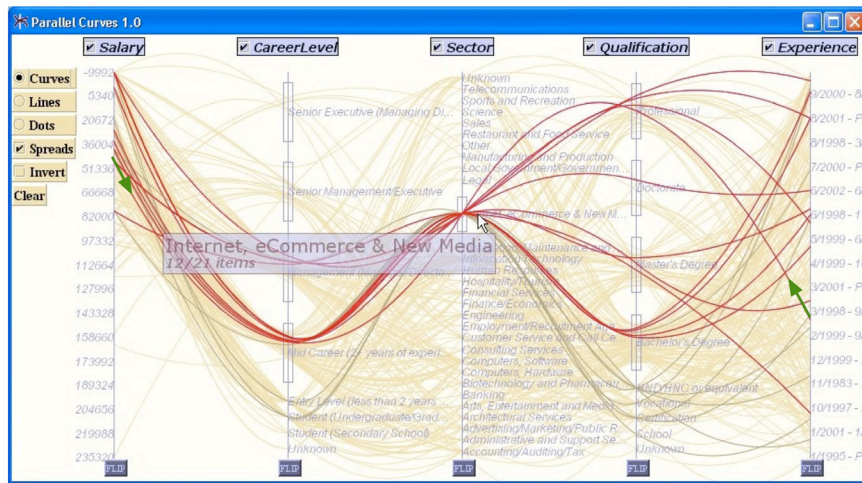


Figure 3-19



(a) Parallel coordinate plot. (b) Curving the lines of the plot to help the user follow individual lines [Graham and Kennedy 03]. Note that green arrows mark the ends of the same data record in each diagram.

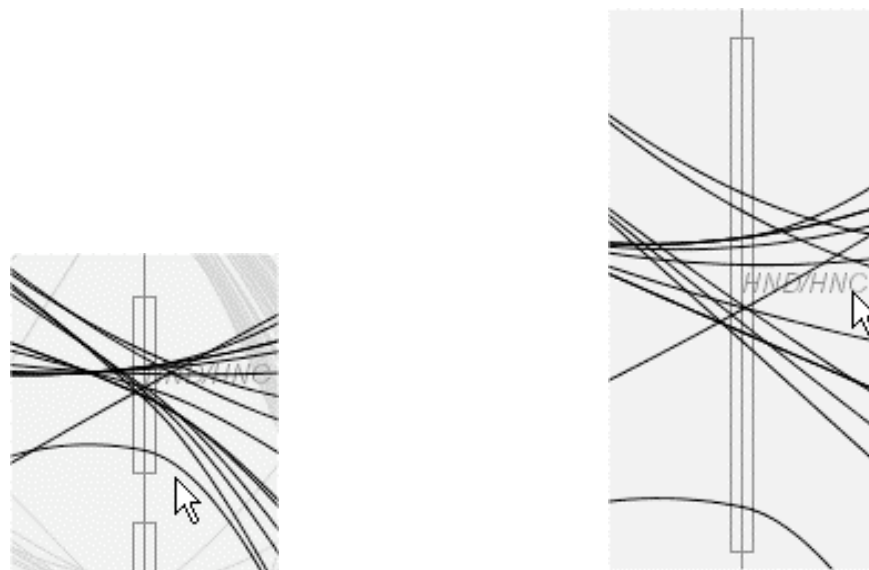


Figure 3-20

Topological distortion along axes of a parallel coordinate plot which stretches region with many lines crossing to help disambiguate the paths of the lines. From [Graham and Kennedy 03]

**G3<sup>+</sup>** As with F3, reducing the size of a point can reduce the ability to perceive its colour and hence be able to discriminate between points, if this is based on colour. A side effect of geo-spatial semantic zoom [Woodruff et al. 98b] is that outliers by definition tend to be in sparse areas of the map and hence are given additional prominence if increased in size. This is ideal if you want to identify these particular points, otherwise it may present a distorted view of the data distribution.

**G4<sup>+</sup>** Opacity is used with good effect in parallel coordinate plots to discriminate overlapping lines, especially in association with clustering [e.g. Fua et al. 99, Johansson et al. 06] (Figure 3-5). There are other appearance attributes such as colour [Fua et al. 99], blurriness [Kosara et al. 02], and texture [Johansson et al. 06] which would be effective as part of a clutter reduction strategy. These are discussed in more detail in Appendix A.2.

**G5<sup>+</sup>** Clustering algorithms can be used to detect outliers as well as create groups. This is used to good effect by Novotny and Hauser [Novotny and Hauser 06] in their outlier-preserving visualisation of parallel coordinates (Figure 3-18).

**G6<sup>+</sup>** Graham et al. [Graham and Kennedy 03] replace the polylines on parallel coordinates with smooth curves to disambiguate the lines at crossing points (Figure 3-19). Similarly, Wong et al. [Wong 03] claim that curving lines in their EdgeLens technique helps to disambiguate the connected nodes of the graph (Figure 3-3).

**G7<sup>+</sup>** Generally not met as overplotted points are not separated. However, close but not overlapping points can be given more space if the background is stretched and hence this may help to discriminate the points<sup>7</sup>. Graham and Kennedy [Graham and Kennedy 03] distort the parallel coordinate axis scale to give extra space to regions with many lines crossing (Figure 3-20).

**G8, G9** No overplotting.

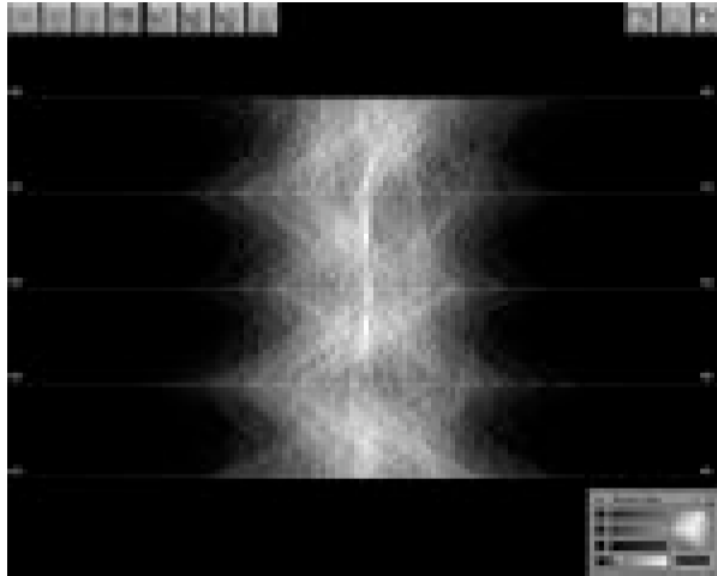
**G10** Only by the fact that lines will cross differently.

**G11** Generally not met but in RSVP applications [Spence 02], individual images are made distinct.

---

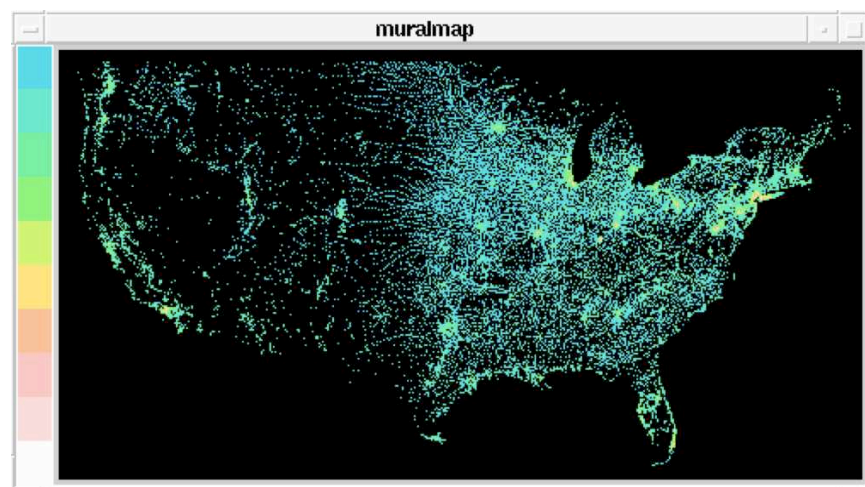
<sup>7</sup> This criterion-technique was change from *no* to *possibly* and the discussion added following the appraisal of Fisheye Menus in Section 3.5.5 and demonstrates one of the strengths of this taxonomy in provoking thought about clutter reduction techniques.

Figure 3-21



Reducing the opacity of lines can indicate the density of the overlapping lines [Wegman and Luo 96]

Figure 3-22



Information Mural utilises the display space by plotting at a pixel level. The use of colour can highlight regions with high overlap density of the original data. [Jerding and Stasko 98]



**can see overlap density**

		1	2	3	4	5	6	7	8	9	10	11
		sampling	filtering	point size	opacity	clustering	point/line displacement	topological distortion	space-filling	pixel-plotting	dimensional reordering	animation
H	<b>can see overlap density</b>	*	*	*	√ <sup>+</sup>	possibly	*	* <sup>+</sup>	* <sup>+</sup>	* <sup>+</sup>	*	* <sup>+</sup>

**H1, H2** No help.

**H3** Could argue that making the points smaller would enable more points to be seen and the user can therefore get some idea of the degree of overlap. However, this is indirect evidence and hence this criterion has been set as not met.

**H4<sup>+</sup>** With careful adjustment, opacity can lead to helpful density maps, which can assist in identifying regions of high overplotting, as shown in Figure 3-21. Fekete and Plaisant [Fekete and Plaisant 02] use the overlap count from the graphic display system mapped to opacity to indicate the density of points in dense scatterplots.

**H5<sup>+</sup>** Clustering does not inherently show the overlap density, however the aggregate value can be displayed, as discussed in **H4**.

**H6** If the data items are displaced to avoid overlap then the spread of the points could indicate the degree of the original overlap density.

**H7<sup>+</sup>** Indirectly. Carpendale's pliable surfaces [Carpendale et al. 95] can indicate the amount of overlap by showing the amount of distortion necessary to spread out the points, however this is not quantifiable. Topological distortion does not separate coincident points. (Figure 3-7).

**H8<sup>+</sup>, H9<sup>+</sup>** Obviously with techniques designed to avoid overlap there is no provision to show the overlap density. An exception to this is Information Mural [Jerding and Stasko 98], which uses an anti-aliasing compression technique to avoid overplotting in regions with more data points than pixels and can make the overlap density perceptible. (Figure 3-22).

**H11<sup>+</sup>** Note that animation-based image browsing applications [e.g. Spence 02] can, because of the height of stack of images, show the number to be browsed. But this is making use of a pseudo-3D effect, which is not part of this classification.

### 3.5. Evaluating the taxonomy

In building the Clutter-reduction Taxonomy of techniques and criteria as summarised in Table 3-6, the main concerns were based on validity – is its method of construction believable, and utility – is the taxonomy useful. We will consider both these issues in

	sampling	filtering	point size	opacity	clustering	point/line displacement	topological distortion	space-filling	pixel-plotting	dimensional reordering	animation
avoids overlap	possibly	possibly	possibly	partly	possibly	✓	possibly	✓	✓	partly	✓
keeps spatial information	✓	✓	✓	✓	partly	✗	possibly	✓	possibly	✓	✓
can be localised	✓	✓	✓	✓	✗	✓	✓	✗	✗	✗	✓
is scalable	✓	✓	✗	✗	✓	✗	✗	✗	✗	✗	✓
is adjustable	✓	✓	✓	✓	✓	possibly	✓	✗	✗	✓	✓
can show point/line attribute	✓	✓	✓	✗	partly	✓	✓	✓	✓	✓	✓
can discriminate points/lines	✗	✗	possibly	✓	✓	possibly	possibly	✗	✗	✗	✗
can see overlap density	✗	✗	✗	✓	possibly	✗	✗	✗	✗	✗	✗

Annotations in the table:

- Red box around 'is scalable' row, 'clustering' column. Arrow labeled 'strong' points to it.
- Red box around 'can show point/line attribute' row, 'clustering' column. Arrow labeled 'weak' points to it.
- Red box around 'can show point/line attribute' row, 'pixel-plotting' column.
- Red box around 'can show point/line attribute' row, 'animation' column.
- Red box around 'can discriminate points/lines' row, 'pixel-plotting' column. Arrow labeled 'popup?' points to it.
- Red box around 'can discriminate points/lines' row, 'animation' column. Arrow labeled 'cycle through attribute values for members of cluster?' points to it.

**Table 3-7** Using the taxonomy to combine the strengths of several techniques to create new visualisations

Data Characteristics	User Task	Recommended Strategy
Small to moderate size	univariate analysis	sorted structure-driven
Small to moderate size	bivariate analysis	raw data-driven
Small to large size	outlier detection	raw or derived data-driven
Moderate to large size	cluster analysis	derived data-driven, e.g., MDS
Small to moderate size	cluster analysis	hierarchical structure-driven after imposing a hierarchy
Small to moderate size	relational link analysis	network structure-driven

**Table 3-8** Sample points in a design space of glyph layout strategies. [From Ward 02, Figure 13]

this section. We will then compare the taxonomy with Ward's taxonomy of glyph placement strategies and Bertini's clutter reduction strategies, paying particular attention to the information on clutter reduction afforded to the visualisation designer. Finally, we examine the importance of criteria in extending the utility of a classification.

### 3.5.1. Validity

In an attempt to ensure the *validity* of the chosen criteria, a systematic and inductive approach has been taken. However, the inductive approach is limited by the techniques that currently exist. New techniques are being and will continue be developed and although many will fall under the broad scope of one or other of the identified classes, others will be radically new; so additional columns could occasionally be added to Table 3-6. In terms of the rows of Table 3-6, criteria tend to be more stable than technology, however it may be the case that tacit criteria are, by their nature, not stated explicitly in existing literature and have therefore not been included. There is of course the scope for adding additional criteria, but as stated at the end of Section 3.3, we need to be careful to differentiate between techniques and the actual visualisations that may employ one or more clutter reduction techniques.

### 3.5.2. Utility

In terms of utility, the categories of techniques and criteria are hopefully broad enough to apply easily. As mentioned previously, the purpose of this work is not to produce a closed categorisation and scoring of techniques, but rather as a tool for thinking about techniques and hence gaining a better insight into their use in different situations.

Just looking at the taxonomy table presented in this chapter (Table 3-6 or Table 3-7) we can see the strengths and weaknesses over a wide range of criteria for eleven clutter reduction techniques which are utilised in many visualisations. So, for instance, if one is dealing with very large datasets and hence requires a technique that scales well then row D would be a good place to start. Reading the accompanying notes for the *is scalable* criterion would make the user aware of some special cases, which may or may not be pertinent to the visualisation task.

An example of the use of Table 3-6 is to see whether techniques can be combined to allow weaker aspects of one to compliment stronger aspects of another. In fact, simple combinations do not always work, and this has often led to a deeper understand of the techniques and their interactions. As highlighted in Table 3-7, clustering is strong in terms of scalability, but weak in terms of ability to show attributes unless those attributes can be reduced to a summary statistic. At first glance, pixel plotting or space filling would seem to be complimentary, but actually, it is very hard to imagine

			examples	clutter reduction	remarks	
<i>data driven</i>	<i>raw</i>	<i>original</i>	Spotfire, scatterplot matrix	ineffective mapping can result in substantial cluttering	1-3 data dimensions used as positional components and/or graphical attributes	
		<i>distorted</i>	random jitter, Gridfit, constant density, zoom	by moving data items, aggregation of points, deletion, replacement (reduce size), level of detail (local zoom)	aim is to distort original placement to reduce clutter. need to maintain data integrity (difference between perceived and actual position)	
	<i>derived</i>	<i>original</i>	multi-dimensional scaling, self-organising maps, spring models		analytic process generates position. coordinates have no semantic meaning	
		<i>distorted</i>				
<i>structure driven</i>	<i>ordered</i>	<i>overlapping</i>			visualisation allows overlap	
		<i>space-filling</i>	Keim's spirals, raster scan [Ward 94]	effective utilisation of screen space	individual points difficult to interpret but patterns and textures visible	
		<i>separation</i>				
	<i>hierarchical</i>	<i>overlapping</i>	Cheops [Beaudoin et al. 96]		controlled overlap to compress large hierarchies	
		<i>space-filling</i>	Treemap, dimensional stacking [LeBlanc et al. 90]	no overlap		
		<i>separation</i>	node-link diagrams	links can cause clutter, also unbalanced trees	white space used to partition data items, lines may link nodes	
	<i>network</i>	<i>overlapping</i>	force-based techniques		can lead to cluttered displays	no discussion of the three classes
		<i>space-filling</i>				
		<i>separation</i>				

**Table 3-9** Expanded version of Ward's taxonomy of glyph placement strategies [Ward 02]. Examples, clutter reduction notes and remarks columns do not appear in the original paper but are assimilated from the text of the paper. Empty cells indicate that no examples or reference to that particular class was made in the paper.

combinations that would preserve clustering's scalability. Most often, as an initial observation, complementary techniques seem to be best combined through overlays, alternative views or drill downs. For example, one could imagine a pop-up pixel-plotting representation of an individual cluster.

Continuing the above line of investigation, we can look for techniques to compliment the weakness of clustering in displaying point/line attributes, such as colour, that cannot easily be aggregated. Animation techniques often do satisfy this criterion and this suggested incorporating animation that would cycle the attribute displayed for a cluster through the various specific values of its constituent members. In retrospect, this seems an obvious solution, and may well be in use elsewhere, but importantly the classification has prompted new ideas. This is again illustrated in Table 3-7.

Of course, the ultimate test of utility is whether other people can use the taxonomy in their own research or practice, and this will only become apparent in the years following its publication in late 2007 [Ellis and Dix 07].

### **3.5.3. Comparison with Ward's taxonomy of glyph placement strategies**

In Ward's paper [Ward 02] there are two classifications: an initial guide for choosing a layout strategy, which is reproduced in Table 3-8, and the taxonomy, which is the first three columns of Table 3-9. The first of these (Table 3-8) is a very general guide that suggests a placement strategy based on the size of the dataset and the user task. So, if the user is interested in detecting outliers then a strategy of raw or derived data-driven is recommended. Unfortunately, no indications are given of what constitutes a moderate or large size.

Now, considering the taxonomy, we can look at the visualisation examples given for data-driven placement and take note of the strengths and weaknesses mentioned in the accompanying text<sup>8</sup>. For example, the major strengths of the raw data-driven strategy (e.g. scatterplots or scatterplot matrices) are its ease of interpretation, low computational overhead and effectiveness of uncovering correlations. Limitations include that only pair-wise relationships may be revealed (as only 2 dimensions are ever plotted against each other in one scatterplot) and that occlusion is a significant concern, especially with inappropriate dimensional mapping. To help alleviate this occlusion, the accompanying text points to various distortion techniques as presented in the raw→distorted row of Table 3-9.

In addition, there is some guidance on displacement – a major trade-off is ease of implementation versus accuracy of the result. This is useful for a designer, but we

---

<sup>8</sup> Note that the additional clutter reduction and remarks columns of Table 3-9 are generally related to clutter reduction and hence do not include all the information presented in the original paper, such as the aforementioned strengths and weaknesses.

strategies	methods	
<i>visual density reduction</i>	<i>suppression</i>	sampling filtering
	<i>subsetting</i>	topological distortion
	<i>level of detail</i>	change point size clustering
<i>spatial organisation</i>	<i>layout</i>	space-filling point/line displacement pixel-plotting
	<i>ordering</i>	dimensional reordering
<i>retinal properties</i>	<i>brightness</i>	
	<i>colour mapping</i>	
	<i>transparency</i>	change opacity
	<i>shading</i>	

**Table 3-10** Comparison of Bertini’s clutter reduction methods [Bertini 07] with the clutter reduction techniques used in the Clutter-reduction Taxonomy. Note that the techniques in green boxes are in the appearance group and those in blue boxes are in the spatial distortion group. Animation is not included in Bertini’s classification.

cannot get an overview of this information from the original taxonomy table (first three columns) and instead we have to search through the text. Furthermore, a fair number of the classes are not mentioned in the text, including overlapping and separation classes within ordered and network structure-driven strategies. This is surprising as these are two of the four main glyph placement issues set out near the beginning of the paper and one might question whether the taxonomy is as comprehensive as hinted at in the paper's abstract.

In terms of clutter reduction, Ward's taxonomy is somewhat limited in its use as demonstrated by the gaps in the clutter reduction notes column of Table 3-9. Obviously, clutter reduction is not the main aim of the classification but is nevertheless an important aspect of visualisation design. He does point out that there is often a trade-off between efficient screen use, degree of occlusion and amount of distortion and that the user should ideally be able to dynamically adjust parameters to achieve an appropriate balance between these factors. In addition, the user should be able to choose more than one visualisation technique. However, achieving this desirable aim would be difficult with his classification as visualisation techniques are not explicit, apart from references to actual visualisations, and criteria with which to make a comparison are not made available.

Ward's taxonomy of glyph placement is based on how the spatial arrangement of a point is determined – either extracted from data values (actual or calculated) or from the structure of the data (e.g. hierarchical). Strengths and weaknesses of many of the classes are given in the accompanying text together with visualisation examples. Some useful guidance on clutter reduction is given but only for some classes. However, in contrast to the Clutter-reduction Taxonomy, visualisation techniques and comparison criteria are not explicitly stated. Ward also presents a guide for choosing a layout strategy based on dataset size and user task, but there is little explanation or discussion of its use.

#### **3.5.4. Comparison with Bertini's clutter reduction strategies**

Table 3-10 shows a comparison of Bertini's classification of clutter reduction techniques with those used in the Clutter-reduction Taxonomy.

We can see from Table 3-10 that Bertini's visual density reduction and spatial organisation strategies match quite closely with the choice appearance group of techniques (green boxes) and spatial distortion group (blue boxes). The only exceptions being topological distortion and change opacity. It is also clear from Table 3-10 that many of Bertini's methods are less specific. These differences will now be discussed.

Under the *visual density reduction strategy* Bertini classifies sampling and filtering examples as suppression where, as discussed in Section 3.4.1, there is an important

strategy	method	examples	clutter reduction	distortion possibilities
<i>visual density reduction</i>	<i>suppression</i>	sampling, filtering	reduce number of data items	
	<i>subsetting</i>	zooming in	reduce number of data items by changing x-y scale	allocate more screen space to detriment of adjacent areas (e.g. Fisheye)
	<i>level of detail</i>	reduce size, aggregation, clustering	reduce screen area covered by data items, reduce number of data items	size of object dependent on screen space (Woodruff's constant density)
<i>spatial organisation</i>	<i>layout</i>	graphs, Treemap	minimise arc crossings (graphs), low average aspect ratio (Treemap)	random displacement (jittering), displacement calculated by optimisation algorithm (e.g. Keim GridFit)
	<i>ordering</i>	pixel plotting, parallel coordinates axis reordering	pixel plots made more effective, reduce line crossings in parallel coordinates	
<i>retinal properties</i>	<i>brightness</i>	Information Mural [Jerding and Stasko 98]	conveys data density	magic lenses provide local distortion, non-linear mapping of colours to graph edges [Herman et al. 00]
	<i>colour mapping</i>			
	<i>transparency</i>		detect hidden objects	
	<i>shading</i>	Vis. Million Items [Fekete and Plaisant 02]	disambiguate overlapping shapes	

**Table 3-11** Expanded version of Bertini's clutter reduction strategies [Bertini 07]. The two columns to the left are the original classification. The content of the three columns to the right (examples, clutter reduction comments and distortion possibilities) have been extracted from the published text and added to the original classification.



difference in that sampling does not require the user to make choices on what data to exclude. Although subsetting is included within the same strategy, it can be argued that subsetting involves a spatial distortion of the background rather than an operation on the data items and hence it should be classified as topological distortion. Furthermore, unlike Bertini who regards both reduce size and clustering examples as level of detail methods, these have been classified separately as change point size and clustering (see Section 3.4.1). Aggregation has not been included as this more as pre-processing, as mentioned in Section 3.2.

Under *spatial organisation* strategy, layout is similar to Ward's hierarchical and network classes of structure-driven strategies. Treemaps are down as being a space-filling technique. Bertini's ordering method is similar to Ward's ordered class. It is difficult to see how the ordering of points in the pixel-plotting example is a clutter reduction method, when in fact pixel-plotting itself is a valid technique. Similarly, the reordering of parallel coordinate axes as dimensional reordering is more specific than Bertini's ordering method.

Of the four methods under *retinal properties* strategy, only transparency has been taken selected and is referred to as change opacity. Brightness and colour mapping are useful to highlight and differentiate data items but these have been excluded as mentioned in Section 3.4.1.

It appears that the point/line displacement technique is not included in Bertini's classification, however in the accompanying text the author suggests that displacement is a form of spatial organisation distortion. In addition, he considers focus+context techniques such as Fisheyes and hyperbolic mapping as subsetting distortion, whereas it can be argued that these are topological distortion techniques, which better model the process by which the data items are given more space.

Animation is mentioned by Ward in the context of smooth transition between original and distorted views but neither authors class animation as a clutter reduction technique. Finally, like Ward, Bertini suggests that some degree of automatic clutter reduction is desirable, provided that changes to the display do not disorientate the user. Automatic clutter reduction is an integral part of this work and is demonstrated later in Chapter 5 through auto-sampling.

We will now consider the usefulness of Bertini's clutter reduction strategies. To aid this, an expanded version of the clutter reduction strategies is given in Table 3-11 that includes examples, together with an explanation (as summarised in the clutter reduction column) taken from the accompanying text of his thesis. In addition, a column has been added summarising the distortion possibilities, discussed by Bertini.

			examples	clutter source	remarks
<i>placement</i>	<i>strategy</i>	<i>partitioning</i>	Treemap, PixelMap	graphic element (line/white space) required to divide the space	non-overlapping but not necessarily space- filling
		<i>overlapping marks</i>	Scatterplot, parallel coordinates	partial or total overlap	
	<i>degree of freedom</i>	<i>fixed</i>	maps, scatterplots	overlapping points/lines	no degree of freedom
		<i>constrained</i>	Treemap, parallel coordinates, TableLens	lines may overlap (parallel coordinates)	Treemap columns and parallel coordinates axes constrained by the order of the original data but may be reordered
		<i>free</i>	ball-and- spring graphs, multidimens ional scaling	graph edges may overlap	no spatial ordering but often relative positioning implies the strength of the relationship
	<i>visual marks</i>	<i>pixel</i>	Keim's pixel plotting	total overlap or visual interference of neighbouring pixels	no partial occlusion as elementary graphic element
<i>line</i>		parallel coordinates	intersection noise, saturation, ambiguous patterns	includes curved and poly lines	
<i>area</i>		scatterplots, Treemaps	partial or total overlap	includes points (>1 pixel)	
<i>text</i>		labels	distraction, overlap other lines/areas	must avoid overlap and be attached to object	

**Table 3-12** Expanded version of Bertini's design space characterisation [Bertini 07]. The examples, clutter source and remarks columns do not appear in the original publication – their content has been extracted from the published text.

As with Ward's taxonomy, there is no attempt to assess or compare the chosen clutter reduction methods and hence, apart from making the designer aware of some of the available methods, it is difficult to imagine how they could "revise and improve the tools they have built", one of Bertini's stated aims. Another aim is to "make them [designers] aware of undesired degradation effects some new designs may have" and whilst there is little in the section on clutter reduction strategies towards meeting this aim, he does provide a useful analysis of the visualisation design space - this is given in the first three columns of Table 3-12. The other columns are summaries of relevant information within the published text. We will now compare this design space characterisation with the Clutter-reduction Taxonomy and Ward's taxonomy.

The visual marks classification provides a useful discussion of the different types of graphic elements, indicating possible sources of clutter. In contrast, the Clutter-reduction Taxonomy is based on a classification and assessment of clutter reduction techniques rather than a list of graphic elements, however the techniques do of course operate on these graphics. For example, displacement of points and lines, opacity, sampling and change point size.

*Text* has not been considered explicitly as a visual mark, but there is no reason why the aforementioned techniques cannot be applied to the text labels on a chart. For example, text can be displaced and the size of text is changed by cartographers to fit the available space, which was behind Woodruff's constant density visualisation [Woodruff et al. 98b]. In semantic zooming maps (e.g. Google maps) the detail which is displayed, including place and street names, is filtered based on the map scale. Thinking of single line display screens, often found on trains or busses or in a doctors waiting room, text is animated by way of a virtual scrolling display if there is insufficient space to hold the complete message. Also, topological distortion has been applied to lists of text in Fisheye Menus [Bederson 00].

We have seen that Bertini's classification is focused on clutter reduction as illustrated by his top-level strategies of visual density reduction, spatial organisation and retinal properties. As demonstrated by the extended version of his classification (Table 3-12), visualisation examples, an explanation of clutter reduction mechanisms and distortion possibilities are included for most of the chosen methods within the accompanying text, but it is difficult to make a comparison between methods. In Bertini's design space characterisation, he discusses how different placement strategies and types of visual marks can produce clutter. This is useful in making a visualisation designer aware of sources of clutter but, as before, a summary table similar to Table 3-12 would add to its usability.

		1	2	3	4	5
		straw bale	rammed earth	stone	wattle and daub	papercrete
A	good insulation	✓	✓	✗	✗	✓
B	high thermal mass	✗	✗	✓	✗	possibly
C	easy to build	✓ <sup>+</sup>	possibly	✗ <sup>+</sup>	✓	✓ <sup>+</sup>
D	decorative	✗	✗	✓	✓	possibly
E	durable	✗ <sup>+</sup>	✗	✓	✗ <sup>+</sup>	possibly
F	good resistance to earth quakes	✓	✓	✗	✗	✗ <sup>+</sup>
G	load bearing	possibly	✓	✓	✗	✗

**Table 3-13** A comparison of natural building techniques for walls. Adapted from information on wall systems by M.G.Smith [The Natural Building Network]

### 3.5.5. Criteria are important

As we have seen in Sections 3.5.3 and 3.5.4, neither Ward nor Bertini's classifications are criteria based, so this section examines the importance of criteria in developing a classification. To help with the discussion we will consider the scenario of constructing an eco-house.

When an architect is designing a building, they have to make many choices, one of which will be the type of wall building material and technique to use. Making this comparison requires a set of criteria, and a table, such as that given in Table 3-13 would presumably be useful to the architect to get an overview of the possibilities.

Some of the criteria, such as *good insulation* can be given as a yes or no (tick or cross) as it is fairly easy to measure that particular property, whereas others, such as *easy to build* are perhaps dependent on various factors that need to be taken into account. Therefore, notes can accompany the comparison table, such as :

- B5 Thermal mass varies with max.
- C1 Construction is relatively quick but requires experienced designers/builders.
- C2 Very labour or machine intensive. Requires formwork.
- E1 Very susceptible to moisture damage. Must be protected from rain at all times.
- E5 New technique. Water resistance is questionable.

In addition, we may give some illustrations of some of techniques in use.

Clearly, the above example has been presented using a similar approach to the taxonomy in Section 3.4 to illustrate several important points. First, the compact table provides a very good overview of the data, so we can get an idea of the benefits (and conversely, weaknesses) of different techniques. Second, the user is presented with additional information where indicated, such as the requirement for experienced designers when building with straw bales. Last, but most importantly, are the criteria, without which comparison is difficult. The original data [The Natural Building Network] is presented in a tabular format (see Appendix E), but with headings such as structural/thermal properties, advantages and disadvantages. Although most of the information is contained within the table, it is not particularly easy to make comparisons between techniques in this form.

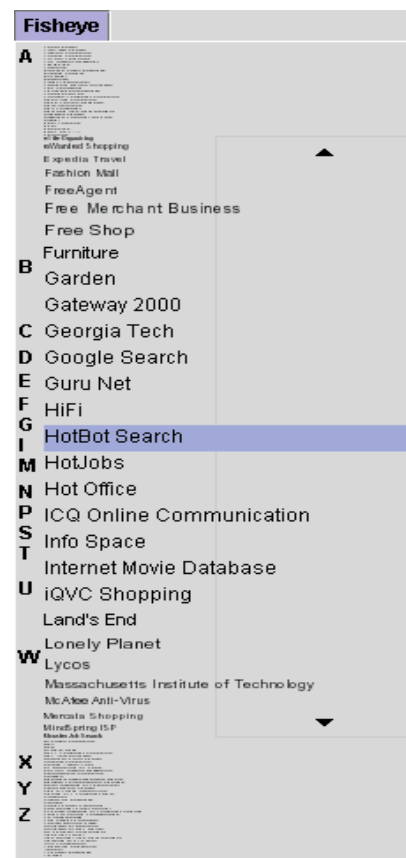
Besides the ability to compare techniques, whether these are for building walls or reducing clutter, the use of criteria have two important functions, both involve promoting thought. Defining the criteria in the first place requires much reflection on what is relevant to the user of such a classification. For the Clutter-reduction Taxonomy, the help of other researchers was enlisted in the form of their publications describing new visualisations. However, as discussed in Section 3.3, it was not easy to come up with a set of generally applicable criteria. Having established the criteria, one is literally forced to fill in the boxes of the table. There is no escape. Each technique

Table 3-14

		7
		topological distortion
A	avoids overlap	possibly
B	keeps spatial information	possibly
C	can be localised	✓
D	is scalable	x
E	is adjustable	✓
F	can show point/line attribute	✓
G	can discriminate points/lines	possibly
H	can see overlap density	x <sup>+</sup>

The clutter reduction taxonomy for the topological distortion technique

Figure 3-23



Fisheye Menu [Bederson 00]

has to be judged against each criterion and in the case of the Clutter-reduction Taxonomy, this invoked much analysis, as shown by the notes, which accompany the taxonomy table.

Therefore, by opting for a criteria based classification, we end up with not just a classification, but also with a systematic, in depth analysis of the techniques, which itself can promote thought about existing visualisations and help with the creation of new visualisations.

### **The Fisheye Menu example**

An example of thinking about a visualisation and understanding its mechanism with the aid of criteria, came about whilst writing Section 3.5.4. The Fisheye Menu is an example of topological distortion being applied to lists of text. Now, let us unpack the Fisheye Menu using the Clutter-reduction Taxonomy criteria. To help the discussion, the relevant part of the taxonomy is given in Table 3-14 and an example screen shot of a Fisheye Menu is given in Figure 3-23.

The text in lists, by design, does not overlap so criteria A and H (avoids overlap and can see overlap density) do not apply. The Fisheye Menu is localised so criteria C is met already, it is adjustable, spatial information is kept, it can show point/line attribute such as a highlight, but it is not scalable to very long lists due to a lower limit on the font size. Note that topological distortion does not change the size of the data items (in this case the font size) directly, but makes more space available for the text to be drawn in a larger font. This is different to the operation of a Fisheye lens which, as noted in the taxonomy discussion, acts as a magnifier.

An earlier version of criteria G suggested that topological distortion cannot discriminate points/lines, since stretching the background does not alter any existing overlap. However, one could argue that data items that are close but do not overlap, in this case lines of text, are given more space, which in turn enables them to be displayed larger, does indeed help to discriminate the textual data items. After thinking about topological distortion further and considering other examples, this particular technique-criterion was modified accordingly.

As mentioned above, this Fisheye Menu example was picked spontaneously whilst writing the previous section, is further evidence that the Clutter-reduction Taxonomy is very useful in prompting discussion about all kinds of visualisations.

## **3.6. Summary and reflection**

This chapter has presented the Clutter-reduction Taxonomy for information visualisation. Its construction is novel in several ways. First, it is based on a thorough survey of the literature, both to select a set of clutter reduction techniques that





represent the wide range of methods used in current visualisations and perhaps more importantly, to generate a set of criteria, expressed as benefits, with which to assess each technique. Second, evidence from the literature and the author's personal experience has assessed each technique whether it satisfies each criterion. Some special cases have been identified together with those which are only met or partially met in certain situations. Discussion of these cases and other comments are logically organised, which makes browsing particularly easy.

The task of selecting a set of clutter reduction techniques is not easy. After the initial survey of the literature resulted in fifteen techniques that were subsequently reduced to eleven which were distinct and represented a wide range in common use. Other techniques such as aggregation and dimensional reduction could well have been included, but on balance they are not used interactively and hence do not feature in the final list. Colour mapping should have perhaps been included, especially as this is referred to in the taxonomy as a method for discriminating between points and lines. However, as a clutter reduction technique, colour mapping is difficult to define.

The differences between *point/line displacement* and *topological distortion* were deliberated over. Although it may appear on the display that those techniques which move points should be classified as displacement, it is important to differentiate between moving a point relative to its original position within the frame of reference, and distorting the frame of reference. It can be argued that the difference from the users point of view is influenced by the Gestalt Law of Common Fate<sup>9</sup>. Therefore an action, which essentially stretches or distorts the background on which the points are set (the frame of reference), is perceived as moving the points collectively rather than individually and hence is easier to comprehend. One only has to watch a user apply a fisheye distortion to a scatterplot or map to realise that this is viewed by the majority as a collective displacement due to a topological distortion of the background on which the points sit.

Following the presentation of the taxonomy, a case was made for its validity and utility. The validity is based on a sound systematic and inductive approach to its creation, whilst the utility was demonstrated by several examples of its use in developing new visualisations and the prospect of gaining a better insight into the use of techniques.

A comparison was then made between the Clutter-reduction Taxonomy and two other classification schemes related to clutter reduction for information visualisation. Ward [Ward 02] and Bertini's [Bertini 07] work are both useful in summarising available

---

<sup>9</sup> "The law of common fate states that when objects move in the same direction, we tend to see them as a unit." [http://infovis-wiki.net/index.php?title=Gestalt\\_Laws](http://infovis-wiki.net/index.php?title=Gestalt_Laws)



methods or classes and grouping these helps the user consider similarities and differences. However, searching the accompanying text for examples and explanations limits their use and it is unfortunate that neither author has extended their tables, as in Tables 3-9, 3-11 and 3-12, to provide a more useful overview and incidentally expose omissions. Their usefulness to a visualisation designer is also limited, especially when it comes to combining more than one technique, as there is no clear method of comparing techniques.

We finally looked at why the use of criteria is important in providing a way to compare techniques. A criteria-based classification often leads to a more accessible overview of the data. The act of choosing the criteria and being forced to think about each technique in relation to each criterion is very beneficial in devising the classification in the first place. The Fisheye Menu example demonstrated the benefit of using a criteria-based classification to think about an existing visualisation.

One of the driving forces behind the development of this taxonomy was to assess where sampling fits into the gamut of clutter reduction techniques. This is addressed in the next chapter. However, the process of its construction has prompted a thorough examination of clutter reduction and has produced a tool that can act as a guide to match techniques to problems where different criteria may have different importance. More importantly, this taxonomy provides a means for information visualisation designers to suggest new techniques and critique existing ones.



## Chapter 4

# Clutter reduction: random sampling and lenses

In Chapter 3 we considered a proposal for a random sampling-based application, the Astral Visualiser, which raised a number of issues relating to sampling, such as display continuity and suggested the Reality Check function to reassure users. The proposed z-index method looked well placed to address many of these issues and in addition, we saw that sampling showed much promise as a clutter reduction technique.

In this chapter, we will describe the implementation of these features through the development of some sampling visualisations, applying the z-index method to generate the necessary data samples. We uncover the strengths and weaknesses of sampling by comparing it with some other clutter reduction techniques and show the viability of combining techniques. Furthermore, the possibility of localised sampling in the form of a lens, is investigated.

We will also encounter some lens re-sampling animations, enabled by the improved interactive performance and describe two novel lenses, one of which demonstrates the effective use of sampling to enhance a standard feature of parallel coordinate applications.

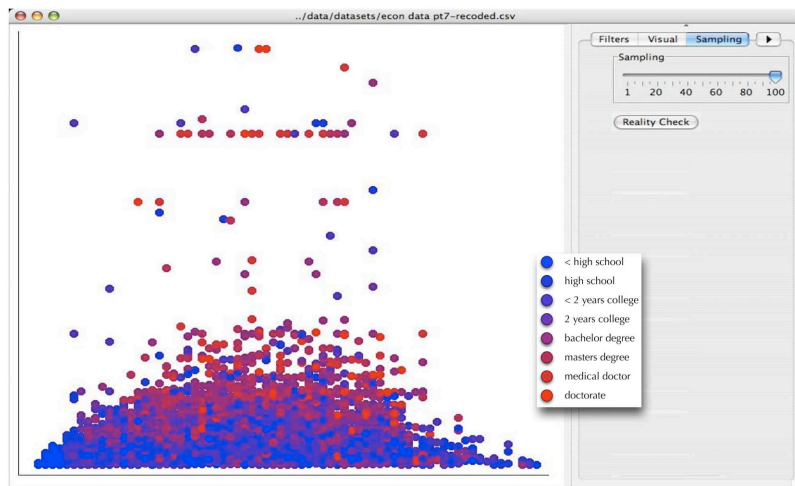
Section 4.1 describing sampling-based scatterplot and parallel coordinate visualisations. The implementation of the Reality Check is also described together with an illustration of its use.

Section 4.2 compares some clutter reduction techniques including change opacity, change point size, filtering and now sampling using the same dataset to assess their effectiveness.

Section 4.3 considers the problem of sampling datasets with a wide range of overlap densities across the plot. A possible solution of a moveable lens with its own sampling control is proposed. The features of this Sampling Lens application are described, together with some implementation details, including the extension of the z-index method to generate the lens samples.

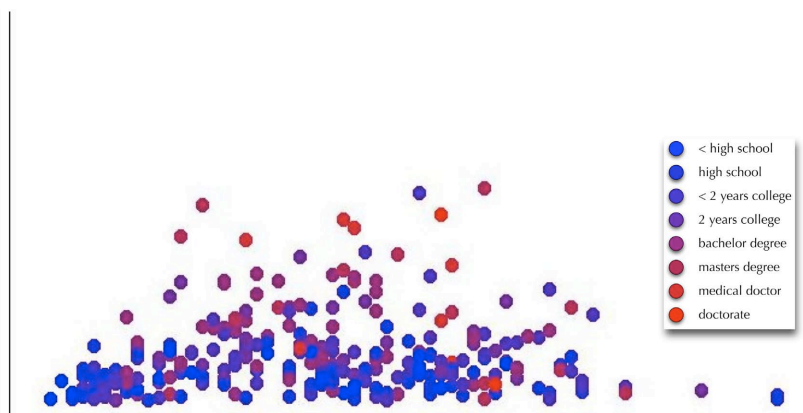
Section 4.4 illustrates some additions to the visualisation, including two special purpose lenses for parallel coordinates (the inter-axis and axis lenses), a related data visualiser and several experimental Reality Check transitions.

Figure 4-1



Sampling-based scatterplot visualisation showing age (horizontal axis) versus monthly income (vertical axis) for a sample of 9432 people [SIPP 2004 dataset]. The point colour represents their highest educational achievement. The key was added later.

Figure 4-2



Scatterplot of the age-income-education data now reduced to 188 points (2% sampling rate). Although there is still some degree of overlap we can clearly see the distribution that not only gives the spatial information but also a sense of number of points in each educational achievement category.

Finally, in Section 4.5, we reflect on the main issues that arise from the development of the sampling-based applications, including the Sampling Lens and summarise the benefits and disadvantages of sampling.

## 4.1. Sampling-based scatterplot and parallel coordinates

The initial sampling visualisation was developed to investigate the basic premise that for visualisations with significant overplotting of points or lines, random sampling can be used to reduce the number of data items and hence the visual clutter. This section describes the first sampling-based applications for scatterplots and parallel coordinate plots. The InfoVis Toolkit [IVTK], a Java open source interactive graphics toolkit was used to develop the sampling applications, although a large amount of low-level code was added to achieve the required functionality. A description of the InfoVis Toolkit, including its architecture is given in Appendix D.2.

### 4.1.1. Basic sampling

The first application to be built was a scatterplot visualisation as shown in Figure 4-1. A set of tabbed control panels on the right of the display are generated automatically by the InfoVis Toolkit which allow the user to set dynamic filter ranges for all the variables and also visual attributes of the data points (e.g. colour, size, opacity). In addition, eccentric labels [Fekete and Plaisant 99] can be shown and a fisheye lens if so desired. When not all the control panel tabs can be displayed due to lack of space, they can be selected from a drop-down list. A sampling control panel was added with a sampling rate slider ranging from 100% (no sampling) down to 1%, in 100 steps.

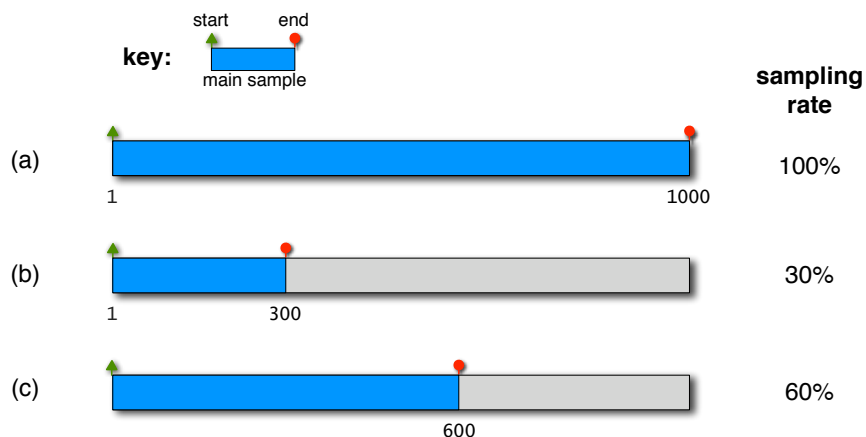
The data used for this scatterplot visualisation was extracted from the SIPP 2004 dataset<sup>1</sup>, and shows age (horizontal axis) versus monthly income (vertical axis) for a sample of 9432 people, with the point colour representing their highest educational achievement (the key was added later to the screen shot). We can see that monthly income tends to increase with age for the first 10 or so years (the data starts at age 17) and reduces noticeably when people retire. Also, it appears that more qualified people tend to be paid more, but with almost 10,000 plotted points there is considerable overlap, hence we cannot be certain of this fact as the trend depends on the order the points are plotted. However, in this example, they are plotted in a random order so the distribution would tend to be more representative, although as discussed in Section 2.4 (types of sampling), the trend is dependent on the number of points in each education achievement category.

Figure 4-2 illustrates the reduction in display density when reducing the number of data points to 188 by taking a 2% random sample of the original data. Not only can we

---

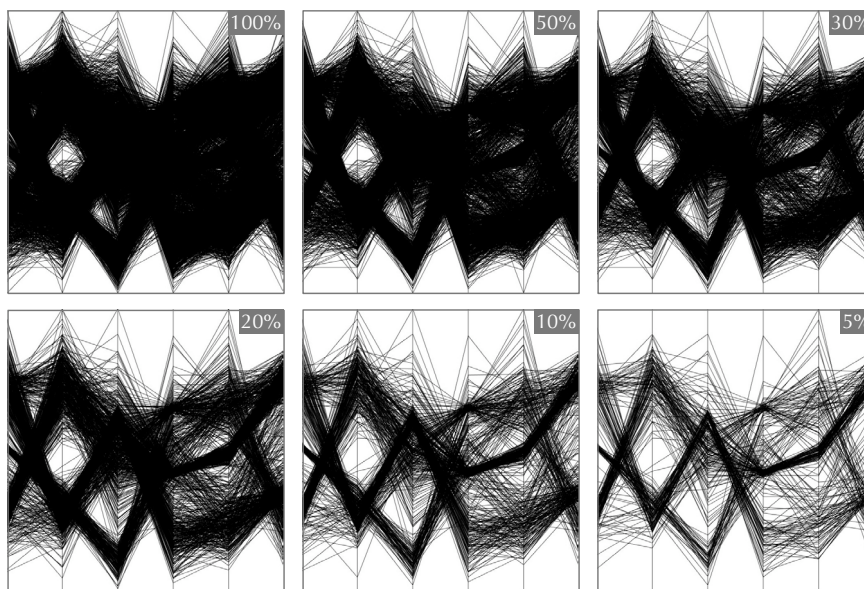
<sup>1</sup> A full description of the data is given in Appendix B.2

Figure 4-3



(a) the full dataset of 1000 randomly sorted records is displayed. (b) sampling rate reduced to 30% by moving the end of the sample window (c) sampling rate increased to 60%, again by moving the end of the sample window, and thus ensuring display continuity

Figure 4-4



Parallel coordinates visualisation at sampling rates from 100% to 5% [5K Synthetic dataset (details in Appendix B.4)]

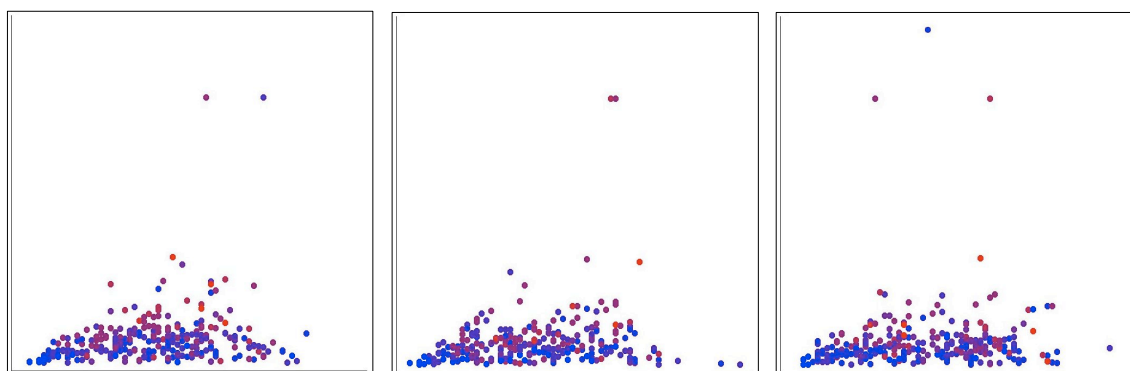


Figure 4-5 Scatterplots following successive Reality Checks showing that despite the 2% sampling rate, the distribution appears visually consistent in the more dense regions of the plot. [SIPP data (9432 records)]



see the distribution of points (although there is still some degree of overlap), we can get a sense of number of points in each educational achievement category.

The technique for choosing the items in the sample is called the z-index method, as described in Section 2.2. After loading the data into the visualisation, a *sampling column* is created and populated with random values using Java `Math.random()`. The data columns are then indexed using this *sampling column*, which essentially orders the data items randomly. The sampling rate control changes the size of a window on the randomised set of data items and hence selects the points to be displayed. As discussed in the previous chapter, to provide display continuity, it is important to redisplay deleted points in the reverse order to which they were removed, thus avoiding users being disoriented with points appearing or disappearing haphazardly. This feature is achieved quite easily by changing the upper limit of the sample window and is illustrated in Figure 4-3, where diagram (a) represents the full randomly ordered 1000 records dataset. The following diagrams (b & c) show how the end of the sample window is moved to reach the desired sample size. Note that as the end of the sample window is moved from 30% to 60%, the previously removed data items reappear in the reverse order, hence ensuring display continuity.

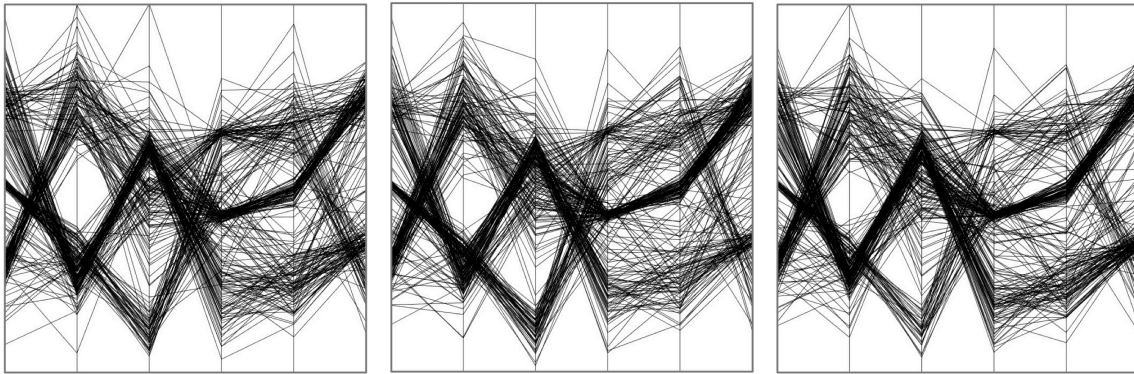
Parallel coordinates<sup>2</sup> is a commonly used technique for visualising multivariate data, however, with even relatively small datasets the display can be cluttered due to the number of crossing lines. A sampling-based parallel coordinate visualisation was developed reusing the same sampling control panel and code as the scatterplot version. Figure 4-4 demonstrates the effectiveness of sampling in making patterns within the data visible to the user. Giving the user the ability to adjust the sampling rate smoothly down to 1% appears to be an important feature as particular artefacts are more evident at different sampling rates. For example in Figure 4-4, patterns start to appear towards the left of the plot at around 30%, whereas in more cluttered central region, a sampling rate of 5% is more appropriate in order to discern trends in the connection of lines between the axes. In addition, the animation effect of lines being added/removed as the sampling rate changes increases the users awareness of structures within the data displayed. Therefore, the display continuity provided by the z-index method is vital.

#### 4.1.2. Reality Check

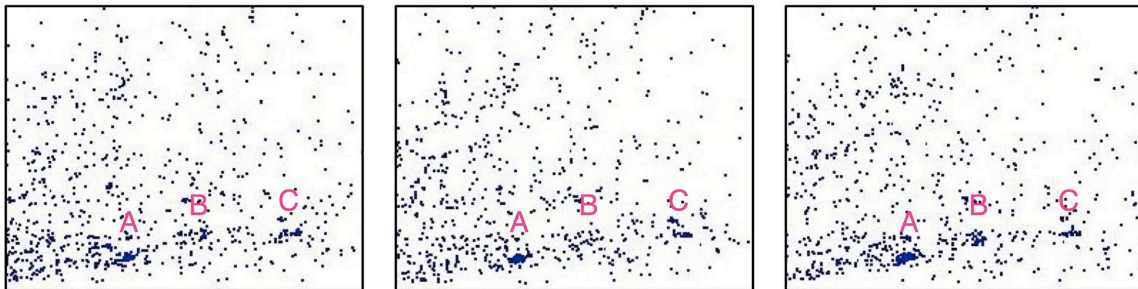
A Reality Check function was added to give the user some confidence in the fact that a particular pattern was real as opposed to an artefact of the sampling. This essentially displays a completely new sample of the data (or as much of a new sample as possible if the sampling rate is greater than 50%).

---

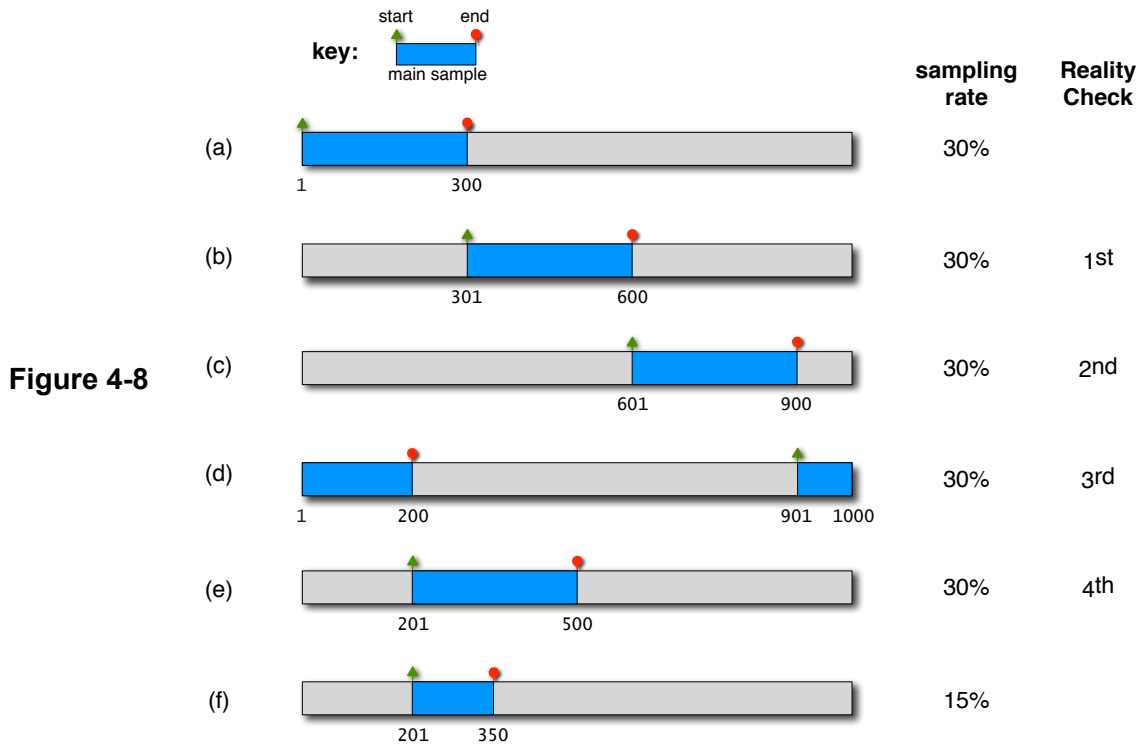
<sup>2</sup> An excellent review of parallel coordinates is given in [Siirtola and Raiha 06]



**Figure 4-6** Parallel coordinate plots following successive Reality Checks. [5% sampling rate - 5K Synthetic dataset]



**Figure 4-7** The same section from three scatterplots following successive Reality Checks. Whereas the small area below A and C appear in all three plots, the artefact below B is not apparent in the middle plot and this suggests that this is an artefact of the sampling rather than the data. [20% sampling rate - Parcels dataset]



**Figure 4-8**

Different states of the sample window after reducing the sampling rate and Reality Checks. (a) initial state with 30% sampling rate and 1000 data items. (b to e) four successive Reality Checks - note that the main sample wraps around after the 3<sup>rd</sup> Reality Check. (f) reducing the sampling rate further to 15%.

To illustrate the Reality Check in action, Figure 4-5 and Figure 4-6 show plots following three successive Reality Checks at low sampling rates. The scatterplots in Figure 4-5 each show a different 2% of the full dataset and the more dense regions near the bottom of each plot show fairly consistent distribution of data points. This demonstrates that even small samples of data provide a representative plot and suggests that the patterns present are artefacts of the data and not artefacts of the sampling. In sparse regions, it is expected that there will be differences between the plots, in this case, not just in the position of points but also in the colour of points in this example as there are 8 categories of educational achievement within the dataset. The successive parallel coordinate plots in Figure 4-6 are very similar and strongly suggests that the clusters are real. This is not wholly surprising as this particular synthetic dataset has been created to test clustering algorithms. However, each plot is a relatively small random sample of the data (5%) and a visual inspection would suggest that the sampling technique gives a representative set of data.

To illustrate a situation where Reality Check identifies sampling artefacts, Figure 4-7 shows the same section from a scatterplot following successive Reality Checks on a 20% sample of data from the Parcels dataset (details in Appendix B.3). This plots package weight against volume for a selection of 7760 parcels delivered by the German postal service. We are interested in the small areas of the plot just below the letters A, B and C. The groups of points at A and C appear to be consistent in all three plots suggesting that they are inherent in the data. However, the group of points at B are not so apparent in the middle plot and hence could well be an artefact of the sampling and should be investigated further.

Now we will look at how the Reality Check is implemented using the z-index method. We saw in Figure 4-3 how the data items are randomly selected for different sample sizes ensuring display continuity. Figure 4-8 follows on to illustrate how Reality Check samples are generated. Diagram (a) shows a 30% sample of a 1000 record dataset. A Reality Check event attempts to produce a sample of new data items and hence moves the start of the new sample to the next item following the end of the previous sample and then calculates the end of the new sample based on the size of the new sample. As mentioned earlier, only if the sample size is less than 50% will this result in a completely new sample; otherwise, some of the previous data items will obviously be included.

Successive Reality Checks are shown in (b) to (e). Note that when the sample window reaches the end of the data set it wraps around and continues from the start of the data set. Given the start and end points of the sample window, the visualisation can very easily calculate if an item should be displayed. Finally, in Figure 4-8(f) the sampling rate has been reduced to 15%, which is easily achieved by moving the sample



end point. To summarise, the sampling rate control moves the end (🔴), whilst the Reality Check moves the start (🟩) to the previous end, a new end point is then calculated based on the sample size.

The z-index method efficiently generates new samples whilst maintaining display continuity or in the case of Reality Check, display discontinuity. However, several issues that arise from its use. If the sampling rate is set to an equal proportion of the full dataset (e.g. 50%, 25%, 20% etc.) successive Reality Checks will eventually wrap around and replicate previous samples. One solution would be to reshuffle the data items at the point when no new samples are possible (e.g. after four Reality Checks at 25% sampling rate). This could be achieved by filling the *sample column* with new random values (as it done when the visualisation initialises) to generate a new index by which the items are ordered. This would avoid the user being presented with reoccurring samples but then raises an issue when the sampling rate does not give an equal proportion of the dataset.

Referring to Figure 4-8(d) as an example; with a sampling rate of 30% and 1000 records, the sample after the third Reality Check includes 100 unseen data items plus 200 from wrapping around to the start. There might well be some features in the display which prompt the user to say that they have seen this sample before and hence a case could be made for reshuffling whenever the sample wraps around but would this be necessary when say the unseen component of the new sample is about half? The processing overhead of producing a new random index needs to taken into account as this is approximately 300msecs for the full cars dataset (almost 6000 records). One solution to the reshuffle or not dilemma would be to provide a reshuffle button so the user could initiate the action, but it was felt that this might be an unnecessary source of confusion and so was not included.

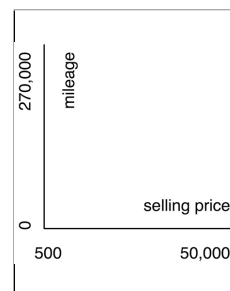
Another issue was whether to provide a back function for the Reality Check. This would enable the user to go back to review the previous state or even states of the sampled display. However, if the user had changed the sampling rate following the Reality Check (or the system had changed the rate as with auto-sampling described in the next chapter) a return to the previous state would not be possible. In addition, the function of the Reality Check is “convince me that the patterns I see are real” and is not a navigation function. The back feature was therefore not implemented.

## 4.2. Comparing clutter reduction techniques

We will now apply change opacity, change point size, filtering (which are built-in functionalities of the InfoVis Toolkit) and sampling to the same dataset to assess their effectiveness at reducing clutter.

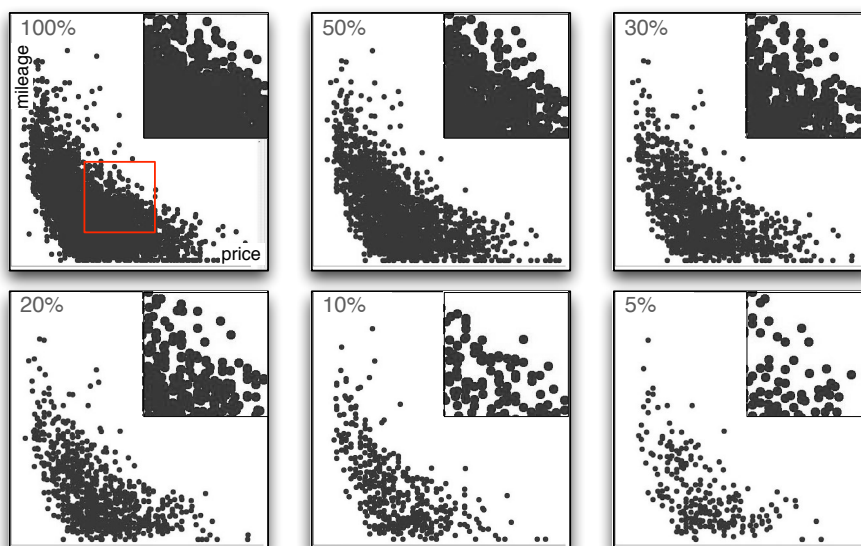
Table 4-1

vehicle type	Sedan	SUV	External Cab
records	2150	1455	510
% of data	53%	35%	12%
point colour	red	purple	blue



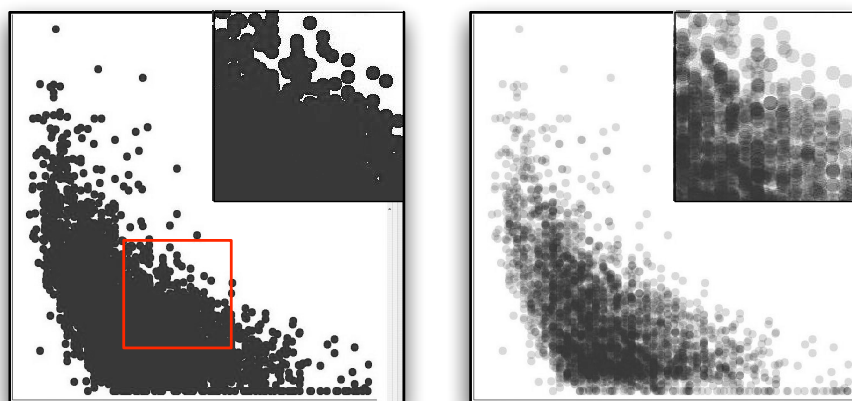
Details of the subset of the Portland cars data used to illustrate basic clutter reduction techniques. A description of the axes for all the plots is given on the right.

Figure 4-9



Reducing the overlap of points using random sampling (sampling rates from 100% down to 5%). The insert shows an enlarged view of the section of the plot outlined in red. [subset of Portland cars dataset (3925 records)]

Figure 4-10



Reducing the opacity of the points gives a useful density map. (20% opacity)

To illustrate the techniques, a 3925 record subset of the Portland new and used cars dataset (details in Appendix B.1) containing the three most popular vehicle types. All screen shots are using the initial sampling application and show mileage (10 to 270000) vs. selling price (\$500 to \$50,000) for the three vehicle types. Details are given in Table 4-1. Note that the point colour only refers to the plots where colour is being used to represent the vehicle type (Figure 4-13 onwards). In the examples given in this section, most of the plots show an insert in the top right corner. This is an enlarged view of the section indicated by a red outlined box on the first plot.

### 4.2.1. Sampling

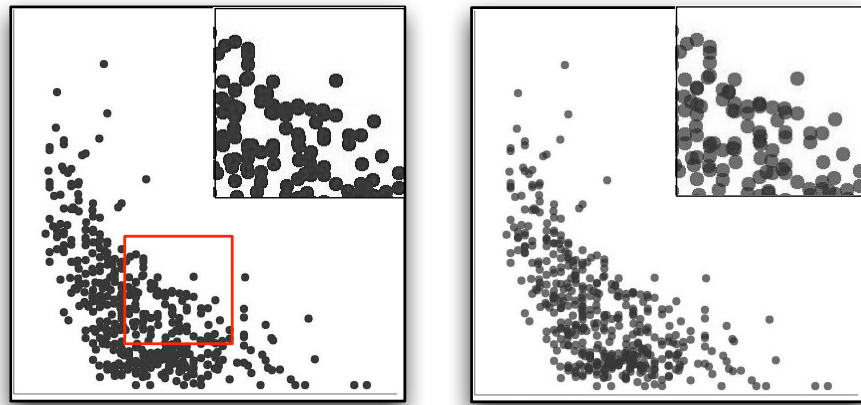
The aim of random sampling is to reduce the number of overlapping data items and this is illustrated in Figure 4-9 which plots mileage against selling price for all the vehicles. As the sampling rate is reduced, the number of discernible points increases as there are less points partly overlapping. We need to say partly, as there may be vehicles at exactly the same price and mileage that would be plotted as coincident points. Without sampling, the only real relationship we can make out is that the sale price of vehicles is approximately inversely proportional to the mileage; however, there is a large spread and we cannot be more precise. There are some obvious outliers, notably vehicles with high mileage yet higher than normal prices. Reducing the sampling rate by way of the slider control gradually removes points at random. The user can finely adjust this control until the plot is sufficiently de-cluttered to enable patterns to be seen. At sampling rates of 10% and below (Figure 4-9), we can see that the majority of the vehicles seem to be low mileage (about 30,000) but there is a wide range of prices. This spread of selling prices is not unexpected considering that we have a large range of manufacturers, models and ages.

### 4.2.2. Opacity

Examples from the literature presented in Section 3.4.1 criteria H, illustrate the use of reduced opacity to give a density map of overlapping lines. An alpha channel slider on the Visual control panel of the InfoVis Toolkit was used to change the opacity of the plotted points. The left-hand plot in Figure 4-10 suggests overplotting, but we cannot see the extent of this. Reducing the opacity of the points to 20% enhances the plot by presenting further information to the viewer as a density plot. In this example, where there is considerable depth of overlapping points, one has to be careful not to reduce the opacity too much otherwise all single points may become very faint and even disappear from view. Of course, this would be helpful if the viewer is only interested in the dense regions.

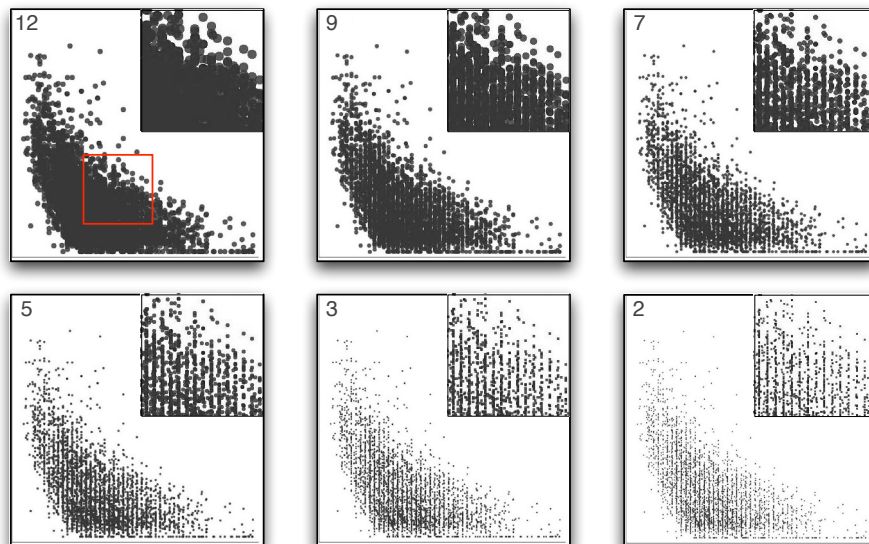
Reducing the opacity of the points slightly when the density has been reduced by sampling clearly has benefits, as illustrated in Figure 4-11. As the depth of overlap is

Figure 4-11



A combination of sampling to reduce overlap and opacity to see the overlap. (10% sampling, 70% opacity)

Figure 4-12



Effect of the size (given in pixels) of the plotted points on the perceived density [100% sampling, 80% opacity]



now only two or three points, the opacity can be set higher and hence avoids the problem of disappearing single points noted above. It is easy to see where overlap occurs, which helps the user to adjust the sampling to an optimum rate. In addition, individual points are better discriminated which enhances the display.

Reducing the opacity of plotted points appears to be advantageous, especially when the amount of overlap is low, however the optimal opacity setting was very dependent on the amount of overlap. Although not included in this investigation, it might be useful for the system to set the opacity automatically dependent on the overlap density.

### **4.2.3. Point size**

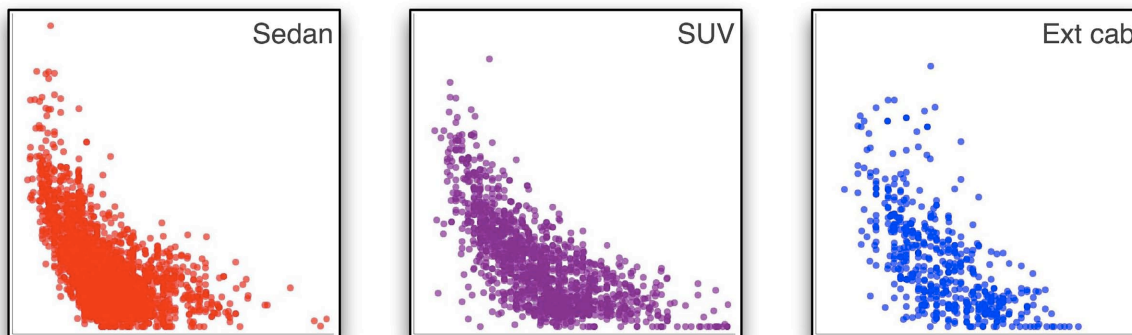
Sampling reduces the overlap of points by reducing the number of plotted points. Reducing the size of the points may also reduce the overlap, if the points are not coincident or less than a point diameter apart. Figure 4-12 illustrates the change in the perceived density of the plot caused by reducing the size of the points. The appearance of vertical lines at point sizes of seven pixels and below shows a noticeable reduction in overlap in the horizontal direction. It turns out that in this dataset, the sale price of a vehicle is very often set to the nearest \$1000 (or more specifically \$5 less, so \$15995 and \$19995 are common). Looking at the enlarged view inserts in Figure 4-12, we can detect white space appearing between some points, in the vertical direction, at point size five and below which indicates reduced overlap. In terms of the whole plot, reducing the point size does seem to show more detail, with a distinctive denser region for low mileage vehicles. However, this may well be due to the sellers setting prices in between the normal \$1000 boundaries and hence could be misleading.

Another consideration is visual acuity. The smaller the point the more difficult it is to perceive the colour [Ware 04]. In Figure 4-12, the points have been plotted with opacity of 80%, however this is hardly noticeable at sizes less than nine. Similar tests were conducted using coloured points (as in Figure 4-14) and dissimilar colours could be discriminated down to a size of four pixels, but below this, the colours tend to merge giving an overall colour pattern to the user.

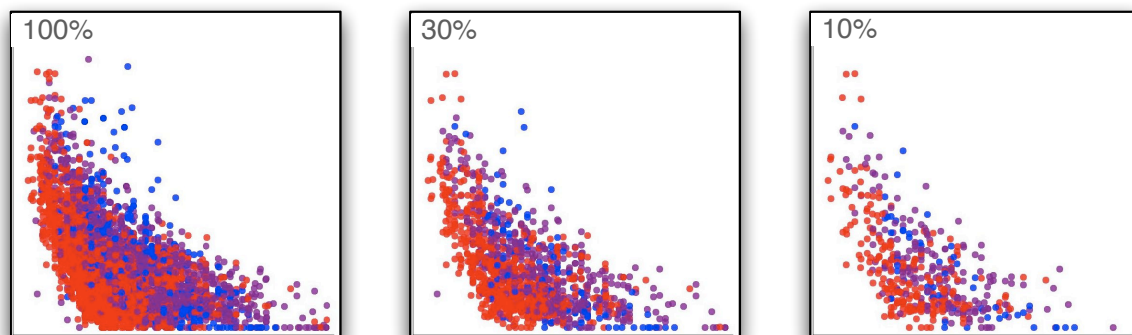
We might conclude that reducing the point size does reduce overlap but it does not help that much with high-density plots and in fact might be misleading. In addition, very small point sizes may not be appropriate if reduced opacity is used or colour is used to represent an additional attribute value.

### **4.2.4. Filtering**

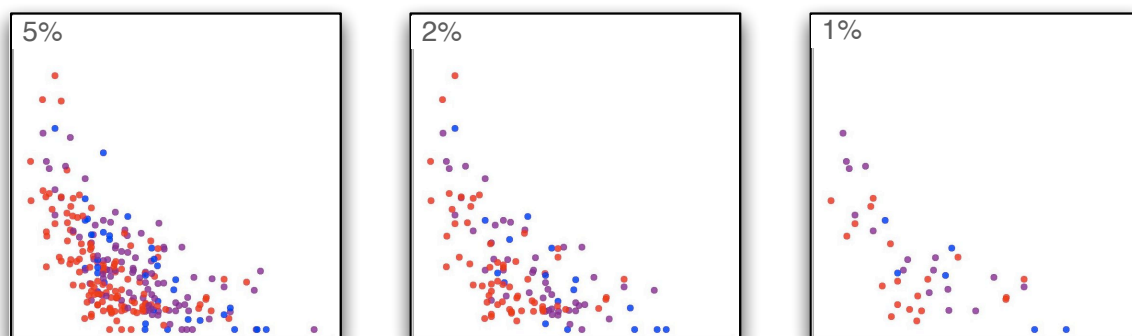
Many visualisations employ dynamic query filters to reduce the number of data items, however, there is a subtle difference between the filtering and sampling techniques.



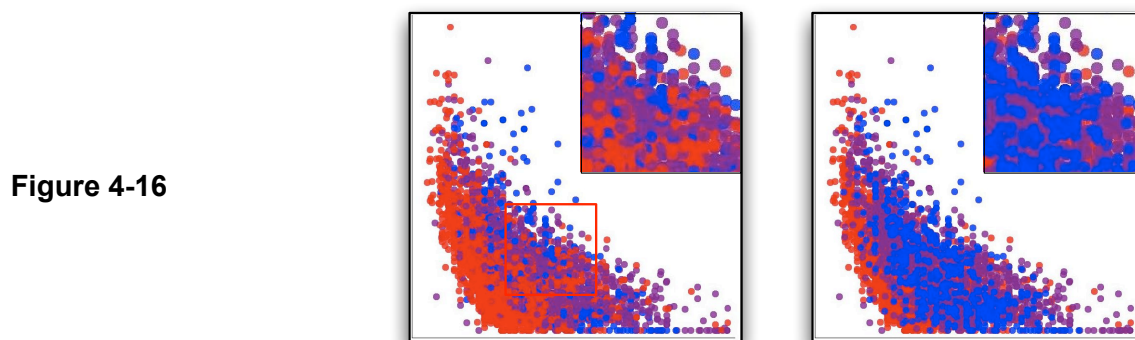
**Figure 4-13** The data filtered on vehicle type. The percentage of the total data for the three plots is (left to right) 53%, 35% and 12%. [80% opacity]



**Figure 4-14** The effect of sampling on showing the distribution of the three vehicle types. (Colours as in Figure 4-13) [80% opacity]



**Figure 4-15** Note that the choice of colours is also a contributing factor in visualising this data. Unfortunately the InfoVis Toolkit only allows the start and end colours to be specified, rather than giving a more balanced palette. Hence the use of purple to represent SUVs, which is less pronounced than the more pre-attentive red and blue colours.



**Figure 4-16**

The effect of inappropriate sorting of the data (right-hand plot) over-emphasises the number of blue points (12% of the dataset).

Sampling is the random selection of a subset of the data whereas filtering is the selection of a subset of data that satisfies a given criteria (e.g.  $\text{year} \geq 2000$  and  $\text{filmtype} = \text{"humour"}$ ). If the user needs to look at a specific set of data or has an idea of what might be interesting then filtering is ideal, otherwise sampling provides an alternative way of exploring the data.

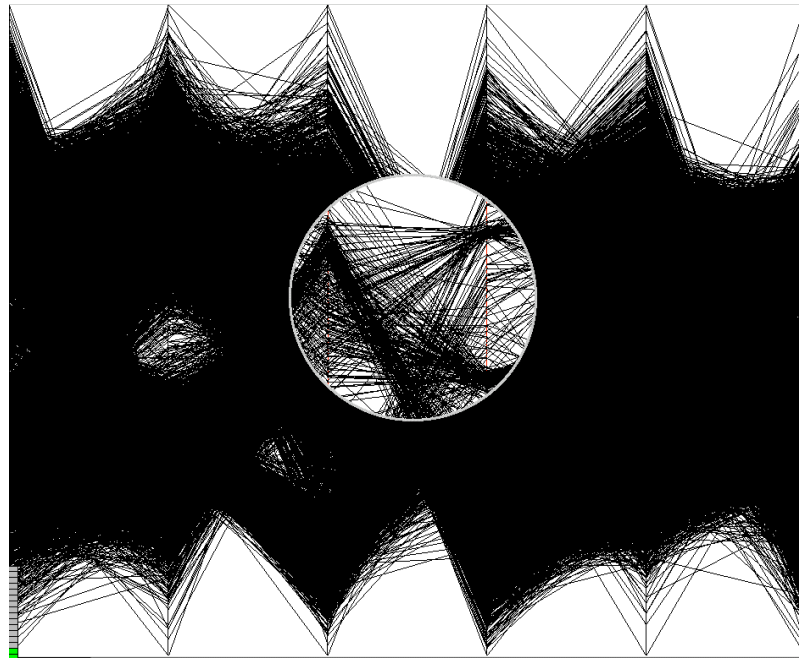
Say we wish to look at the differences in mileage/selling price trends for the three vehicle types in the dataset. Filtering on the vehicle type results in the three plots shown in Figure 4-13. These indicate that selling price for SUVs tends to be higher than for Sedans and External Cab vehicles are even more expensive. Note that the percentage of each vehicle type for each year is fairly constant so this is not a contributory factor to the trend. In addition, there are less External Cabs for sale than the other types. Note that most scatterplot visualisation tools do not allow the user to view plots side by side to make this valuable comparison.

#### 4.2.5. Filtering vs. Sampling

To compare the filtering approach with sampling we can represent the vehicle type by the point colour (same colours as used in Figure 4-13). Figure 4-14 shows the full dataset and plots at sampling rates of 30% and 10%. Although the plots indicate that Sedans tend to be less expensive than the other types, the distribution is not that obvious, even at a sampling rate of 10%. Note that the points are plotted in a random order. The results of reducing the sampling rates further are shown in Figure 4-15. Even though there are fewer plotted points, removing the overlap may well give a clearer view of the proportion of the different vehicle types. In terms of viewing the distribution, this depends on the proportion of each vehicle type. External Cabs only account for 12% of the data and hence, in this example at low sampling rates, there are not many points to adequately describe their distribution. It might be beneficial in this situation to adopt attribute dependent sampling, as described in Section 2.3.2, which in this case would adjust the sampling rate of each individual vehicle type to give each the same number of points.

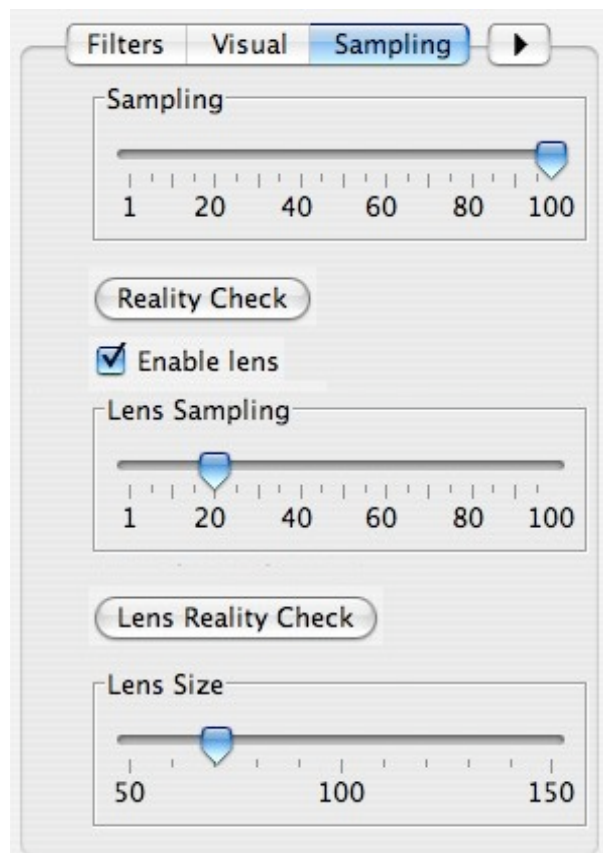
For this particular task and dataset, filtering is more successful in showing the extent of each vehicle type and when all three can be seen next to each other (as in Figure 4-13) a comparison is fairly straightforward. Viewing each filtered set, one after the other may not be so successful though. Returning to Figure 4-13, the plots towards the left are still overplotted, with little detail within the main body of the pattern. It is important to realise that filtering on its own will not necessarily reduce overcrowding on its own and other measures, such as sampling, may be required to reveal detail.

Figure 4-17a



An example of a sampling lens on a parallel coordinate plot [5K Synthetic clustering dataset]

Figure 4-17b



The sampling control panel on an early version of the Sampling Lens

### 4.2.6. Sorting issue

The order in which the data points are plotted is significant where there are overlapping points as only the top items are displayed in full – randomness is therefore used to give a representative view, as also noted by Keim [Keim et al. 04].

To illustrate the importance of the sort order, Figure 4-16 presents a randomly sorted plot (on the right) together with one which has been sorted by vehicle type value, promoting the External Cabs to the top and hence presenting a misleading view of the data.

A summary of the main points revealed by the comparison of clutter reduction techniques is given in Table 4-2 at the end of this chapter.

## 4.3. Sampling Lens

Using the sampling visualisation described in Section 4.1, the user can adjust the sampling level to reduce the data density, consequently revealing features that are otherwise hidden. However, the density across the plot is often non-uniform and as a result, the low sampling rate required to investigate denser regions can make the data in less dense regions disappear. Alternative solutions include non-uniform sampling and localised sampling. Non-uniform sampling means adjusting the sampling rate for different areas of the screen and whilst this can be used with scatterplots [Dix and Ellis 02, Bertini and Santucci 05], it is not an option with parallel coordinates as a polyline (joining points on successive attribute axes) represents the data for a record. Therefore, following the tradition of visualisation lenses that apply transformations or add information to the area under focus, the Sampling Lens was developed.

The moveable region with its own sampling control allows the user to investigate dense regions of a plot by reducing the lens sampling rate to an appropriate level within the lens, with the potential to reveal informative patterns and trends whilst still retaining the context of the lens area within the overall plot. An example of a round sampling lens on a parallel coordinate plot is shown in Figure 4-17a.

This rest of this section describes the features of the Sampling Lens application, and explains how the lens sample and Reality Check is generated. Some implementation details are then given and finally further examples illustrate the use of the lens.

### 4.3.1. Lens features

The features of the Sampling Lens will now be described with reference to the sampling control panel shown in Figure 4-17b.

**main sampling and Reality Check** These controls are associated with density reduction of the main display and as such do not control the lens itself. However, it



should be noted that the sampling rate of the lens is a percentage of the main display. For example, if the main sampling rate is set to 50% and the lens sampling rate is also set to 50%, the data within the lens will be sampled at 25%.

**lens on/off** This toggles the lens on and off. Initially, the lens appears in the centre of the display. Turning off and then on again makes it appear in the same position. In addition, the settings of all controls are to be maintained.

**lens movement** The centre of the lens tracks the mouse position when the mouse button is down (i.e. the lens can be dragged around the screen). The edge of the lens is allowed to go off the screen, and is clipped appropriately.

**lens radius** The radius of the lens can be changed by way of the slider, thus allowing areas of different sizes to be explored. A large lens is useful if possible patterns in a large area of the display need to be investigated, whereas a small lens allows a focussed examination, retaining more of the context of the surrounding area of the display. From a performance point of view, a smaller lens requires fewer points or lines to be drawn and thus movement of the lens is generally smoother. A circular lens was initially chosen, as it seemed more akin to an investigators spyglass, although in later versions square, inter-axis and axis lenses were developed. The last two are solely for use with parallel coordinates and are discussed in Section 4.4.

**lens sampling rate** The sampling rate is the percentage of data items shown within the lens and can range in value from 100 down to 1. The user can adjust the sampling rate by the slider to alter the display density within the lens<sup>3</sup>. As noted, above, the lens sampling is relative to the main sampling rate.

**lens Reality Check** This generates a new sample of the data within the lens and is an important feature that allows the user to gauge the trustworthiness of patterns within the lens. The implementation of this function is described in the next section.

### 4.3.2. Generating the lens sample

The lens data sample is generated in a similar way to the main sample, using the z-index method described earlier, but using its own lens window. Hence, the lens sampling rate control changes the size of the lens window on the randomised set of data items and selects the points to be shown on the screen. Display continuity (redisplay deleted points in the reverse order to which they were removed) is achieved again, quite easily by changing the upper limit of the lens sample window (↑).

---

<sup>3</sup> The prototype Sampling Lens application changed the sampling rate in increments of 1%. There is scope for using log scales or sub-sampling smaller ranges to give a finer resolution, especially at low sampling rates.

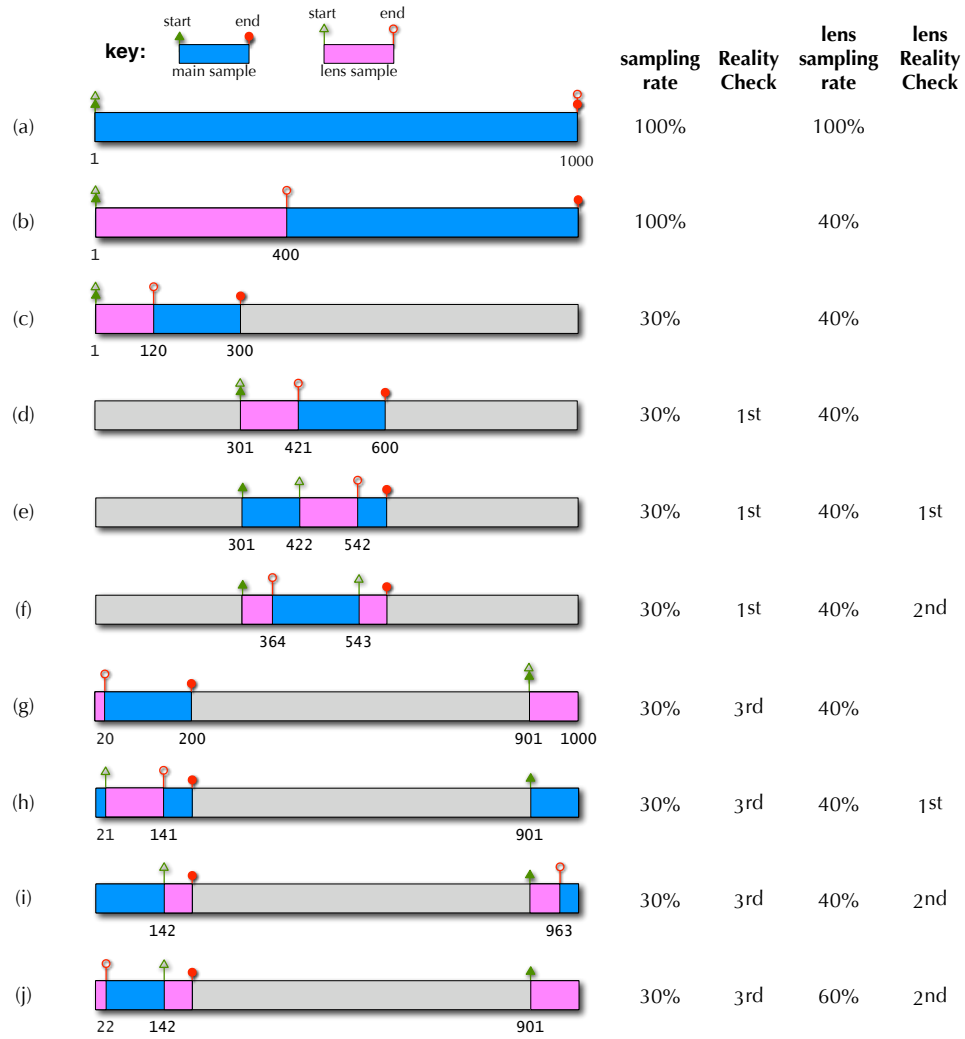


Figure 4-18

Different states of the lens sample window. (a) initial state with 100% sampling and lens sampling rates (1000 data items). (b) lens sampling rate reduced to 40% (c) sampling rate reduced to 30%; the lens sample is 40% of the new value. (d) overall Reality Check. (e) lens Reality Check. (f) further lens Reality Check; note that the lens sample wraps around after the 2nd lens Reality Check. (g) two further overall Reality Checks; the main sample wraps. (h & i) two further lens Reality Checks. (j) increasing the lens sampling rate to 60% causes the lens sample to wrap around again.



Figure 4-18(a) to (d) demonstrate how the lens sample is generated, based on the main sample. Diagram (a) shows the full dataset of 1000 randomly ordered data items with no sampling. (b) is the result of setting the lens sampling rate to 40%. The action of the user sliding the lens control down to 40, is matched by the end value of the lens sample window sliding from 1000 to 400. The next diagram, (c) shows the state of the z-index when the user has finished reducing the main sampling rate to 30%. The end of the lens sample window moves in tandem with the end of the main sample window as the lens sampling rate is relative to the main sampling rate. In (d), the user has completed a Reality Check which shifts the main sample window and the accompanying lens sample window (↑ & ↓).

The rest of the diagrams in Figure 4-18 illustrate six of the eight possibilities that can occur with wrapping of the main and lens sample windows.

Diagrams (e) & (f) are the result of successive lens Reality Checks. Note that the lens sample window wraps in (f). Following two main Reality Checks, the main sample window and the lens sample window are wrapped - see diagram (g). Two lens Reality Checks lead to states (h) and (i) - note that in the latter, the lens sample wraps by *jumping across* to the start of the main sample window. Finally, (j) shows the result of the user increasing the lens sample rate to 60%, which results in the lens sample window wrapping around the start of the dataset again - one of the more complicated states.

The system only records the start and end values of the main and lens windows and if they are wrapped. The algorithm for calculating the lens end value (`setLensSampleEndPosition()` method) was a particularly challenging. Whereas, it is fairly straightforward and efficient to calculate if a particular data item should be plotted in the background or in the lens, which was a major consideration with the design of the sampling function.

### 4.3.3. Implementing the lens

Implementing the lens involved adding a sizeable amount of code at the Java paint level as well as the z-index calculation to determine which points of lines to display. First, the background is drawn and then the circle representing the lens is cleared. The sampled scatterplot is fairly easy to draw as it is only necessary to calculate the points falling within the circle representing the lens. The parallel coordinates version is more problematic as lines have to be clipped to a circle (not a Java API) and as a further complication, more than one vertical attribute axis can cross the lens and hence a clipped line does not necessarily traverse the whole lens (see Figure 4-17a).

Interaction with the lens was poor for a parallel coordinate plot with only 1000 records and four attributes (approximately one second to redraw) but this was improved

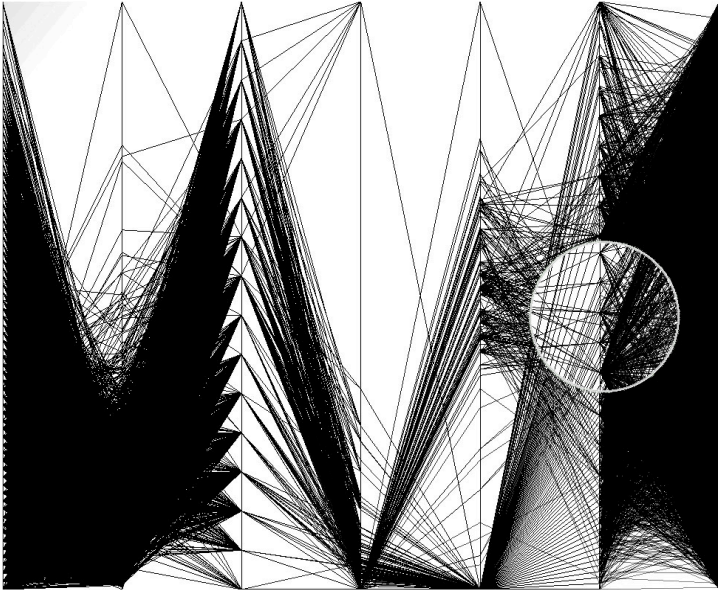
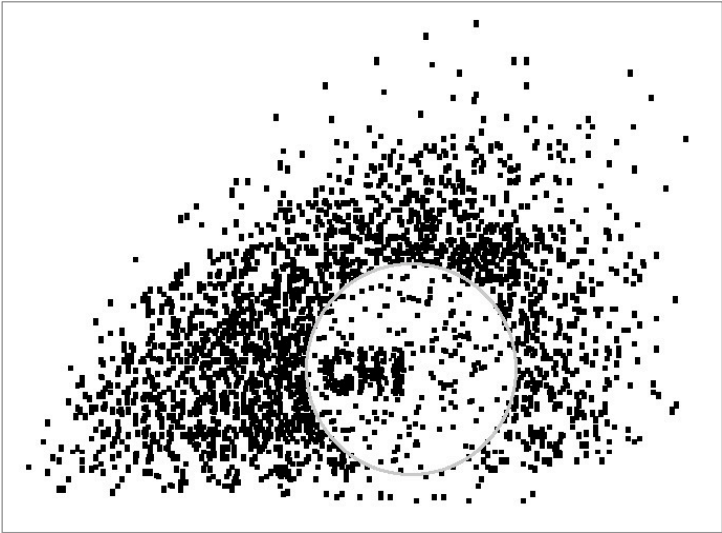


Figure 4-19



Screen shots of the early version of the Sampling Lens application prepared for the CHI'05 paper. The scatterplot used synthetic data, designed with a high density "CHI" pattern that would be exposed by the lens. [Ellis et al. 05]

considerably by utilising an off screen buffer holding a copy of the background lines. Hence the background was only redrawn when the overall sampling rate was changed. Acceptable performance could then be obtained as long as the number of lines within the lens was fairly small (less than 50) and the lens was small (50 pixels radius).

Further implementation issues are discussed in Appendices D.2.2 and D.3.2, including the development of a version of the Sampling Lens with much improved interactive performance.

#### **4.3.4. Examples of the lens on scatterplots and parallel coordinates**

Figure 4-19 presents some screen shots of the early version of the Sampling Lens application prepared for the CHI'05 paper [Ellis et al. 05]. The parallel coordinate plot used the 10K People dataset [Appendix B.5] that results in several high density regions, in which the lens proves effective in revealing patterns. The scatterplot example uses a synthetic dataset that has particularly dense areas in the form of the letters "CHI", which are revealed as the lens passes over them. An effective demonstration and a crowd pleaser.

The Sampling Lens application was shown to people in the lab and at the CHI conference, and the majority understood the principle behind clutter reduction by random sampling and the purpose of the lens. Some even commented that it was analogous to an x-ray of a person, to reveal details of the skeleton inside. As predicted, display continuity proved an important aspect of the design when moving the lens around the display and adjusting the sampling rate. Users welcomed the function of the Reality Check, although a few did not appreciate the name.

### **4.4. Other lenses and techniques for parallel coordinates**

The lens used in the early versions of the Sampling Lens was circular, principally because it looked and felt like an investigative spyglass, in the tradition of Sherlock Holmes. This section describes two special purpose lenses, the inter-axes and axis lenses, which have been implemented for parallel coordinate plots. Closely related to the axis lenses is a novel technique called RaDar (**Related Data visualiser**) that provides the user with related information. Again, examples of its use are given below. Lastly, two techniques for animating the transitions during a lens Reality Check (described earlier in Section 4.3) are illustrated and discussed.

#### **4.4.1. Inter-axis**

Useful feedback following the InfoVis'06 presentation included the possibility for looking at the lines connecting any two attribute axes on a parallel coordinate plot. This makes sense as the pattern of lines connecting two axes indicates the type of

Figure 4-20a

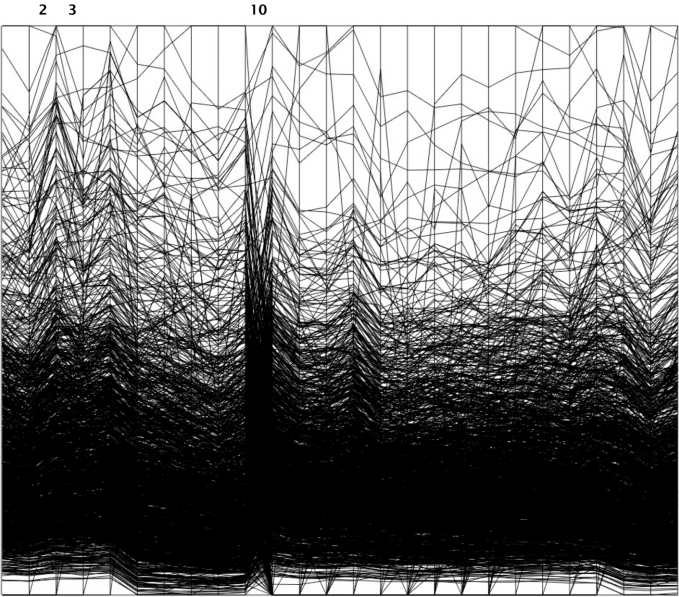


Figure 4-20b

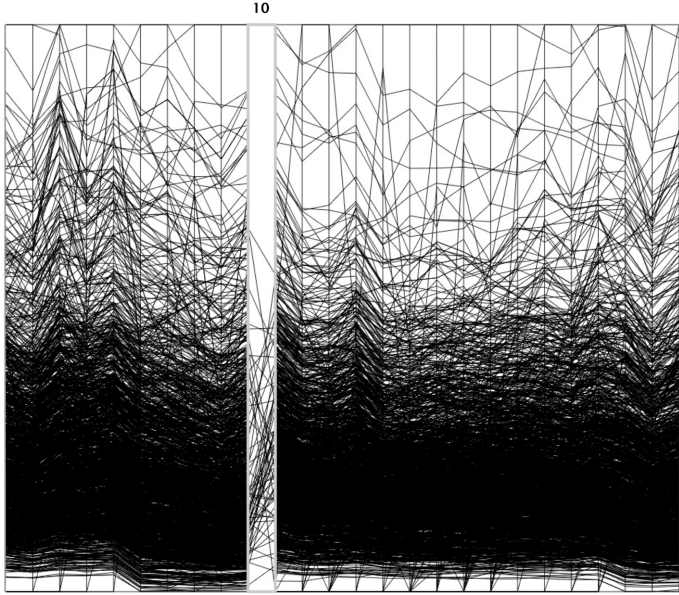
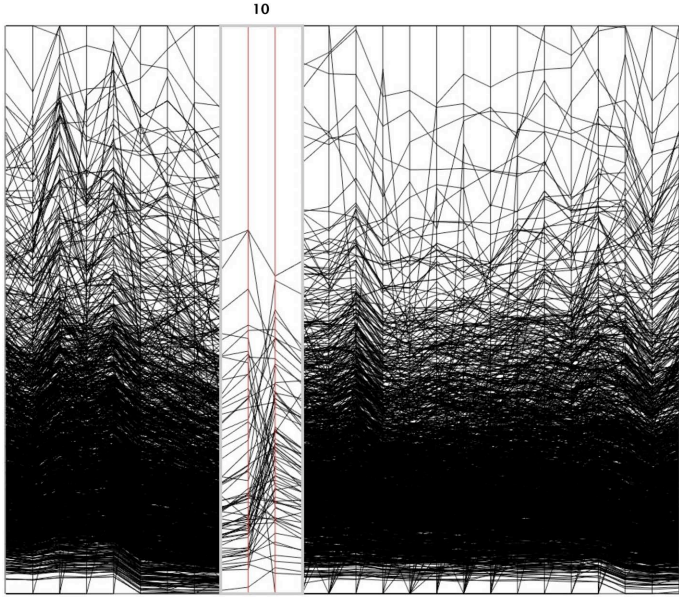


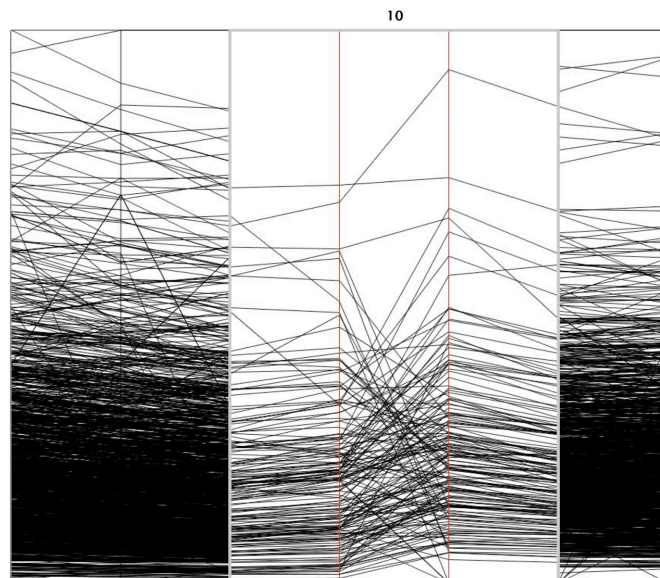
Figure 4-20c



relationship between these attribute values.

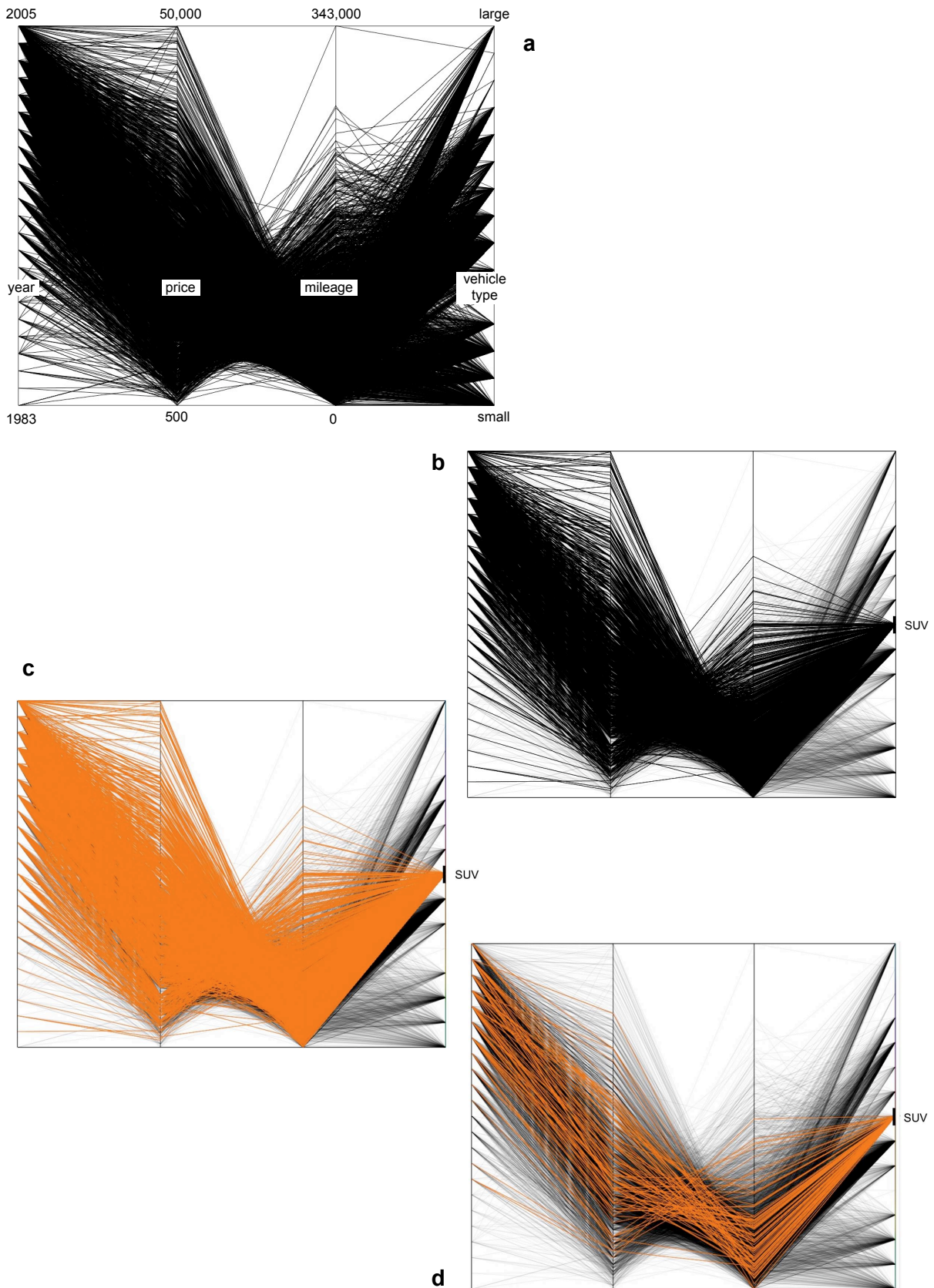
The first step was to provide a square lens that could be resized to include the required region of the plot. However, resizing and position the lens was not that straightforward. Therefore an *inter-axes* lens was created, which occupies the whole region between the axes. Dragging the mouse to an adjacent region or clicking on another part of the plot moves the lens.

The screenshots in Figure 4-20 illustrates the usefulness of the inter-axis lens, in this case for time-series data. Figure 4-20a is a parallel coordinate plot of weekly closing prices for Dow Jones stocks over a 26 week period in the second half of 2001. The overall plot shows that the higher priced stocks fluctuated from week to week (the display is too cluttered to see what happened to the lower priced stocks) with a noticeable general rise in week 2, followed by a fall in week 3, repeated the next two weeks. However, week 10 stands out as being unusually cluttered over most of its height. Dragging an inter-axis lens (with auto-sampling) over this particular week clearly reveals the reason for the clutter – a high proportion of share prices have either risen sharply or fallen sharply (see Figure 4-20b). An available option is to shift-click on any other region of the plot to widen the lens. For example, in Figure 4-20c the lens has been widened to show the previous and following weeks that emphasises the dramatic changes in week 10. A zoomed in version of a wide lens centred again on week 10 is shown in Figure 4-20d. We will return to this dataset later, to see how an axis lens and RaDar can assist the interpretation of the data.



**Figure 4-20d**

Parallel coordinate plot of weekly closing prices for Dow Jones stocks over a 26 week period in 2001. (a) note the abnormal activity during week 10 giving rise to a particularly cluttered region, (b) inter-axis lens over week 10 that shows that a many share prices have either risen sharply or fallen sharply, (c) widening the lens to include the preceding and following weeks, (d) a zoomed in view of a wide lens centred about week 10 [Stockmarket data – see Appendix B.6]



**Figure 4-21** Parallel coordinate plot of the Portland cars dataset (5850 records) showing the advantage of axis lens in reducing display clutter. (a) full dataset with no lens, (b) axis lens, with no sampling, selecting the SUV vehicle type, note that the data not selected is shown in light grey, (c) colouring the selected data to make it stand out against the background, (d) axis lens in auto-sampling mode reduces the clutter and enables some structure to be seen in the SUV data.

#### 4.4.2. Axis (filter)

The axis lens is similar to a range filter (sometimes referred to as brushing) found on most parallel coordinate applications in that the user can restrict the data to only those records that have an attribute value within the set minimum and maximum<sup>4</sup>. However, this lens acts as a regular sampling lens with a sampling rate control and auto-sampling mode, but sticks to the nearest axis when being dragged around the display with the mouse. Instead of clipping the lines to the lens interior, the full polylines of the lines passing through the lens are displayed. Hence the axis lens can be employed to select a particular set of data and sample this selection to reduce clutter. An example of its use is given in Figure 4-21, which shows parallel coordinate plots of the 5850 record Portland cars dataset. Figure 4-21a is an annotated plot of the full dataset. Figure 4-21b shows an axis lens selecting all the SUV vehicles (no sampling), whilst data not selected is shown in light grey. The next screenshot, Figure 4-21c uses colouring to make the selected data stand out against the background, and clearly displaying almost 1500 records leads to a cluttered display. However, putting the axis lens into auto-sampling mode (Figure 4-21d) reduces the clutter and enables some structure to be seen in the SUV data.

The axis lens has proved successful in other parallel coordinate plots for reducing clutter and auto-sampling generally works very well. This is somewhat surprising as estimating the occlusion is achieved within a very thin, 6 pixel lens, using the same algorithm as the other lens shapes and takes no account of the density of the lines displayed, which are outside the lens. Certainly, a more accurate estimate of occlusion would result from considering the lens as covering the whole display area, especially with the use of binning (Section 5.6.2), but this would result in an efficiency trade-off.

Implementing multiple axis lenses would be possible as the lines within each lens could be easily determined even when sampling, but providing sampling controls (e.g. auto on/off, desired occlusion) for more than one lens would require careful consideration.

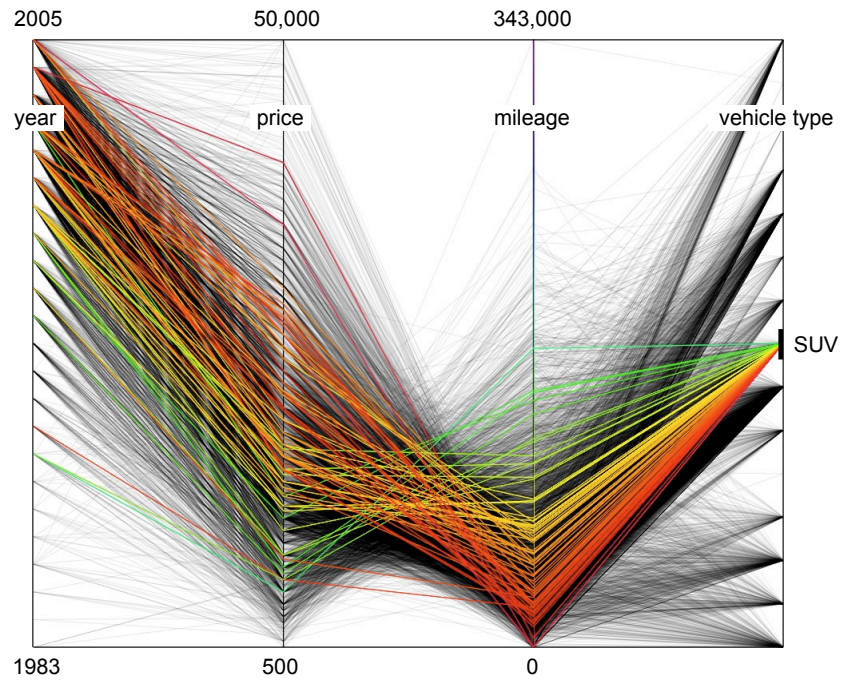
#### 4.4.3. RaDar

With parallel coordinate plots, it is often useful when browsing to select a subset of the data using one or more axis filters. As we have seen in the last section, an axis lens can reduce a cluttered selection to expose underlying patterns. It is usual to highlight the lines selected by a range filter in a different colour, but additional information can be provided to the user if the colour of the lines represented the value of one of the attributes. Hence, RaDar was implemented which colours the particular axis selected

---

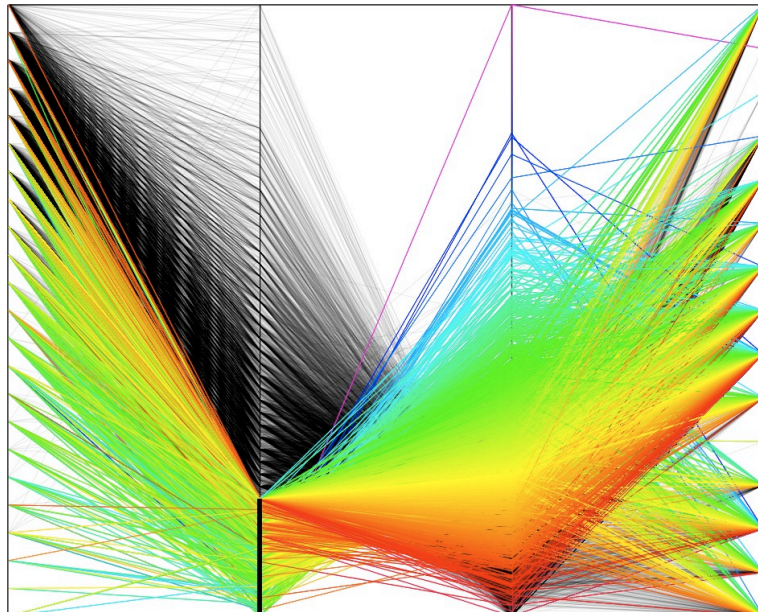
<sup>4</sup> Most parallel coordinate applications allow the user to set multiple range filters on the same attribute axis or over different axes.

Figure 4-22



Same SUV data as in Figure 4-21d, selected with the axis lens, but now each line is coloured using a rainbow scale, according to the mileage of the vehicle.

Figure 4-23a



Further example of the use of an axis lens in conjunction with the rainbow colouring of RaDar. The axis lens (without sampling) highlights vehicles for sale at less than \$10000 and the RaDar axis is the mileage. [Portland cars]

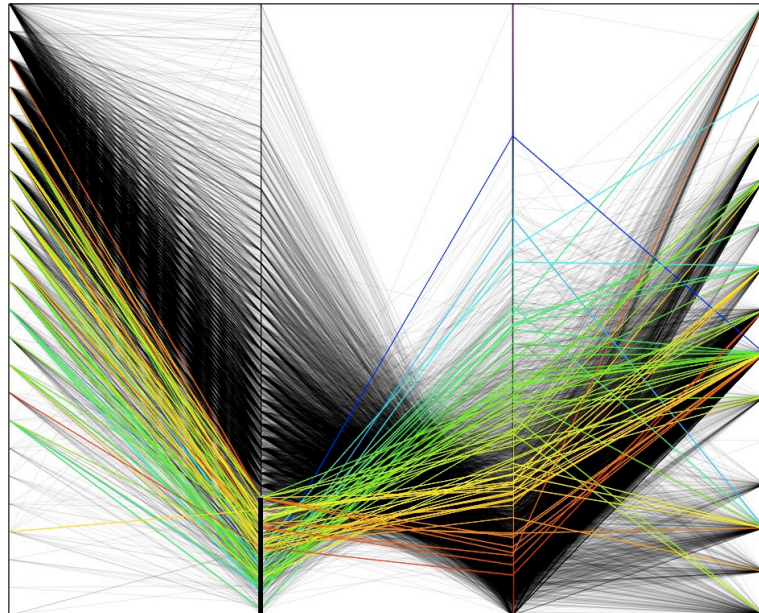


by the user (ctrl-click on axis) in rainbow colours - from red through yellow, green, blue to purple and hues in between<sup>5</sup>. A line then takes on the colour of the point where it crosses the current RaDar axis. The colour represents the value of that particular attribute and can thus be compared to any other attribute value. Note that many parallel coordinate applications do permit the user to change the axis order, thus allowing direct comparison of attributes on adjacent axes but they do not provide the added functionality of RaDar - related data at a distance.

The example given in Figure 4-22 demonstrates this technique in action using the same data presented in Figure 4-21d. The user has selected the mileage of the vehicle as the RaDar axis (second from the right), and we can immediately see that the majority of SUVs are low mileage. Looking at the year and price axes, this generally means younger vehicles at moderate to high prices. However, there are some green lines signifying high mileage amongst the younger cars that breaks this trend, although they seem to follow the noticeable inverse relationship between mileage and price. Again there are some exceptions, such as low mileage, low price SUVs that may still be worth pursuing even though they are older vehicles.

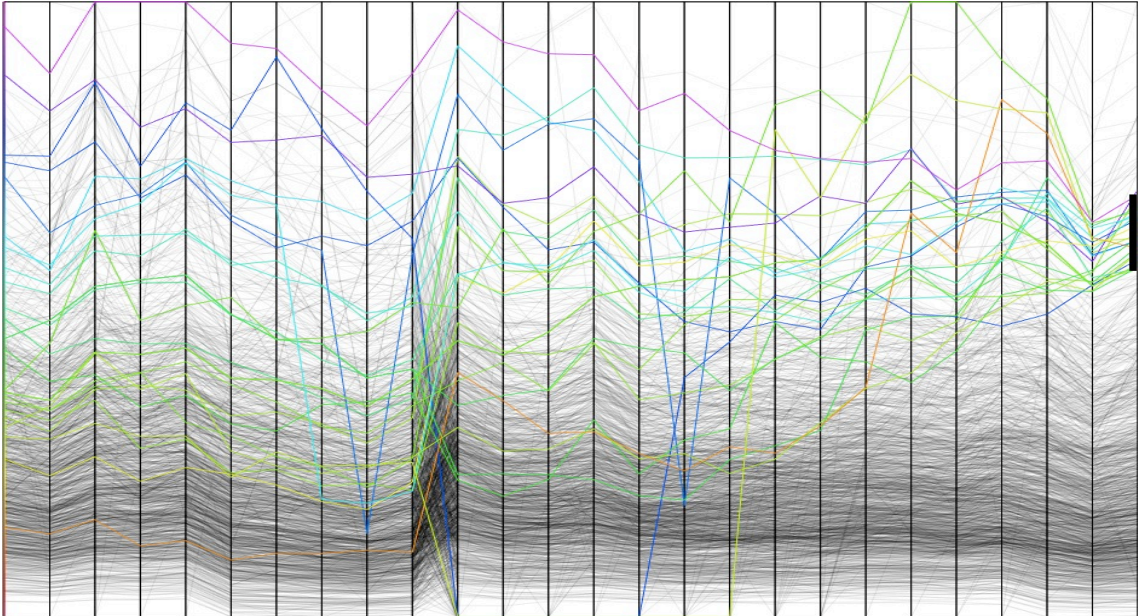
Figure 4-23 gives a further example, which illustrates the strengths of an axis lens in reducing clutter and RaDar emphasises relationships between attributes, including non-adjacent attributes. Furthermore, the use of colour enables lines to be discriminated as discussed in Section 3.4.1, criteria G.

**Figure 4-23b**

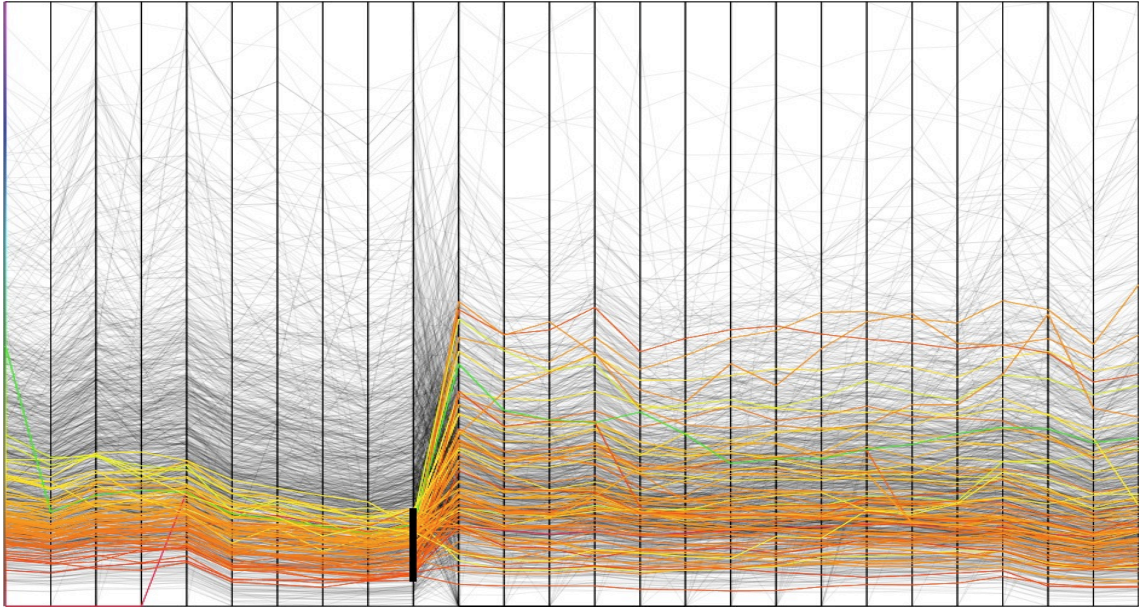


The axis lens in auto-sampling mode, reduces the clutter. (See Figure 4-23a)

<sup>5</sup> According to Ware [Ware 08] a rainbow sequence is often erroneously presumed to be a perceptual sequence. The author suggests that a sequence of colours with increasing luminance would be a better solution.



**Figure 4-24a** Parallel coordinate plots of weekly closing prices for Dow Jones stocks over a 26 week period in 2001 that illustrate the use of an axis lens and RaDar colouring in revealing patterns in otherwise very overcrowded plots.



**Figure 4-24b** As Figure 4-24 but with the axis lens highlighting the lower priced shares at the start of week 10.

Returning to the stock market data examples first seen in Figure 4-20, we can apply RaDar colouring and an axis lens to the plot. Figure 4-24a shows a sample of medium to high stocks fifteen weeks after the dramatic changes in share prices in week 10. As the line colours are based on the stock prices at the start of the half year period, clearly many of the stocks have not changed a great deal on year end (i.e. green-blue colour), despite wild fluctuations. There are some green, yellow and red lines in the sample that indicate a price rise (the red being a substantial rise) and some in the blue, purple part of the rainbow spectrum indicating a drop over this period. Figure 4-24b places the axis lens at the start of week 10 and highlights the progress of these lower priced shares over the following fifteen weeks. This is a good example of the colouring scheme discriminating between lines, but does of course rely on the lens reducing the display clutter first.

As the visualisation is interactive, sliding the lens up and down the axis and to other axes reveals a wealth of information that is not possible to show in these static illustrations. However, the usefulness of sampling, axis filtering and RaDar are hopefully demonstrated through the examples presented here.

#### 4.4.4. Fade and Twinkle – Reality Check transitions

Reality Check is a feature of the Sampling Lens that allows the user to view a new sample of data items within the lens and hence gain more confidence that observed patterns are real or an artefact of the sampling (full description in Section 4.1.3). In the earlier versions of the Sampling Lens, one sample was replaced by the new sample and it was felt that a more gradual transition may be more appropriate as reported earlier

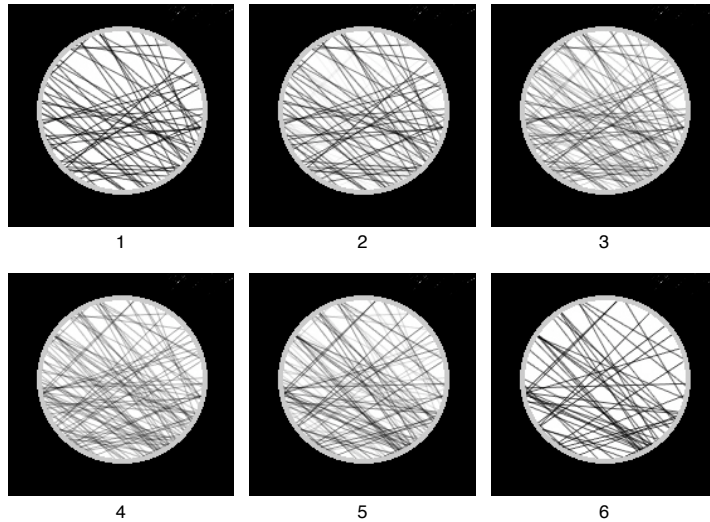


Figure 4-25a

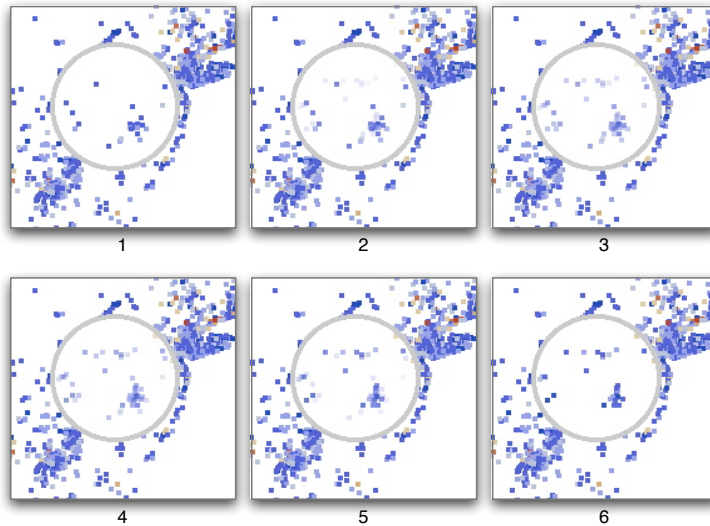


Figure 4-25b

Fade transitions for (a) parallel coordinate plot and (b) scatterplot.

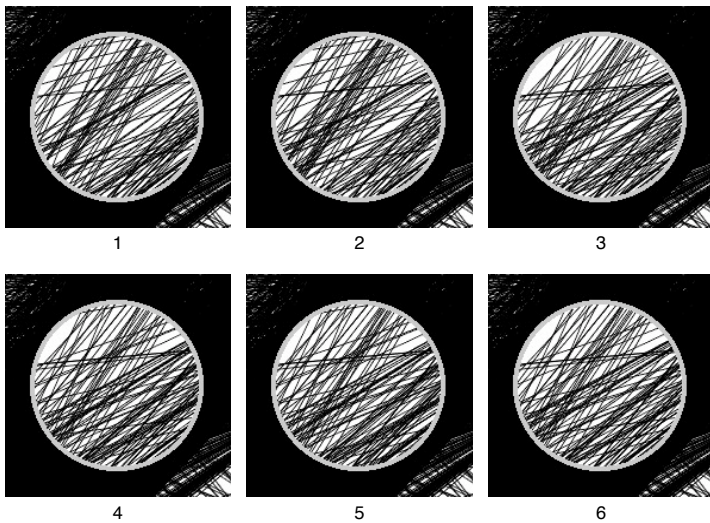


Figure 4-26

Twinkle transition on a parallel coordinate plot. The frames are taken at 2 second intervals with 1 or 2 lines added/removed per second.

in Chapter 3 (also in Appendix A.1.10). Two different transitions were implemented, a traditional fade and a novel technique called twinkle. Both techniques were only possible in the OpenGL version due to the increased display speed over the original Java2D versions (see Section 6.4.1 for details).

Animation is understandably difficult to illustrate with static pictures, however, Figure 4-25 shows six frames from the full 20 frame fade transition for a parallel coordinate plot and a scatterplot. Over a two second period, the original data items fade out, whilst the data items for the new sample fade in. Although no user testing has been carried out to assess the benefits of this, informal studies suggest that users prefer smooth transitions as they can more easily see persistent patterns. However, an abrupt change (no smooth transition) does reinforce the notion of a new sample. Further work is required to assess the effectiveness of such transitions.

The other Reality Check transition, is called twinkle after the Astral Visualiser example (described in Chapter 2), which is based on the notion of star gazing. The idea is to successively add a new randomly selected data item whilst removing another randomly selected item, so the lens sample slowly changes. If there are real patterns present within the lens, then these will tend to persist within a background of noise, similar to twinkling stars. In practice, this is achieved by shifting the lens sample window one point at a time to the right at a given rate (Figure 4-18 in Section 4.3.2 showed the sample windows used with the z-index method). In the prototype, the rate can be varied between 1 and a maximum of about 15 points per second (dependent on the dataset size and lens sample). Remember that only the points in the z-index lens sample that appear in or pass through the lens are actually displayed, so the observed twinkling rate is proportional to the size of the lens and inherits some temporal randomness. Again, a static example, such as Figure 4-26, does not illustrate the effect very well.

The benefits of the twinkling transition are questionable. It is entertaining to watch but somewhat distracting as a single new point or line is often easily detected by our visual system. This could be improved by fading in each new point so changes are less distracting and a more gradual change is perceived. However, this is difficult to implement unless the graphics card supports this feature.

## 4.5. Summary and reflection

Examples of sampling-based scatterplot and parallel coordinate visualisations demonstrate the effectiveness of sampling in clutter reduction. We saw how the implementation of the z-index method, proposed in the previous chapter, successfully achieves interactive sampling rate adjustment and maintains display continuity. The ability to visually explore with the sampling rate appears to help the user identify

	strength	weakness
opacity	<ul style="list-style-type: none"> <li>▶ can show overlap density</li> <li>▶ can discriminate individual points</li> </ul>	<ul style="list-style-type: none"> <li>▶ can lose colour information if low opacity and/or high overlap</li> </ul>
point size	<ul style="list-style-type: none"> <li>▶ can reduce overlap</li> </ul>	<ul style="list-style-type: none"> <li>▶ not effective if not too overcrowded</li> <li>▶ problem with discriminating small points if reduced opacity and/or coloured points</li> </ul>
filtering	<ul style="list-style-type: none"> <li>▶ good at isolating a subset of data</li> </ul>	<ul style="list-style-type: none"> <li>▶ cannot always be adjusted to reduce overplotting adequately</li> </ul>
sampling	<ul style="list-style-type: none"> <li>▶ good at reducing overplotting</li> </ul>	<ul style="list-style-type: none"> <li>▶ colour scheme is an important consideration if colours represent an attribute value</li> <li>▶ an awareness of the distribution of point attribute values may be required</li> </ul>

**Table 4-2** Strengths and weaknesses of sampling and some other clutter reduction techniques built-in to the InfoVis Toolkit.

structures within the data. We also saw how the z-index provides the Reality Check function, which allows the user to test if perceived patterns are representative of the dataset as a whole. Reality Check even seems to work with relatively small data samples as shown by some examples.

A simple experiment, as described in Section 4.2, was carried out to compare opacity, point size, filtering and sampling in reducing clutter using the same dataset and visualisation type. Table 4-2 highlights their main strengths and weaknesses.

The importance of randomising the sorting (plotting) order of the data was also demonstrated and we saw that combining reduced opacity with sampling provides the user with feedback of the overlap density when adjusting the sampling rate. However, achieving the optimal opacity is very dependent on the density – setting the opacity automatically, based on the density, would be valuable. In addition, the ability to compare plots side by side would be a useful addition to visualisations.

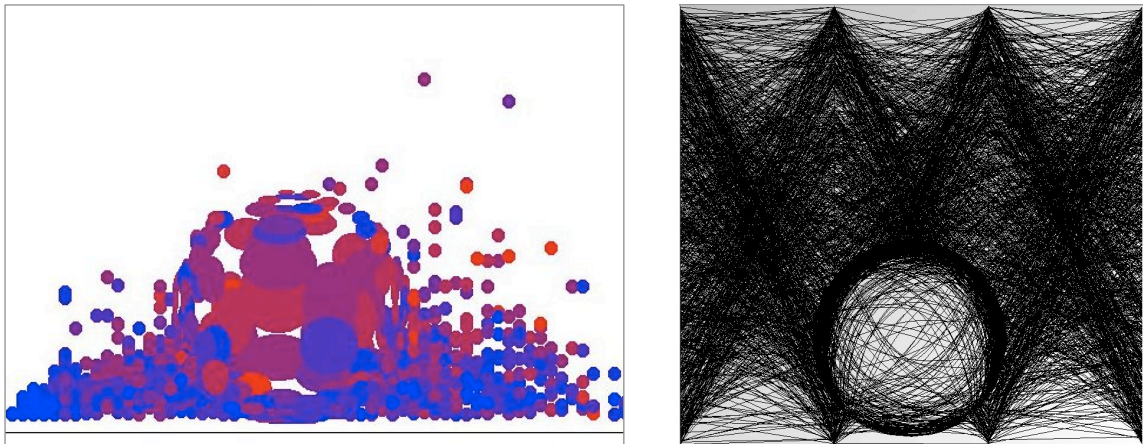
The Sampling Lens visualisation prototype was successful in showing patterns in regions of a plot that were otherwise too overplotted to reveal anything. However, it was noted that the interactive performance on a parallel coordinate plot was only acceptable with relatively small lens and a small number of lines. An extended version of the z-index method was shown to cope very well with generating the lens sample.

When the Sampling Lens application was shown to some users, they quickly understood the principle of random sampling and appreciated how useful it was in reducing display clutter. The utility of the lens was readily identifiable and the function of the Reality Check, in confirming that an observed patterns was real, was welcomed by the users.

The version of the Sampling Lens visualisation described in this chapter, required the user to adjust the sampling rate within the lens using a manual slider control. This proved to be awkward and would be unnecessary if the system could adjust the sampling rate automatically depending on the density of the points or lines in the lens. Auto-sampling provides this functionality and is the subject of the next chapter.

The final section illustrated a range of novel lenses and other features added to the Sampling Lens. A parallel coordinate plot of New York stock market data showed the effectiveness of the inter-axis lens in revealing significant trends in a particularly overcrowded section of the time-series graph. Similarly, a sampling version of a range filter found on most parallel coordinate applications, namely the axis lens, was demonstrated to be equally effective in showing structure in a subset of the data.

A novel enhancement of the axis lens, RaDar, colours the lines passing through the lens, based on the value of a selected attribute and hence provides related information. Two Reality Check transitions were described. A simple fade provides a smooth



**Figure 4-27** A fisheye lens provided by the InfoVis Toolkit applied to a scatterplot and parallel coordinate plot.



transition and may give a better sense of persistent patterns, however this is untested. The twinkle transition is entertaining but could distract the user from spotting real patterns, which is the function of the Reality Check. However, this technique may prove to be useful in visualising uncertainty, as discussed in Section 7.4.2.

An important finding demonstrated by the last section is that sampling can be used with good effect in conjunction with other techniques such as a parallel coordinate range filter and RaDar, the related data visualiser.

The InfoVis Toolkit provides a fisheye lens function and Figure 4-27 demonstrates its use on a scatterplot and parallel coordinate plot. With reference to the Clutter-reduction Taxonomy in Chapter 3, the topological distortion accompanied by non-uniform increase in the point sizes (together with a distortion of the shape) seen in the scatterplot does not add to a greater understanding of the region within the fisheye. In fact, the extra display space given to the central points means that the points towards the edge of the lens are given less space and hence appear more cluttered. In the parallel coordinate example, we can clearly see that the fisheye lens curves the lines connecting attributes and relationships are no longer apparent.



## Chapter 5

# The provision of auto-sampling

Chapter 4 described the development of the first sampling-based visualisations. We saw how the Sampling Lens application with a moveable lens that has its own sampling control allows the user to explore dense regions of a scatterplot or parallel coordinate plot, whilst retaining the context of the whole visualisation. However, it became apparent that adjusting the lens sampling rate as the lens was moved between regions of high and low density was an unnecessary burden on the user.

In this chapter, auto-sampling is proposed whereby the system adjusts the sampling rate to give a reasonable uncluttered view within the lens. However, unlike the constant-density zoom of the Astral Visualiser, described in Chapter 2, where the sampling rate is calculated based on the magnification factor, with auto-sampling the sampling rate for the lens is dependent on a measurement of the clutter or occlusion within the lens.

There is little guidance in the literature that helps with such a measurement, especially for the crossing lines found in parallel coordinates, hence the need for defining an occlusion measure. From a set of proposed occlusion measures based on quite different raw data values, the most accurate measure is determined through an empirical study. However, by constructing a theoretical model based on the notion of scattering the plotted points<sup>1</sup> randomly across the lens area, we discover that all the set of proposed occlusion measures are in fact related. We formulate three quite different methods, based on direct measurements and binomial distributions for calculating the selected occlusion measure. We test the accuracy of these algorithms under a wide range of conditions and whilst there is very good agreement in most situations, two extreme cases are identified and solutions are proposed.

The method for calculating occlusion, which coincidentally is based on randomness, is both accurate and extremely efficient for implementing auto-sampling within a lens on parallel coordinates.

Section 5.1 first gives an overview of the literature for clutter metrics and defines an occlusion measure, in terms of the number of plotted points on each screen pixel, which could be used with both scatterplot and parallel coordinate visualisations.

---

<sup>1</sup> The lines in the parallel coordinate plot are divided up into pixel points.



Section 5.2 describes an initial implementation of auto-sampling based on the above occlusion measure and discusses some problems as highlighted during experiments with parallel coordinates. Two further occlusion measures are proposed, one based on plotted points and the other on plotted points and available pixels.

Section 5.3 describes an empirical study to determine which of the three proposed occlusion measures is the most accurate for lens-based sampling on parallel coordinates. A theoretical model, based on randomly distributing the points over the available pixels, is devised that demonstrates a relationship between the methods. One of the occlusion measures is subsequently chosen.

Section 5.4 considers three very different ways of calculating the selected occlusion measure, derived from plotting points on a raster grid – one is a direct calculation, one is based on a purely random distribution and the other is partly probabilistic/partly a direct measure.

Section 5.5 discusses the results of a series of experiments that compare the accuracy of the three methods of estimating the occlusion of crossing lines within a lens on parallel coordinate plots. Despite good agreement between the methods, problem cases are found. The efficiency of the methods are demonstrated, based on empirical data, and we consider ways in which the calculation of each method can be made faster.

Section 5.6 details experiments to determine if non-uniform density across the lens is responsible for inaccuracies in the occlusion measurements. A solution is proposed, evaluated and is deemed successful.

Finally, in Section 5.7, we summarise the main issues raised in this chapter.

## **5.1. Defining a clutter measure**

We will first review the literature to uncover measures of density or occlusion that could be used for overlapping points (as found on scatterplots) or overlapping lines (as found on parallel coordinates). We will then define a measure for occlusion.

### **5.1.1. Existing metrics for display clutter and density**

Here are the metrics that have been proposed in the literature for measuring display clutter.

**Rosenholtz et al.** [Rosenholtz et al. 05] provide a useful discussion of display clutter but note that most of the metrics have problems and few have been implemented so far. They also describe a new measure of clutter, based on predicting the level of feature congestion in maps, using image values such as luminance and colour contrast; however, this is not readily applicable to either parallel coordinate or scatter plots.



**Tufte's** [Tufte 83] *data ink ratio* and Frank and Timpf [Frank and Timpf 94] *ink per unit area* give a measure of crowdedness for traditional graphs and maps but they do not include any notion of hiddenness and are not directed towards computer displays.

**Brath** [Brath 97] in his “Metrics for Effective Information Visualisation” defines some metrics for characterising 3D digital images. Although most are not relevant in the context of this work, he does include several metrics for 2D plots. These are:

data density = no. of data points / no. of available pixels

occlusion percentage = no. of points completely obscured / no. of points.

**Bertini and Santucci's** work on reducing clutter in scatterplots [Bertini and Santucci 05] use Brath's metrics, as base measures, which they refer to as data density and collisions per area. However, they divide the plot into small squares and attempt to preserve the relative data density between all the squares in the plot, and in this process use some more complex quality measures.

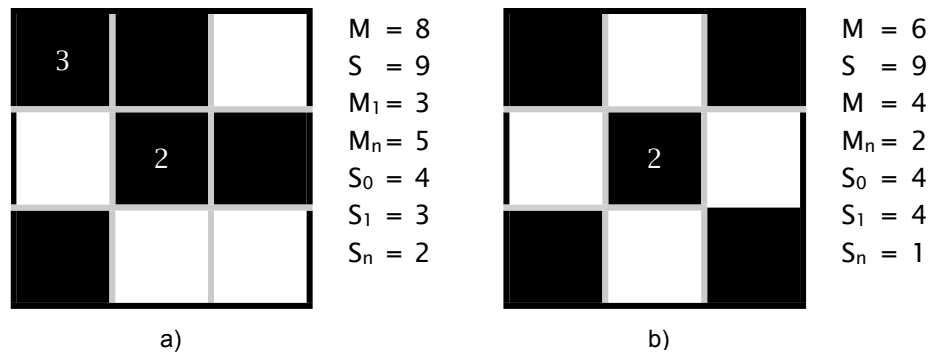
**Miller and Wegman** [Miller and Wegman 91] consider the plot densities for lines constructed in parallel coordinates. They study theoretical density plots rather than individual data points and use these to produce visualisations of density functions over high dimensional spaces. This work does not measure clutter per se, indeed with a theoretical probability distribution they effectively have an infinite number of infinitely thin lines, but plot density is clearly closely related to clutter.

### 5.1.2. Defining a measure for occlusion

From the literature review, it is clear that there is no commonly agreed measure for display density or clutter on scatterplots or parallel coordinates. Different things have an effect on perceived clutter, such as hidden or partially hidden screen objects, the closeness of adjacent objects and the merging of different coloured objects [Ware 04]. In order to have a computationally tractable measure, a fairly simple measure is adopted based on hidden points, as the important question for the user is “how much of the data cannot be seen?” Depending on the visualisation, a drawn object may be a single pixel point, point symbol, glyph, line or some text; all may result in many pixels being plotted on the screen, some of which may be on top of existing pixels. This investigation will only be considering clutter in relation to hidden or occluded objects and will not be taking into account other effects on perceived clutter mentioned above. More specifically, the object in question will be the scatterplot points and parallel coordinate polylines and these will be formulated in terms of screen pixels, as this is something that can be readily measured.

For a given screen region (in particular the interior of the sampling lens), let  $S$  be the total number of available pixels and  $M$  be the number of plotted data points. Note that in this context each plotted point is the size of a screen pixel.

Figure 5-1



(a) Example of a scatterplot with overplotting occurring at 2 of the pixels.

(b) Example of overplotting with 2 lines crossing at the centre point. Corresponding raw data values are given for each example.



So, if the lens is circular with radius  $R$  pixels,

$$S = \pi R^2$$

For a parallel coordinate plot the number of plotted points can be specified in terms of the number of lines crossing the lens,  $L$  and the average number of pixels per line,  $P$ , so that

$$M = P * L$$

Note that, in general,  $M$  is not the number of actual pixels with points plotted on them as some points will be overplotted on the same pixel. Therefore, the number of plotted pixels is usually less than the number of plotted points.

The following raw data values can be defined from which the occlusion measures will be obtained:

$M_1$	number of plotted points on their own pixel
$M_n$	number of plotted points sharing a pixel
$S_0$	number of empty pixels
$S_1$	number of pixels with 1 plotted point (same as $M_1$ )
$S_n$	number of pixels with more than 1 plotted point

Note that  $M = M_1 + M_n$  and  $S = S_0 + S_1 + S_n$  and always  $M_1 = S_1$ , but  $M_n \geq 2 S_n$  as each overplotted pixel contains two or more overplotted points. To illustrate these raw data values in practice, examples for a scatterplot and parallel coordinate plot are now given.

Figure 5-1a shows an example of a 3x3 pixel area that may be found in a scatterplot. There are 8 plotted points, resulting in one pixel overplotted with 3 points, one pixel overplotted with 2 points and 3 more pixels with single points. On the other hand, Figure 5-1b is more typical of a parallel coordinate plot, with two lines crossing at the centre pixel. Note that the lines are assumed to be extremely thin.

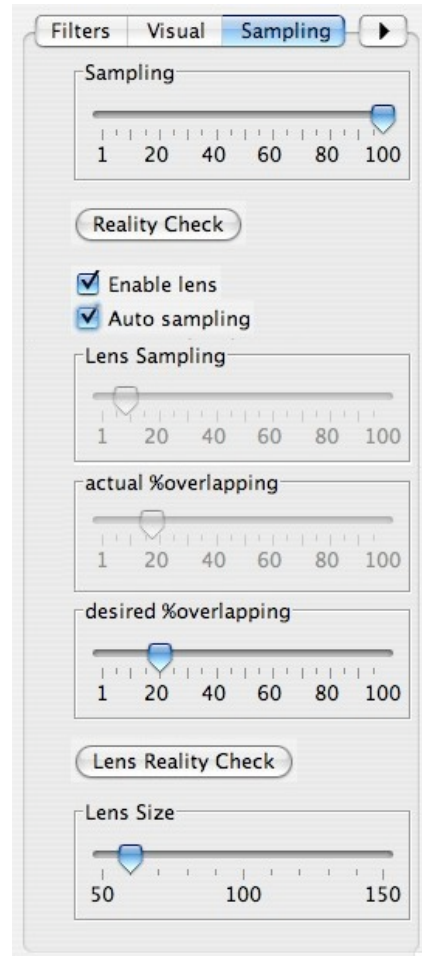
The initial clutter measure called %overlapping was defined as the percentage of pixels with more than 1 plotted point. This was based on the notion of giving the viewer an indication of proportion of the pixels that were actually overplotted. For the raw data values this is given by  $100 * S_n / (S_1 + S_n)$ . The range being 0 (all plotted points on their own) to 100 (no single plotted points).

## 5.2. The first attempt at auto-sampling

Auto-sampling was initially implemented for the scatterplot visualisation, utilising the occlusion measure defined in the previous section. As the points within the lens were plotted, a list was constructed of all the points within the lens that occupied each plotted screen position<sup>2</sup>. From this it was relatively easy to count up the number of

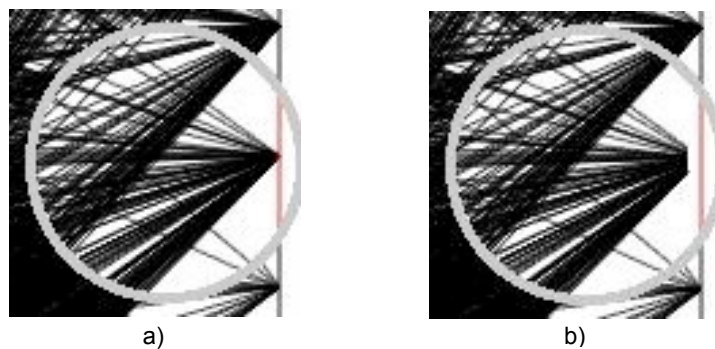
<sup>2</sup> Initially this was at a pixel level, but later on, this was amended to take into account the size of the plotted points.

Figure 5-2



Sampling control panel for the auto-sampling version of the Sampling Lens. Note the addition of the desired overlap control and the feedback of the actual overlap value.

Figure 5-3



(a) Lines meeting at a point on an attribute axis (b) Setting a non-overlap zone near to an attribute axis so that lines meeting at a point on the axis are not counted as overlapping. Note that for illustration purposes the clipping has been increased to 5 pixels (normally set to 1).

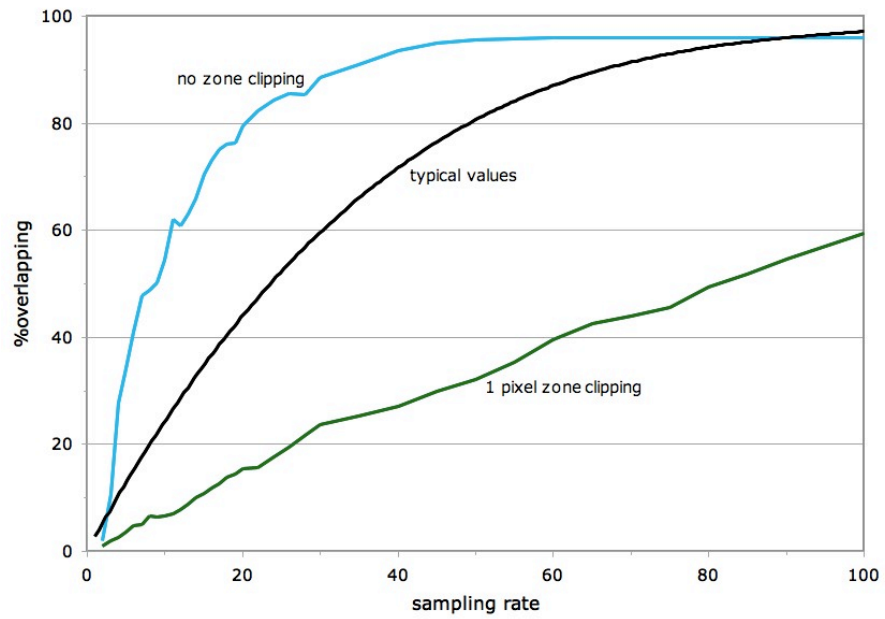
plotted points that overlapped, determine  $S_n$  and  $S_1$  and hence calculate %overlapping. This was implemented as an iterative process - the painting class that *knows* about the points and lines to be plotted, notified the sampling control of the new %overlapping, which in turn adjusted the lens sampling rate to move towards a preset value. It turned out that the feedback mechanism was unstable so algorithmic damping was applied to stop over-swings. However, feedback loop patterns were still present in some circumstances that showed as a flickering display within the lens. A successful solution was to recognise repeating patterns and break the loop.

The sampling control panel was initially updated to include an auto-sampling on/off switch and a slider to give feedback to the users on the calculated %overlapping value. Obviously in auto-sampling mode, the user cannot alter the lens sampling rate manually. In use, aiming for a %overlapping value of 20 gave a reasonably uncluttered view of the data with the test data, however, it was felt that the user should be able to adjust this target value so a *desired %overlapping* slider control was added. This control was subsequently found useful. The control panel is shown in Figure 5-2.

Having implemented useable auto-sampling in the scatterplot application, providing the same functionality with parallel coordinates was investigated. When drawing the lens region, the extent of all the lines cross the lens had to be calculated, so, the initial attempt made use of this data. A model was produced that uses the length and angle of all the lines crossing the lens to estimate the line overlaps and then combines these to give an overall %overlapping value. This is referred to as the *lines* algorithm and is described fully in Section 5.4. Although auto-sampling based on this measure worked reasonably well to enable details to be seen in dense regions of the plot, it did not give consistent results across the whole of the parallel coordinate plot being tested. One problem region was where many lines met at a vertical axes as revealed in Figure 5-3a.

This led to a dramatic increase in the occlusion measure as the number of lines intersecting increased significantly. To negate this effect, a non-overlap zone either side of each axis was defined that effectively clips the lines before they meet on the axis. An exaggerated view of this line clipping is given in Figure 5-3b. The width of this zone in pixels can be set between 0 and 20 with a slider control (shown in Appendix D. 1) however, a 1-pixel setting improved the behaviour of the *lines* measure as expected due to the assumption in the line intersection algorithm that lines are extremely thin. Plots of occlusion measure against sampling rate are given in Figure 5-4 that clearly show this decrease in the *lines* occlusion measure when zone clipping is applied. Note that the amount of extra estimated occlusion is dependent on the type of attribute value. For example, if nominal data (e.g. year of manufacture) then there is likely to be fewer individual points on the axis and hence greater chance of many lines meeting at the same point. The justification of using this zone clipping is discussed further in

Figure 5-4



Behaviour of the *lines* algorithm with and without zone clipping in the exceptional case where many lines meet on a vertical axis. These can be compared to typical values obtained with the lens is at other regions of the plot.

<p>overplotted%  <math>100 * S_n / (S_1 + S_n)</math></p>	<p>The percentage of pixels with more than 1 plotted point. The range is 0 (all plotted points on their own) to 100 (no single plotted points)</p>
<p>overcrowded%  <math>100 * M_n / (M)</math></p>	<p>The percentage of plotted points that are in pixels with more than 1 plotted point. The range is 0 (all plotted points on their own) to 100 (no single plotted points)</p>
<p>hidden%  <math>100 * (M_n - S_n) / (M)</math></p>	<p>The percentage of plotted points that are hidden from view due to being overplotted. Note that pixels with more than 1 plotted point will be showing the top plotted point. The range is 0 to just less than 100, depending on the number of pixels.</p>

Key:  $M$  = total points plotted,  $M_n$  = plotted points sharing a pixel,  $S_1$  = pixels with one plotted point,  $S_0$  = pixels with more than one plotted point.

Table 5-1

Definition of the occlusion measures

Section 5.5 where different methods of calculating occlusion are investigated.

Having undertaken this somewhat informal investigation of measuring occlusion within the lens for parallel coordinate plots, it was decided to take a more formal, engineering approach to providing auto-sampling. First we look at other occlusion measures than %overlapping and consider using a more direct measurement based on pixels rather than the partly probabilistic-based *lines* measure.

### 5.2.1. Defining further occlusion measures

Going back to Brath's *occlusion percentage* (i.e. number of points completely obscured divided by the number of points) a measure called *hidden%* was defined as the percentage of plotted data points that are obscured. Considering the pixel view, the previous %overlapping measure was given the more appropriate name of *overplotted%*. In addition, another measure called *overcrowded%* was defined that had some flavour of the other two. These measures are described with their formulae in Table 5-1. When calculated for the example 3x3 plots presented earlier, Figure 5-1a gives *overplotted%* = 40, *overcrowded%*  $\approx$  62 and *hidden%*  $\approx$  37 whereas in Figure 5-1b *overplotted%* = 20, *overcrowded%*  $\approx$  33 and *hidden%*  $\approx$  17. All three occlusion measures have some level of face validity, however, they can vary substantially.

To examine whether this is an issue in practice, experiments were conducted to compare these measures empirically using a range of lens positions on a parallel coordinate plot.

## 5.3. Investigating occlusion measures for parallel coordinates

This section describes a series of experiments to determine the best occlusion measure, of the ones defined above, for estimating the density of lines in a parallel coordinate plot.

### 5.3.1. The experiments

To investigate the relationship between the proposed occlusion measures, we require the raw values  $M_1$ ,  $M_n$ ,  $S_0$ , etc., defined in 5.1.2. As these are in terms of plotted pixels, the parallel coordinate lines have to be rasterised to a pixel grid and the number of plotted points on each pixel counted. The Sampling Lens application was instrumented to gather this data and calculate values for the three occlusion measures. In addition, the rasterised lens region was visualised as a *lens overlap density map* in a separate window that proved useful during the experiments. Two other control panels also formed part of this instrumentation, one to set various parameters and the other to control the automatic collection of data. A description of the instrumentation of the Sampling Lens and some of the practical issues this raised are presented in Appendix D.1.

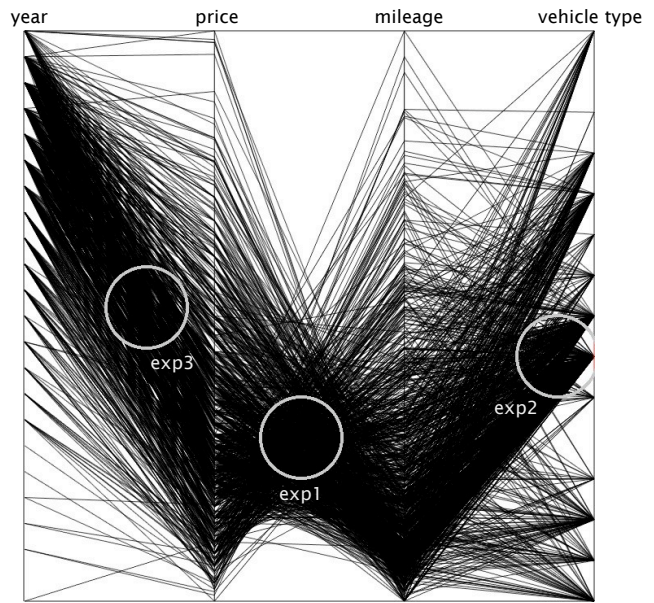


Figure 5-5

Parallel coordinate plot using 1K car dataset (labels and lens positions for exp1, 2 & 3 are superimposed)

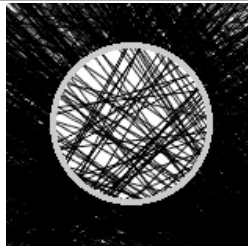
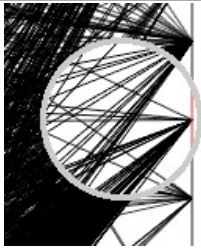

exp1	exp2	exp3
Most lines crossing each other at almost acute angles	Many lines ending at a point	Most lines cross at shallow angles
		

Table 5-2

Lines within the lens at 10% lens sampling rate

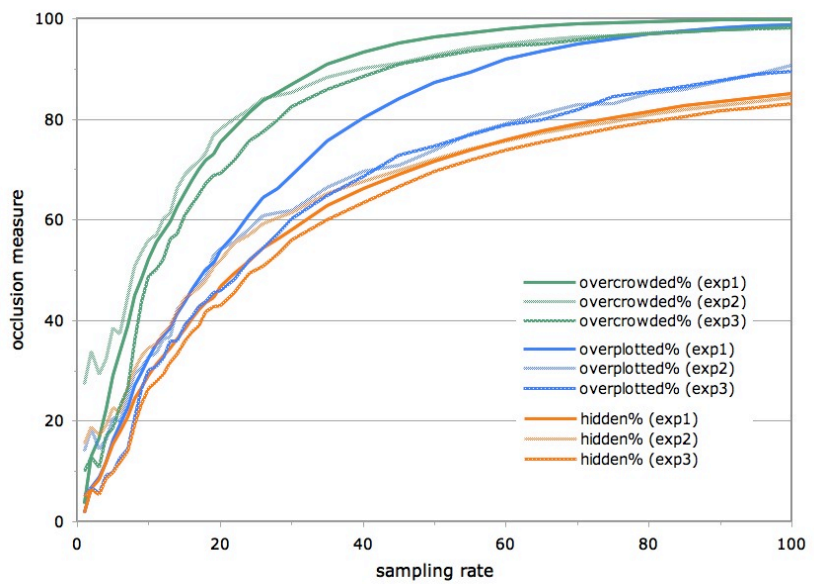


Figure 5-6a

Occlusion measures for exp1, exp2 and exp3 plotted against sampling rate

Most of experiments use the Portland cars dataset, described in Appendix B.1. The data plotted on the parallel coordinates plots is: year of manufacture, price, mileage and vehicle type (given as an integer code). All values are in ascending order, so for example, a high mileage car would appear towards the top of the third attribute line and probably towards the bottom of the second attribute line as a low cost car. Figure 5-5, is a screen shot of the parallel coordinate visualisation of 1000 records of the cars dataset. The majority of experiments used this random subset as the occlusion was very high with the full set and reduced the sensitivity of the measurements. Experiments were also conducted on the full dataset and datasets up to 10,000 records to verify that the results do scale for larger and different datasets. An overview of all experiments is provided in Appendix C.

As the aim of this investigation was to evaluate different occlusion measures it seemed appropriate to choose a wide range of line crossing patterns within the lens, especially as the results of several scenarios calculated earlier (Section 5.1.2) varied substantially. Hence the choice of lens positions for the first set of experiments, exp1 to exp3, shown in Figure 5-5 and described in Table 5-2.

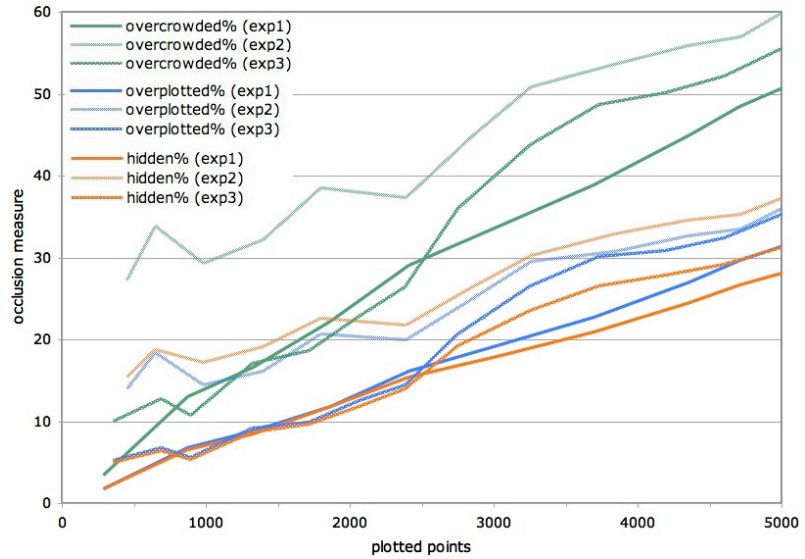
### 5.3.2. Empirical results

Figure 5-6a shows the calculated occlusion values over a range of sampling rates for the lens positions described above. For each measure, there appears to be a definite trend with `hidden%` giving the lowest estimate of occlusion over the range of sampling rates and `overcrowded%` giving the highest estimate. The values of exp1 are consistently above those of the other experiments.

The computation of the occlusion measures is based on the number of pixels and the number of plotted points (see Table 5-1). As the lens size is the same in all experiments, the number of available pixels is constant, however, the number of lines within the lens (and subsequent number of plotted points) is different due to the change in lens position between the experiments. To account for this, the graph can be re-plotted using the number of plotted points as the x-axis. As seen in Figure 5-6b, the curves for each experiment are generally aligned more closely, especially for `overplotted%`, although the exp1 results for the latter measure still diverges noticeably from the results for the other two experiments.

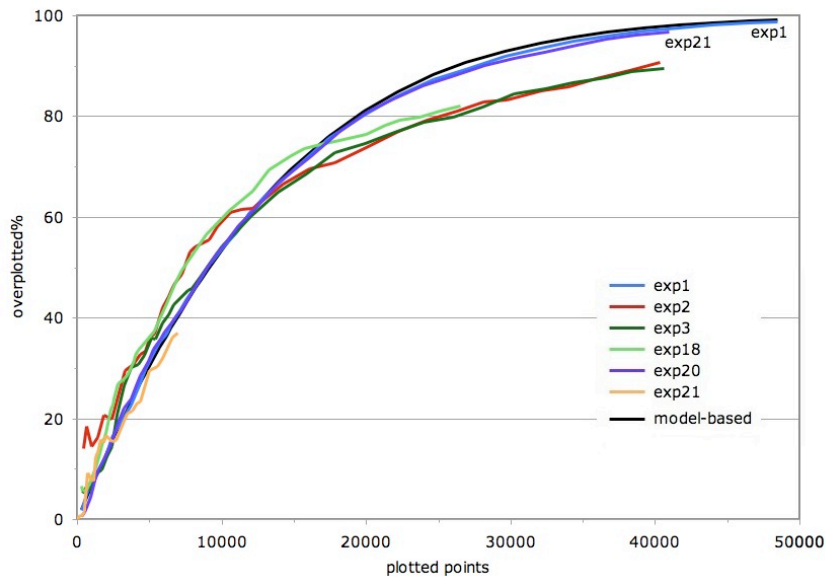
Looking at the behaviour of the different measures at low densities (approximately 20%) where the user will typically be working, it appears that `overcrowded%` is more sensitive to small changes in the raw data than the other measures. Figure 5-6c shows a zoomed in section of Figure 5-6b to illustrate this. As `overplotted%` appears marginally better at the lower densities, three more experiments were carried out with the lens at different, more extreme regions of the parallel coordinate plot. All six lens

Figure 5-6c



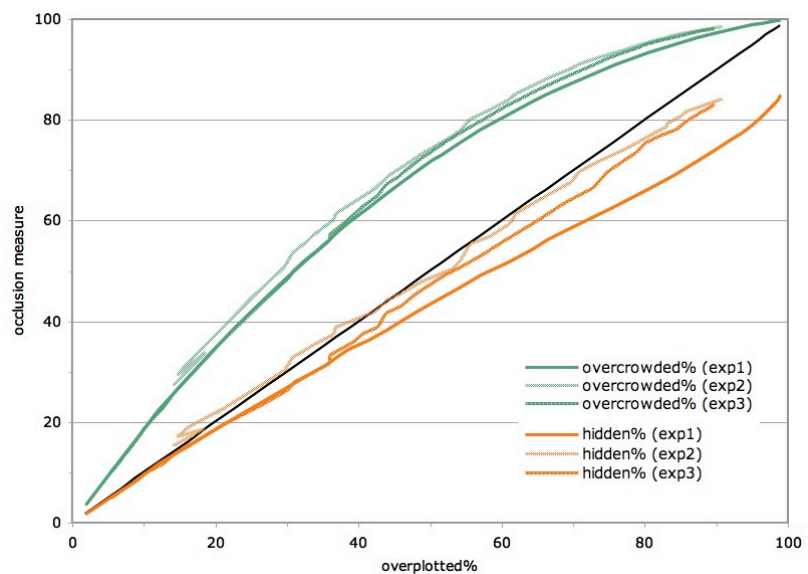
Behaviour of the occlusion measures at low densities

Figure 5-7



Overplotted% occlusion measures for a wide range of line crossing patterns (experiments 1, 2, 3, 18, 20 and 21)

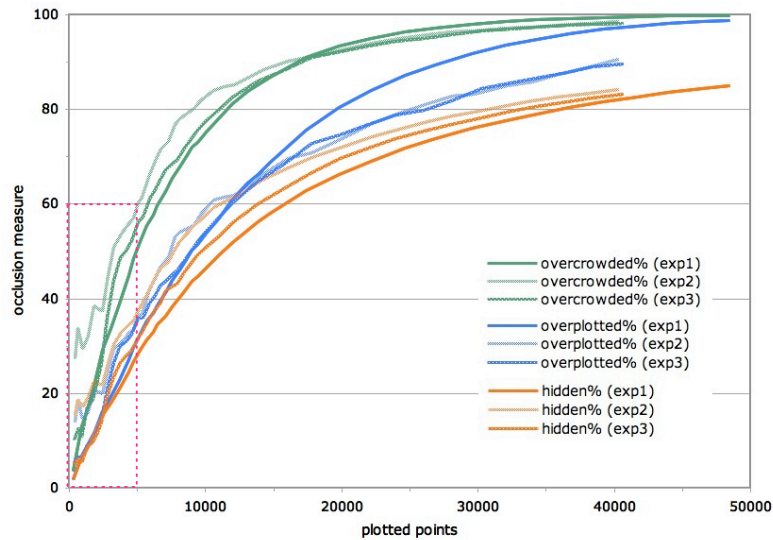
Figure 5-9



Occlusion measures for experiments 1, 2 and 3 normalised against overplotted%



Figure 5-6b



Occlusion measures for exp1, exp2 and exp3 plotted against plotted points. The red dotted box indicates the enlarged section shown in Figure 5-6c.

regions are shown in Figure 5-8 and the experimental results are given in Figure 5-7.

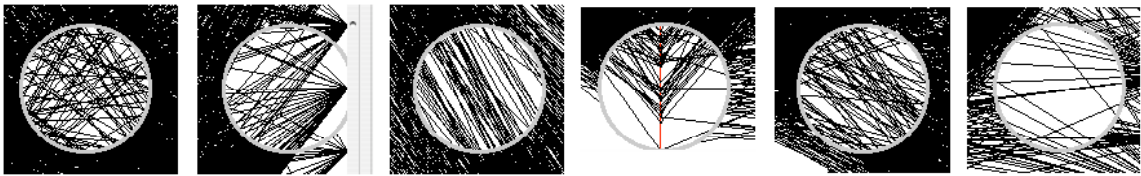


Figure 5-8

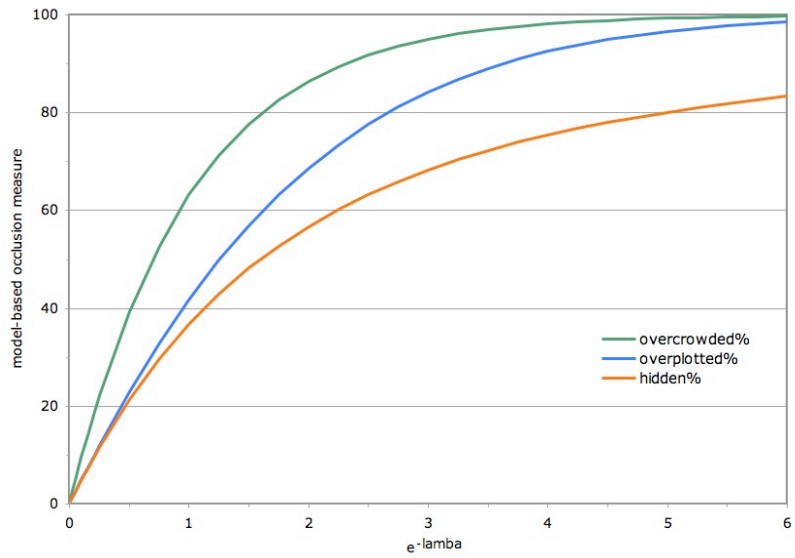
Lenses at 10% sampling rate for experiments 1, 2, 3, 18, 20 and 21

The lines for the overplotted% measure seem to split into two distinct groups when the density is higher. The group with lower occlusion values includes the lens positions for experiments 2, 3 and 18 (second, third and fourth from the left in Figure 5-8) that tend to have more lines running parallel or at shallow angles to each other, rather than the criss-cross patterns seen in the other lens positions. For the same number of plotted points, this results in a greater proportion of single point (and empty) pixels and hence lower occlusion values, given the equation  $100 * S_n / (S_1 + S_n)$ .

To see the relationship more clearly, the data was normalised against one of the measures, overplotted%. Figure 5-9 shows the resulting graph. Note that the overplotted% values lay on a straight line (drawn in black). The overcrowded% values (in green) are almost coincident for all three lens positions however, the values are markedly higher than overplotted%. Two of the hidden% experimental lines (in orange) lie close to overplotted%, however, the other one (exp1) is slightly underestimating the occlusion measure at higher densities. This anomaly is possibly due to the increase in the proportion of single point pixels as mentioned earlier.

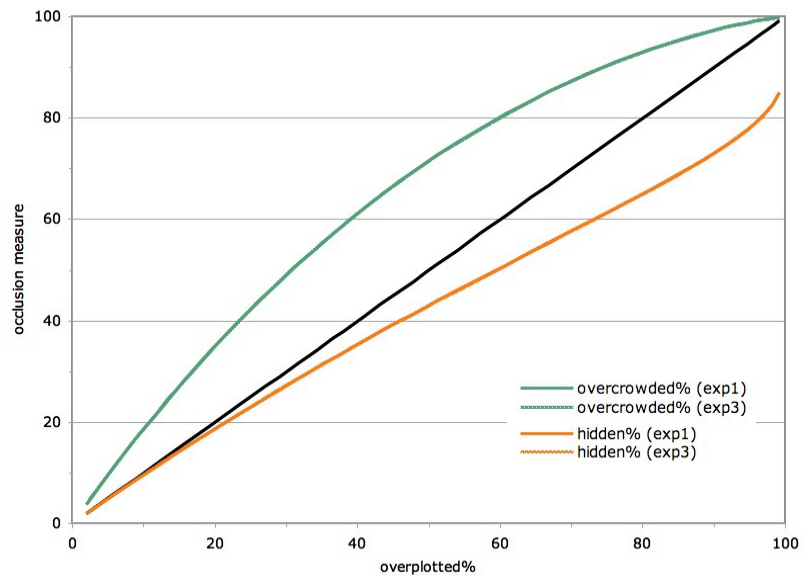
However, overall there is a good agreement over a range of patterns across the lens and between the three measures, which is unexpected as the measures are based on

Figure 5-10



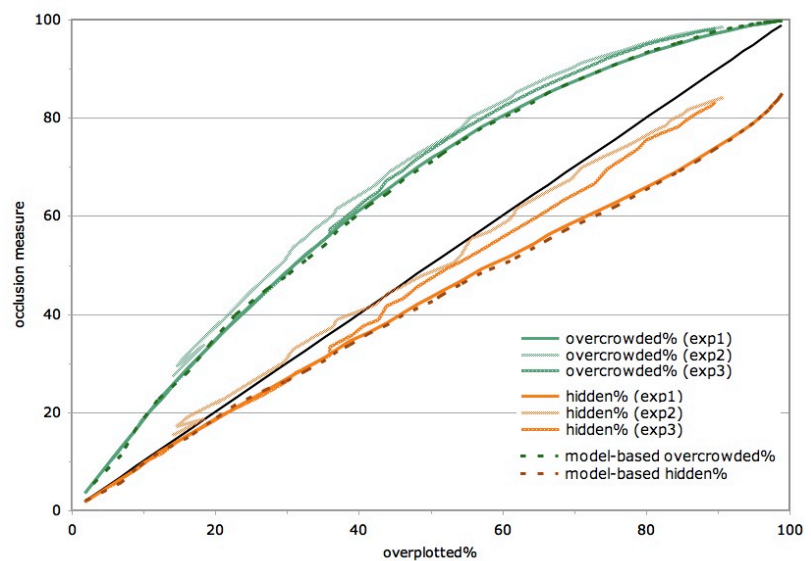
Model-based measures, overplotted%, overcrowded% and hidden%

Figure 5-11a



Theoretical curves for measures based on random point placement

Figure 5-11b



Comparing theoretical and empirical results

different raw data values. To investigate the relationship further the next section develops a simple theoretical model.

### 5.3.3. Theoretical model

The following model was devised by the author and supervisor. Imagine randomly plotting the  $M$  points over  $S'$  pixels (not necessarily all  $S$  pixels). Assuming that  $M$  and  $S'$  are quite large (so that combinatorics approximate to exponentials) and  $\lambda$  is defined as the ratio  $M/S'$ , it is possible to derive the expected values for the raw measures (5.1.2) as:

$$\begin{aligned} E(M_1) &= M e^{-\lambda} \\ E(M_n) &= M (1 - e^{-\lambda}) \\ E(S_0) &= S e^{-\lambda} \\ E(S_1) &= M e^{-\lambda} = S \lambda e^{-\lambda} \\ E(S_n) &= S (1 - (1+\lambda) e^{-\lambda}) \end{aligned}$$

Using the above equations, the expected overplotted%, overcrowded% and hidden% values can be calculated as:

$$\begin{aligned} \text{overplotted\%} &= 100 (1 - \lambda e^{-\lambda}) / (1 - e^{-\lambda}) \\ \text{overcrowded\%} &= 100 (1 - e^{-\lambda}) \\ \text{hidden\%} &= 100 (1 - (1 - e^{-\lambda}) / \lambda) \end{aligned}$$

Plotting the equations for these model-based measures gives the chart in Figure 5-10.

Utilising only the number of plotted points and number of available pixels from experiments 1 and 3, Figure 5-11a shows the expected values for the measures normalised against overplotted%. The similarity with Figure 5-9 is striking and to demonstrate the closeness, Figure 5-11b combines the results of the theoretical model with the empirical results. Although the real parallel coordinates points are not randomly placed (indeed they are in lines.), the relationship between the different measures of occlusion is very similar in practice as in this very simple theoretical model.

Having looked at the different occlusion measures we now have to decide which one is the best. The overplotted% value is based on the number of occupied pixels (pixels positions), so in a sense it is a more viewer-centric measure whilst overcrowded% and hidden% are based on the number of plotted points, hence they are more data-centric measures. Overplotted% was chosen as the occlusion measure as it is more viewer-centric and is also the middle of the three measures when plotted. In the end, it does not matter too much as, although all these measures are different, they are functionally related in practice, so a measure of any one is a measure of them all.



## 5.4. Methods for calculating occlusion

We will look at ways of calculating the chosen occlusion measure, with the aim of finding a method that is both accurate and efficient. Recall that the purpose of this investigation is to enable automatic clutter reduction within the Sampling Lens for parallel coordinates and facilitate the user to detect interesting patterns within cluttered displays. An occlusion measure has been selected from a set of possible occlusion metrics, so, now an effective method of calculating this is required, remembering that a rapid response is needed to maintain reasonable interactivity of the lens.

The previous experiments used a very direct data-driven pixel counting method to calculate occlusion, the *raster* algorithm. Two other way of estimating overplotted% have been devised, which are more model-based approaches, embodying simplifying assumptions about the data. In brief, the methods are:

*raster* algorithm - This rasterises the lines on a grid of a given cell-width (in pixels) and counts the number of plotted points on each grid cell to get an estimate of overplotted%. In the case when the cell-width is one pixel, this corresponds exactly to the desired value. It is thus the gold standard as it is based on the actual overlap of the lines being displayed, or more accurately the overlap of the plotted points that make up the lines.

*random* algorithm - This treats every plotted point as if it was randomly placed in the viewable pixels and calculates the overplotted% using probability. Here, only the number of plotted points comes from the data; everything else is based on a theoretical model.

*lines* algorithm - This estimates the intersection volumes of all the lines crossing the lens. This is partly data-driven in that it uses actual lines, but is partially model-based in the way the line overlaps are combined to give an overall overplotted% value.

The three different algorithms are now described in more details, including their theoretical underpinning.

### **raster**

This is clearly the simplest method and the only one that corresponds directly to the desired measurement in terms of plotted pixel points. In effect, this amounts to emulating the action of the graphics processing in drawing the lines across the area of the lens and counting the number of plotted points on each pixel.

### **random**

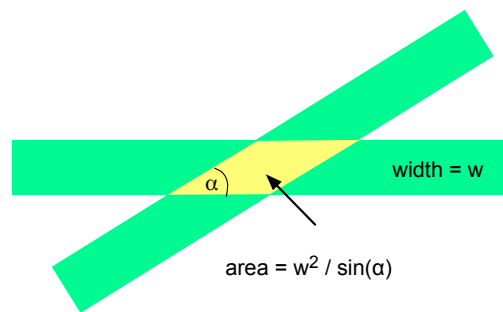
The second algorithm simply assumes that all the points on all the lines are individually randomly scattered over the available pixels. Given this very simplistic model, the number of points plotted in each pixel follows the binomial distribution

**Table 5-3**

$M_1$	number of plotted points on their own pixel
$M_n$	number of plotted points sharing a pixel
$S_0$	number of empty pixels
$S_1$	number of pixels with 1 plotted point (same as $M_1$ )
$S_n$	number of pixels with more than 1 plotted point

Raw data values for occlusion measures, reproduced from Section 5.1.2

**Figure 5-12**



Line overlap proportion

where  $p$ , the probability of a single point being plotted in a particular pixel is  $1/S$  (recall that  $S$  = number of available pixels). Hence, we can calculate expected values for the different raw values as follows:

$$\begin{aligned} E(M_1) &= M(1-p)^{M-1} \\ E(M_n) &= M - E(M_1) = M(1 - (1-p)^{M-1}) \\ E(S_0) &= S(1-p)^M \\ E(S_1) &= S * M(p)(1-p)^{M-1} = M(1-p)^{M-1} \\ E(S_n) &= S - (E(S_0) + E(S_1)) \end{aligned}$$

( $M_1, M_n$  etc. are defined in Table 5-3)

A value for the overplotted% can be obtained from:

$$\text{overplotted\%} = 100 * (1 - ((1-p)^M + M/S(1-p)^{M-1})) / (1 - (1-p)^M)$$

This algorithm is very cheap to calculate, as it only requires an estimate of the total number of points to be plotted, which is readily obtained from adding up the number of plotted points in all the drawn lines or, as it turns out, just counting the number of drawn lines. However, it is the least realistic, basically treating each line as a collection of points to be randomly distributed over the available pixels.

### lines

Here the lines crossing the lens (or a sample of the lines, in the case of a denser region) are taken and the overlap between each pair of lines is estimated by first checking the end points of the pair to verify whether the lines intersect and if they do, the overlap on one of the lines is calculated as:

$$\text{line overlap proportion} = \max(1.0, \text{wid} / (\text{len} * \sin(\alpha)))$$

where  $\text{wid}$  and  $\text{len}$  are the width and length of the chosen line in pixels and  $\alpha$  is the angle between the two lines (see Figure 5-12). Note that if the crossing lines are nearly parallel, they have a higher overlap than if they cross at right angles. The data (start and end points of each line) is freely available as part of the lens drawing method.

An average overlap,  $p_1$ , is computed by combining the line overlap proportion for all intersecting lines and weighting by the total length of the lines (non-intersecting lines are not included). Although there are many pairs of lines (almost  $L^2$  possible pairs; a line is not compared to itself) only an estimate is required, so it is sufficient to use a sub-sample to calculate this overlap proportion.

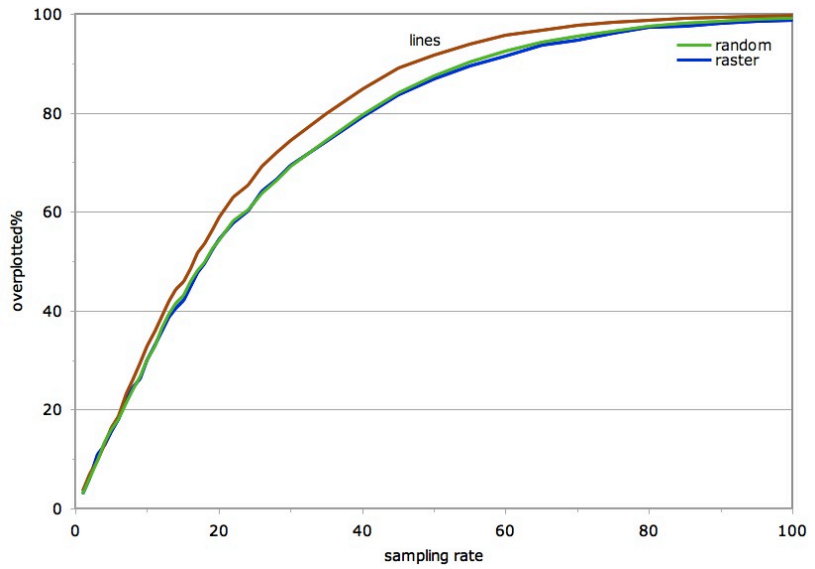
$p_1$  effectively tells us how a point plotted on a line is likely to be overplotted by one other line. To estimate  $p_{\text{free}}$ , the likelihood that a pixel will not be overplotted by any line, we can use:

$$p_{\text{free}} = (1-p_1)^{L-1}$$

Using the definitions from Table 5-3 we see that:

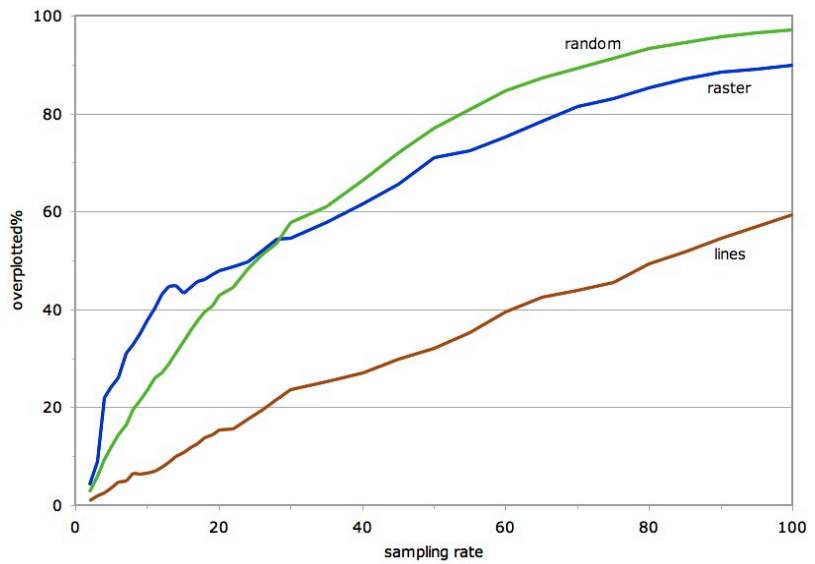
$$p_{\text{free}} = E(M_1) / M$$

Figure 5-13



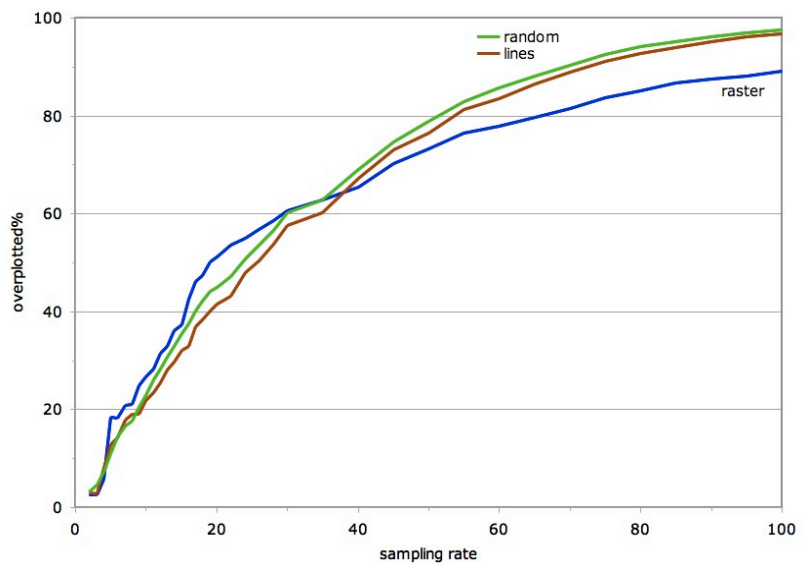
Three different occlusion algorithms (exp1)

Figure 5-14a



Three different occlusion algorithms for exp2

Figure 5-14b



Three different occlusion algorithms for exp3



where  $M$  is the total of the line lengths in pixels. We then use formulae similar to the *random* algorithm, but taking into account that the lines do not cover all pixels with equal probability. Inverting the equation for  $E(M_1)$ , we get an ‘effective’ number of pixels  $S'$  (which would be expected to be smaller than  $S$ ). Hence:

$$S' = 1/q \quad \text{where} \quad q = 1 - p_{\text{free}}^{1/(M-1)}$$

Note that this algorithm uses some of the same assumptions as the *random* algorithm, but bases it on some more direct measures of the lines as they actually fall.

## 5.5. Comparing the occlusion algorithms

The principal dataset used to compare the three algorithms was a 1000 record sub-sample of the Portland cars dataset as described previously in Section 5.3.1. However, larger and different datasets were also used to verify the results. The accuracy of the methods is considered first and then their computational efficiency is discussed.

### 5.5.1. Accuracy : which is good enough?

Figure 5-13 shows the results for exp1. It demonstrates very good agreement between the gold standard *raster* plot, based on actual screen pixels, and the *random* plot [1% above,  $sd=1.3$ ]. This is somewhat surprising as the latter is based on a standard distribution of a given number of plotted points and available pixels. The *lines* calculation is still fairly close, overestimating the overplotted% by about 6% [ $sd=3$ ].

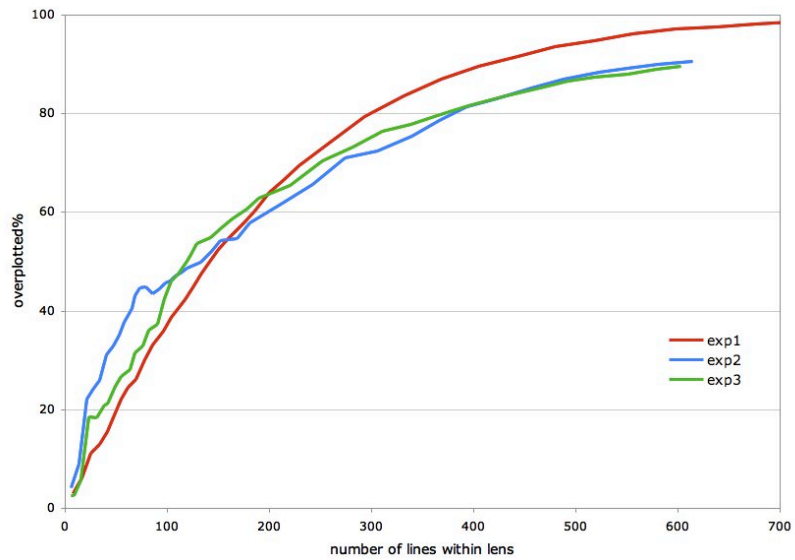
Similar graphs have been plotted for exp2 and exp3 and these are shown in Figure 5-14. The results for exp3 are again in fairly close agreement, although the *raster* values are lower than the other two algorithms by about 10% at higher line densities (a reason for this is suggested later on). The *lines* algorithm appears to cope well with the very different line patterns in the lens regions for exp1 and exp3, namely criss-cross as opposed to shallow angle crossings. This means that the 8% of the lines in exp3 that mostly intersect at shallow angles results in a total intersection area, which is similar to the much greater number of intersection (48%) at almost right angles as occurs in exp1. The number of lines was fairly similar at 740 and 600 for exp1 and exp2 respectively. Contrast these good estimates of overplotted% (in terms of agreement with our *raster* standard) with the results for exp2 where many points meet at a point on a vertical axis (Figure 5-14a). The obvious departure from the other occlusion values was discussed earlier (Section 5.2) where it was felt that including these axis point intersections unnecessarily overestimates overplotted% and hence setting a non-overlap zone has ignored them. Unfortunately this result is a sizeable underestimate and it could be argued that the situation has not been improved. Alternative solutions are proposed later.

**Table 5-4**

	average pixels per line	stdev
exp1	68.27	1.98
exp2	68.14	0.81
exp3	71.83	2.86

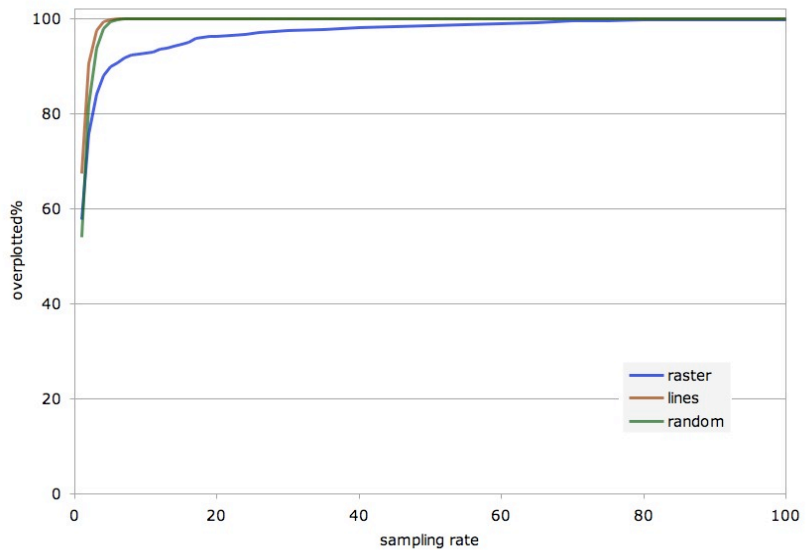
The average pixels per line of all the lines cross the lens is very consistent for a range of lens positions

**Figure 5-15**



*Raster* values for the three experiments, plotted against the number of lines crossing the lens

**Figure 5-16**



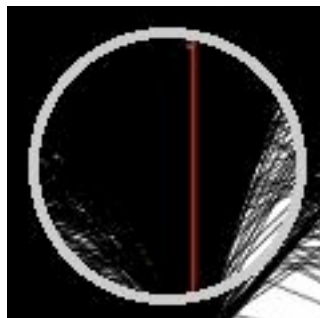
The three occlusion measures, *raster*, *lines* and *random* for a dense region of a 10,000 record dataset (exp7)

The data for this set of experiments was analysed further, such as comparing the results for each of the occlusion measures separately. The *random* curves show very good agreement in all regions of the parallel coordinate plot and when plotted against the number of lines within the lens as opposed to the sampling rate, result in practically coincident lines. As *random* overplotted% is directly proportional to the number of plotted points, by definition, this suggested that the average number of pixels per line is fairly constant. In practice it turns out to be true as set out in Table 5-4. This finding was useful when considering ways of speeding up the calculations. Incidentally, other line graphs were redrawn using the number of lines within the lens and this gave a similar small convergence of the curves.

In comparing the *raster* plots for the three experiments we are again faced with a slightly divergent pattern (see Figure 5-15) that was discussed in connection with the empirical results from the investigation into occlusion metrics (5.3.2). Remember that exp2 and exp3 are regions with few criss-crossing lines and it was suggested that this caused the inconsistency.

However, during a series of experiments with a large 10,000 record dataset (People 10K) it was noticed that the *raster* calculation was consistently giving an underestimate. Figure 5-16 shows a graph of the three measures from exp7. Due to the number of lines in this particularly dense region of the parallel coordinate plot, the occlusion measures are understandably high and only reduce noticeably at very low sampling rates. However, the *raster* value starts to drop away from the others at much higher sampling rates. Looking at a screen shot of the lens region (Figure 5-17) it is apparent that there is an almost empty area to the right. Going back to the equations for the measures, we see that the *lines* and *random* calculations do not take empty pixels into account, whereas the *raster* calculation does. This behaviour under non-uniform density is investigated in Section 5.6 and a solution is found.

**Figure 5-17**



Lens position for exp7 showing the small low density region to the right

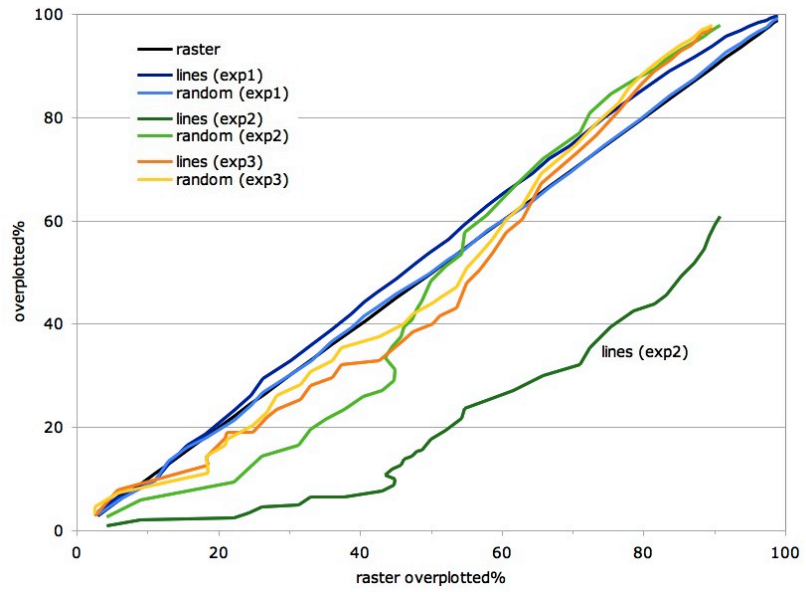


Figure 5-18

Exp1, 2 and 3 normalised against *raster* values

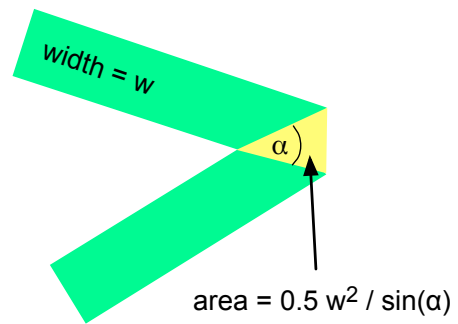


Figure 5-19

Modification of the original *lines* algorithm to deal with the special case of lines meeting at their end points

So far we have compared the three methods of calculating occlusion for three quite different lens positions and discussed particular problems with the *lines* and *raster* algorithms in certain extreme conditions. To compare all three experiments, it is useful to normalise the results against the gold standard *raster* value. This is shown in Figure 5-18.

The extent to which the *exp2* values deviate, especially for the *lines* algorithm can now be seen clearly. Recall that a fix has been applied that ignores the intersection of lines that meet on an attribute axis. This counteracts the severe overestimate but in turn has led to this underestimate. Alternative solutions have been considered although not implemented. One possibility is to assume the lines have a thickness when determining if they intersect so that lines at shallow angles to each other contribute to the occlusion measure – so far the lines are infinitely thin and hence even almost parallel lines do not intersect if their end points are clipped even a very small distance from the axis. Another possible solution to improve the robustness of its estimates, is to allow lines to meet on the axis but reduce the calculated overlap region for line ends meeting at a point. A modification to the parallelogram in Figure 5-12 to achieve this is given in Figure 5-19.

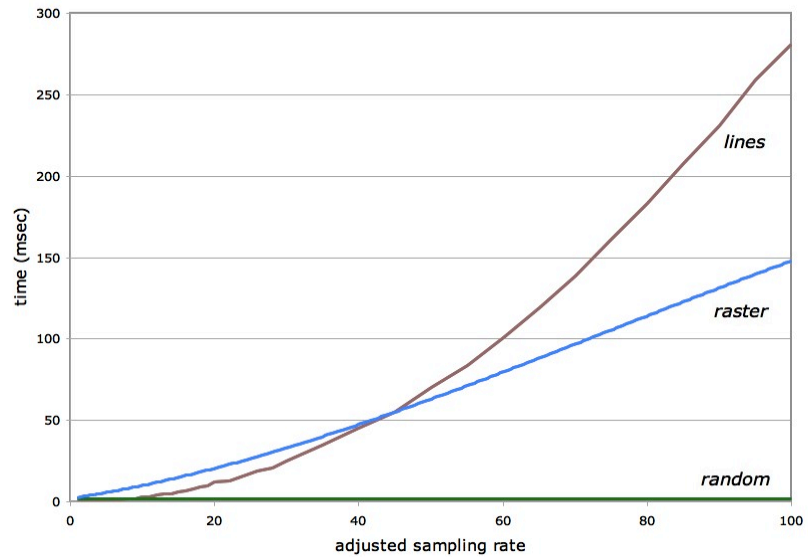
Apart from the *lines* curve for *exp2*, the *random* curve also noticeably underestimates *overplotted%* when compared to the *raster* normal values. It was thought that this may be due to the particular pattern of lines at this position on the plot and was investigated further. Studying the curves for *exp3* in Figure 5-18, these also show *overplotted%* being underestimated at low sampling rates and overestimated at higher sampling rates. However, looking back at the *raster* plots for *exp3* and *exp2* (Figure 5-15) and evidence from other experiments such as *exp7*, discussed earlier, it is likely that the skewness of the graph lines is due to the behaviour of the *raster* calculation in situations where there is noticeable non-uniform density across the lens. Details are in the section on dealing with non-uniform density across the lens (5.6).

To summarise, all three algorithms yield comparable results except in extreme situations. This is particularly noteworthy for the *random* algorithm as it embodies a fairly rudimentary model of the data. The *random* algorithm also performs passably in the difficult case when the lens overlaps an attribute axis, a case where even the direct measurement is problematic.

### 5.5.2. Efficiency: which is fast enough?

So far, whilst the *lines* algorithm has some problems in difficult cases, all the algorithms are potential contenders as estimates of the *overplotted%* occlusion measure. The *lines* algorithm uses the intersection volumes of all the lines crossing the lens, so in terms of efficiency, is  $L^2$  in the number of lines. The *raster* algorithm

Figure 5-20



Calculation times for the three algorithms

Table 5-5

exp	records	lines	painting	clipping	raster	lines
1	1000	750	2	7	150	280
4	5850	4130	30	50	1100	1600

The major components of time taken (in msec) to redraw the lens (painting, clipping and occlusion calculation) for the same dense region of a parallel coordinate plot with different number of records.

rasterises the lines to a grid of pixels, thus the time taken is proportional to the number of lines and the number of cells crossed by the lines. The *random* algorithm only requires an estimate of the number of plotted points and the number of pixels in the lens area and hence is orders of magnitude faster than the other methods. Figure 5-20 shows times (in msec) to perform the three algorithms at sampling rates between 1% and 100%. These were taken during *expl* from the Java implementation running on a 867MHz G4 PowerBook. The quadratic growth time for the *lines* algorithm is evident.

Now, for interactive feedback a response time of between 100 and 200 msec is acceptable [Card et al. 86] and would provide reasonably smooth movement of the lens. The lens redraw time is made up of three main components - clip the lines to the lens and any attribute axes going through the lens, calculate the occlusion value and finally draw the lens background and the clipped lines on the display. For the experimental setup for *expl* (i.e. 50 pixel lens radius and dense region of the 1000 record cars dataset) clipping takes about 7 msec and the drawing time is 3 msec. These values are presented in Table 5-5. There are approximately 750 lines drawn with no sampling. Increasing the lens radius to 100, increases the drawing time to 30 msec, which is consistent with the fact that line drawing time in Java2D is proportional to the length of the line (as later discovered when comparing OpenGL and Java2D - Appendix D.3). So for small datasets the major component of lens redrawing is in the occlusion calculation.

Of course, a 1000 record dataset is relatively small and as mentioned earlier, was chosen to give a larger range of occlusion measurements over the range of lens sampling rates. Using the full cars dataset, but otherwise the same settings as *expl* (*exp4*), increases the clipping and drawing times to 30 and 50 msec (up from 7 and 2). This is not surprising as there are now 4130 lines crossing the lens. Similarly, the *raster* and *lines* calculations take much longer at 1100 and 1600 msec respectively (up from 180 and 280). It is noteworthy that during the *lines* calculation, over 17 million line pairs are checked and the volume of nearly 7 million resulting intersections are summed. It should be pointed out that the timings are the worst case of no sampling within the lens. At sampling rates of 10%, redrawing is faster by at least a factor of 10. However, it is disconcerting for the user if the lens seems to stick when being moved from a low density to a high density region of the plot and they have to wait for auto-sampling to reduce the sampling rate and hence speed up the lens redraw.

Before looking at ways of speeding up the raster and lines calculations, it should be noted that there is a difference between the modes of use of the two model-based algorithms compared with the data-driven *raster* algorithm which affect the lens redraw times. Given initial data for a lens position (average crossing area for *lines* and

Figure 5-21

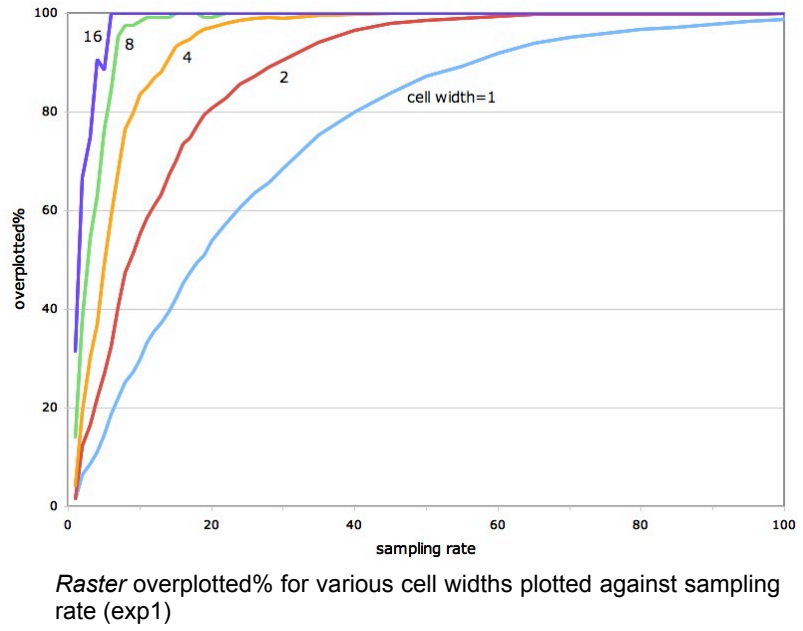
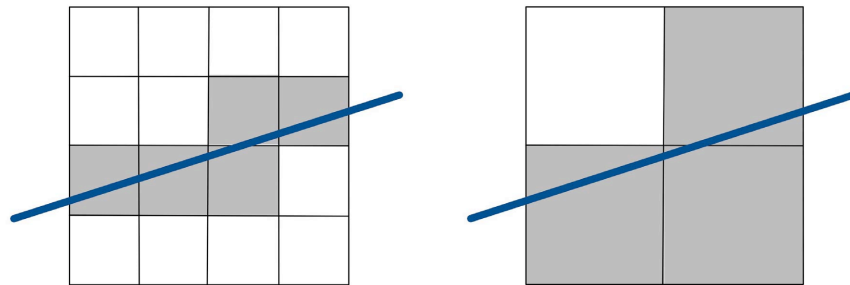
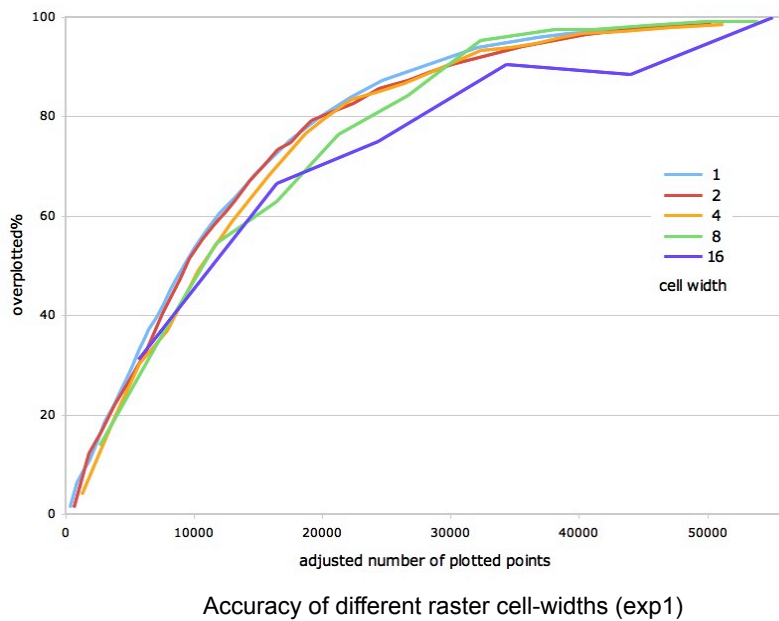


Figure 5-22



When rasterising a line, the proportion of cells crossed increases with the cell size. In the left diagram 5/16 (31%) of the cells are crossed, whereas in the right diagram 75% (75%) are crossed.

Figure 5-23





number of pixels for *random*) overplotted% can be calculated for any sampling rate as the equations of the curves are known. Thus, the appropriate sampling rate can be chosen directly to give the desired occlusion measurement, overplotted%. However, the *raster* calculation is based on the actual lines plotted at a given sampling rate. The *raster* calculation therefore, has to be used in an iterative cycle, adjusting the sampling rate and recalculating the measure at each iteration. The actual time taken to use the *raster* algorithm is perhaps 5-10 times that given for a single iteration.

### speeding up the *raster* calculation

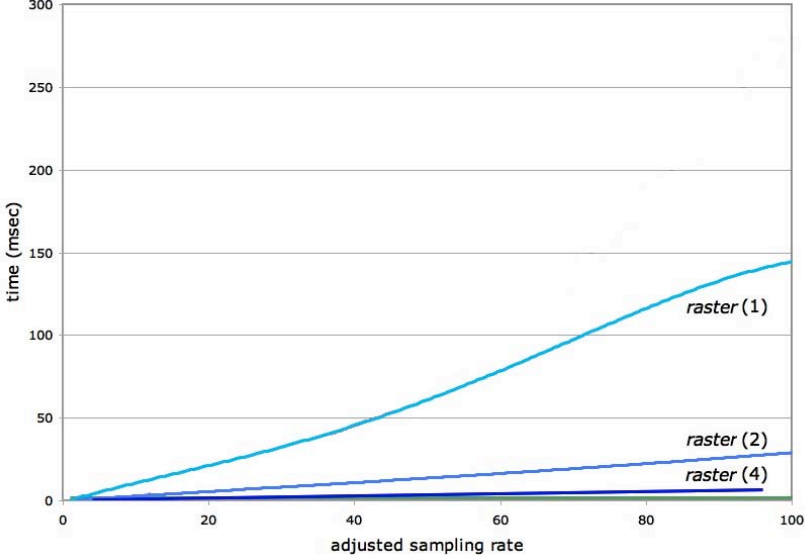
Rasterising lines is of course performed by the Java2D graphic libraries when line drawing, but Java does not make overplotting data available. The more sophisticated graphics processors do however, rasterise lines and also maintain a count of the number of points plotted on each display pixel, but at the time of undertaking this work, it was not possible to use data from the graphics card. So faced with single pixel counting it was thought that using raster grids greater than 1 pixel wide would reduce the calculation time considerably and hence this was implemented in the software. Some of the experiments were re-run, with data collected for the *raster* algorithm for cell-widths of 1, 2, 3, 4, 5, 6, 8, 10, 12, 16 and 20. Figure 5-21 shows the results for *expl*, although this is typical of other experiments.

It appears that the *raster* calculation is systematically overestimating overplotted% with increasing cell width. However, a given line will cross a greater proportion of the grid cells when the cells are larger - the proportion of cells crossed being on average proportional to the grid size, as illustrated in Figure 5-22. Therefore, to maintain the correct proportions, we need to sub-sample the lines when using coarser grids.

Figure 5-23 shows the results of adjusting the x-axis to account for the necessary sub-sampling and evidently little change to the accuracy occurs at a cell-width of 4 and even at a cell-width of 8 the values would be acceptable. Note that due to the adjustment for the number of plotted points, as detailed above, there are less points on the curves for higher cell-widths and inaccuracies are magnified.

An additional benefit of sub-sampling the lines together with fewer grid cells plotted per line, is reduced calculation time that should result in a roughly  $N^2$  speed increase with larger cell sizes. The measured time to perform the *raster* calculation for a range of cell widths is given in Figure 5-24 (the *lines* and *random* times have been included for comparison). At 4 pixels, the calculation time has decreased considerably, as predicted and hence increases the usefulness of the *raster* method in our search for an efficient and effective way to measure occlusion. Graphs for other lens positions, such as those given in Table 5-2, show a similar behaviour.

Figure 5-24



Reduction in the calculation times of the *raster* algorithm with increasing cell widths. The *lines* and *random* times have been included for comparison.

### speeding up the *random* calculation

In contrast to the *raster* algorithm, the *random* calculation depends only on a count of the number of points to be plotted. This comes almost for free as a side effect of other calculations, but in the case of very large numbers of lines, it can be easily estimated. As mentioned earlier (Table 5-4), line lengths have a standard deviation of less than 3% of their average, so sampling the lengths of even 1000 lines would give an error of less than 0.1%.

### speeding up the *lines* calculation

It has already been mentioned that the *lines* calculation time is dependent of the number of lines crossing the lens. In Figure 5-24 the longest time of 280 msec is based on 750 lines, that is over half a million line pairs and it can be seen that the time decreases exponentially with sampling rate that is itself proportional to the number of lines. Therefore, using a sample of the lines within the lens would reduce the *lines* calculation times by several orders of magnitude whilst not substantially altering its accuracy<sup>3</sup>.

### 5.5.3. The winner

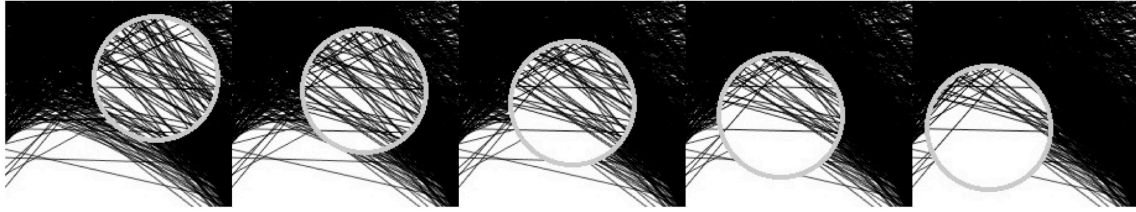
Combining efficiency with our accuracy measurements from Section 5.5.1, we can see that *raster* algorithms with cell sizes up to 4 pixels has little noticeable effect on accuracy and leads to a significant decrease in processing time of at least 90% compared with the single pixel raster. The *lines* algorithm is at first sight slower, but being model-based, it can use sampled data and does not require an iterative process. However, it is the least robust method in extreme situations. So the clear winner is the *random* algorithm as it is not only very accurate in normal cases and reasonably stable in difficult situations, but it is also almost instantaneous compared with both other algorithms. However, we have seen that noticeable uneven density across the lens leads to inaccuracies. The next section describes the subsequent investigation into this and pursues a solution.

## 5.6. Dealing with non-uniform density

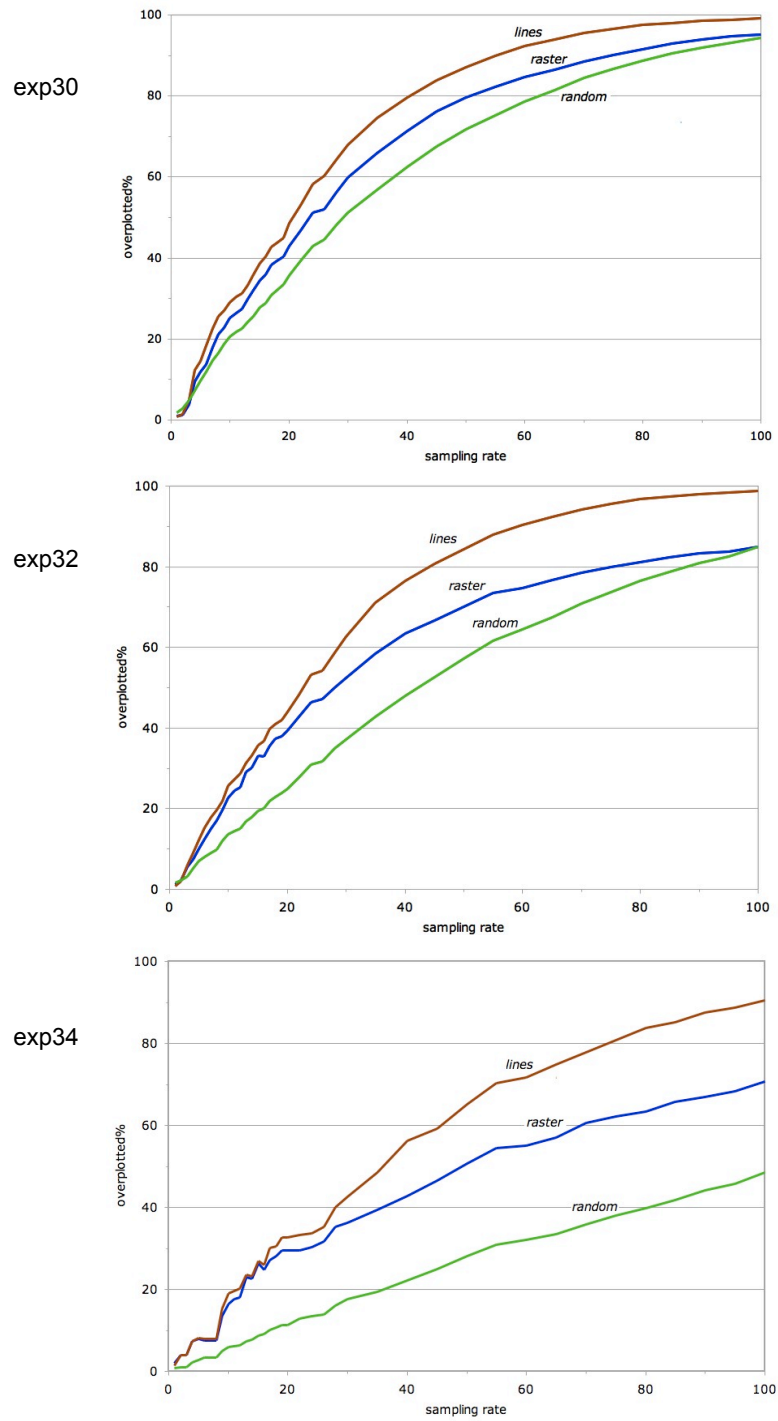
So far we have seen how the *random* algorithm is surprisingly accurate and also very efficient to calculate. However, it is based on a very rudimentary model of the plotted points, so we would expect to find cases where it breaks down. One example is in exp2 near the axis where many lines converge that gives an underestimation of the overplotted% occlusion measure by the *random* algorithm. *Raster* also seems to have problems in exp2, exp3 and exp7 all of which have, to some degree, non-uniform densities across the lens area.

---

<sup>3</sup> A limit on the number of lines used in the calculation was incorporated into the Sampling Lens software, to avoid the lens freezing when crossing very dense regions of a plot.



**Figure 5-25** Lens patterns (at 10% sampling rate) used to investigate non-uniform density across the lens – left to right, exp30, exp31, exp32, exp33 and exp34. Note that these are the same lens positions used for exp35 to exp39.



**Figure 5-26**

*Lines, raster and random overplotted% values at different sampling rates for lens positions exp30, exp32 and exp34, showing increasing divergence with increasing empty space within the lens*

The main simplification of the *random* algorithm is to assume that the points are randomly and *uniformly* scattered over the available area. The fact that the points are on lines of course means that the points are not randomly scattered; but when using real data, this lack of randomness is clearly not an overriding problem. However, the lines themselves do not always lie uniformly over the lens area. The *raster* algorithm on the other hand does not assume a uniform scattering of points across the lens but in using  $S_1$ , the number of pixels with one plotted point will be unduly influenced by even small regions of low density (single plotted points), resulting in an underestimate of occlusion.

The next section details an investigation into the effect of non-uniform density through a series of experiments that systematically change the degree of uniformity. Section 5.6.2 proposes a solution and discusses the results of its implementation in the Sampling Lens visualisation.

### 5.6.1. Identifying the problem

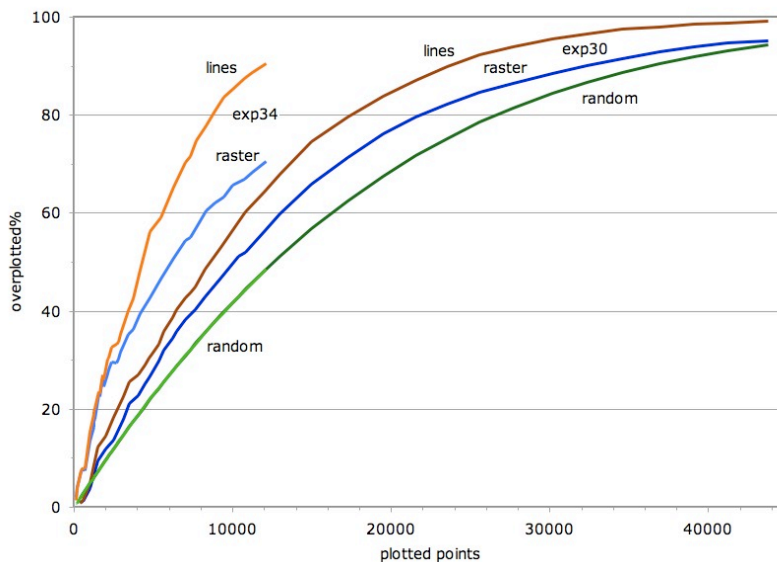
As noted, problems occur in areas where there is a marked difference in density across the lens. Figure 5-25 shows a series of lens positions at just such a boundary, taken from exp30 to exp34. These lens positions move from being in an area of fairly uniform line coverage (exp30) to one where less than a quarter of the area is covered (exp34).

Figure 5-26 shows plots at three of these positions (exp30, exp32 and exp34). Each figure has overplotted% curves for the *lines*, *raster* and *random* algorithm at different sampling rates. In relation to the gold standard *raster* calculation, the *lines* algorithm overestimates overplotted% and the *random* algorithm slightly underestimates this occlusion measure. However, as the coverage of the lens becomes less, there is a noticeable widening and by exp34 the *random* measure is underestimating by nearly 50%. Whilst in most areas the *random* algorithm is surprisingly good, in this extreme case it is no longer accurate. Also, in these situations, it may be unwise to use *raster* as the standard.

There are a number of effects at the edge of a dense area. For example, the lines tend to be lying in the same direction, hence less likely to cross, but when they do cross the overlap is greater due to the shallow angle. Also, by definition such areas are at the edges as far as the data set is concerned and may have unusual properties. Note that the lens is not over an attribute axis so many lines meeting at a point do not affect the lines calculation.

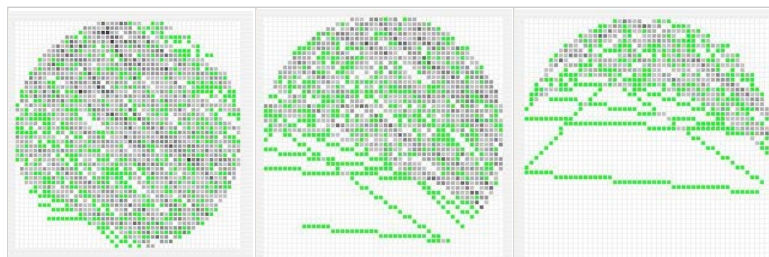
Whilst more of the lens becomes empty in moving from exp30 to exp34, there are less lines within the lens and it is not surprising that the curves in Figure 5-26 become less steep. To compensate for this, the results from exp30 and exp34 (the two extremes for

Figure 5-27



*Lines, raster and random* overplotted% values for lens positions exp30 and exp34 plotted against number of plotted points. The random curves are coincident as the calculation is dependent on the number of plotted points and pixels within the lens.

Figure 5-28



Overlap density maps for exp30, exp32 and exp34 at a sampling rate of 20%. Green squares represent pixels with only 1 plotted point. A grey scale represents 2 to 9 plotted points, with black representing 10 or more plotted points.

this particular test) have been plotted against the number of plotted points (Figure 5-27). As expected, the random curves coincide as this measure is solely based on the number of plotted points and the lens area, which remains constant. The *lines* and *raster* values for the partly covered lens show the same divergence as noted before as well as higher estimates of occlusion. Again, this is not unexpected as there are a greater proportion of pixels with only one plotted point (illustrated in Figure 5-28 and discussed below).

As part of the investigation, a density visualiser was developed for the Sampling Lens that displays, in a floating window, a representation of the number of plotted points in each raster cell. At cell widths of 1 this corresponds to screen pixels. These *overlap density maps* were a great help in understanding the behaviour of the different algorithms. Figure 5-28 shows the corresponding *maps* for the left, centre and right lens positions shown in Figure 5-25 and displays the distribution of pixels with only one plotted point (represented in green) that play a significant part in the overplotted% metric (i.e.  $100 * S_n / (S_1 + S_n)$ ).

### 5.6.2. The solution - using multiple bins

Because the goal is to automatically reduce overcrowding in dense regions within the lens, a measure of the occlusion of lines in the denser regions is required and hence it seemed fairly obvious that the *random* and *raster* methods could be improved by splitting the lens area into a number of smaller areas or bins. The occlusion could be measured for each bin and then a weighted average calculated using the number of plotted points per bin. Code was written to perform binning (GridBin class) and to calculate the weighted averages using both *random* and *raster* algorithms. The bin width and lens width were supplied as parameters. Figure 5-29 illustrates the bins for a lens diameter of 100 and bin width of 30. Note that the bins start from the centre of the lens to give more even coverage, although the size of part bins is known and the weighting function takes this into account anyway.

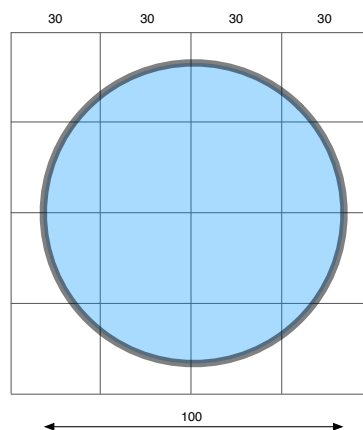
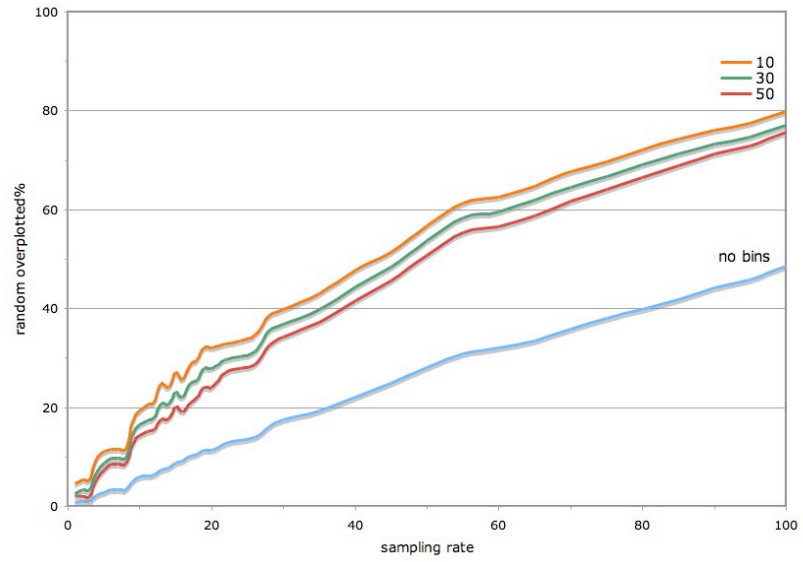


Figure 5-29

Example of dividing a 100 pixel wide lens area into bins 30 pixels wide. The size of each bin is calculated using the lens mask.

Figure 5-30



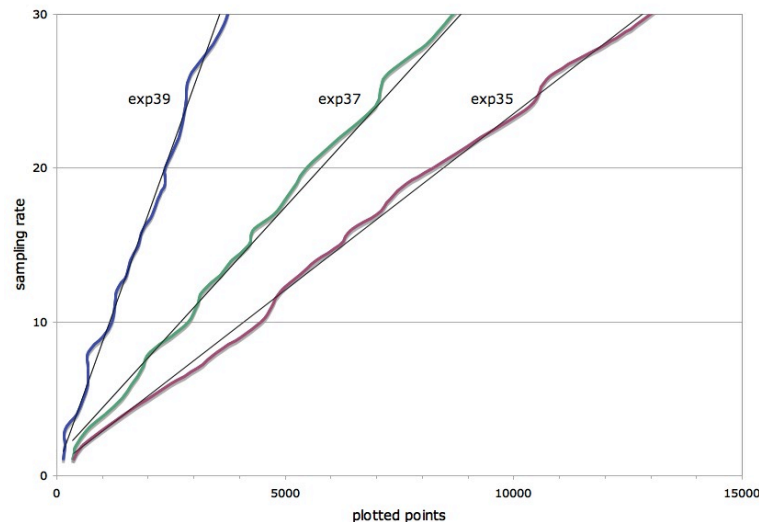
The positive effect of binning on random overplotted% in correcting for a partly covered lens (exp39). Bin widths of 50, 30 and 10 correspond to dividing the lens into 4, 16 and 100 bins.



A series of experiments were conducted (exp35 to exp39) that replicated the conditions of the previous set (exp30 to exp34). Data was collected for bin width of 50, 40, 30, 20 and 10 with corresponding number of bins from 4 up to 100. Figure 5-30 demonstrates the success of binning on the *random* calculation where the lens is only partly covered as in exp39 (same lens position as exp34 in the previous tests). Dividing the lens area into quarters (bin width of 50) has a very marked effect on the calculated value with subsequent further divisions having much smaller cumulative effect.

The unevenness of the curves, especially at low sampling rates is due to the sensitivity of the *random* algorithm to the change in the number of lines crossing the lens. Recall that the lens sampling algorithm takes a random sample of the currently displayed data items (all the data if the overall sampling rate is set to 100%) and not a sample of the lines crossing the lens. Hence it is likely that as the lens sampling rate is reduced, the actual number of lines is not exactly proportional to the sampling rate. In addition, the lengths of the lines (and hence number of points in the line) will not be the same that will cause some unevenness in the point count. This last effect should be even more pronounced in partly covered lenses, such as exp34 & 39, where lines cut across near to the edge of the lens. To test this, the number of plotted points for exp35, exp37 and exp39 were plotted against the sampling rate and the results can be seen in Figure 5-31. Straight lines have been fitted to each curve in order to emphasise the error. The results suggest that partly covered lens (as in exp39) are not liable to more errors contrary to an earlier prediction.

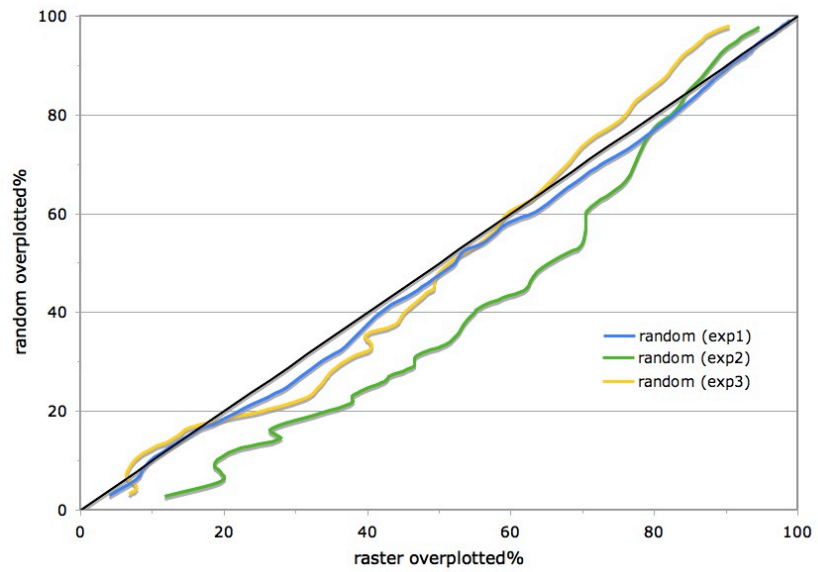
**Figure 5-31**



As expected, the number of plotted points is not exactly proportional to the sampling rate.

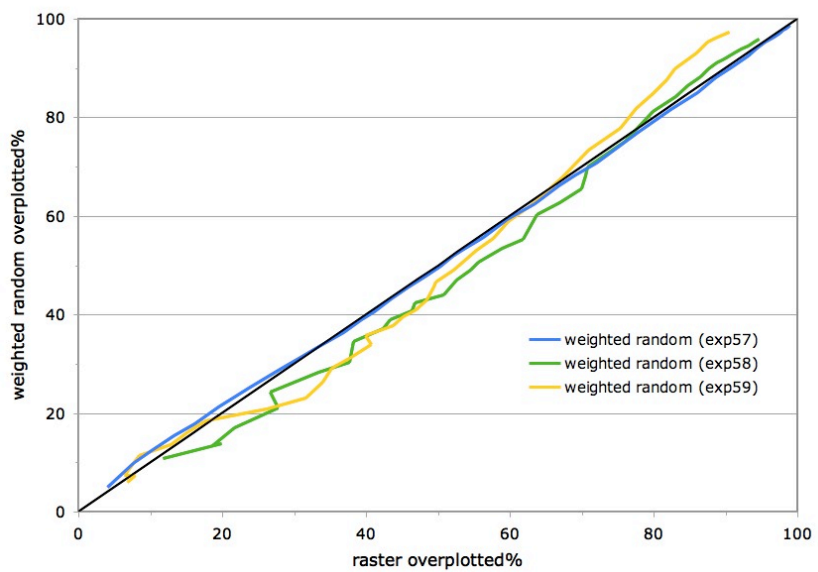
To further illustrate the effect of the bin-based correction, Figure 5-32 shows the same data as plotted in Figure 5-30 but this time the calculated *random* values have been normalised against the true *raster* calculation for different sampling rates. We can see that whilst the original *random* algorithm under-approximates the true value, the

Figure 5-33a



The advantage of using binning for lenses with non-uniform density such as in exp2. *Random overplotted%* values normalised against *raster overplotted%* for exp1, 2 and 3.

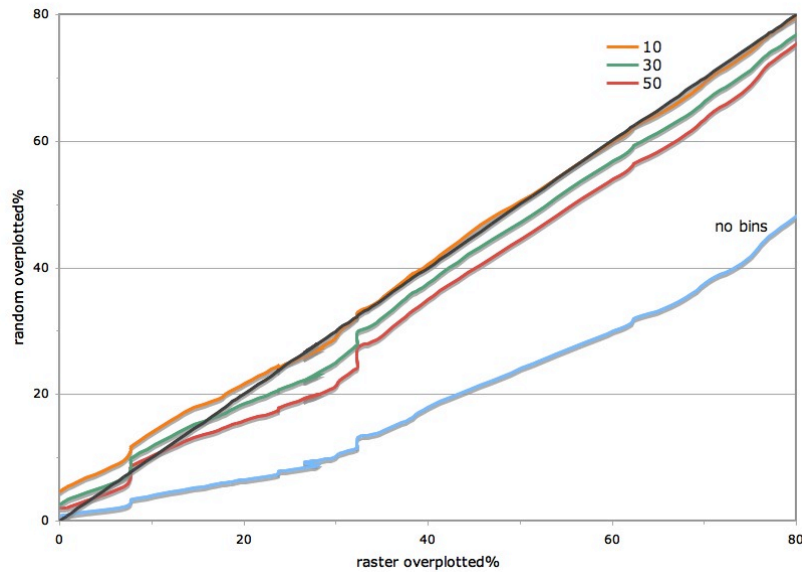
Figure 5-33b



The advantage of using binning for lens with non-uniform density. Weighted *random overplotted%* values normalised against *raster overplotted%* for exp57, 58 and 59 (bin width=10)

weighted value estimates lie progressively closer to the perfect 45-degree line as the number of bins increase.

**Figure 5-32**



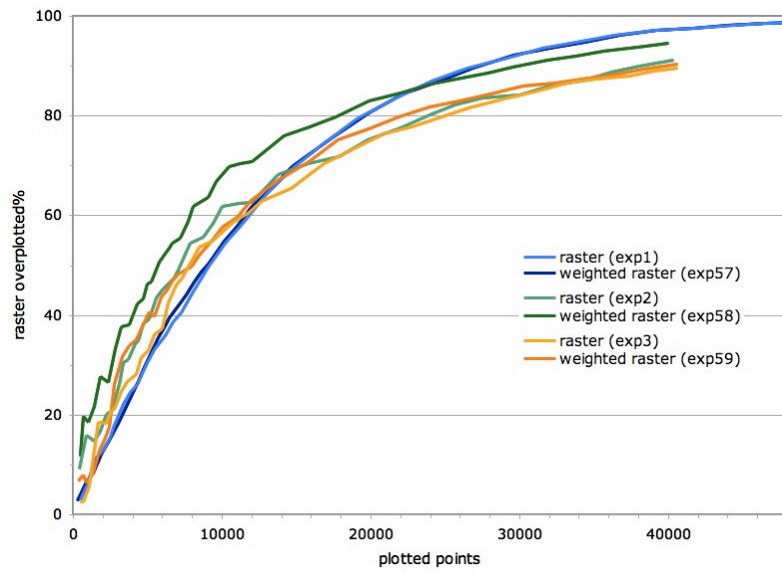
*Random* occlusion measure normalised against the *raster* standard for a partly covered lens (exp39) showing the closeness of the *random* values to the standard with smaller bin widths (i.e. more bins).

Experiments were also conducted on synthetic datasets. The dataset with a wide variation in density confirmed the necessity of binning and even the dataset with a near constant density, shows that binning is desirable. Details of these experiments are give in Appendix C.5.

We have seen that binning is effective in correcting the *random* calculation in the special case of non-uniform density across the lens, but is the calculation efficient enough to warrant its use? The weighted calculation time is proportional to the number of bins (time  $\propto 1/(\text{lensdiameter}/\text{binwidth})^2$ ). However, given that the *random* calculations are very fast (the total line length within each bin is all that is required), the time is very small indeed and even with 100 bins, the time is still too small to measure against the other calculation methods (shown in Figure 5-24).

To conclude this section on dealing with non-uniform densities across the lens, we return to the experimental data that begun this particular part of the investigation. As discussed in Section 5.5.1, the *random* algorithm underestimates the density when the lens is partly covered, as in exp2. A series of further experiments (exp57, 58 & 59) were carried out with exactly the same conditions as the original exp1, 2 & 3 but with the lens binning function in operation. The data has been normalised as before against the standard *raster* overplotted% (in this case the weighted *raster* values) and the results are shown in Figure 5-33b. (The original results for exp 1, 2 and 3 are shown in Figure 5-33a). Clearly the extra effort to calculate weighted *random* values is worthwhile as it effectively deals with non-uniform densities across the lens.

Figure 5-34



The effect of binning on the calculated *raster* overplotted% values. The most noticeable difference occurs with a non-uniform density across the lens (exp2/exp58)

Considering the *raster* values, it was anticipated at the end of section 5.5.1 that binning might also correct these calculations. Plotting the *raster* data for all the experiments above (Figure 5-34), we can see that the weighted values are noticeably different for the exp2/exp58 lens position and to a lesser extent for exp3/exp59 where the non-uniformity of density is less.

## 5.7. Summary and reflection

Through the creation of an occlusion metric and a thorough empirical study of the accuracy and efficiency of three very algorithms for calculating the metric, we now have a reliable way of estimating the density of a region of overlapping lines. This is vital for the implementation of an auto-sampling lens for parallel coordinates and hence help the user discover meaningful information within cluttered plots.

We have come quite a long way since the initial idea to provide auto-sampling for parallel coordinates. Three occlusion measures were proposed and using an instrumented version of the Sampling Lens application, an empirical study discovered good agreement between the measures but also problem cases. A simple theoretical model, based on a random distribution of plotted points over the lens area demonstrates a close agreement with the empirical data.

Having chosen the *overplotted%* measure, three methods of calculating this were devised, very different in their approach and theoretical underpinning: the *raster* algorithm being a direct gold standard measure, the *random* algorithm being based on a purely binomial distribution and the *lines* algorithm being somewhere in between the two. All three algorithms yield comparable results except in extreme cases: many points meeting at a single place on an attribute axis and exaggerated non-uniform density within the lens. The efficiency of each method was measured and ways of speeding up the *raster* and *lines* calculations were found. The *random* calculation is almost instantaneous. Finally the errant behaviour of the *raster* and *random* algorithms under non-uniform density was fully investigated and an effective solution, based on dividing the lens into multiple bins, was successfully demonstrated.

So the best method to calculate a measure of occlusion for use in parallel coordinates is the weighted *random* algorithm. It is easy to implement, surprisingly accurate over a wide range of line patterns and extremely fast to calculate. This finding is unexpected, as the model behind the algorithm is based on treating the lines as a series of points and scattering these in a random fashion across the lens.



## Chapter 6

# Evaluation of sampling

Throughout this work, the word clutter has appeared numerous times. But what is clutter? In the Introduction chapter, visual clutter was defined as an overwhelming number of items making it hard to comprehend the data. Then at the beginning of Chapter 5, when looking at ways to measure clutter, a number of metrics were mentioned such as Tufte's data ink ratio and Brath's data density and occlusion percentage.

Before going on to reflect on sampling as a clutter reduction technique, we should perhaps step back and review some other definitions of clutter and check that we have a consensus. Starting with dictionary definitions, clutter is referred to as a confused or disordered state or collection, a jumble and a confused multitude of things. Note that few of these definitions of clutter mention the number of objects, apart from multitude, which is itself rather imprecise. Instead, the focus is on the organisation (or more probably disorganisation) of the objects.

Information visualisation does provide some context for the organisation or placement of data items - for example, the position of a point on a map represents its latitude and longitude. From the information visualisation literature there are certainly definitions that refer to disorganisation, for example disordered visual entities [Peng et al. 04], a state in which the number of objects cause confusion [Rosenholtz et al. 05], however, the emphasis is on the perception of structure. Some examples of this are clutter obscures the structure of the visual display and hinders visual analysis [Cui et al. 06], clutter *makes it difficult to perceive patterns in subsets of objects* [Chuah et al. 95] and clutter corresponds to all the factors that interfere with the process of finding structures [Peng 05].

So, one might say that to gain insight we need to perceive patterns<sup>1</sup> in an understandable organisation of data. Anything that inhibits seeing patterns, such as an overcrowded display should be avoided, and just as important, data should not be hidden, either by occlusion or removal by some means, which the user is unaware of, as it distorts a user's perception of the data space [Chuah et al. 95].

This work is all about the use of random sampling to help users perceive these important structures in large datasets. This chapter presents an analytical evaluation

---

<sup>1</sup> In certain circumstances, the lack of correlation manifested by the absence of trend lines or clusters will be the *pattern* that provides the insight.





of sampling using the Clutter-reduction Taxonomy in Chapter 3, and a practical examination of global and lens-based sampling of a large map-based dataset using a scatterplot version of the Sampling Lens application.

The Sampling Lens was consolidated through Chapters 4 and 5 and its development process will be revisited. We will discuss how the functionality embodied in the Sampling Lens application has informed the use of sampling for clutter reduction and postulate the incorporation of sampling into other types of visualisations. This research was inspired by the Astral Visualiser proposal as described in Chapter 2 and we will see its implementation here and how well it works in practice.

Section 6.1 discusses the problems of user evaluation of information visualisations and suggests feasible alternatives. The objectivity of a criteria-based evaluation of sampling is discussed. Section 6.2 compares sampling with other clutter reduction techniques using the Clutter-reduction Taxonomy in Chapter 3. An example where quality measures have been used to compare sampling to clustering is also given.

Section 6.3 explores the use of global and lens-based sampling on a Sampling Lens scatterplot, which maps the household income across the USA. Section 6.4 reflects on the use of the Sampling Lens and discusses its functionality for clutter reduction. With the aid of examples, Section 6.5 discusses whether dynamic sampling can be added to other visualisations to reduce clutter. Finally, Section 6.6 revisits the Astral Visualiser in its manifestation as a zooming, constant-density feature of the scatterplot Sampling Lens visualisation and finds that automatic clutter reduction works well in practice.

## **6.1. User evaluation issues**

At the beginning of Chapter 3, it was noted that effective evaluation of information visualisation was fraught with problems. Consequently, one of the reasons for developing a taxonomy of clutter reduction was to enable a comparison to be made between sampling and other techniques. As we have seen in the discussion so far, the taxonomy has proved to be very useful, not just as an opportunity to look at sampling in a wider context, but to look critically at visualisations and even suggest new visualisations.

This section reviews current thinking on the evaluation of information visualisation and offers alternatives to quantitative user studies. We will then reflect on the decision not to undertake a user evaluation of the Sampling Lens.

### **6.1.1. User evaluation of information visualisations is problematic**

Formal quantitative studies are often used in related areas such as HCI and visual cognition where it is often possible to specify low-level tasks and measure quantities such as time to complete a task, interaction details and accuracy. Whilst a similar



approach is taken to evaluate visualisations, the usefulness of the results is often questionable. The major reason reported by researchers at the BELIV'06 workshop [BELIV 06], is that information visualisations are often used, quite rightly, for exploratory rather than specific tasks. Even if we would like to instruct the participant to “find something interesting in their dataset with this tool”, it is difficult to identify the criteria with which to measure success [Stasko 06]. Deliberately choosing simple tasks and/or simple datasets is not going to tell us much about the visualisation's potential with realistic tasks and real datasets, especially as low-level tasks do not scale up to complex displays [Kosara et al. 03]. We need to be aware that offering the user too much freedom to explore the system can confuse participants and can result in unanticipated behaviour [Henry and Fekete 06]. Plaisant, in *The Challenge of Information Visualisation Evaluation* [Plaisant 04], goes further, noting that discoveries occur very rarely, but are often very important and that the active intellectual engagement required for such discoveries is particularly difficult to instigate and control.

Visualisations are often developed for experts with corresponding requisite domain knowledge, yet many studies use students. Whilst convenient, they usually do not have a deep enough understanding of the problem that the visualisation tool is attempting to solve and the necessary understanding of the data [Andrews 06]. In such cases, the chances of assessing the usefulness of the tool will be slim, as found during informal testing of the Sampling Lens [Ellis et al. 05]. Users liked the lens-based tool as it revealed patterns within an overcrowded parallel coordinate plot. However, when asked what the patterns meant, they did not understand; they just thought it was cool.

As suggested by several researchers [e.g. Kosara et al. 03, Ellis and Dix 06a], comments from participants and close observation are often more important than quantitative data, since they provide valuable insight into what is happening. Instead of simple tasks with generally novice users (30 minutes training is unlikely to turn a novice into an expert), researchers [Kosara et al. 03, Tory and Möller 05] have suggested that better information can be obtained by using either a small number of domain experts involved in more qualitative studies, expert visual designers or HCI expert reviewers<sup>2</sup>. Of course, it is more difficult to get access to such a group of people but it is less likely that such participants will become bored or overwhelmed [Henry and Fekete 06], unduly influenced by novelty or biased by the familiarity of traditional interfaces [Andrews 06]. Also, for a novice, the usability of the system can strongly affect its perceived utility [Kobsa 01].

---

<sup>2</sup> Zuk et al. [Zuk et al. 06] even suggest an Extreme programming approach with domain and evaluation experts working in pairs.



It is relatively easy to choose a task that makes a visualisation tool look better and think this is definitive – indeed, it is natural to choose tasks (and datasets) that suit a novel technique i.e. ones that it is good at. Furthermore, as a researcher, there is a temptation to deliberately choose the tasks that make one’s method look good.

On the question of formal quantitative studies, Kosara et al. [Kosara et al. 03] found that not only are studies very time consuming to undertake and analyse, they are terribly difficult to design with sufficient experimental rigour. Due to this they abandoned the idea of a formal study all together in one particular project and obtained a wealth of useful information from observing two domain experts using their visualisation.

An alternative method suggested by Plaisant [Plaisant 04] is in-depth, ethnographic case studies, where users in their workplace setting undertake real tasks using the new visualisation, have the potential to convince the manager that information visualisation is a worthwhile investment. Although, she notes that the results may not be generalised to other domains, she also promotes longitudinal studies. She acknowledges that the latter are more difficult to conduct and in a later paper with Shneiderman [Shneiderman and Plaisant 06], they propose a new paradigm for evaluations, Multi-dimensional In-depth Long-term Case studies (MILC). Again, ethnographic observation methods are used, plus engagement with participants through interviews, surveys and automatic logging of users activity. The downside of this approach is that such case studies have to be conducted over several months in order to lead to refinement and a better understanding of the visualisation tool.

Zuk et al. [Zuk et al. 06] review the use of heuristics in information visualisation evaluation and propose an *optimal list* comprising of perception, usability and discovery heuristics<sup>3</sup>. A case study found that the range of heuristics were useful but required the participation of experts from visualisation and usability to be of full benefit.

Amar and Stasko’s [Amar and Stasko 04] higher-level holistic evaluation of entire systems with the focus on closing the *analytic gaps*<sup>4</sup> (obstacles to decision making and learning) also require the participation of domain experts.

---

<sup>3</sup> Perception: perceptual and cognitive heuristics from Bertin, Tufte and Ware [Bertin 83, Tufte 90, Ware 04]; usability: Shneiderman’s Visual Information Seeking Mantra [Shneiderman 96]; discovery: Amar and Stasko’s Knowledge and Task-based Framework [Amar and Stasko 04].

<sup>4</sup> The two categories of *analytic gaps* are i) Rational Gap – gap between perceiving a relationship and expressing confidence in the correctness and utility of that relationship and ii) Worldview Gap – gap between what is being shown and what actually needs to be shown to draw a representational conclusion for making a decision. [Amar and Stasko 04].



### 6.1.2. Possibility for evaluating the Sampling Lens with users

Faced with the problems of information visualisation evaluation discussed in the previous section plus those identified through the survey of user studies [Ellis and Dix 06a and Appendix F], evaluating the Sampling Lens with users with the aim of obtaining some meaningful insights was going to be difficult.

Colleagues and students could have been persuaded to participate in some low-level task-based experiments, but it was felt that the effort would give little evidence on the value of sampling. Other schools at Lancaster University and the business unit (KBC) were approached but were unable to find suitable datasets or tasks. In fact, finding large, multivariate datasets was a surprising problem during this work and the lack of suitable data/benchmark repositories was a subject discussed at BELIV'06. For example, to demonstrate the first version of the Sampling Lens at CHI'05, the Portland Cars dataset [Appendix B.1] was created by downloading and processing the results of numerous price range searches for cars for sale in Oregon, by way of an online advertising site.

Undertaking case studies was not appropriate given that the sampling-based visualisation was only a prototype and lacked the functionality required by an analyst.

The use of sampling for clutter reduction was certainly novel when this research started, thus there was no real understanding of sampling-based techniques for clutter reduction. The Sampling Lens appears to do an excellent job in reducing clutter with a variety of datasets as demonstrated by many examples throughout this work. However, do we actually understand the mechanism by which sampling reduces clutter?

At a simple level we do. The number of data items is reduced and this in turn reduces the degree of overplotting that can lead to an uncluttered view of trends or patterns within the data. But is sampling better than say filtering or clustering, which also reduce the number of data items? To answer such questions we can either undertake user studies or continue down an analytical route.

In this work an analytical approach was adopted by devising a taxonomy, which because of its criteria-based underpinnings, enable us to start a comparison of clutter reduction techniques on an even footing. As discussed in Chapter 3, this is useful for visualisation designers and it may also, in time, shed some light on possible mechanisms for a range of visualisation techniques.

An alternative way forward is to undertake micro user studies, which by their nature are controlled and designed to answer specific questions rather than the high level quest of user studies for investigating "why is technique X better than technique Y". For example, the Reality Check transitions described in Section 4.4.4 would be

Criterion	Strength	Weakness	Objectivity
avoid overlap	<ul style="list-style-type: none"> <li>▶ can make hidden data items visible</li> <li>▶ an acceptable amount of overlap can be tolerated</li> </ul>	<ul style="list-style-type: none"> <li>▶ cannot avoid overlap altogether as sampling randomly removes data items</li> <li>▶ amount of overlap is difficult to measure</li> </ul>	medium
keeps spatial information	<ul style="list-style-type: none"> <li>▶ position of data items are not moved</li> </ul>	<ul style="list-style-type: none"> <li>▶ none</li> <li>▶ note: assuming that the removal of a data item is not considered to be a loss of spatial information</li> </ul>	high
can be localised	<ul style="list-style-type: none"> <li>▶ can be restricted to a particular region with a lens</li> </ul>	<ul style="list-style-type: none"> <li>▶ none</li> </ul>	high
is scalable	<ul style="list-style-type: none"> <li>▶ can cope with very large datasets</li> </ul>	<ul style="list-style-type: none"> <li>▶ limited by computational/hardware resources</li> </ul>	high
is adjustable	<ul style="list-style-type: none"> <li>▶ highly adjustable (from full dataset to single data item)</li> </ul>	<ul style="list-style-type: none"> <li>▶ requires a mechanism to give required control sensitivity</li> </ul>	high
can show point/line attribute	<ul style="list-style-type: none"> <li>▶ enables mapping of one or more attributes as the characteristics of data item are unchanged</li> </ul>	<ul style="list-style-type: none"> <li>▶ none</li> </ul>	high
can discriminate points/lines	<ul style="list-style-type: none"> <li>▶ may indirectly discriminate data items as a result of reducing clutter</li> </ul>	<ul style="list-style-type: none"> <li>▶ cannot help to see individual data items</li> </ul>	medium
can see overlap density	<ul style="list-style-type: none"> <li>▶ possibility of estimating overlap indirectly by viewing the sampling rate</li> </ul>	<ul style="list-style-type: none"> <li>▶ no indication of overlap density</li> </ul>	medium

**Table 6-1** The strengths and weaknesses of a sampling approach to clutter reduction in relation to the criteria used in the Clutter-reduction Taxonomy and the objectivity in assessing each criterion.



candidates for perception experiments. This could be undertaken by showing users a series of pre-prepared lens sample transitions, either instant, fade or indeed twinkle, and we could then note which transition type favours the recognition of patterns within the dataset. A synthetic dataset made be used as it offers greater control on the embedded patterns. A similar type of experiment could be attempted to assess the efficacy of combining clustering with other techniques, as suggested by the Clutter-reduction Taxonomy (see Section 3.5.2).

The advantages of these types of micro user studies are several fold. The users do not need any domain knowledge or experience of visualisation, they are low cost in terms of time and effort and answer specific, often quantifiable questions. The knowledge gained can be fed back to visualisation designers, using the taxonomy's grid structure, and gradually our understanding of clutter reduction techniques will evolve.

### **6.1.3. Objectivity of criteria-based evaluation**

As discussed in Section 6.1.1, it is fairly easy to select a combination of users, data and tasks to make the result of a user study a forgone conclusion. Does this lack of objectivity apply to criteria-based evaluation?

Table 6-1 lists the strengths and weaknesses of the sampling clutter reduction techniques against the criteria used in the Clutter-reduction Taxonomy and gives an assessment of objectivity for each criterion. A high objectivity means that the criterion can be judged or measured without being influenced by personal feelings or opinions. As we can see, five out of the eight criteria are rated high as there is little or no doubt in determining whether or not these criteria have been met. Note that it has been assumed that removing a data item from display is not counted as a loss of spatial information. The remaining three criteria have been rated as medium objectivity as the assessment of sampling against these criteria is not so clear cut. For instance, sampling does not avoid overlap, however, it does reduce overlap. Measuring the overlap of co-incident points or lines is straightforward but partial overlap is more difficult to determine. Likewise for the lines in parallel coordinates, as seen in Chapter 5. Sampling does not directly help the user to see either the overlap density or individual points but reducing the number of points does reduce clutter, making points distinct. The sampling rate will also indirectly show the data density - a low sampling rate to reduce clutter indicates a high original density.

We have seen that the objectivity of sampling has been assessed as generally high, however, we will now look at examples where techniques are far less objective for some criteria. Reducing point size and opacity can both avoid overlap to some extent but colour blending can occur, especially with small points, and the associated subjectivity of the user would suggest a low objectivity. For displacement techniques,

		1	2	3	4	5	6	7	8	9	10	11
		sampling	filtering	point size	opacity	clustering	point/line displacement	topological distortion	space-filling	pixel-plotting	dimensional reordering	animation
A	avoids overlap	possibly	possibly	possibly	partly	possibly	✓+	possibly	✓+	✓+	partly	✓+
B	keeps spatial information	✓	✓	✓	✓	partly	✗+	possibly	✓+	possibly	✓	✓
C	can be localised	✓	✓	✓	✓	✗+	✓	✓	✗	✗+	✗	✓+
D	is scalable	✓	✓	✗	✗+	✓	✗	✗	✗	✗	✗	✓+
E	is adjustable	✓	✓	✓	✓	✓	possibly	✓	✗+	✗+	✓	✓+
F	can show point/line attribute	✓	✓	✓	✗+	partly	✓	✓	✓	✓	✓	✓
G	can discriminate points/lines	✗	✗	possibly	✓+	✓+	possibly	possibly	✗	✗	✗	✗
H	can see overlap density	✗	✗	✗	✓+	possibly	✗	✗+	✗+	✗+	✗	✗+

**Table 6-2** Clutter-reduction Taxonomy (reproduced from Section 3.4)

the relative position of the points may be more important than the absolute position, especially when the user is searching for patterns, and hence the objectivity is reduced for the *keeps spatial information* criteria. The degree to which a technique can discriminate points or lines is also subject to users judgement and hence displacement and topological distortion techniques will have low objectivity. The next section gives a more detailed discussion of other clutter reduction techniques when comparing them to sampling. Note that this assessment (as shown in Table 6-1) is based on scatterplots and parallel coordinates.

## 6.2. Comparing sampling to other clutter reduction techniques

In Section 3.5, the Clutter-reduction Taxonomy was compared with Ward's taxonomy of glyph placement strategies [Ward 02] and Bertini's clutter reduction strategies [Bertini 07], the two major published classifications dealing with clutter reduction. The discussion showed that the Clutter-reduction Taxonomy could help visualisation designers think about the effectiveness of existing or new applications for clutter reduction. In addition the Fisheye Menu example (Section 3.5.5) demonstrated the usefulness of the taxonomy in helping to understanding the mechanism behind an interactive visualisation. This section will focus on the sampling technique by using the taxonomy (reproduced here in Table 6-2) to make a comparison with other techniques.

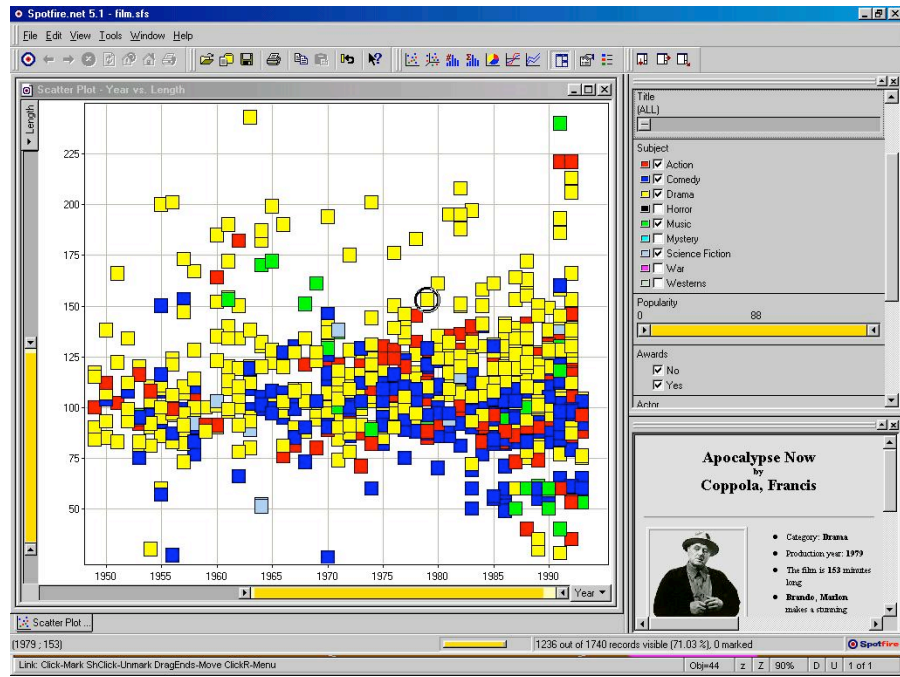
From Table 6-2, we can see that sampling meets many of the criteria. However, this taxonomy should not be used simply as a means of rating techniques based on the number of ticks. Instead, we will discuss the strengths and weaknesses of sampling in de-cluttering overcrowded visualisations by considering each criterion in conjunction with the discussion that accompanies the taxonomy.

### 6.2.1. Criteria based evaluation of sampling

#### **avoid overlap**

In Chapters 4 and 5, there were many examples where sampling has been used to reduce the number of data items that, in turn, reduced the number of overplotted points or lines. As sampling removes points at random, there is no user control over which points are selected. Contrast this with clustering that by design can be selective, although the algorithm parameters often need to be adjusted by the user to achieve the optimal effect. As noted in Chapter 3, clustering replaces a group of data items by a single representative point or line and hence detail is lost due to this reduction in granularity, even though the cluster point will often have attributes such as colour, shape or size to represent the group's features. Sampling can however, be partially selective, either spatially, as in non-uniform sampling or statistically, as in stratified

Figure 6-1



FilmFinder application in Spotfire [Ahlberg 96]

sampling (see Section 2.4) so the output can be shaped algorithmically whilst retaining a desirable degree of randomness.

Sampling is similar to filtering (columns 1 and 2 of the taxonomy table have the same number of ticks and crosses) but as addressed in Section 4.2.5, sampling is better for explorative visualisation as the user does not have to decide what is *uninteresting* and hence can be removed. Although the user has some control over what data is removed, dynamic filtering does not necessarily reduce overplotting. For example, with the often cited FilmFinder [Ahlberg 96] (see Figure 6-1), reducing the release date interval (horizontal axis) would not necessarily avoid overplotting as the axis scale is *stretched*, whilst restricting the film categories (check boxes) or duration of the film (vertical axis) might avoid overplotting.

We also need to consider the difference between partial overlap and coincident points or lines and the effectiveness of the different techniques in this regard. Sampling will not necessarily avoid coincident point overlap, but will tend to negate this problem as the number of points is reduced. Change point size, change opacity and topological distortion do not disambiguate coincident points, apart from the use of reduced opacity to inform the user that points are overlapping. Space-filling and pixel-plotting are of course the best at dealing effectively with potential coincident points as no overlap is permitted. This holds true for those point/line displacement algorithms that also ensure a similar outcome (e.g. Gridfit [Keim and Herrmann 98]). Space-filling and pixel-plotting are spatially dimensionless, however, with displacement techniques applied to spatial data, there is a trade-off between the original (or actual) position and perceived position<sup>5</sup> and this requires careful thought. Again, sampling does not suffer from this, unless one considers that the deletion or suppression of a data item as a total loss of data integrity.

This leads on to RSVP animation (Rapid Serial Visual Presentation, see Appendix A.1.10) that displays one image (or perhaps a few) in rapid succession, thus saving screen space at the expense of the user only getting a fleeting view. RSVP is thus only suitable for specific tasks such as searching or browsing images. However, RSVP does have an advantage over sampling in that no data item is hidden forever, but it might take a while for a particular image to appear and there is still the chance that the user may blink at an inappropriate time.

### **keeps spatial information**

If there is a spatial dimension to the dataset on display, such as a map, scatterplot or timeline then preserving this data or at least preserving the perception of this data, is important. Sampling, filtering, changes to point size and opacity do not alter the

---

<sup>5</sup> Ward refers to this as *maintenance of data integrity* and is discussed in Section 3.1.1.



location of the data item, so meet this criterion. As before, it could be argued that removing a point (as with sampling) does not preserve its location, however it would be surprising to find a user of GoogleMaps who thought that all the small roads had literally disappeared when zooming out from a street level view to a country view.

Some techniques such as space-filling, pixel-plotting, dimensional reordering and animation would not be used to plot geo-spatial data, however, they do preserve some spatial information so would generally meet this criterion. All the techniques, including some versions of space-filling preserve an order (if present) e.g alphabetic, numeric or temporal (for animations). Clustering, point/line displacement and topological distortion all shift points where the amount is generally dependent on the degree of overplotting. With topological distortion, some form of landmarks or reference points (e.g. country boundary, regular imposed grid) are necessary to help maintain data integrity. (Further examples in Appendix A.1.6).

### **can be localised**

This criterion questions whether a technique can be applied in the form of a lens to a restricted region of the display. This work has certainly demonstrated that sampling can be localised effectively and Bertini and Santucci [Bertini and Santucci 06] showed that sampling could be applied non-uniformly by essentially dividing the display into many rectangular sampling lenses, each with its own sampling control. Similarly, point size can be changed locally and non-uniformly - Woodruff's constant density [Woodruff et al. 98b] is an example of the latter.

Filtering, opacity, point/line displacement, topological distortion and indeed animation exist or can be envisaged as functions of magic lenses. Recall from Chapter 3 that a fisheye lens (used as a magnifier) is non-uniform topological distortion, whilst zooming is uniform topological distortion. As mentioned in the Section 3.5.5, Fisheye Menus are interesting as they employ both topological distortion and change point size.

### **is scalable**

Scalability, in terms of the ability to handle very large datasets, is a criterion where sampling excels, only matched by filtering and clustering. The author would argue that sampling has the edge over clustering in both ease of implementation and adjustability, as has been demonstrated with the Sampling Lens visualisation. It is hard to imagine a clustering algorithm that would provide the interactive possibility enabled by the z-index method (Section 4.1).

Even though pixel-plotting, such as Keim's spirals [Keim 97] and Fekete and Plaisant's achievement of visualising a million items [Fekete and Plaisant 02] obviously cope with very large datasets, they are limited to the number of pixels on a display screen and





hence are not ultimately scalable (the emergence of Giga-pixel displays does extend their usefulness though). Whilst discussing scalability, it is common practice to use aggregation to reduce large datasets to a manageable size. But as mentioned in Section 3.2, aggregation is not considered as a clutter reduction technique as it is normally a pre-processing operation and does not lend itself to be visually interactive.

### **is adjustable**

As discussed earlier, sampling is highly adjustable and suitable for interactive exploration of large datasets. Many other techniques are also adjustable within imposed limits - such as minimum font size or the atomic pixel. Space-filling algorithms cannot be adjusted to control the clutter apart from folding hierarchical structures (to hide detail), which is more akin to filtering. There are few visualisations that use point/line displacement for clutter reduction that give the user control of the displacement. The original display density normally dictates the amount of displacement. One exception is Waldeck's Mobile 2D scatterplot [Waldeck and Balfanz 04] where the displacement is controlled by the stylus pressure on the PDA screen.

The only adjustable quantity in animation based clutter reduction is the animation rate or speed across the display with a *conveyor belt* type animation. However, it is debatable whether a higher rate reduces or increases clutter. For instance, speeding up the presentation of RSVP images, reduces the time required to see all the images that might be regarded as reducing clutter. However, this increases the human visual and memory processing requirement, which may be thought of as increasing the clutter as it requires more effort.

### **can show point/line attribute**

It is often advantageous to utilise the shape, colour or size of a point to show one or more data attributes, so this criterion indicates whether the clutter reduction technique can accommodate this. As we have seen, sampling does not affect the characteristics of plotted points (apart from removing them.), so meets the criterion. In fact all the techniques apart from change opacity and clustering also meet this criterion. Although as noted in the taxonomy discussion, we need to be aware of the colour perception problems associated with small points and thin lines, especially if closely packed [Tuft 90].

Reducing opacity causes similar perception problems due to reduced contrast and colour mixing when one or more points overlap. Clustering only allows an aggregate or representative of the group's attribute values, hence only partly meets the criterion.

### **can discriminate points/lines**

Sampling does not meet this criterion. One could argue that reducing clutter by reducing the number of data items does help to discriminate the items left on display.



Better techniques exist such as reducing the opacity, displacing lines making it easier to follow their path and clustering to bring outliers to the fore. In addition, recall the Fisheye Menu example (Section 3.5.5), where stretching the background can separate close lines of text and hence make them distinct.

### **can see overlap density**

Sampling does not satisfy this criterion too. A density map can only really be achieved by reducing opacity (overplotted data items are darker) or colouring points based on the calculated degree of overplotting (applies to clusters as well as individual points). Other techniques provide indirect measures of overlap density. For example, Carpendale's pliable surfaces show the degree of distortion required to separate points. It is noteworthy that the sampling rate control of the auto-sampling interface (Figure 5-2, Section 5.2) also indirectly indicates the degree of overplotting of the original dataset.

### **can identify outliers**

One criterion that did not make it to the final list was *can identify outliers*, principally because it is not really clutter reduction. However, it is a question that often crops up when demonstrating the Sampling Lens. Because only a fraction of the dataset is being displayed it is often assumed that, all the outliers, which by definition are few in number, will disappear. It is true that if we have 10,000 data items of which 1% are outliers and display a 10% sample, then the number of outliers will drop from 100 to approximately 10. Using the Reality Check feature (Section 4.1.2) ten times will show all the data items, including all the outliers. With very low sampling rates, such as 1%, one would not expect the user to go through a hundred Reality Checks. However, as random samples are being displayed, a modest number of Reality Checks will give a good indication as to the proportion of outliers, without having to go through all of them, assuming that the outliers are hidden amongst the overplotting, e.g. in a parallel coordinate plot. Later on in Section 6.3.2 we will see examples that demonstrate that a sampling lens is useful in identifying outliers.

## **6.2.2. Comparison between sampling and clustering**

A comparison of sampling and clustering is reported by Cui et al. [Cui et al. 06]. In one experiment, a parallel coordinate plot (1160 records of 14 dimensions) is sampled<sup>6</sup> to give a visually acceptable reduction in clutter (8% sampling rate) and in another plot; the same data is clustered to give the same number of data items. Two quality measures, often used in pattern recognition are calculated. The Histogram Difference Measure, which minimises the difference between the distributions of two datasets, is much better for sampling than clustering and the authors suggest that sampling

---

<sup>6</sup> No details are provided on the type of sampling or its implementation.

- ▶ excels at scalability
  - ▶ is highly adjustable
  - ▶ is suitable for interactive applications
  - ▶ does not affect the characteristics of the plotted data items
  - ▶ does not distort the position of points
  - ▶ is very suitable for exploratory visualisation (do not have to pre-judge the data)
  - ▶ is random yet adaptable (spatially and statistically)
- also indirectly:
- ▶ indicates overlap density via sampling rate control when in auto-sampling mode
  - ▶ identifies outliers when used with the Reality Check

**Table 6-3** A summary of the benefits of sampling for clutter reduction based on the taxonomy.

maintains the relative density of the dataset much better than clustering. This is explained by the fact that clustering displays one representative for each cluster irrespective of the size of the cluster and hence relative density is lost<sup>7</sup>. On the other hand, a Nearest Neighbour Measure, which minimises the distance between datasets, is marginally higher for clustering than sampling and from this it is suggested that clustering maintains the outliers a little better than sampling. This seems plausible and Novotny et al. [Novotny and Hauser 06] have used clustering to detect outliers. In another experiment, scatterplots are presented and the authors conclude that sampling is useful at discovering patterns in cluttered plots and furthermore state that high values of their quality measures can give confidence that patterns found in sampled data are valid. Unfortunately, they do not provide any evidence that low quality measures indicate the contrary. It appears that the quality measures are calculated across the whole plot and hence it would be helpful to localise the measures (e.g. a lens) so that the user could assess the quality and hence validity of particular patterns within a plot.

Cui et al. acknowledge the earlier work of Dix and Ellis [Dix and Ellis 02] on sampling and visualisation and they note that view continuity is important (Section 2.2) and that re-sampling through Reality Check (Section 4.1.3) can help verify a pattern discovered in a previous sample.

### 6.2.3. Advantages and disadvantages of sampling

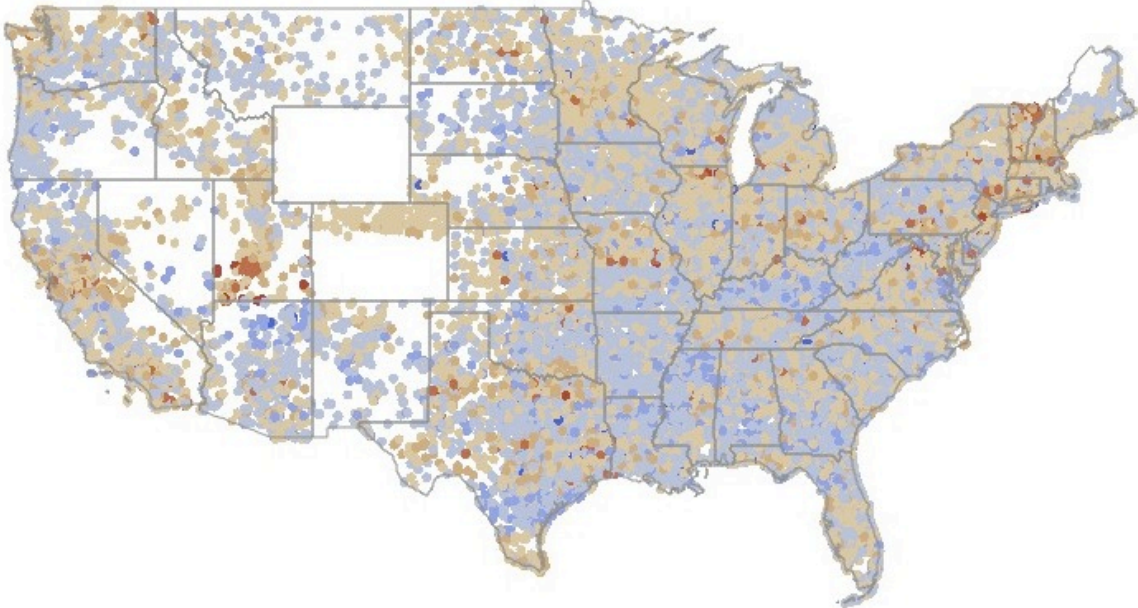
Table 6-3 summarises the benefits of sampling for clutter reduction based on the Clutter-reduction Taxonomy. In presenting this list, we should remember that the selection of the criteria in the taxonomy is based on the experience of many researchers and not biased towards sampling.

By virtue of its randomness, sampling maintains an overview of the data and hence is not designed to answer specific queries such as “where do the highest income households live?”. A filter operation is obviously a better solution, although, as noted in Section 4.2.5, filtering cannot necessarily reduce overplotting to the desired level. For other tasks such as identifying similar items, cluster is better, and related techniques such as hierarchical clustering in parallel coordinates (Appendix A.1.4) are more suitable.

One of the drawbacks of sampling as highlighted by the taxonomy is the fact that sampling does not avoid co-incident points or lines. However, we have seen examples in Chapter 4 and will see further examples in Section 6.3 where reducing the opacity of points can help with this issue.

---

<sup>7</sup> Although not mentioned by the authors, factoring the size of each cluster into the HDM algorithm would conceivably have given a closer agreement with the sampling measure.



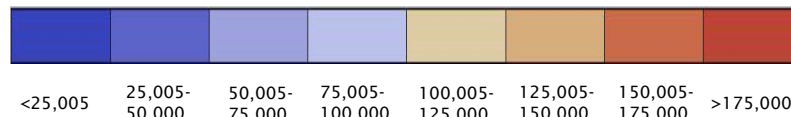
**Figure 6-3** Full 155K dataset showing the distribution of household income across the USA. There is severe overplotting, especially in cities, however, the order of plotting the points is randomised so a representative sample is displayed [USA Census 2000 household income]

Sampling thus appears to show promise. Furthermore, the findings from Cui et al. [Cui et al. 06] suggest that sampling maintains relative density much better than clustering and when backed up by quality measures, sampling is useful in discovering patterns in cluttered plots.

### 6.3. Further exploration of sampling-based scatterplots

In Chapter 4, the Sampling Lens application was introduced with examples of reducing clutter on scatterplots and parallel coordinate plots both with global sampling and with lens-based sampling. Chapter 5 dealt with auto-sampling and due to the particular problem of estimating the occlusion of lines in a parallel coordinates lens, attention shifted away from scatterplots. In Section 4.4, we saw some novel lenses and colouring techniques, again for parallel coordinate plots. In this section, we return to look at the humble scatterplot.

The dataset for the following examples is from the USA 2000 census and consists of just over 155,000 data items representing the median income for blocks of households across the country (see Appendix B.7 for more details). The discussion will initially focus on global sampling, with a comparison being made to opacity and filtering, before moving on to consider the usefulness of lens-based sampling. Note that the examples use the OpenGL version of the Sampling Lens application (details in Appendix D.3), which was enhanced to draw the state boundaries.

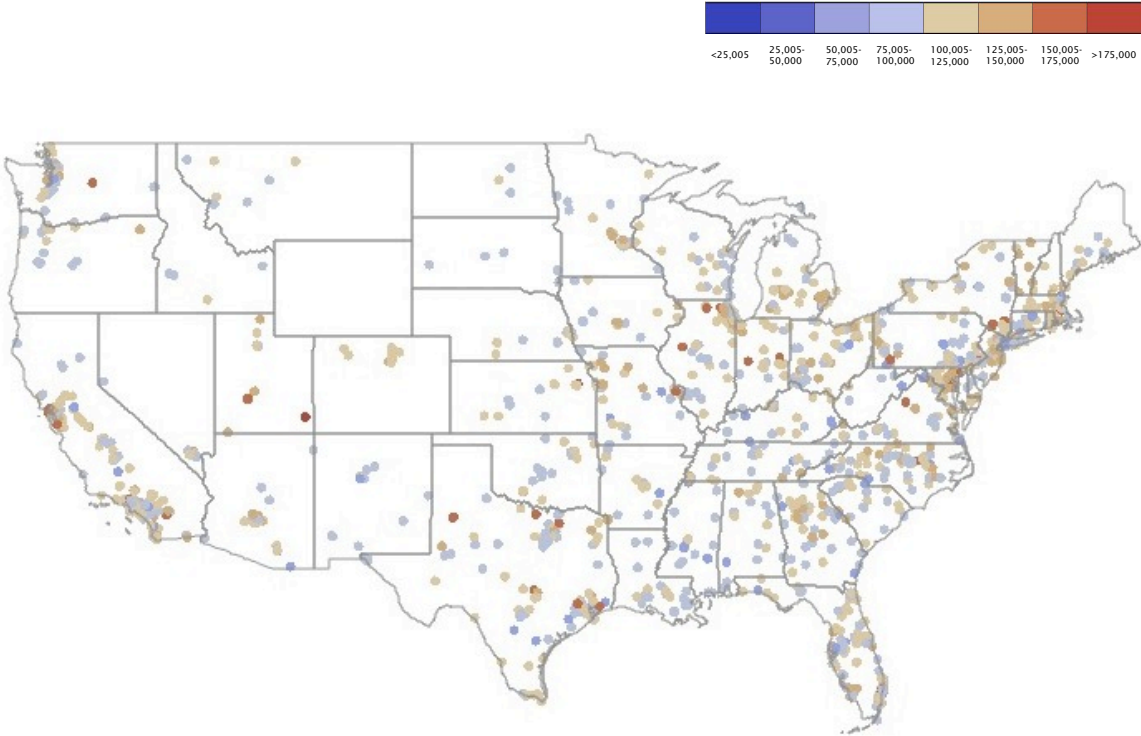


**Figure 6-2** Colour scale for USA household income scatterplots. Income levels in US dollars. [USA Census 2000]

#### 6.3.1. Global sampling

A scatterplot of all the household income dataset is shown in Figure 6-3. With approximately 155,000 points, there is considerable overplotting, apart from some sparsely populated regions towards the west of the country. As the data items are plotted in a random order they should show a representative sample, so small areas of high income (>\$150,000 shown in dark brown) stand out, as well as large areas of moderate income (in light blue). Relatively few areas of very low income feature on the map. The income colour scale for all the maps is given in Figure 6-2. Note that the income distribution is a relatively normal distribution (see Appendix B.7).

One striking feature of Figure 6-3 is that there are no plotted points in Wyoming or the lower half of neighbouring Colorado. As the population of these states in 2000 were 493 thousand and 4.3 million respectively, one must assume that the dataset is missing this data. However, this omission does highlight an advantage of sampling in

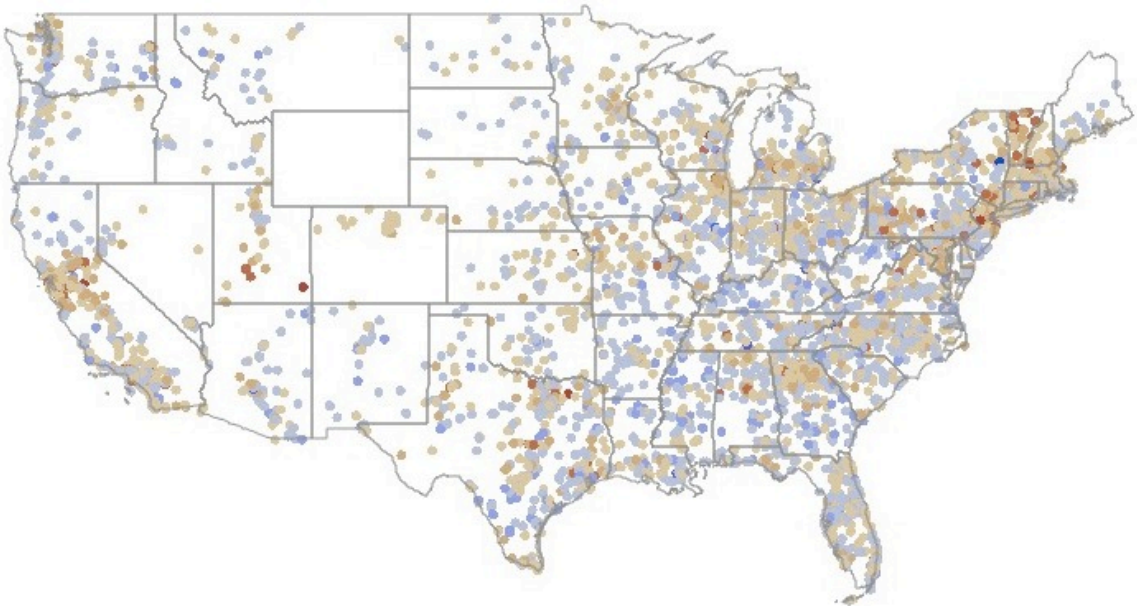


**Figure 6-4b** 1% sample of the original 155K dataset loses the income data in all but the major urban areas [USA Census 2000 household income]



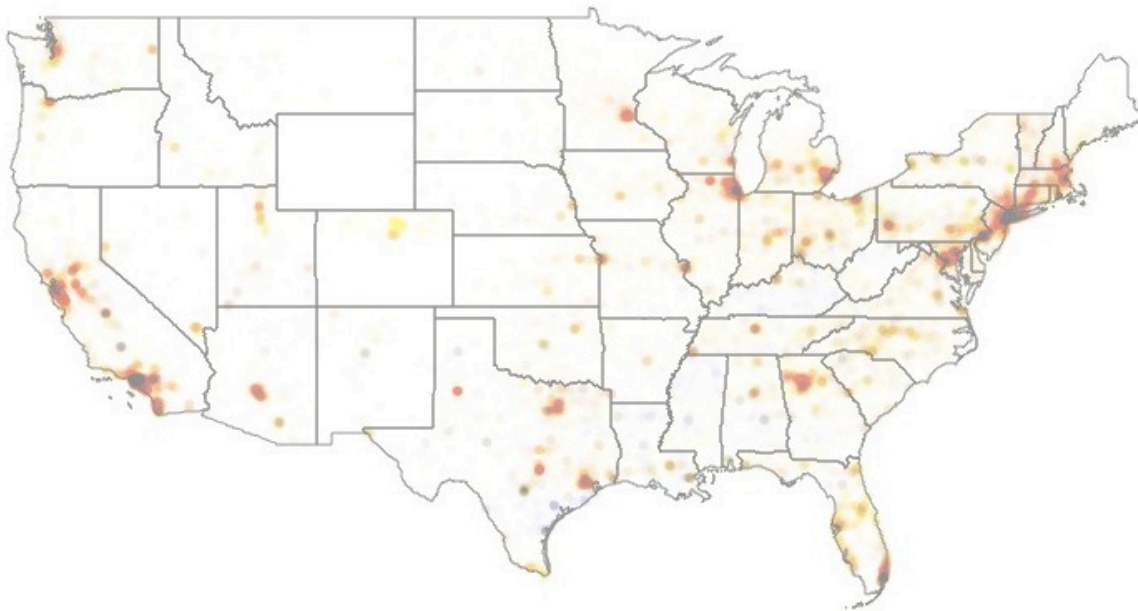
giving an overview of the data. A filtering interface with the income set to the range 0 to \$100,000 would have given the user the incorrect result and led them to conclude that all households in Wyoming are fairly wealthy, compared to the rest of the country.

The effect of sampling can be seen on the scatterplots in Figure 6-4. A 5% sample (shown in Figure 6-4a) gives an indication of the population density and the smaller 1% sample highlights the major urban areas, but much of the income data for the less populated regions has been lost. One could argue that as long as the data is plotted in a random order, the map in Figure 6-3 gives a better view of the income distribution than the sampled plots. But sampling does have the advantage of lending itself to being used interactively and is very adjustable (Table 6-3). As mentioned before, the act of changing the sampling rate not only reduces the amount of data on the display but importantly, emphasises the structure of the data. Moreover, as a user, one finds that the ability to easily adjust the sampling control (i.e. decreasing and increasing the data on view) is an essential aspect of exploring the dataset.

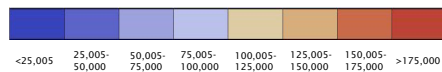


**Figure 6-4a** 5% sample of the original 155K dataset [USA Census 2000 household income]

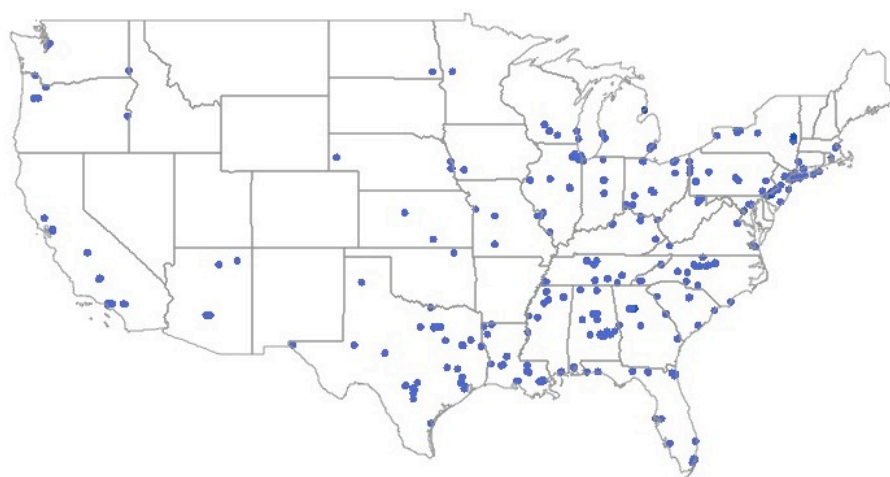
Whilst considering this particular dataset, the author thought it would be valuable to look again at alternative methods of reducing clutter, reducing opacity and filtering. Figure 6-5 shows two plots, one with the points set to opacity of 4% (Figure 6-5a) and the other set the 1% (Figure 6-5b). The former gives a good indication of regions of higher population density (more points = more households) and the blending of colour gives a rough guide to the average income. Reducing opacity further, loses this income information, as shown in Figure 6-5b, although major urban centres are more prominent. Therefore, opacity appears to be useful, especially if adjustable, and later on in this section we will see the successful combination of a sampling lens on a reduced opacity scatterplot.



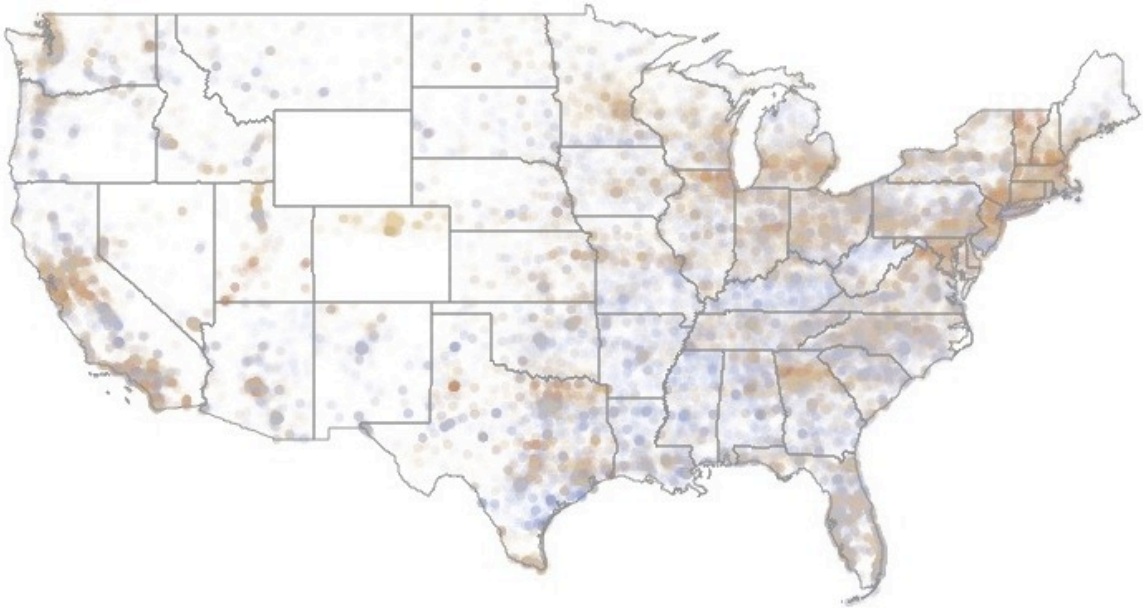
**Figure 6-5b** Reducing opacity to 1% highlights the major population centres but the other information is lost. [USA Census 2000 household income]



**Figure 6-6a**

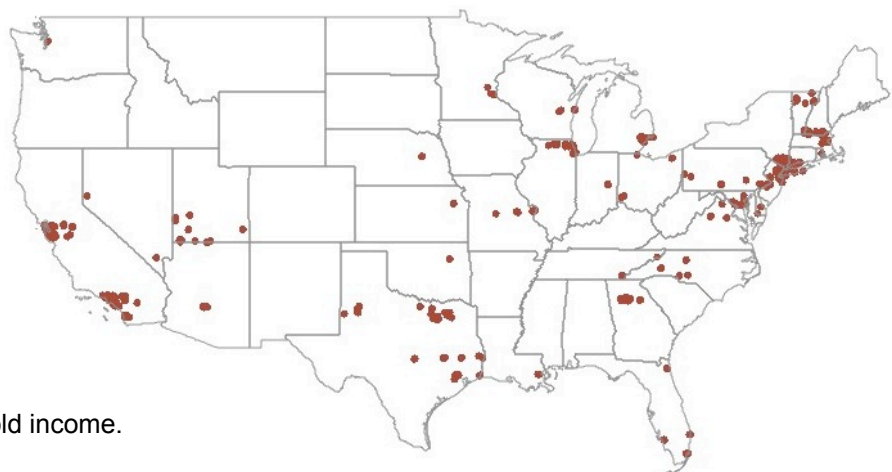


Low median household income. Filtering highlights areas of particular income categories but cannot show overall income distribution or regions of adjacent high and low income. [USA Census 2000 household income]



**Figure 6-5a** Reducing the opacity of plotted points to 4% gives a good indication of the higher population density areas (more households) and gives an approximate average income through colour blending. [USA Census 2000 household income]

As for filtering, if the user wishes to find regions of a particular income group then setting a dynamic filter will give the information. For example, the two plots in Figure 6-6 are the result of filtering the dataset on high and low incomes. However, showing the overall income distribution or finding adjacent high and low income areas is not possible with simple filtering. In addition, recall the finding in Chapter 4 (Table 4-2) that filtering is good at isolating a subset of data but cannot always be adjusted to reduce overplotting adequately.

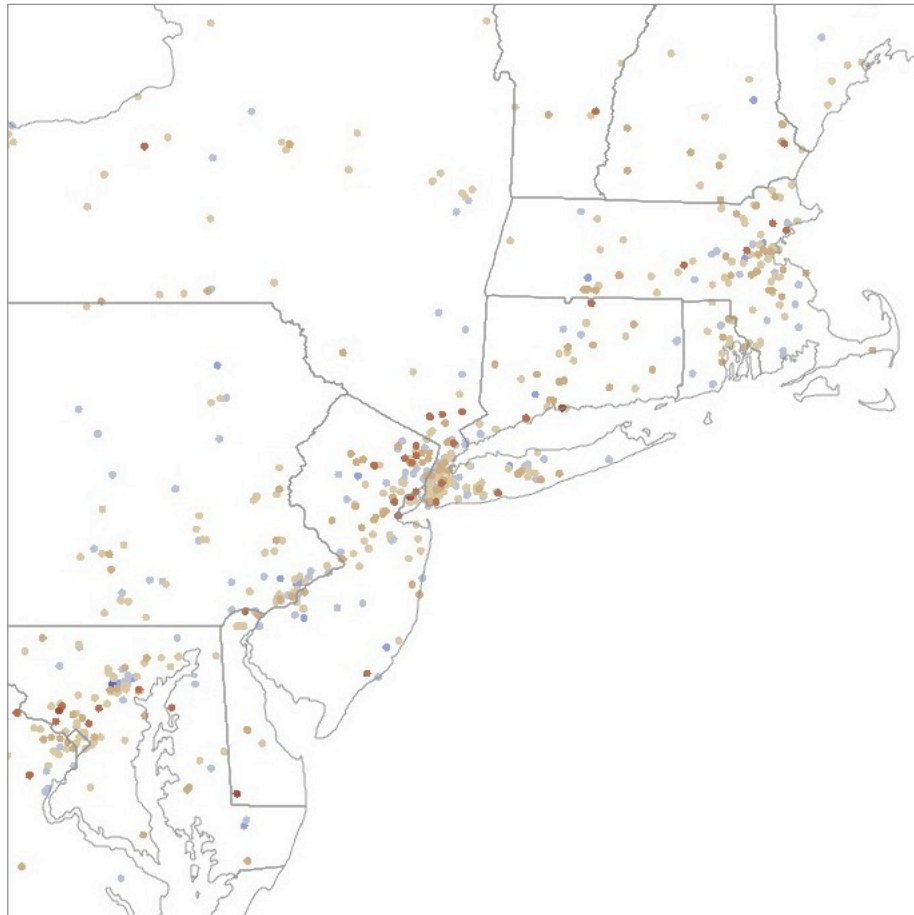


**Figure 6-6b**

High median household income.

For the next example, we zoom in on the North-eastern states of the USA, with the city of New York in the centre and Washington, DC towards the bottom left. In the previous scatterplots, the data was plotted in a random order, however, now the data has been sorted, so the points representing the highest incomes are plotted last and hence are

Figure 6-7b



Reducing the sampling rate to 2% gives a representative sample.

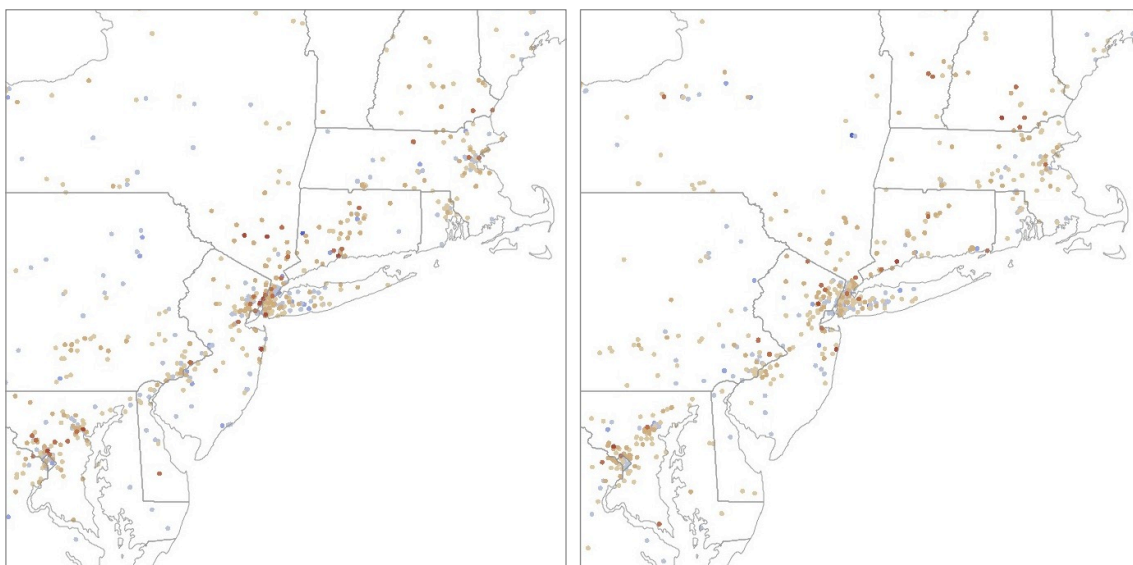
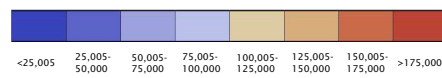


Figure 6-8 Two successive Reality Checks following on from Figure 6-7b, demonstrate that different 2% samples present representative views.

Figure 6-7a



North-eastern states (centred on the city of New York – the data is sorted so that the points representing the highest incomes are plotted last. [USA Census 2000 household income])

*on the top*. As shown in Figure 6-7a, this presents a very different and distorted view in areas with overplotting. Reducing opacity of the points does not help in this situation, as the brown coloured points still tend to hide the light and dark blues representing lower incomes.

Reducing the sampling rate to remove much of the overplotting (a 2% sample), as in Figure 6-7b, now gives a representative sample and hence a more truthful view. Detail is lost in the less populated regions, but as the sampling rate is easily adjusted, the user can increase the sampling rate to show more data in less populated regions if they wish. To check if a 2% sample does give a representative view, a series of Reality Checks were made. The first two of these (see Figure 6-8) demonstrate that even at low sampling rates, random samples show distinct overall income distribution patterns.

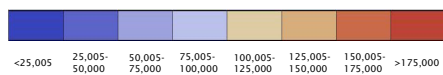
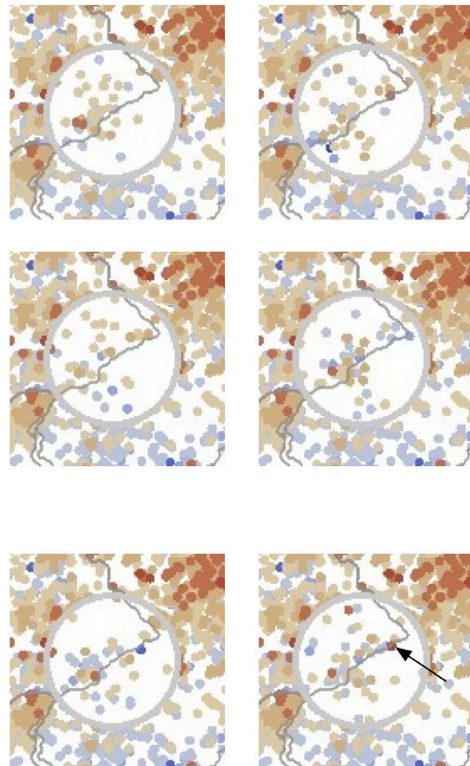
It would be possible to undertake non-uniform sampling on such a scatterplot (as discussed in Section 2.4). This would remove the population density information but would indicate the income spread across the whole map. We will now consider the use of a sampling lens, in exploring the North-eastern USA dataset.

Figure 6-9a



A sampling lens over Philadelphia reduces the overplotting so that the lower income data can be seen. For comparison, the inset picture is the lens region without sampling.

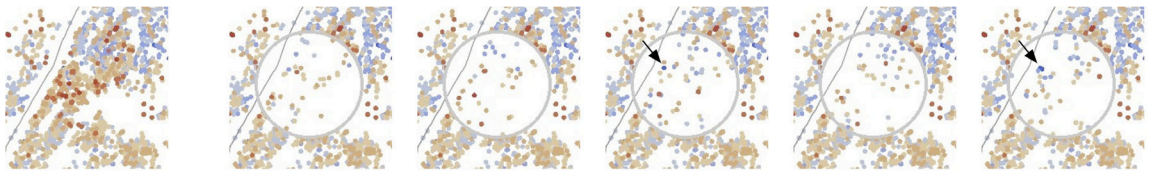
Figure 6-9b



Four successive Reality Checks (top) and two other samples (bottom). The lens sampling rate is 2%.

### 6.3.2. Lens-based sampling

Figure 6-9a illustrates the use of a sampling lens to reduce clutter in a localised area (in this case Philadelphia) rather than the global sampling seen in Figure 6-7. One obvious benefit is that the rest of the plot is unchanged, so data for the lower density regions can still be seen. As before, successive Reality Checks (top four pictures in Figure 6-9b) reveal a distribution of modest and higher incomes across the city as well as a higher population density along the river. More specifically, there does seem to be an area of lower income households south of the river, towards the bottom of the lens. However, a more significant finding, highlighted by further Reality Checks is that there is a wide range of incomes in many localities within Philadelphia. For example, the region of Figure 6-9b (bottom right) indicated by the arrow, has very high and very low incomes in adjacent housing blocks. Contrary to many people's belief that sampling removes outliers<sup>8</sup>, it became apparent whilst using the Sampling Lens that successive samples actually highlight anomalies within the data. In this situation, the extremes of the income scale (Figure 6-2) are saturated colours and hence pop-out pre-attentively. Even without such assistance, it appears that sampling is good at finding outliers in overplotted areas (with the value represented by the colour of the point). The lens also has the additional benefit of focusing attention on a specific region.



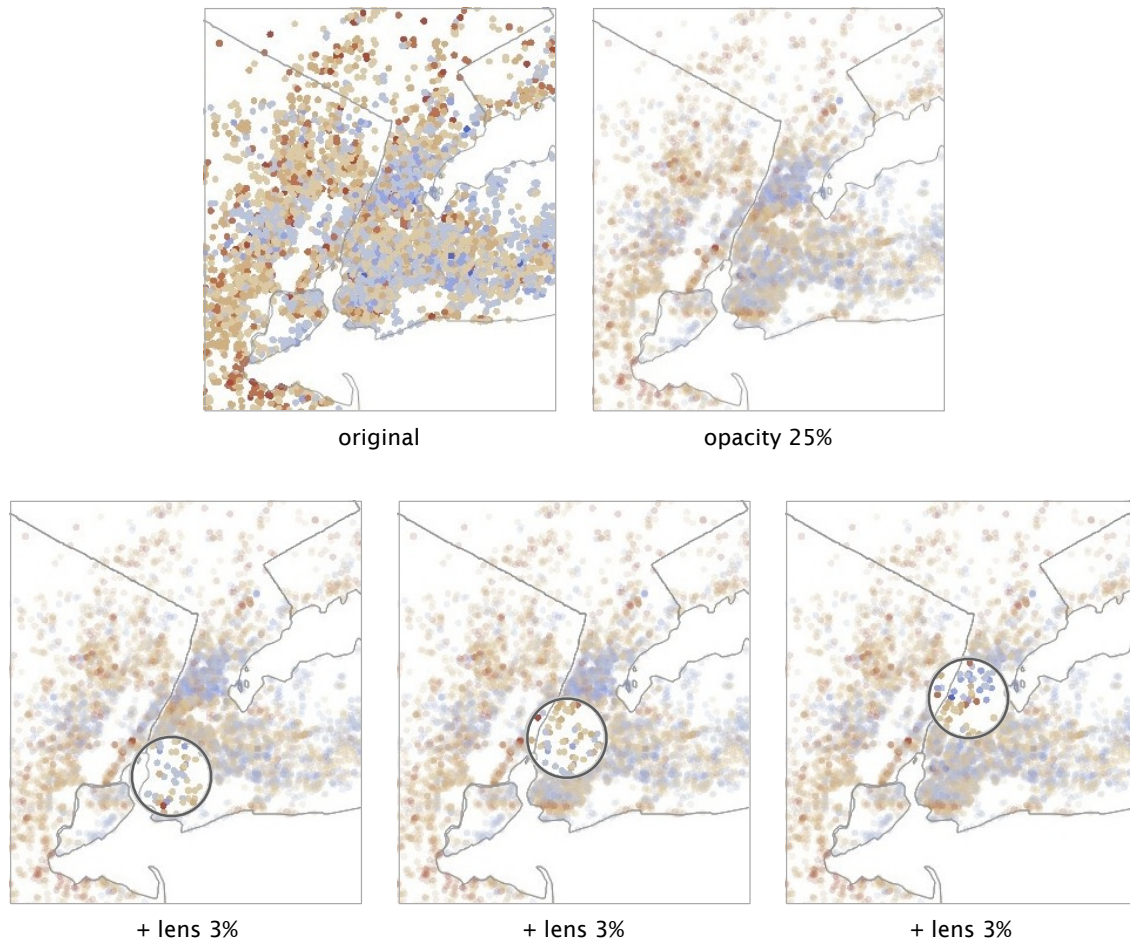
**Figure 6-10** Reality Check samples for the lens on a densely populated part of Manhattan, New York illustrating both the general trend and the diversity of income distribution. The full dataset for this region is shown on the left.

Another example where a sampling lens shows both a general trend and diversity is given in Figure 6-10. Here we see a densely populated region in Manhattan, New York with a high degree of overplotting. A lens sampling rate of 5%, avoids much of the occlusion and indicates that the northern region covered by the lens is occupied by lower income households (USA Census 2000). However, there are some surprising findings, such as the low income groups of households picked out by the arrows.

The last example on the use of a sampling lens on scatterplots suggests that the combination of reduced global opacity and a lens can be effective in viewing cluttered scatterplots. The sequence of screen shots in Figure 6-11 illustrates the story.

The original scatterplot of southern New York State and part of New Jersey, is very overcrowded (note that in contrast to the previous examples, the points are plotted in a random order). Changing the opacity of the plotted points to 25% highlights the

<sup>8</sup> This was a common comment made about sampling during demonstrations of the Sampling Lens.



**Figure 6-11** The effective combination of reduced opacity and a sampling lens on a cluttered scatterplot. Southern New York State and part of New Jersey. [USA Census 2000 household income]

dataset	records	attributes	no. lines ( $\times 10^3$ )	line length ( $\times 10^6$ )	Java2D msec	OpenGL msec	OpenGL from display list msec
stockmarket	1430	52	72.9	2.3	1150	350	85
cars 5K	5840	4	17.5	8.7	5000	150	57
cars 1K	1000	4	3.0	1.5	870	20	3
parcels	7321	4	21.9	10.8	5400	190	70

**Table 6-4** Comparing the performance of Java2D and OpenGL versions of the Sampling Lens with parallel coordinates datasets



densest regions and the resulting colour mixing gives a rough indication of the overall household income distribution in these regions. Exploring the data with a sampling lens (bottom row) provides a clear insight into that localised dataset as the opacity within the lens is set at 100%.

The effect is more pronounced in actual use rather than the printed screenshots, which conveys the sense of a focussed set of points within a slightly hazy overall map<sup>9</sup>. This idea may be worth pursuing with overcrowded scatterplots as it offers a density map, an averaging of the point attribute values (albeit, very dependent of the plotting order and colour scheme) and a means of viewing a representative sample in a localised region.

## 6.4. The Sampling Lens synthesis

The Sampling Lens application has progressed through many versions during this research. It started as the humble scatterplot and parallel coordinates version with a global sampling control and Reality Check button. It was then augmented with a lens and instrumentation to undertake the experiments. Finally, an OpenGL version was produced with a variety of novel lenses, re-sampling transitions and a constant-density zoom function.

This section reflects on the development and use of the Sampling Lens application and discusses its functionality for clutter reduction.

### 6.4.1. Development of the Sampling Lens application

The main objective of this research was to investigate sampling as a clutter reduction technique and not to create a production version of visualisation software. Hence, the prototypes do not contain all the functionality one would expect. For instance, the parallel coordinate does not have axis labels, the ability to reorder and resort axes and multiple range filters. The range of the sampling controls was in the integer range 1 to 100, hence the resolution with very large datasets was limited and should be extended, especially at low sampling rates. A logarithmic scale might be the answer.

As mentioned in Chapter 4, the drawing times for the Java2D versions were slow and hence the interactive performance was only acceptable with a relatively small datasets (approximately 1000 records). An OpenGL version of the Sampling lens was subsequently developed and compared to the Java2D version for a range of parallel coordinate datasets. The results in Table 6-4 show that the OpenGL version proved to be 30-40 times faster at drawing the parallel coordinate plots. Utilising the OpenGL display lists can boost the performance an additional 3 or 4 times. It is apparent that

---

<sup>9</sup> In some way this is similar to use of visual sharpness by Kosara et al. to make points stand out from the background (an illustration of this is given in Appendix A.2.3).



the Java2D drawing times tend to be proportional to the total length of the lines drawn whilst the OpenGL times are proportional to the number of lines drawn. Details of the comparison experiments and OpenGL development of the Sampling Lens can be found in Appendix D.3.

There are certainly opportunities to improve the performance when dealing with very large datasets. For instance, the use of a database back-end and associated queries to select the data to be displayed, especially within the lens, would have made the lens more responsive with the 155,000 record dataset used in the USA household income scatterplots (Section 6.3). More use could have been made of the graphics processor, through OpenGL programming. For example, frame buffer clipping to mimic the z-index selection, pbuffers to save the background image and viewports to simulate zooming.

Looking back at the development of the Sampling Lens, the use of the InfoVis Toolkit was certainly helpful. Its internal data structures are well crafted to give good dynamic query performance and the scatterplot and parallel coordinate prebuilt visualisations and associated control panels were used. Open source code meant that it could be modified although a lack of documentation and the sheer complexity of the code added to the challenge, especially when incorporating OpenGL. There were other problems associated with its single thread architecture and reliance on repaint methods that were not conducive to implementing auto-sampling within the lens. However, the toolkit is very powerful at building higher-level visualisations, such as NodeTrix [Henry et al. 07].

#### **6.4.2. Functionality of the Sampling Lens visualisation**

Some benefits of the sampling technique have already been identified through the development of the Clutter-reduction Taxonomy in Chapter 3 and the comparison of sampling with other techniques in Section 6.2 (see Table 6-3). This is an opportune time to reflect upon the Sampling Lens visualisation, using examples from this and previous chapters.

Looking back to the definitions of clutter mentioned at the start of this chapter, clutter is anything that inhibits seeing structure or patterns within the data. As a result, overcrowded displays should be avoided. There are plenty of examples (see Sections 4.1, 4.2, 4.4, 6.3) that illustrate how sampling reduces display clutter at both a global level and localised as various shapes of lenses. Moreover, informal evidence suggests that users readily understand the notion of random sampling, including lens-based sampling (Section 4.3). The fact that users comprehend the principles of sampling is important, remembering that we should endeavour not to distort a user's perception of the data space. [Chuah and Roth 95].



The excellent adjustability of sampling has been emphasised, which is ideal for interactive exploration. In addition, it has been noted (Sections 4.1, 6.3) that the ability to decrease or increase the sampling rate smoothly heightens the awareness of structures within the display. This is linked to the maintenance of display continuity (Chapter 4), so when the sampling rate is increased, data items reappear in the reverse order to which they were removed. This can be likened to panning a camera across a scene and such display continuity is also provided when moving the lens around the display.

The focus+context approach of the sampling-based lenses has the advantage of not changing the entire plot whilst investigating a particular region. This is particularly useful when large differences in density occur, as was seen in Section 6.3. The case was made for auto-sampling within the lens (Chapter 5), which has been successfully demonstrated. Auto sampling allows the user to set the desired degree of overplotting and this will be maintained by the system.

Along with round and square lenses that can be dragged anywhere on the display, novel lenses have been devised for parallel coordinate plots (Section 4.4). The usefulness of the inter-axis lens has been shown to reduce clutter in one or more regions between attribute axes, so relationships are easier to spot. In addition, the axis (filter) lens, a sampling version of a standard range filter, was seen to be effective in showing structure in a subset of a large dataset (Section 4.4.2). This example, together with the opacity+lens example in Section 6.3, are important in demonstrating that sampling can be combined to good effect with other techniques.

The Reality Check function generates new samples (either globally or within the lens) and has proved to be invaluable in gaining confidence in the sampling approach, and is a feature readily understood and welcomed by users (Section 4.3). Not only does this feature suggest whether a pattern is real or an artefact of the sampling, but as we saw in Section 6.3, Reality Checks are actually good at finding outliers within dense plots, especially when using a lens to focus attention on a particular area.

For clutter reduction, sampling, as demonstrated by the Sampling Lens application, is easy to use; informal user studies suggest it is intuitive; and technically, it is straightforward to add (using the z-index method) to visualisations that use a flat data structure. Sampling more complex data structures is not so straightforward, as we will see in the next section.



## 6.5. Can sampling be incorporated into other visualisations?

In this work, sampling has been incorporated into scatterplot and parallel coordinate plot visualisations, so the user (or indirectly by the system) can dynamically adjust the amount of data on display to reduce clutter. Three examples of other visualisations that utilise random sampling are given in Section 2.3 - The Influence Explorer (generates data from an infinite parameter space), Bertini and Santucci's scatterplot (non-uniform sampling) and ALVIN (sampling network graphs). Also, according to Cui [Cui 07], sampling has been incorporated into the XmdvTool [Xmdv], "to simplify data", although no implementation details are provided. In all these interactive applications, the user has some control on the amount of data displayed, although not necessarily the dynamic sampling controls of the Sampling Lens.

Other uses of randomness for visualisation are presented in Section 2.3, that include sampling non-uniform space and reducing the size of the dataset. However, all these use sampling as a pre-processing step to speed up some calculation and/or limit the data on display. This approach could be applied to many visualisations with an excess of data [Ellis and Dix 04], but the author is particularly interested in how we might add interactive sampling, so the user can reduce display clutter dynamically.

Note the suggestion above, that the sampling approach could be applied to *many* and not *all* visualisations. Whilst contemplating this issue, Calendar applications, such as DateLens [Bederson et al. 03] came to mind. Typically, it would show appointments and events such as trips to the theatre and birthdays. Imagine the trouble explaining to your nearest and dearest, that due to a cluttered display, the sampling algorithm had randomly removed their birthday from the display and hence you had forgotten to buy a present. Of course, this is why we have reminders built into electronic calendars, but all the same, we do need to be aware of the context in which the visualisation is used when opting to reduce clutter through sampling. In fact, sampling might not be considered the first choice when dealing with a cluttered display. For example, in the calendar application, it would seem advisable, to have a filtering option such as *show only birthdays*. On the other hand, sampling would be a useful addition to a photo-browser application, where there are too many photos to fit on the display whilst maintaining a sensible size. Even if the owner of the photo collection has rated each photo so that the *best* could be selected, it may be valuable to include a random selection of the others.

We will now look at adding sampling to applications that visualise hierarchical data and then generalise to visualisations that, by design, avoid overplotting. A suggestion is then made for using sampling to display representative data and finally conclusions are presented on whether sampling can be incorporated into other visualisations.

Figure 6-12a

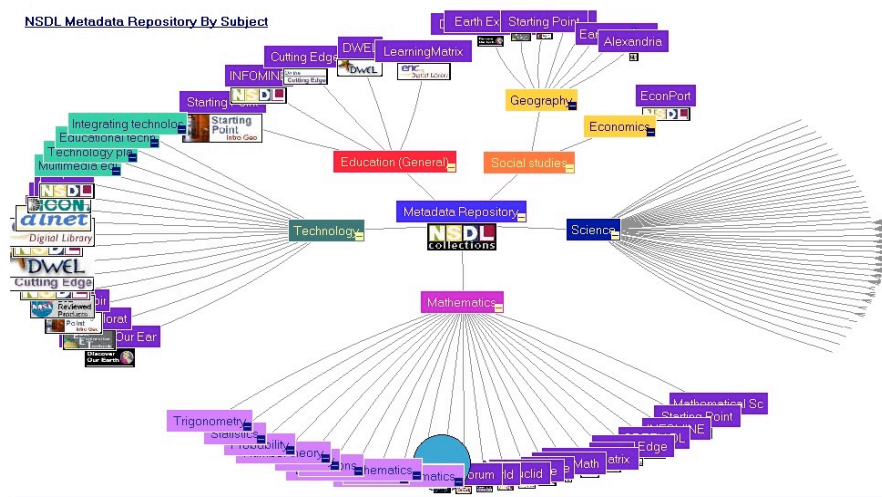
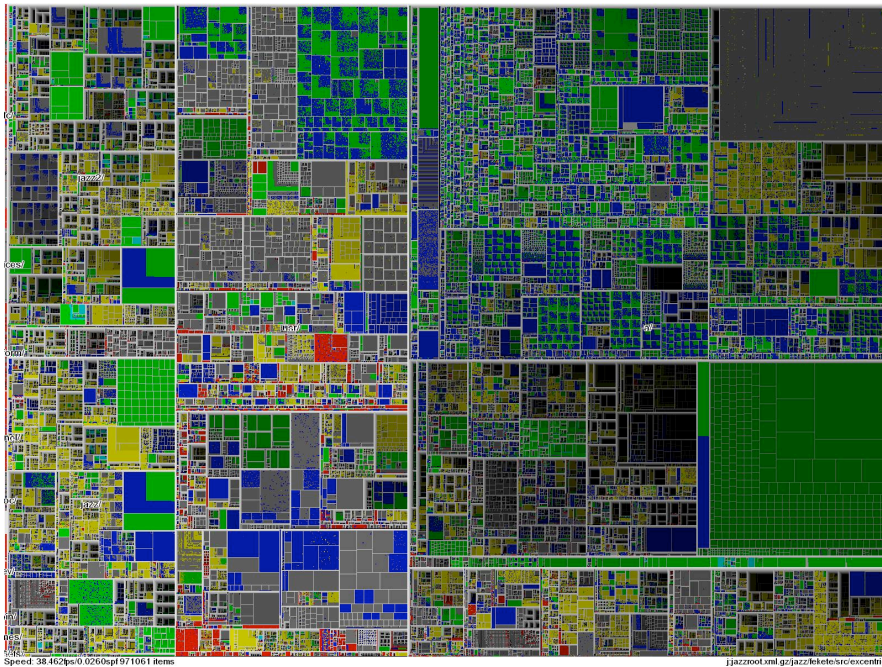


Figure 6-12b



Visualising large hierarchies with (a) Hyperbolic Browser [Lamping and Rao 96] and (b) Treemap [Fekete and Plaisant 02]



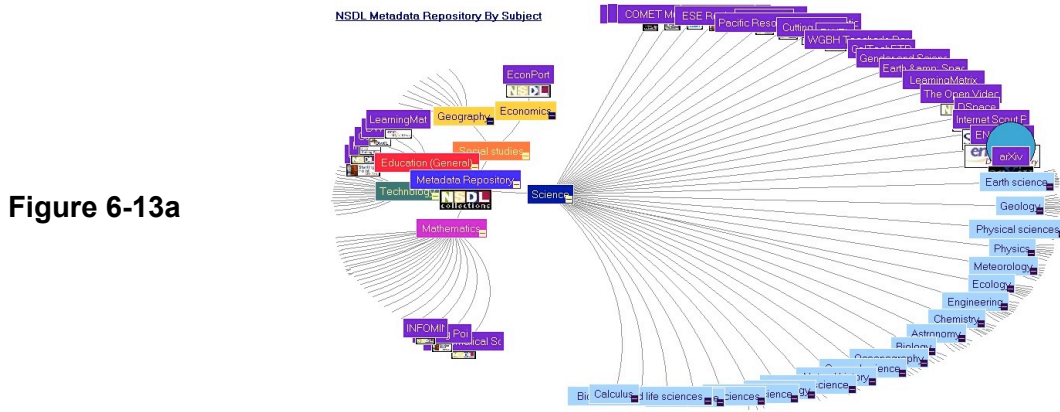


Figure 6-13a

Changing the focus in a Hyperbolic Browser to expand lower nodes. First to Science category and then in Figure 6-13b to Engineering. The initial state of the plot is given in Figure 6-12a.

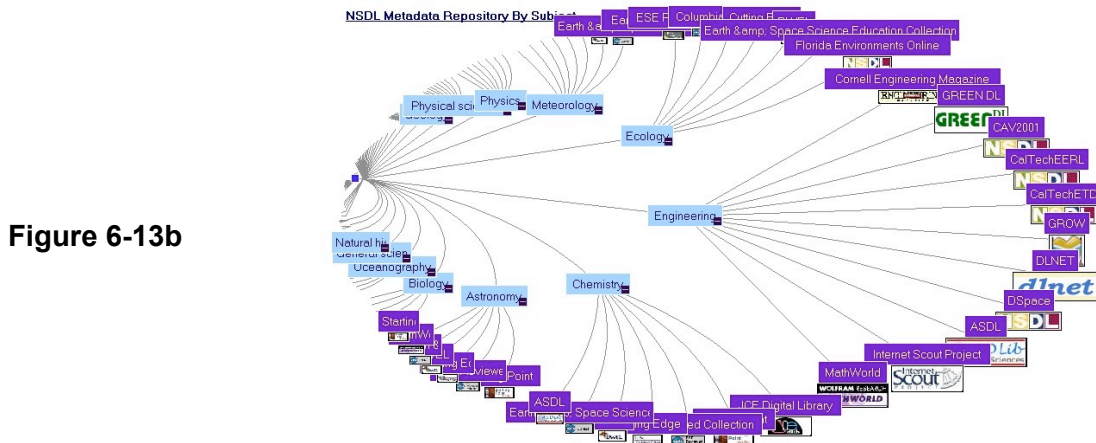


Figure 6-13b

Changing the focus in the Hyperbolic Browser example to expand to the Engineering category

### 6.5.1. Hierarchical data structures

There are various types of applications that visualise hierarchical data – acyclic graphs, either flat (e.g. SpaceTree [Grosjean et al. 02]) or spatially distorted (e.g. Hyperbolic Browser [Lamping and Rao 96]); radial segments (e.g. Sunburst [Stasko and Zhang 00]) and Treemaps [Shneiderman 92]. None of these allow overplotting, but all can have a very large number of leaf nodes, as illustrated in Figure 6-12. Hierarchical data can be self-pruning in that, if the visualisation has no space to draw the next level of the hierarchy, then it can stop at that point. It might add an indicator, to show that more data is available, as in the far right of the Hyperbolic browser plot in Figure 6-12a.

Functions to view the lower levels exist. For example, the Hyperbolic Browser allows the user to move the focus point and give more display space to the nodes below, as illustrated in Figure 6-13. Detail is reduced in other parts of the tree.

Figure 6-14

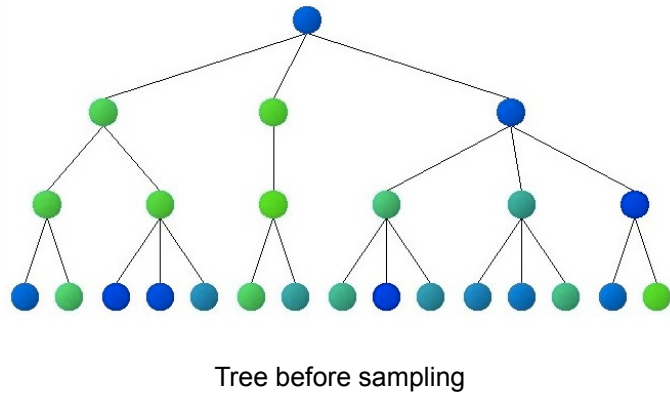


Figure 6-15a

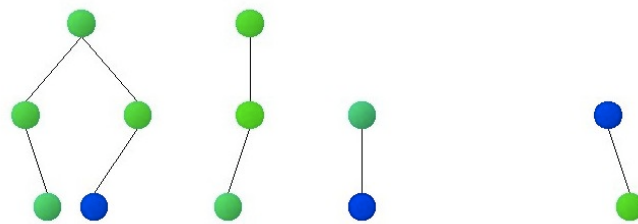
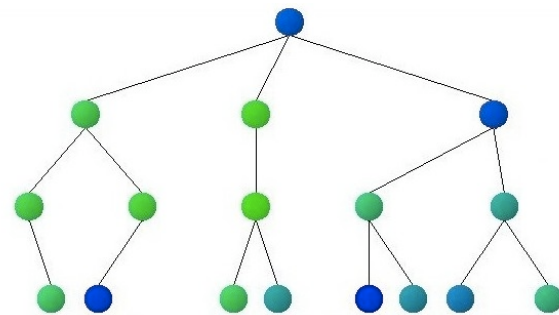
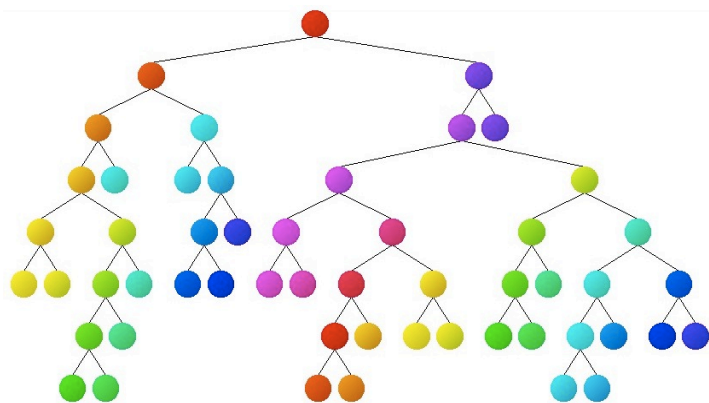


Figure 6-15b



Sampling the tree (50% sampling rate). (a) any node and (b) only leaf nodes.

Figure 6-16



Acyclic tree with different path lengths

But, can we sample the hierarchical data that is displayed in these graphs in a meaningful way? To answer this question, we will now look at different ways of sampling, using the simple acyclic tree shown in Figure 6-14 as the example. The Sampling Lens application was modified to draw a tree through one of the standard visualisations available in the InfoVis Toolkit and thus inherits the sampling control. Note that in reality, we would only be sampling if the display was cluttered in the first place. Clutter for a tree is as much to do with the ability to identify the name labels associated with each node (see Figure 6-13) as the patterns made by the nodes and edges.

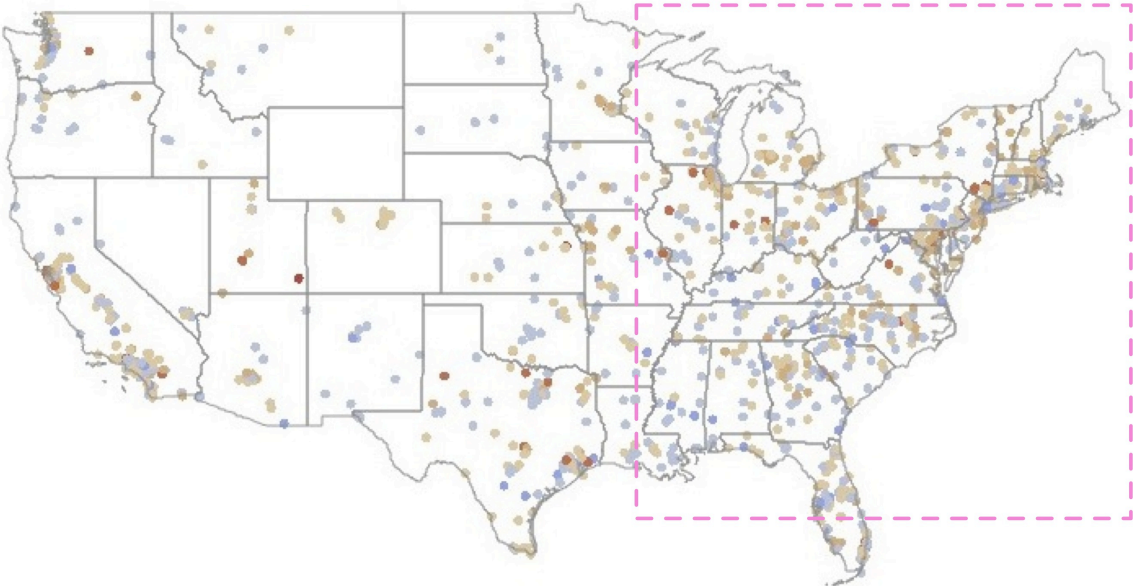
In the initial version, adjusting the sampling control to 50%, removes a random selection of the nodes, however, the result is incongruous as shown in Figure 6-15a. The *floating* nodes could be removed but in this case, the root node is missing, hence all the nodes would have gone. Another problem with removing these floating nodes is specifying the sampling rate. If, by chance, the root node was the first to be removed, then the sampling rate would either be 96% (notionally there are still 24 out of 25 left) or 0% (none on display).

Let us now try a different approach. For this particular dataset, all the leaf nodes are at the same level in the hierarchy, so a sample of these particular nodes would probably make more sense. A 50% sample of the leaf nodes is shown in Figure 6-15b (actually 68% of the total nodes). Notice that when the two leaf nodes on the far right have been removed, their parent node become a leaf node at a higher level, and it was decided to remove that as well, as the focus was the lower leaves. However, with this algorithm, removing all the leaf nodes would remove the entire tree.

A third approach would be to leave higher nodes in their place until all the lower nodes had been removed by sampling, and then sample the new set of leaf nodes. This might be more logical as the sampling rate would then correspond to the proportion of the original nodes on display.

It should be noted that no attempt has been made to balance the layout of the tree. This is possible, but as each node is removed or added, the layout shifts and would probably distract the user. An alternative algorithm would be to slowly change the layout after the user has stopped adjusting the sampling rate for a short time.

For fixed level tree structure, we can imagine sampling generating a selection of leaf node objects of the same type that would enable the user to explore their ascendants. Also, if the colour and/or size of a node represented some other attribute, then sampling would provide each node with more display space (i.e. drawn larger) and hence enable the user to detect patterns in the data. Now, if we look at a tree where the leaf node are at different levels, as in Figure 6-16, does it make sense to sample the



**Figure 6-17** Zooming in on a scatterplot with automatic adjustment of sampling rate. Full map area at 1% sampling rate. Dotted box shows next zoom region. [USA Census 2000 household income 155,000 record dataset]

leaf nodes in the same way?

Stratified sampling could be considered where the number sampled at each level is ideally proportional to the number of leaf nodes at that level, but the justification of this must be down to the particular dataset and the significance of each level.

Compared to acyclic graphs, Treemaps are far more economical in display space, as demonstrated by Fekete and Plaisant's visualisation of a million data items [Fekete and Plaisant 02] (see Figure 6-12b). As mentioned earlier, a Treemap is self-pruning in that the next level is not drawn if there is insufficient display space. Similar arguments as those discussed for drawing trees apply to sampling Treemaps. Should the lowest level leaf nodes be sampled first, or should all the leaf nodes be sampled? Again, this probably depends on the particular dataset. With the layout, a Treemap can be very sensitive to changes in the data. This is an important consideration, as unlike the acyclic graph example, a Treemap will have to be redrawn each time the size of the dataset is altered to avoid *holes* in the map. Given these significant issues, it is unclear whether it is indeed possible, let alone desirable to sample Treemaps.

### 6.5.2. Visualisations that avoid overplotting

Treemaps plot hierarchically structured data and avoids overplotting. Other space-filling and pixel-plotting visualisations, such as Keim's spirals [Keim and Kreigal 94] and TableLens [Rao and Card 94] also avoid overplotting. Sampling Keim's spirals (Appendix A.1.8) would not work, as the layout is sensitive to changes to the data. TableLens (Appendix A.1.8) on the other hand could benefit from sampling, but only to fit the number of records on a page of the display, thus the interactive element would not be particularly useful. Similarly, with any visualisation that sorts the data (TableLens sorts on columns) the user is often most interested in the extremes of the dataset (top or bottom records), hence sampling may not be appropriate.

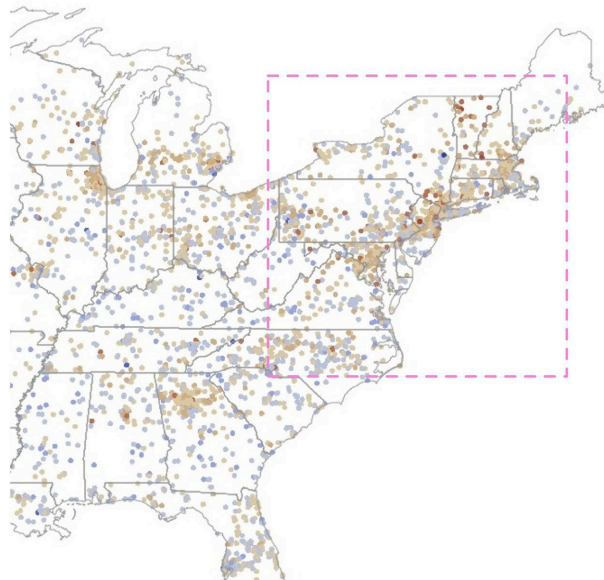
### 6.5.3. Visualisations that provide representative data

Lens-based sampling, provided by the Sampling Lens, reduces clutter by visualising a sample of the data within a particular region of the display. In Sections 8.3 and 8.4 we have seen examples showing the benefits of providing the user with a representative sample. Apart from the examples of sampling-based applications given in Section 2.3, there are few visualisations that utilise sampling to provide representative samples for the user.

The Scatter/Gather Browser [Pirolli et al. 96] presents the user with the title and keywords of the three most typical documents, although it is not clear whether this is a random selection from the document cluster or the result of some weighting function. However, this last example does suggest that sampling could be used to generate a representative sample, when the display space is limited or when we do not

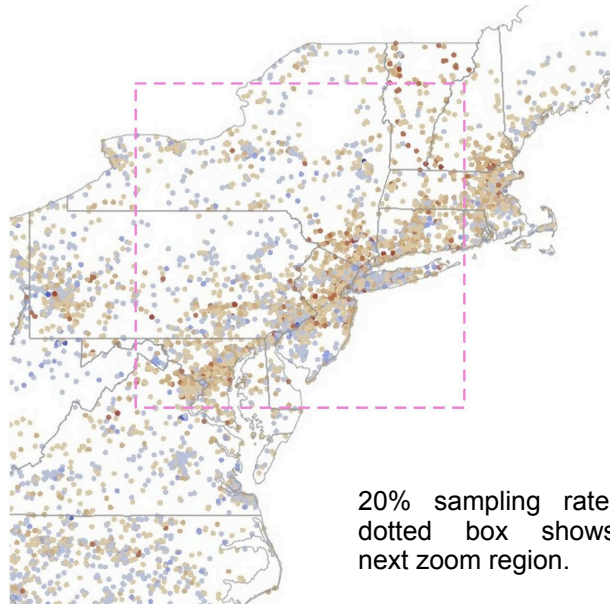
**Figure 6-18a**

Zooming in on a scatterplot with automatic adjustment of sampling rate. 5% sampling rate, dotted box shows next zoom region. [USA Census 2000 household income 155,000 record dataset]



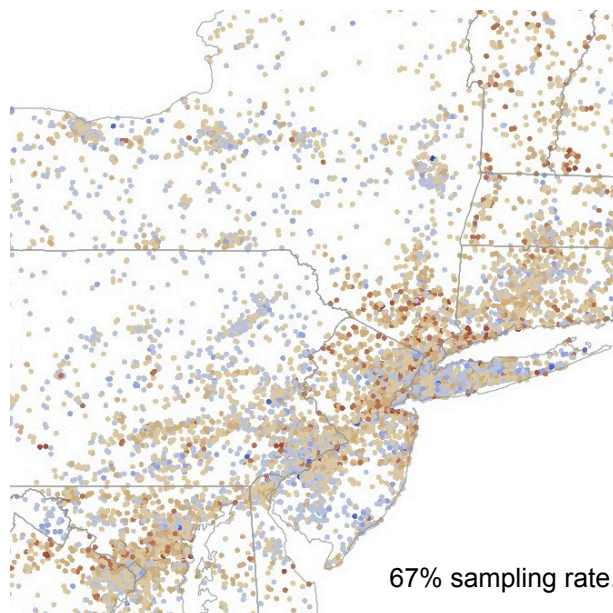
**Figure 6-18b**

20% sampling rate, dotted box shows next zoom region.



**Figure 6-18c**

67% sampling rate.



want to overwhelm the user with an extensive list of data items. For example, in a hierarchical clustered parallel coordinate plot (Appendix A.1.4), the user may select a particular cluster and then ask for a representative sample of this cluster. Similarly, on a 2D cluster plot, the user can request a sample of a particular cluster, and perhaps even utilise a lens to both select the cluster and see the results.

## 6.6. Astral Visualiser revisited

This work was inspired by the Astral Visualiser proposal as described in Chapter 2. At the time only a simple demonstrator was produced to illustrate the concept. So at this final stage it would be fitting to implement the visualisation and assess how well it works in practice.

The significantly higher performance OpenGL version of the scatterplot Sampling Lens (details in Appendix D.3) was modified so the user can stretch a dotted box over a particular area of interest<sup>10</sup>. The system calculates the scale factor, based on the decrease in area, and displays the new region, adjusting the sampling rate accordingly to maintain a notional constant density. For instance, if the user selects a quarter of the area of the display, the sampling rate will increase by a factor of four. The user may continue zooming in, but the sampling rate will not go beyond 100%. The system maintains a record of the size of each *zoom* region and hence the user can backtrack to previous zoom states. Note that the automatic adjustment of the sampling rate also occurs when the user zooms out, this time decreasing the sampling rate. The user is free to adjust the sampling rate at any time.

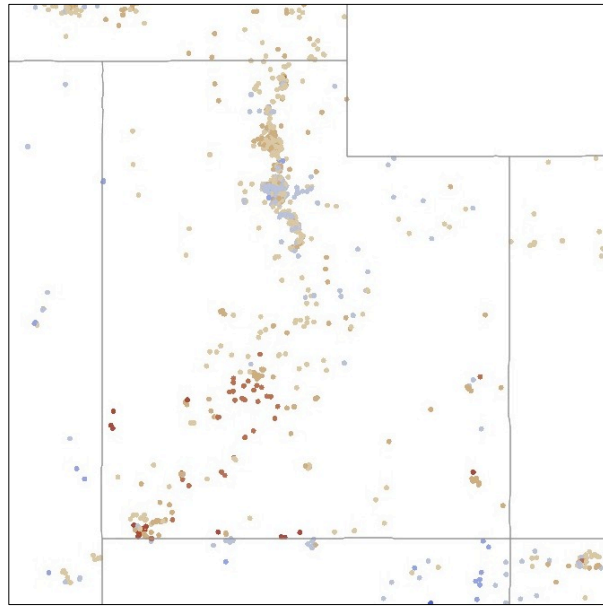
We will first consider a user zooming in to a map of the USA showing household income data, using the same dataset as in Section 6.3. Figure 6-17 shows the original map with a sampling rate set to 1% as the density of the points is very high, especially to the eastern side of the country. The user selects the region, shown by the dotted box and the system zooms in, displaying the map shown in Figure 6-18a. The user continues to zoom in (see Figure 6-18b) until the display shows the screenshot in Figure 6-18c. At each stage the system has calculated the scale factor and has set the sampling rate accordingly (at 5%, 20% and 67%). The series of screenshots clearly shows that this method of adjusting the sampling rate maintains the plot density rather well. For comparison, the last screenshot (Figure 6-18d) shows the same map region as in Figure 6-18c, but at the original sampling rate of 1%.

This mechanism also works when zooming out. Figure 6-19a displays the full set of data for the state of Utah. Note that apart from the Salt Lake City region to the north, Utah is sparsely populated due to mountain ranges, deserts and national forests. The

---

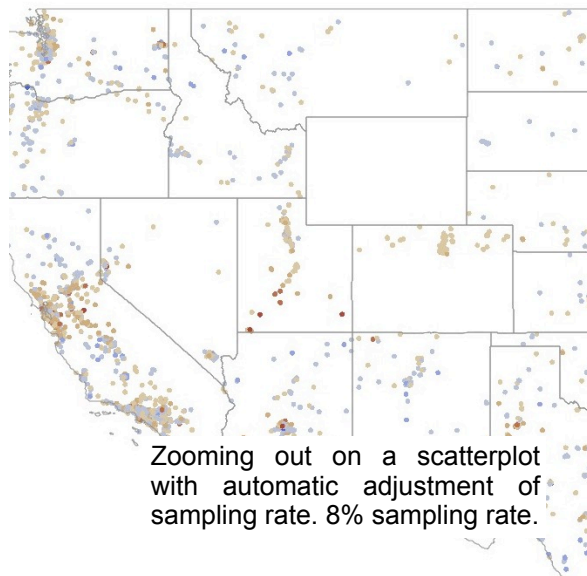
<sup>10</sup> Note that the zoom box is constrained to the width/height ratio of the display to avoid distorting the map.

Figure 6-19a



Full dataset (Utah).

Figure 6-19b



Zooming out on a scatterplot with automatic adjustment of sampling rate. 8% sampling rate.

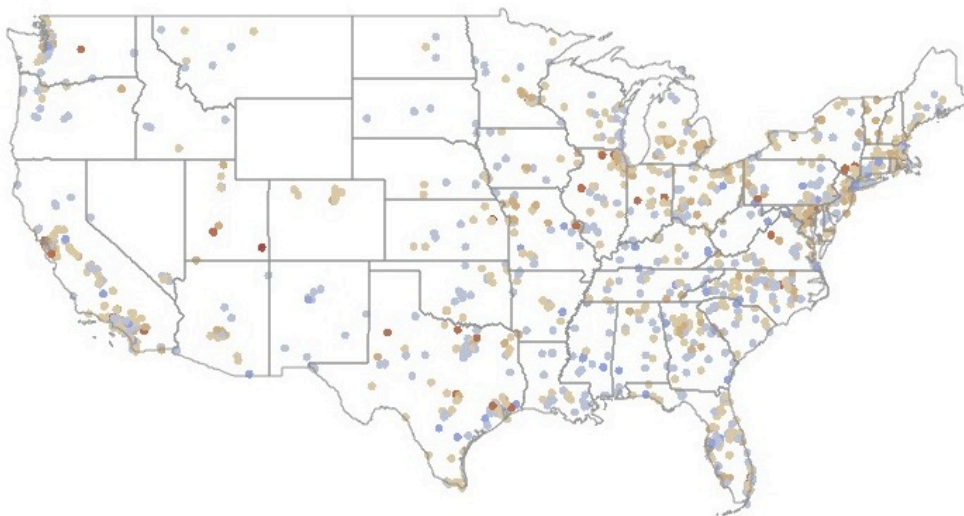
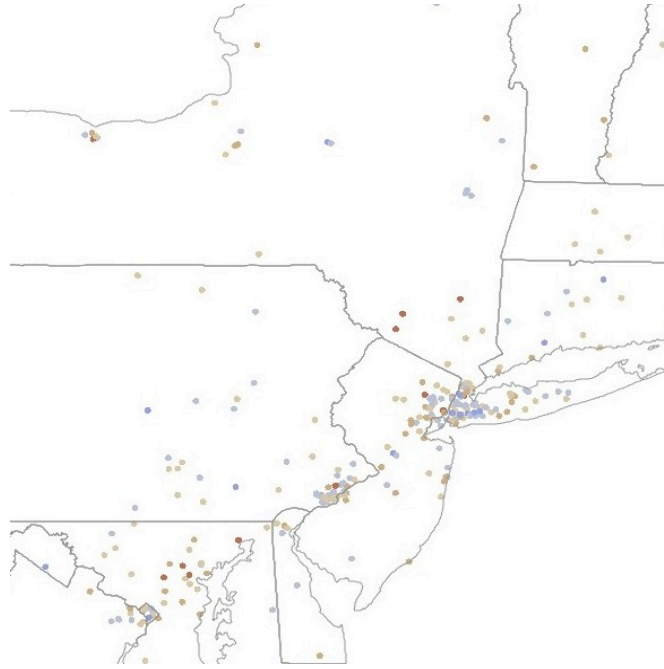


Figure 6-19c

Zooming out with automatic adjustment of sampling rate. Now at to 1%





**Figure 6-18d** Same region as previous map (Figure 6-18c) but at 1% sampling rate.

user zooms out to a previous display region and the system adjusts the sampling rate to 8% to compensate for the fact that there is now less screen area per unit map area to display the data points (Figure 6-19b). The user zooms out to the full size map and the sampling rate decreases proportionately to 1% (Figure 6-19c).

In this case, the method of automatically adjusting the sampling rate, based on the scaling factor, appears to be effective.

An alternative approach to setting the sampling rate is to treat the whole display area as a sampling lens and set the sampling rate accordingly. This was tested, but due to the very high overplotting density in urban areas, this method kept the sampling rate low until zoomed in on one of the urban regions. Consequently, little data was shown in large areas of the display.

Thus, the automatic adjustment of sampling whilst zooming in and out of a scatterplot, proposed in the design of the Astral Visualiser, appears to work well in practice.

### 6.6.1. Thinking about the Astral Visualiser

In a similar fashion to the Fisheye Menu example (Section 3.5.5) we will now reflect on the Astral Visualiser using the Clutter-reduction Taxonomy. Table 6-5 shows the sampling and topological distortion techniques from the taxonomy table, both of which are used for clutter reduction in the Astral Visualiser. The (uniform) topological distortion in this instance is zooming in.

**Table 6-5**

		1	7
		sampling	topological distortion
A	avoids overlap	possibly	possibly
B	keeps spatial information	✓	possibly
C	can be localised	✓	✓
D	is scalable	✓	✗
E	is adjustable	✓	✓
F	can show point/line attribute	✓	✓
G	can discriminate points/lines	✗	possibly
H	can see overlap density	✗	✗+

The Clutter-reduction Taxonomy for the sampling and topological distortion techniques

We can see from the example maps in the previous section (Figures 6-17, 6-18 and 6-19) that spatial information is kept. Although this criteria is marked as *possibly* for topological distortion, the fact that this is a uniform distortion and has easily recognisable landmarks in the form of state boundaries and coastlines, means that users should be able to keep track of the map locations. Of course if we zoom in to a region sufficiently to lose all boundaries then this would not be the case, unless a semantic zoom feature drew lower level features such as county boundaries.

The Astral Visualiser zoom can obviously be localised, is adjustable and does not affect the characteristics of the points on display. There is a limit to the extent of the zoom, for the reasons just mentioned regarding landmarks and hence the *is scalable* criteria is not met. Sampling is scalable (down to very few points<sup>11</sup>), adjustable; localisation is not utilised in this application. Both clutter reduction techniques do not avoid overlap all together and due to the auto-sampling function of the Astral Visualiser (i.e. increasing the sampling rate when zooming in), the example maps do not show a reduction in overlap, apart from the map of the whole of the USA that obviously benefits from a sampling rate of 1%. It is worth noting, that for co-incident points, sampling will tend to avoid overlap whilst topological distortion will not. In terms of discriminating points, zooming in does separate non-overlapping points, although the auto-sampling feature again, tends to negate this effect. Neither clutter reduction techniques enable the user to see the overlap density.

Could we improve the Astral Visualiser? We have already noted that in order to keep spatial information, we need to ensure that the user has sufficient landmarks. In addition smooth animated zooming would tend to preserve the users model of the data. Sampling and topological distortion do not meet criteria H, *can see overlap density*, however, looking at the taxonomy table (reproduced in this chapter as Table 6-2) we see that opacity is good at this task so adding an opacity control would be of benefit.

This example has demonstrated that the Clutter-reduction Taxonomy (table plus accompanying notes) is a useful tool for prompting discussion of a visualisation application as well as suggesting new features.

## 6.7. Summary and reflection

In this chapter we have reflected on many aspects of sampling as a clutter reduction technique; its advantages, drawbacks, evaluation and application to other visualisations.

---

<sup>11</sup> Assuming the sampling rate control is sufficiently sensitive, as noted in Section 6.4.1



User evaluation of information visualisation was discussed with the evidence stacking up against the value of user studies. The exploratory nature of information visualisation together with non-trivial domains means that low level tasks, often encountered in user interface evaluation studies are not appropriate. In-depth, longitudinal case studies or heuristic analysis show promise but it is generally agreed that these requires domain experts and for case studies, a substantial amount of time and analysis effort. The use of the Clutter-reduction Taxonomy in conjunction with targeted micro user studies was seen as an opportunity to add to our understanding of clutter reduction techniques. The objectivity of using a criteria-based evaluation of sampling was demonstrated to be high for the majority of the criteria.

The Clutter-reduction Taxonomy prompted a discussion of clutter reduction techniques, which highlighted the many benefits of sampling and identified some disadvantages, leading to the conclusion that sampling definitely shows promise.

The use of global and lens-based sampling to explore a large spatial dataset was demonstrated and we found that as well as reducing clutter to reveal structure in the data, sampling may reveal outliers. Reduced opacity was found to be useful in highlighting regions of dense overplotting and could favourably be combined with a *sharp view* from a sampling lens. Some of the disadvantages of filtering were highlighted and the importance of a random order of plotting data items was shown.

We reflected on the use of a visualisation toolkit to implement the Sampling Lens application and all things considered, the use of the InfoVis Toolkit was worthwhile. The functionality of the Sampling Lens was examined, which highlighted its desirable features such excellent adjustability, display continuity, re-sampling, various lenses and associated auto-sampling function.

We considered adding sampling to visualisations other than scatterplots and parallel coordinates. Visualisations that have data dependent layout algorithms are problematic, as the ensuing rearrangement of the data items every time the sampling rate is changed, would distract the user. For hierarchically structured data, we need to consider closely which leaf nodes should be sampled and the impact on the rest of the tree when they are removed. In addition, when using sampling to reduce clutter, it is important to be aware of the context in which the visualisation is being used.

The feasibility of auto-sampling zooming interface of the Astral Visualiser that prompted this work was demonstrated and the Clutter-reduction Taxonomy was successfully used to discuss the visualisation and to make suggestions for improvement.

A summary of the properties of sampling that make it applicable to clutter reduction is presented in the final chapter.



# Chapter 7

## Conclusion

We have come a long way since looking at the stars with the Astral Telescope Visualiser! The development of the Sampling Lens application has demonstrated the usefulness of random sampling in reducing display clutter both across the whole plot and localised by way of a lens. In addition, auto-sampling adds greatly to the utility of the lens and the increased performance from programming in OpenGL allowed the exploration of much larger datasets coupled with the invention of an additional lens and some further techniques for parallel coordinates. Evaluating sampling was also a challenge. However, pursuing an alternative approach to traditional user studies resulted in a taxonomy, which has proved useful in understanding sampling and other techniques and would be of benefit to the visualisation community.

Chapter 6 has already reflected on many of the issues arising from this work. To conclude, this chapter provides an overview of the thesis, highlighting some of the more noteworthy issues, as well as summarising the benefits and drawbacks of sampling as a clutter reduction technique. Some suggestions for further work prompted by this investigation are also considered.

Section 7.1 summarises the salient issues raised in each of the chapters of this thesis together with the main outcomes.

Section 7.2 provides a summary of sampling as a clutter reduction technique, bringing together the outcomes from the analytic evaluation, practical examination and discussion of sampling in Chapter 6, and observations made in Chapters 2 and 3.

Section 7.3 shows how the objectives set out in the Introduction have been met and highlights some of the major aspects this work. The main issues raised in meeting these objectives as well as the major outcomes and deliverables are summarised in the accompanying tables.

Section 7.4 discusses future work based on sampling relational data and complex data structures. The idea of the semantics of sampling is also introduced. Furthermore, the re-sampling Reality Check and the *twinkle* transition presented in Chapter 4 suggest a possible solution to visualising uncertainty.

main issue	outcome
can sampling be used to reduce clutter?	possibility for a zoomable auto-sampling interface shown
how should data items be removed and added as the sampling rate is changed?	importance of display continuity shown
how do we give users confidence that patterns are real and not artefacts of the sampling?	Reality Check proposed
need for a method for generating random samples that provides display continuity and re-sampling	z-index solution proposed
are there existing visualisations that utilise sampling?	three different examples that solve particular visualisation problems are described
need to be aware of data distribution and the different types of sampling available	examples given for non-uniform, constant density, stratified and relative sampling
database support for sampling	list of requirements for a database management system presented

**Table 7-1a** Main issues and outcomes from Chapter 2 - Sampling as a clutter reduction technique

main issue	outcome
how to choose a representative set of clutter reduction techniques	achieved through an extensive literature survey
what is the best method of assessing techniques?	assembled a set of criteria expressed as benefits
how to decide whether each technique meets each criteria	use evidence from the literature
how does each technique reduce clutter?	illustrated by a wide variety of examples
how to deal with special cases	in-depth discussion of special cases given
demonstrating the utility of the Clutter-reduction Taxonomy	logical organisation of taxonomy and accompanying discussion including examples of generating new visualisations given
how does the Clutter-reduction Taxonomy compare with other classification schemes?	comparison with Ward and Bertini's taxonomies related to clutter reduction highlights the usefulness of the Clutter-reduction Taxonomy to visualisation designers
can the Clutter-reduction Taxonomy be used to critique existing visualisation?	successfully demonstrated through Fisheye Menu example
are criteria-based classifications beneficial?	demonstrated the importance of criteria in thinking about visualisations and in constructing a classification

**Table 7-1b** Main issues and outcomes from Chapter 3 – Clutter-reduction Taxonomy



## 7.1. Main issues and outcomes

This section gives an overview of the main issues discussed in each chapter of this thesis and their outcomes in the accompanying tables (Table 7-1a to Table 7-1e).

### Chapter 2 - Sampling as a clutter reduction technique

Random sampling was proposed as a technique to reduce display clutter caused by overcrowding. A prototype visualisation, the Astral Visualiser, was described that raised various important issues. The z-index method was proposed as a solution for generating random samples, which ensures display continuity and provides a re-sampling function, the Reality Check, that confirms apparent patterns in the displayed sample.

When this work started there were no visualisations that used sampling to provide dynamic clutter reduction per se. Three different visualisations where sampling have been employed were considered, two of which have some interactive controls that further demonstrates the usefulness of sampling.

This chapter promotes an awareness of different types of statistical sampling in relation to the distribution of data and requirements from an analysis. Sampling from databases is considered and the list of requirements devised by Olken [Olken 93] is extended. Many of these requirements are not supported by current database management systems. The Astral Visualiser scenario highlights some advantages of sampling, thus making the exploration of sampling for clutter reduction plausible.

### Chapter 3 - Clutter-reduction Taxonomy

Chapter 3 presents the Clutter-reduction Taxonomy for for information visualisation as a matrix of common clutter reduction techniques against a set of criteria, expressed as benefits. The validity of the taxonomy was demonstrated through a systematic approach to its construction, based on the findings from an extensive literature survey. The importance of criteria in developing a classification was highlighted.

The taxonomy allows the user to see at a glance whether a particular technique satisfies a criterion and vice versa. Some cases are clear-cut while others are more complex and warrant further explanation; these are discussed in the easily accessible notes that are illustrated using example visualisations.

The taxonomy was motivated by the difficulties that are inherent with evaluation through user studies in information visualisations. This analytical approach not only presents a broad examination of techniques against benefits but it also offers advantages over existing classifications, given its accompanying in-depth discussion and effective organisation.

main issue	outcome
need for effective generation of data samples	z-index method successful in ensuring display continuity and generating Reality Checks
is sampling a useful technique for clutter reduction?	sampling-based scatterplot and parallel coordinate examples illustrate that sampling reduces clutter
how does sampling compare to other clutter reduction techniques?	a table of strengths and weaknesses of sampling and other clutter reduction techniques (available through the InfoVis Toolkit)
is there a focus+context solution to cope with a wide density range?	successful implementation of lens-based sampling for scatterplot and parallel coordinates
how to generate of a lens data sample	z-index method adapts well
can sampling be incorporated into existing parallel coordinate tools?	a sampling version of the parallel coordinate range filter is demonstrated
can novel tools be created to aid further understanding of the data?	RaDar adds relational information function to parallel coordinates
are lens Reality Check transitions helpful?	two different lens transitions are tested but only one is found to be useful

**Table 7-1c** Main issues and outcomes from Chapter 4 – Clutter reduction: random sampling and lenses

main issue	outcome
very few clutter metrics exist	invented three different measures of occlusion
how to compare occlusion metrics for overlapping lines	through an empirical study and probabilistic model (found close agreement)
efficient calculation of the occlusion metric for overlapping lines	empirical study successfully compared three very different methods of calculating occlusion metric
how to deal with non-uniform densities across the lens	devised a weighted measure through binning
what is the best solution for estimating occlusion of overlapping lines?	a binominal distribution based algorithm is surprisingly accurate and very fast

**Table 7-1d** Main issues and outcomes from Chapter 5 - The provision of auto-sampling

Visualisation designers can use the taxonomy as a means of critiquing existing visualisations and looking at possibilities for combining techniques to create new visualisations. An example of using criteria in thinking about an existing visualisation is demonstrated through the Fisheye Menu.

#### **Chapter 4 - Clutter reduction: random sampling and lenses**

Chapter 4 describes the implementation of two sampling-based visualisations, scatterplot and parallel coordinates, illustrating that sampling is a useful technique for clutter reduction. The z-index method was successful in providing display continuity and the Reality Check function. Using a simple scatterplot example, some strengths and weaknesses of sampling and other clutter reduction techniques such as, change opacity, change point size and filtering were highlighted. A focus+context approach was developed to cope with sampling a plot with a wide density range. The moveable sampling lens was effective in exploring dense regions of scatterplots and parallel coordinates whilst retaining the context of the whole plot. The z-index method adapts well to generating the lens samples. Several novel sampling-based lenses are demonstrated to assist the user in discovering significant information in overcrowded parallel coordinate plots. These include the axis lens, which adds sampling to the traditional range filter.

#### **Chapter 5 - The provision of auto-sampling**

Auto-sampling maintains an uncluttered view within the lens without user intervention, even when moving the lens into a high density region. This requires rapid calculation of density, the determination of which proved particularly elusive for the overlapping lines in parallel coordinate plots. An extensive empirical study combined with an algorithmic experimentation provided a suitable occlusion metric and devised an accurate and efficient method to calculate its value, even in extreme cases. The solution, based on a random distribution model, was found to be very accurate and fast.

main issue	outcome
problematic nature of user studies for information visualisation evaluation	discussion of problems and possible solutions to user evaluation studies
validity of criteria-based evaluation	demonstrated through an assessment of the objectivity of criteria-based evaluation of sampling
how to evaluate sampling	demonstrated analytically through comparison with other clutter reduction techniques using the taxonomy and demonstrated practically through an exploration of a sizeable dataset
is sampling useful in understanding a dense scatterplot map?	successfully demonstrated through examples and Reality Check function was shown to be effective at highlighting outliers
how to evaluate the functionality of the Sampling Lens application	demonstrated through examples of the utility of the Sampling Lens functionality
can sampling be incorporated into other types of visualisations?	discussion of difficulties and possible solutions
is a zoomable constant-density scatterplot possible?	demonstrated through an implementation of Astral Visualiser

**Table 7-1e** Main issues and outcomes from Chapter 6 – Evaluation of sampling

▶ reduces amount of data	Chapter 2
▶ good for explorative visualisation	Table 6.3
▶ scalable to very large datasets	Table 6.3
▶ does not affect characteristics of the plotted data items	Table 6.3
▶ can be combined with other techniques	Section 6.4.2
▶ can be applied to a localised region	Section 6.4.2
▶ random yet adaptable (spatially and statistically)	Table 6.3
▶ has a simple interactive user control and is highly adjustable	Table 6.3
▶ relatively easy to implement	Section 6.2.2
▶ re-sampling adds confidence	Section 6.4.2
also indirectly:	
▶ can indicate overlap density	Table 6.3
▶ can identify outliers	Table 6.3

**Table 7-2** A summary of the benefits of a sampling approach to clutter reduction

## Chapter 6 – Evaluation of sampling

The problematic nature of user studies for information visualisation was reviewed and justifies the evaluation of sampling using the Clutter-reduction Taxonomy. Opportunities for micro user studies were also suggested that, in conjunction with the taxonomy, may contribute to a better understanding of clutter reduction. The objectivity of the criteria used in the taxonomy were assessed for sampling that demonstrated the validity of criteria-based evaluation.

An analytical evaluation of sampling using the Clutter-reduction Taxonomy is presented together with a practical examination of global and lens-based sampling of a real, large (155,000) map-based dataset using a scatterplot version of the Sampling Lens visualisation. An exploration of a scatterplot map, demonstrates the effectiveness of both global and lens-based sampling in gaining a better understanding of a large dataset.

The implementation of the Sampling Lens application was discussed and possible improvements were suggested. The functionality of the Sampling Lens was examined, which highlighted its desirable features such excellent adjustability, display continuity, re-sampling, a range of lenses and invaluable auto-sampling function.

The possibilities of adding sampling to other visualisations, apart from scatterplots and parallel coordinates, are discussed. This revealed that visualisations with data dependent layout algorithms may be problematic and sampling hierarchically structured data needs careful consideration. The proposal for a sampling visualisation (Astral Visualiser) that started off this work was implemented, which showed the feasibility of zoom-dependent sampling. In addition, the Clutter-reduction Taxonomy was used to promote a discussion of the clutter reduction techniques used by the Astral Visualiser.

## 7.2. Summarising sampling as a clutter reduction technique

Table 7-2 summarises the main benefits of sampling as a clutter reduction technique based mainly on the in-depth evaluation presented in Chapter 6. The right-hand column of the table indicates which section of the thesis the item originated from.

Note that the benefits have been put into an approximate order; those dealing with data towards the top and those more concerned with implementation towards the bottom. An overview of each of the benefits of Table 7-2 is given below.

**reduces amount of data** – one of the basic methods of reducing clutter.

**good for explorative visualisation** – especially when the user is unsure what questions to ask or just wants to browse the data; random sampling reduces clutter without prejudice.



**scalable to very large datasets** – literally any size dataset can be sampled.

**does not affect characteristics of the plotted data items** – the shape, colour, size, positions, etc., are unchanged by sampling (assuming that visibility is not a characteristic)

**can be combined with other techniques** – sampling only affects the selection of the data to be displayed, hence it can be combined with almost any other technique. Examples of sampling+opacity and sampling+filtering (axis lens) have been demonstrated (See Sections 6.3 and 4.4.2). Note that techniques that change the layout (e.g. space-filling) will have to cope with a re-organisation when the sample is changed (see Section 6.5.2).

**can be applied to a localised region** – useful for a focus+context approach such as a lens or for non-uniform sampling (see Section 2.3.2).

**random yet adaptable (spatially and statistically)** – random samples have the advantage of being random yet can be shaped algorithmically. For example, differential sampling rates can be applied across a plot (e.g. Bertini and Santucci's non-uniform sampling – see Section 2.3.2) and statistical methods can be used to deal with particular types of data (e.g. stratified or attribute dependent sampling – see Section 2.4).

**has a simple interactive user control and is highly adjustable** – a sampling control can be any interface widget (or remote control for that matter) that adjusts the sampling rate between 100% and 0% (or very near to zero). The number of steps can be as great as the number of data items. The ability to smoothly and rapidly alter the sample size can heighten the awareness of structures within the display.

**relatively easy to implement** – the z-index method has proved both effective and efficient in generating global and lens samples, and re-sampling without replacement<sup>1</sup> in the form of the Reality Check function.

**re-sampling adds confidence** – the Reality Check function generates a new data sample and patterns that persist are likely to be *real* as opposed to an artefact of the sampling.

**can indicate overlap density** – in auto-sampling mode, the lens sampling rate is adjusted to give a relatively constant density. By observing the lens sampling rate control (shown in Figure 5-2), the user gets an impression of the un-sampled density in that particular lens location.

---

<sup>1</sup> Strictly, this is only the case with small samples and a large dataset where the sample does reach the end and wraps around to the start.

▶ need to be aware of the context	Section 6.5
▶ cannot avoid co-incident points	Section 6.2.1
▶ complex data structure may be a problem	Section 6.5
▶ no pattern enhancement	Appendix A.1

**Table 7-3** A summary of some disadvantages of a sampling approach to clutter reduction



**can identify outliers** - in addition to giving confidence in persistent patterns, the Reality Check can identify outliers, in other words, those *odd* points that appear in unexpected locations and/or with unexpected characteristics, such as colour.

Despite the many advantages of sampling noted above, some drawbacks have also been identified. These are summarised in Table 7-3 and are discussed below.

**need to be aware of the context** - one should not assume that it is wise to sample all data. Recall the missing birthday problem mentioned in the introduction to Section 6.5.

**cannot avoid co-incident points** - sampling cannot guarantee to prevent co-incident points or lines (unless there is only one left).

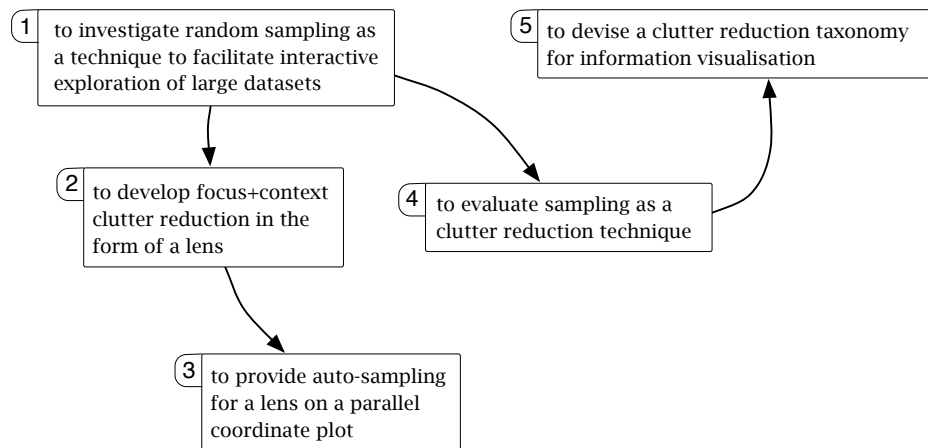
**complex data structures may be a problem** - Section 6.5.1 considered adding sampling to hierarchically structured datasets and concluded that it was not straightforward and further research into this area was necessary.

**no pattern enhancement** - some clutter reduction techniques not only reduce display clutter but also, by design, enhance patterns or trends within the data. For example, hierarchical clustering on parallel coordinates will identify groups of similar polylines (Appendix A.1.4) and Keim's pixel-plotting spirals (Appendix A.1.8) can identify trends in multivariate data. The last disadvantage is perhaps more of a difference of purpose between sampling and some of the other techniques. It could even be thought of as an advantage that sampling does not attempt to enhance patterns, bearing in mind that clustering algorithms often need to be tuned to reveal sensible results.

Having described the benefits and drawbacks of sampling, based on analytical and practical evaluations, it is interesting to revisit the summary of foreseen advantages of sampling, presented the end of the Chapter 2, following the proposal for the Astral Visualiser sampling-based application.

1. reduces the amount of data, which will generally lead to a reduction in clutter.
2. is relatively easy to implement.
3. is good for explorative visualisation where we are not sure of the question.
4. has a simple interactive user control.
5. can be adapted to take account of sampling distributions.
6. can provide a particular density mapping across the display.
7. can offer automatic density control when zooming.
8. is a natural operation - the majority of data is a sample anyway.

All these perceived benefits, apart of 5 and 6 that deal with non-uniform sampling, have been realised in the Sampling Lens. Sampling relational data to take sample distributions into account is one of the suggestions for further work in Section 7.4.



**Figure 7-1** Relationship between the objectives

main issue	outcome
<ul style="list-style-type: none"> <li>▶ the need to devise an occlusion metric and an efficient way of calculating its value</li> </ul>	<ul style="list-style-type: none"> <li>▶ demonstrated auto-sampling lens for parallel coordinates (Chapter 5)</li> <li>▶ found an accurate and fast method for calculating occlusion of lines (Chapter 5)</li> <li>▶ binomial distributions prove to be very useful in finding a good enough solution (Chapter 5)</li> </ul>

**Table 7-4a** Main issues and outcomes for objective 3 - to provide auto-sampling for a lens on a parallel coordinate plot

main issue	outcome
<ul style="list-style-type: none"> <li>▶ efficient generation of the lens data sample</li> </ul>	<ul style="list-style-type: none"> <li>▶ implemented lens enabled sampling visualisations for scatterplot and parallel coordinates (Chapters 4, 5 and 6)</li> </ul>
<ul style="list-style-type: none"> <li>▶ manual adjustment of lens sampling rate is undesirable</li> </ul>	<ul style="list-style-type: none"> <li>▶ generated objective 3, need for auto-sampling</li> </ul>
<ul style="list-style-type: none"> <li>▶ good interactive performance is required</li> </ul>	<ul style="list-style-type: none"> <li>▶ demonstrated improved performance through OpenGL programming (Chapter 6)</li> </ul>
<ul style="list-style-type: none"> <li>▶ can sampling be added to other tools?</li> </ul>	<ul style="list-style-type: none"> <li>▶ showed that sampling can be added to existing tools (e.g. parallel coordinate axis lens) (Chapter 4)</li> </ul>

**Table 7-4b** Main issues and outcomes for objective 2 - to develop focus+context clutter reduction in the form of a lens

### 7.3. Meeting the objectives of this work

This section revisits the objectives as laid out in Chapter 1 (see Figure 7-1) and shows how each one have been met. In pursuing these objectives, various issues were raised, and these together with the outcomes are presented in the accompanying tables (Table 7-4a to Table 7-4e).

We will adopt a bottom-up approach, starting with objective 3, which has largely been met followed by objective 2 (which actually prompted objective 3). We will then consider objective 5, followed by objective 4 and finally objective 1 that started off this work.

**Objective 3:** to provide auto-sampling for a lens on a parallel coordinate plot

The need for auto-sampling for a lens on a parallel coordinate plot became apparent early on in this work (see Section 4.4). The empirical and theoretical work described in Chapter 5 culminated in providing a successful solution, thus meeting this objective.

Some of the important achievements related to the provision of auto-sampling includes: innovations (occlusion measures, measurement methods), theoretical models (occlusion metric, *random* and *lines* measures), software instrumentation of the Sampling Lens (see Appendix D.1) and a full programme of experiments and concerted analysis of the results.

Deliverables:

- occlusion metric for overlapping lines published as AVI'06 paper [Ellis and Dix 06b]
- issues related to the provision of auto-sampling for parallel coordinates published in Trans. Visualization & Computer Graphics /InfoVis'06 conference paper [Ellis and Dix 06c]

**Objective 2:** to develop focus+context clutter reduction in the form of a lens

This objective was driven by the requirement to reveal detail in otherwise overcrowded regions of scatterplots and parallel coordinate plots, whilst retaining the context of the plot as a whole. The idea for a lens follows the tradition started some fifteen years ago by Bier, Stone and Fishkin<sup>2</sup>. This objective has been met to a large extent in that sampling-based lenses have been successfully demonstrated for scatterplots and parallel coordinates. However, to fully meet this objective, one would have to implement lenses on a wider range of information visualisations, although the choice appears to be limited. For instance, global sampling of a Treemap was deemed inappropriate in Section 6.5.2 and restricting the sampling to a lens region would not improve this.

---

<sup>2</sup> Bier et al. 93 and 94, Stone et al. 94, Fishkin and Stone 95. Note that distortion-based focus+context techniques were first suggested before this. According to Spence [Spence 07], the first was the bifocal display demonstrated in the Imperial College Studio in 1980, although Card et al. [Card et al. 99] cite Furnas's fisheye views [Furnas 81] at being the earliest.

main issue	outcome
<ul style="list-style-type: none"> <li>▶ how does one compare a wide range of clutter reduction techniques?</li> </ul>	<ul style="list-style-type: none"> <li>▶ produced the novel Clutter-reduction Taxonomy for information visualisation aimed at visualisation designers (Chapter 3)</li> </ul>
<ul style="list-style-type: none"> <li>▶ how does one demonstrate validity and utility of the taxonomy?</li> </ul>	<ul style="list-style-type: none"> <li>▶ demonstrated the efficacy of the taxonomy in suggesting new and critiquing existing visualisations (Chapters 3 and 6)</li> </ul>

**Table 7-4c** Main issues and outcomes for objective 5 - to devise a clutter reduction taxonomy for information visualisation

main issue	outcome
<ul style="list-style-type: none"> <li>▶ user studies for evaluating information visualisation are fraught with problems</li> </ul>	<ul style="list-style-type: none"> <li>▶ led to objective 5, the Clutter-reduction Taxonomy</li> <li>▶ investigated alternative evaluation methods (Chapter 6)</li> </ul>
<ul style="list-style-type: none"> <li>▶ how to evaluate sampling?</li> </ul>	<ul style="list-style-type: none"> <li>▶ analytical evaluation of sampling using for Clutter-reduction Taxonomy (Chapter 6)</li> <li>▶ practical demonstration of sampling (Chapters 4 and 6)</li> </ul>

**Table 7-4d** Main issues and outcomes for objective 4 - to evaluate sampling as a clutter reduction technique

Implementing the lenses, especially for parallel coordinates has been a challenging task but it has been very satisfying to explore the use of the lenses on a variety of datasets. The z-index method for generating the lens data samples (described in Section 4.3.2) is a smart and efficient solution, as is drawing the lens sample for parallel coordinates (see Appendix D.2.2). Other highlights include implementing an OpenGL version of the InfoVis Toolkit (see Appendix D.3.2) and the effectiveness of the range of parallel coordinate lenses (demonstrated in Section 4.4), especially the axis lens in adding sampling to the traditional range filter.

Deliverables:

- Sampling Lens implementation published in ACM CHI'05 conference paper [Ellis et al. 04]

**Objective 5:** to devise a clutter reduction taxonomy for information visualisation

Meeting this objective is a major achievement of this work (in addition to auto-sampling), due to the practicality of the taxonomy in helping visualisation designers (as illustrated in Section 2.5) and the perseverance required to achieve this criteria-based classification. As demonstrated in Section 3.5.5 criteria are very important, and its use in constructing the taxonomy has given a much clearer understanding of visualisation techniques.

Deliverables:

- taxonomy for clutter reduction published in Trans. Visualization & Computer Graphics/InfoVis'07 conference paper [Ellis and Dix 07]

**Objective 4:** to evaluate sampling as a clutter reduction technique

Various methods were employed to establish the utility of sampling as a clutter reduction technique. A comparison of sampling with several other techniques was undertaken early on using an initial scatterplot version (described in Section 4.2), and informal feedback were obtained (see Section 4.3.4). A detailed literature review of how researchers evaluated their visualisations suggested that user studies of information visualisations are problematic (see Section 6.1), and this led to a critique of user evaluation studies and the reasons why effective user studies of information visualisations are particularly difficult. An alternative evaluation method was therefore pursued, which resulted in the creation of the Clutter-reduction Taxonomy described in Chapter 3.

Deliverables:

- critique of user evaluation studies for information visualisation published as BELIV'06 workshop paper [Ellis and Dix 06a]

main issue	outcome
▶ how to reduce clutter in large datasets	▶ demonstrated through sampling-based visualisations for scatterplot and parallel coordinates with large datasets (Chapters 4, 5 and 6)
▶ how to maintain display continuity and provide re-sampling	▶ demonstrated an efficient implementation was possible (Chapters 4 and 6) through the z-index method
▶ how to deal with large variation in density across a plot	▶ generated objective 2, focus+context solution
▶ need to evaluate sampling	▶ generated objective 4, evaluation method

**Table 7-4e** Main issues and outcomes for objective 1 - to investigate random sampling as a technique to facilitate interactive exploration of very large datasets

**Objective 1:** to investigate random sampling as a technique to facilitate interactive exploration of large datasets

When this research started there was no real understanding of sampling-based techniques for clutter reduction. Through the development of two very different sampling-based visualisations, namely scatterplots and parallel coordinates, we have seen how clutter reduction through random sampling is possible and more importantly, desirable for the exploration of large datasets visualised. The sampling-based visualisations offer users a simple, yet high resolution dynamic control, a method of checking the reality of the representative sample, the option of global and/or localised clutter reduction using a variety of lenses, and the option of automatically maintaining a reasonable view of the data. Although sampling has not been demonstrated for many visualisations, the concept has been assessed through an analytical evaluation, which has revealed many benefits of the technique. We have also demonstrated how sampling can be combined with other techniques.

## 7.4. Future directions

In this work we have seen random sampling being applied to provide a solution to the problem of clutter reduction for scatterplots and parallel coordinates. We will now consider two areas of research, one based on an extension of this work and the other, in the field of visualising uncertainty, where sampling could offer exciting opportunities.

### 7.4.1. Sampling structured and relational data

One of the advantages of sampling, which differentiates it from filtering, is that it gives the user the freedom to explore a large dataset without prejudgement. With single record data, such as the Portland cars dataset used in this work, we can take a random sample of all the records and obtain a representative set of cars for sale.

There are cases, as described in Section 2.4, where we wish to compare certain attributes of the data items and can therefore apply stratified sampling to take the distribution of that particular attribute into account. For example, with the cars dataset, we may wish to compare each vehicle type but the fact that there are far more SUVs and Sedans might distort our view by hiding less popular vehicle types. We can of course apply sampling to reduce any overplotting but there is the distinct possibility that most, if not all the least popular vehicles will be removed. A solution is to randomly select the same number of cars from each vehicle type (e.g. 50 SUV, 50 Sedan, 50 Minivan) and obtain a better comparison. This is a similar approach to the non-uniform sampling adopted by Bertini and Santucci [Bertini and Santucci 05] but they are sampling based on spatial attributes.





If there are relationships between data items, as with relational data (e.g. actors star in films) or more complex data structures (e.g. network graph), then the way we sample becomes important as it influences the meaning of the resulting displayed data. For instance, considering the actor-film example, we have the option to sample by actor, film or an instance of an actor starring in a film. The distribution of data within a relationship has a marked effect on the result as follows:

**actor** - if a small proportion of the actors star in many films, then we have to deal with the long tail problem<sup>3</sup>. The probability of selecting any actor is the same but the resulting number of films selected may vary considerably, as it depends on whether one of the prolific actors is selected.

**film** - the probability of selecting any film is the same, but the chance of selecting a particular actor is dependent on the number of films that actor is in (i.e. there is more chance of selecting one of the prolific actors).

**actor\_film** - the probability of selecting a particular film is based on the number of actors in that film, likewise the probability of selecting a particular actor is based on the number of films that the actor is in.

This raises several questions for the visualisation designer:

- ▶ Do we let the system decide what to sample based on some algorithm (which may take the distribution of data into account) or allow the user some control?
- ▶ In case the user is allowed some control, what information should we then offer so they can make an informed decision on how to explore a particular dataset?
- ▶ What controls should actually be available to the user to make the selection?

In Section 6.5.1 we considered adding sampling to other types of visualisations, in particular, to acyclic graphs. Again, there are options on what to sample. For example, we saw the effect on the acyclic graph when sampling any node and when sampling just the leaf nodes, and discussed other possibilities.

Rafiei and Curial's work (described in Section 2.3.3) uses random sampling to simplify the display of network graphs and they faced similar problems. In fact they gave the user the option of three modes of sampling - sampling the nodes and join with appropriate edges, sample the edges and add the connected nodes, and a combination of these.

Sampling presents an opportunity to reduce clutter when visualising relational or structured data by reducing the amount of data on display. Sampling also has the flexibility to adapt to different data distributions and relationships within the data. However, the practical implications are uncertain and this provides one of the future challenges for clutter reduction through random sampling.

---

<sup>3</sup> Explored in detail by Hernandez-Campos et al. [Hernandez-Campos et al. 02]



### 7.4.2. Visualising uncertainty

Uncertainty: *the degree to which the lack of knowledge about the amount of error is responsible for hesitancy in accepting results and observations without caution* [Hunter and Goodchild 93].

Uncertainty is an essential issue for both information analysts, who aim to make sense of data and decision makers, who act on the information presented. This can range from personal decisions such as whether or not to take an umbrella, to international decisions involving the military. There are many facets to uncertainty that make its definition difficult. These range from statistical measures (such as accuracy, error, precision, completeness) to measures where the levels of uncertainty may themselves be ill-defined (such as credibility, subjectivity, interrelatedness and provenance). In addition to uncertainty during data acquisition, we also have to be aware of further uncertainty introduced during transformation processes and if visualised, from the visualisation process itself.

Uncertainty visualisation has been recognised as one of the top challenges in visualisation [Lundstrom et al. 07, MacEachren et al. 05]. Indeed, in a book published this year the chapter on Visualising Uncertainty in Natural Hazards states “a prevalent shortcoming in the scientific and information visualisation communities is where data is visualised without any indication of their associated uncertainties” [Pang 08]. There is a good body of work reported in the recent literature on a wide range of techniques for mapping uncertainty onto graphical attributes (e.g. colour, size, transparency, fuzziness), adding graphical elements (e.g. textures, labels, iso-surfaces), using animation (e.g. blinking, motion blur) and even sound and touch. However, geographers, cartographers and medical practitioners dealing with flow and 3D volume visualisations are carrying much of the work out. There has been little work by information visualisation specialists who develop tools to make sense of more abstract, often highly multi-dimensional data, such as discovering fraud in financial transactions or significant patterns in intelligence data.

In this thesis we have already seen the Reality Check re-sampling function assists the user in gauging the reality of patterns or trends made available through sampling. This could be considered as a crude form of visualising uncertainty. Using simple charts such as a histogram or line graph, we could animate the display of a sequence of data samples. Consequently, the deviation between each sample would provide the user with an indication of the uncertainty in the distribution within the data. To achieve a smoother animation, we could adopt the idea of the twinkle Reality Check transitions (Section 4.4.4) where the sample window is moved by adding a new point and removing the oldest point. Imagine a line fitted to a set of scatterplot data – it would wiggle about; the amount of movement being dependent on the errors within the data.



Clearly, there are many issues to be resolved in pursuing this idea of visualising uncertainty. For instance, one might establish the potential of this technique by undertaking a simple micro user study whereby participants estimate the errors in the data using a series of static *sampled* charts and compare this with more traditional means of representing errors, such as error bars and box plots. This could be followed by experiments with animated versions.

The need for visualising uncertainty is obviously apparent. There is plenty of scope due to the many facets of uncertainty that can be shown, and the variety of visualisations that require adaptation. Applying a sampling approach could lead to a novel solution to this pressing problem.

## 7.5. Final remarks

With the increasing size and abundance of datasets the need for techniques to reduce clutter will prevail for many years. With the arrival of gigapixel displays, more data can be displayed at once and although impressive, these devices come with their own user interaction problems [North 08].

In Chapter 2, the idea for a sampling-based visualisation was proposed that laid the groundwork for the implementation of scatterplot and parallel coordinate visualisations and then the Sampling Lens applications described in the ensuing chapters. Certainly, the idea was plausible and the Sampling Lens appears to do an excellent job in reducing clutter with a variety of datasets as demonstrated by the many examples throughout this work. Sampling is also flexible as shown by the novel lenses and the fact that it can be added to existing tools.

Datasets are often a sample of the real world and hence further sampling is a natural operation anyway. But if we are trying to detect an extremely small number of fraudulent transactions within hundreds of million records processed per day, such as credit card transactions [Keim 08] then uniform sampling is clearly not appropriate. However, incorporating sampling in a visual analytics approach might be a solution worth pursuing. Understanding the semantics of the data and the sampling then takes on its own importance.

Some of issues identified in the future work section need to be researched, for example how best to sample relational data whilst keeping the user informed. There are also many opportunities for sampling in visualising uncertainty.

This work led to a novel method for estimating the occlusion of overlapping lines that in itself is very useful for providing auto-sampling for parallel coordinates, but may well find a use in other areas of visualisation, and may also inspire others to try out binomial approximations when faced with difficult problems.

**random** - chosen without method or conscious decision in an indiscriminate, haphazard, erratic or accidental manner<sup>5</sup>

**sample** - a small part or quantity intended to show what the whole is like

Random sampling may be taken literally as above but in the sense of the work here, randomness is in fact very controlled.

---

<sup>5</sup> Based on definitions in the Oxford English dictionary

Finally, the Clutter-reduction Taxonomy is in its early days and its use to visualisation designers is yet to be established. But it has already proven to be useful in assessing sampling and a wide range of other techniques, and has demonstrated that criteria are important when building classifications.

I think we can say that sampling has a future in this data rich digital world.





## References

- Ahlberg et al. 92      Ahlberg, C., Williamson, C., Shneiderman, B. "Dynamic Queries for Information Exploration: An Implementation and Evaluation". *Proc. CHI'92*, Monterey, California, May 1992, ACM Press, pp. 619-626
- Ahlberg and Shneiderman 94      Ahlberg, C., Shneiderman, B. "Visual Information Seeking: Tight Coupling of Dynamic Query Filters with Starfield Displays". *Proc. CHI'94*, Boston, Apr 1994, ACM Press, pp.313-317
- Ahlberg 96      Ahlberg, C. "Spotfire: an information exploration environment". *SIGMOD*, 25(4), Dec 1996, ACM Press, pp.25-29
- Amar and Stasko 04      Amar, R., Stasko, J. "A knowledge task-based framework for design and evaluation of information visualizations". *Proc. InfoVis'04*, Austin, Texas, Oct 2004, IEEE, pp.143-149
- Andrews 06      Andrews, K. "Evaluating Information Visualisations". *Proc. BELIV'06, AVI Workshop, Beyond Time and Errors: Novel Evaluation Methods for Information Visualization*, Venice, Italy, May 2006, ACM Press, pp. 9-14
- Artero et al. 04      Artero, A.O., Ferreira de Oliveira, M.C., Levkowitz, H. "Uncovering Clusters in Crowded Parallel Coordinates Visualizations". *Proc. InfoVis'04*, Austin, Texas, Oct 2004, IEEE, pp.131-136
- Asimov 85      Asimov, D. "The Grand Tour: a tool for viewing multidimensional data". *Sci. Statist. Comput.*, 6, 1985, pp.128-143
- Beaudoin et al. 96      Beaudoin, L., Parent, M-A., Vroomen, L. "Cheops: a compact explorer for complex hierarchies". *Proc. Visualization'96*, San Francisco, Oct 1996, IEEE, pp. 87-92
- Bederson 00      Bederson, B.B. "Fisheye Menus". *Proc. UIST'00*, San Diego, CA, Nov 2000, ACM Press, pp.217-226
- Bederson et al. 02      Bederson, B.B., Shneiderman, B., Wattenberg, M. "Ordered and Quantum Treemaps: Making Effective Use of 2D Space to Display Hierarchies". *Trans. Graphics*, 21(4), Oct 2002, ACM Press, pp.833-854

References

- Bederson et al. 03      Bederson, B.B., Clamage, A., Czerwinski, M.P., Robertson, G.G. "A fisheye calendar interface for PDAs: providing overviews for small displays". *CHI'03 Extended Abstracts*, Florida, Apr 2003, ACM Press, pp. 618-619
- BELIV 06      BELIV '06. *AVI workshop on Beyond Time and Errors: Novel Evaluation Methods for Information Visualization* (eds. Bertini, E.,Plaisant, C.,Santucci, G.), Venice, Italy, 2006, ACM Press
- Bertin 83      Bertin, J. *Semiology of Graphics* (Translation: William J. Berg). University of Wisconsin, 1983
- Bertini and Santucci 04      Bertini, E., Santucci, G. "Quality metrics for 2D scatterplot graphics: automatically reducing visual clutter". *Proc. Smart Graphics'04*, Banff, Canada, 2004, Springer Verlag, pp.77-89
- Bertini and Santucci 05      Bertini, E., Santucci, G. "Improving 2D scatterplots effectiveness through sampling, displacement and user perception". *Proc. IV'05*, London, July 2005, IEEE, pp. 826-834
- Bertini and Santucci 06      Bertini, E., Santucci, G. "Give chance a chance - modeling density to enhance scatter plot quality through random data sampling". *Information Visualization*, 5(2), June 2006, Palgrave, pp.95-110
- Bertini 07      Bertini, E. "A Sampling Approach to Deal with Cluttered Information Visualizations". *Ph.D. thesis*, 2007, University of Rome, "La Sapienza"
- Bier et al. 93      Bier, E.A., Stone, M.C., Pier, K., Buxton, W., DeRose, T.D. "Toolglass and MagicLenses: The See Through Interface". *Proc. SIGGRAPH'93*, Anaheim, CA, 1993, ACM Press, pp.73-80
- Bier et al. 94      Bier, E.A., Stone, M.C., Fishkin, K., Buxton, W., Baudel, T. "A Taxonomy of See-Through Tools". *Proc. CHI'94*, Boston, Apr 1994, ACM Press, pp.358-364
- Brath 97      Brath, R. "Concept Demonstration: Metrics for Effective Information Visualization". *Proc. InfoVis'97*, Phoenix, AZ, Oct 1997, IEEE, pp.108-111
- Breunig et al. 01      Breunig, M.M., Kreigel, H-P., Kroger, P, Sander, J. "Data Bubbles: Quality Preserving Performance Boosting for Hierarchical Clustering". *Proc. SIGMOD'01*, Santa Barbara, CA, May 2001, ACM Press, pp.79-90
- Brodbeck et al. 97      Brodbeck, D., Chalmers, M., Lunzer, A., Cotture, P. "Domesticating Bead: Adapting an Information Visualization System to a Financial Institution". *Proc. InfoVis'97*, Phoenix, AZ, Oct 1997, IEEE, pp.73-80

References

- Card et al. 86 Card, S.K., Moran, T.P., Newell, A. "The model human processor: an engineering model for human performance". In *Handbook of Perception and Human Performance*, 1986, Wiley
- Card and Mackinlay 97 Card, S.K., Mackinlay, J. "The structure of the information visualization design space". *Proc. InfoVis'97*, Phoenix, AZ, Oct 1997, IEEE, pp.92-100
- Card et al. 99 Card, S.K., Mackinlay, J.D., Shneiderman, B. *Readings in Information Visualization: Using Vision to Think* Chapters 1 and 2. Morgan Kaufmann, 1999
- Carpendale et al. 95 Carpendale, M.S.T., Cowperthwaite, D.J., Fracchia, F.D. "3-Dimensional Pliable Surfaces: For the Effective Presentation of Visual information". *Proc. UIST'95*, Pittsburgh, Nov 1995, ACM Press, pp.217-226
- Chalmers 03 Chalmers, M. "Informatics, Architecture and Language". In *Designing Information Spaces: The Social Navigation Approach* (eds. Hook, K., Benyon, D., Munro, A.J.), 2003, Springer Verlag, pp.315-342
- Chaudhuri et al. 99 Chaudhuri, S., Motwani, R., Narasayya, V. "On Random Sampling Over Joins". *Proc. SIGMOD'99*, Philadelphia, 1999, ACM Press, pp.263-274
- Chaudhuri et al. 01 Chaudhuri, S., Das, G., Narasayya, V. "A Robust, Optimization-Based Approach for Approximate Answering of Aggregate Queries". *Proc. SIGMOD'01*, Santa Barbara, CA, May 2001, ACM Press, pp.263-274
- Chen and Liu 03 Chen, K., Liu, L. "A Visual Framework Invites Human into the Clustering Process". *Proc. Scientific and Statistical Database Management*, 2003, IEEE, pp. 97-106
- Chi 00 Chi, E.H. "A Taxonomy of Visualization Techniques using the Data State Reference Model". *Proc. InfoVis'00*, Salt Lake City, Utah, Oct 2000, IEEE, pp. 69-75
- Chuah et al. 95 Chuah, M., Roth, S., Mattis, J., Kolojejchick, J. "SDM: Selective Dynamic Manipulation of Visualizations". *Proc. UIST'95*, 1995, ACM Press, pp.61-70
- Chuah and Roth 96 Chuah, M., Roth, S. "On the Semantics of Interactive Visualization". *Proc. Visualization'96*, San Francisco, Oct 1996, IEEE, pp.29-36
- Cockburn and McKenzie 00 Cockburn, A., McKenzie, B. "An Evaluation of Cone Trees". *BCS HCI*, Sunderland, Sept 2000, Springer Verlag, pp.425-236

References

- Cockburn et al. 06      Cockburn, A., Karlson, A., Bederson, B.B. "A Review of Focus and Context Interfaces". *Tech report HCIL-2006-09, Univ. Maryland, 2006*
- Cooper et al. 06      Cooper, K., de Bruijn, O., Spence, R., Witkowski, M. "A Comparison of Static and Moving Presentation Modes for Image Collections". *Proc. AVI'06, Venice, Italy, May 2006, ACM Press, pp.381-388*
- Cui et al. 06      Cui, Q., Ward, M.O., Rundensteinerand, E.A., Yang, J. "Measuring Data Abstraction Quality in Multiresolution Visualization". *Trans. Visualization and Computer Graphics, Oct 2006, IEEE, pp.709-716*
- Cui 07      Cui, Q. "Measuring Data Abstraction Quality in Multiresolution Visualization". *MSc. thesis, May 2007, Worcester Polytechnic Institute*
- Cutting et al. 92      Cutting, D.R., Karger, D.R., Pedersen, J.O., Tukey, J.W. "Scatter/Gather: A Cluster-based Approach to Browsing Large Document Collections". *Proc. SIGIR'92, Copenhagen, June 1992, ACM Press, pp.318-329*
- de Bruijn and Spence 00      de Bruijn, O., Spence, R. "Rapid serial visual presentation: a space-time trade-off in information presentation". *Proc. AVI'00, Trento, Italy, May 2000, ACM Press, pp.51-60*
- Derthick et al. 03      Derthick, M., Christel, M.G., Hauptmann, A.G., Wactlar, H.D. "Constant Density Displays Using Diversity Sampling". *Proc. InfoVis'03, Seattle, Oct 2003, IEEE, pp.137-144*
- DiBattista et al. 98      DiBattista, G., Eades, T., Tamassia, R., Tollis, I. *Graph drawing: algorithms for the visualisation of graphs*. Prentice Hall, 1998
- Dix and Ellis 02      Dix, A., Ellis, G.P. "by chance: enhancing interaction with large data sets through statistical sampling". *Proc. AVI'02, L'Aquila, Italy, May 2002, ACM Press, pp. 167-176*
- Doan et al. 95      Doan, K., Plaisant, C., Shneiderman, B. "Query Previews in Networked Information Systems". *Tech report CS-TR-3524, Univ. Maryland, Oct 1995*
- Domingo et al. 02      Domingo, C., Gavalda, R., Watanabe, O. "Adaptive Sampling Methods for Scaling Up Knowledge Discovery Algorithms". *Data Mining and Knowledge Discovery, 6(2), Apr 2002, Springer, pp.131-152*
- Eades and Hong 04      Eades, P., Hong, S. "Drawing Graphs". In *Handbook of Data Structures and Applications*, 2004, CRC Press, Inc.

References



- Ellis et al. 94 Ellis, G.P., Finlay, J.E., Pollitt, A.S. "HIBROWSE for Hotels: bridging the gap between user and system views of a database". *Proc. IDS'94 Workshop on User Interfaces to Databases*, Lancaster, Apr 1994, Springer Verlag, pp.45-58
- Ellis and Dix 02 Ellis, G.P., Dix, A. "Density control through random sampling : an architectural perspective". *Proc. IV'02*, London, July 2002, IEEE, pp.82-90
- Ellis and Dix 04a Ellis, G.P., Dix, A. "Quantum Web Fields and Molecular Meanderings: Visualising Web Visitations". *Proc. AVI'04*, Gallipoli, Italy, May 2004, ACM Press, pp.197-200
- Ellis and Dix 04b Ellis, G.P., Dix, A. "Visualising Web Visitations: a probabilistic approach". *Proc. IV'04*, London, July 2004, IEEE, pp.599-604
- Ellis et al. 05 Ellis, G.P., Bertini, E., Dix, A. "The Sampling Lens: Making Sense of Saturated Visualisations". *CHI'05 Extended Abstracts*, Portland, USA, 2005, ACM Press, pp.1351-1354
- Ellis and Dix 06a Ellis, G.P., Dix, A. "An Explorative Analysis of User Evaluation Studies in Information Visualisation". *Proc. BELIV'06, AVI Workshop, Beyond Time and Errors: Novel Evaluation Methods for Information Visualization*, Venice, Italy, May 2006, ACM Press, pp. 1-7
- Ellis and Dix 06b Ellis, G.P., Dix, A. "the plot, the clutter, the sampling and its lens: occlusion measures for automatic clutter reduction". *Proc. AVI'06*, Venice, Italy, May 2006, ACM Press, pp.266-269
- Ellis and Dix 06c Ellis, G.P., Dix, A. "Enabling Automatic Clutter Reduction in Parallel Coordinate Plots". *Trans. Visualization and Computer Graphics*, 12(5), Sept 2006, IEEE, pp.717-723
- Ellis and Dix 07 Ellis, G.P., Dix, A. "A Taxonomy of Clutter Reduction for Information Visualisation". *Trans. Visualization and Computer Graphics*, 13(6), Nov 2007, IEEE, pp. 1216-1223
- Engle et al. 06 Engle, S., Shearer, J., Ogawa, M., Haroz, S., Ma, K-L. "Free Your Data! Cenimation: Visualization for Constrained Displays". *InfoVis'06 Contest*, Baltimore, Oct 2006, ACM Press, <http://jshearer.googlepages.com/cenimation>
- Essa 00 Essa, I.A. "Ubiquitous Sensing for Smart and Aware Environments". *IEEE Personal Communications*, Oct 2000, IEEE

References

- Fekete and Plaisant 99 Fekete, J-D., Plaisant, C. "Excentric Labeling: Dynamic Neighbourhood Labeling for Data Visualization". *Proc. CHI'99*, Pittsburgh, May 1999, ACM Press, pp.512-519
- Fekete and Plaisant 02 Fekete, J-D., Plaisant, C. "Interactive Information Visualization of a Million Items". *Proc. InfoVis'02*, Boston, Oct 2002, IEEE, pp.117-124
- Fekete 04 Fekete, J-D. "The InfoVis Toolkit". *Proc. InfoVis'04*, Austin, Texas, Oct 2004, IEEE, pp.167-174
- Fisher 36 Fisher R. "The use of multiple measurements in taxonomic problems". *Annals of Eugenics*, 7, 1936, Blackwell, pp.179-188
- Fishkin and Stone 95 Fishkin, K., Stone, M.C. "Enhanced Dynamic Queries via Moveable Filters". *Proc. CHI'95*, Denver, May 1995, ACM Press, pp.415-420
- Frank and Timpf 94 Frank, A. , Timpf, S. "Multiple Representations for cartographic objects in a Multi-Scale Tree - An Intelligent Graphical Zoom". *Computers and Graphics*, 18(6), 1994, Elsevier Science, pp.823-829
- Friedrich and Eades 02 Friedrich, C., Eades, P. "Graph Drawing in Motion". *Graph Algorithms and Applications*, 6(3), 2002, pp. 353-370
- Fua et al. 99 Fua, Y-H., Ward, M.O., Rundensteiner, E.A. "Hierarchical Parallel Coordinates for Exploration of Large Datasets". *Proc. Visualization'99*, Los Alamitos, CA, Oct 1999, IEEE, pp.43-50
- Furnas 81 Furnas, G.W. "The FISHEYE View: A New Look at Structured Files". *Technical report, AT&T Bell Laboratories*, 1981, Murray Hill, NJ
- Furnas 86 Furnas, G.W. "Generalized Fisheye Views". *Proc. CHI'86*, Boston, Apr 1986, ACM Press, pp.16-23
- Graham and Kennedy 03 Graham, M., Kennedy, J. "Using Curves to Enhance Parallel Coordinate Visualisations". *Proc. IV'03*, London, July 2003, IEEE, pp.10-16
- Grosjean et al. 02 Grosjean, J., Plaisant, C., Bederson, B. "SpaceTree: Supporting Exploration in Large Node Link Tree, Design Evolution and Empirical Evaluation". *Proc. InfoVis'02*, Boston, Oct 2002, IEEE, pp.57-64
- Guha et al. 98 Guha, S., Rastogi, R., Shim, K. "CURE: An Efficient Clustering Algorithm for Large Databases". *Proc. SIGMOD'98*, Seattle, 1998, ACM Press, pp.73-84
- Healey et al. 95 Healey, C.G., Booth, K.S., Enns, J. "Visualizing Real-Time Multivariate Data Using Preattentive Processing". *Trans. Modeling and Computer Simulation*, 5(3), 1995, ACM Press, pp.190-221

References

- Heer and Robertson 07      Heer, J., Robertson, G.G. "Animated Transitions in Statistical Data Graphics". *Trans. Visualization and Computer Graphics*, 13(6), Nov 2007, IEEE, pp. 1240-1247
- Hendley et al. 95      Hendley, R.J., Drew, N.S., Wood, A.M., Beale, R. "Narcissus: visualizing information". *Proc. InfoVis'95*, Atlanta, GA, Oct 1995, IEEE, pp.90-96
- Henry and Fekete 06      Henry, N., Fekete, J-D. "Evaluating Visual Table Data Understanding". *Proc. BELIV'06, AVI Workshop, Beyond Time and Errors: Novel Evaluation Methods for Information Visualization*, Venice, Italy, May 2006, ACM Press, pp.21-25
- Henry et al. 07      Henry, N., Fekete, J-D., McGuffin, M. "NodeTrix: a Hybrid Visualization". *Trans. Computer Graphics and Visualization*, 13(6), Oct 2007, IEEE, pp.1302-1309
- Herman et al. 00      Herman, I., Melancon, G., Marshall, S. "Graph Visualization and navigation in Information Visualization: A Survey". *Trans. Visualization and Computer Graphics*, 6(1), Jan 2000, IEEE, pp.24-43
- Hernandez et al. 04b      Hernandez-Campos, F., Marron, J.S., Samorodniztky, G., Smith, F.D. "Variable heavy tails in internet traffic". *Perform. Eval.*, 58(2-3), Nov 2004, Elsevier Science, pp.261-284
- Herot 80      Herot, C.F. "Spatial management of data". *Trans. Database Systems*, 5(4), Dec 1980, ACM Press, pp. 493-513
- Hinneburg and Keim 99      Hinneburg, A., Keim, D.A. "Clustering Methods for Large Databases: from the Past to the Future". Tutorial *SIGMOD'99*, Philadelphia, 1999, ACM Press
- Holten 06      Holten, D. "Hierarchical Edge Bundles: Visualization of Adjacency Relations in Hierarchical Data". *Trans. Visualization and Computer Graphics*, 12(5), Oct 2006, IEEE, pp.741-748
- Hunter and Goodchild 93      Hunter, G., Goodchild, M. "Managing uncertainty in spatial databases: Putting theory into practice". *Urban and Regional Information Systems Association*, 5(2), 1993, URISA, pp.55-62
- IVC      IVC. "InfoVis Cyberinfrastructure". July 2007, <http://iv.slis.indiana.edu/sw/index.html#ivcfw>
- IVTK      IVTK. "InfoVis Toolkit". Sept 2007, <http://ivtk.sourceforge.net/>
- Jain et al. 99      Jain, A.K., Murty, M.N., Flynn, P.J. "Data Clustering: A Review". *ACM Computing Surveys*, 31(3), Sept 1999, ACM Press, pp.264-323

References

- Jain and Shneiderman 94 Jain, V., Shneiderman, B. "Data Structures for Dynamic Queries: An Analytical and Experimental Evaluation". *Proc. AVI'94*, Bari, Italy, May 1994, ACM Press, pp. 1-11
- Jerding and Stasko 98 Jerding, D.F., Stasko, J.T. "The Information Mural: A Technique for Displaying and Navigating Large Information Spaces". *Trans. Visualization and Computer Graphics*, 4(3), 1998, IEEE, pp.257-271
- Johansson et al. 06 Johansson, J., Ljung, P., Jern, M., Cooper, M. "Revealing Structure in Visualizations of Dense 2D and 3D Parallel Coordinates". *Information Visualization*, 5, 2006, Palgrave, pp.125-136
- Johnson and Shneiderman 91 Johnson, B., Shneiderman, B. "Tree-maps: a space-filling approach to the visualization of hierarchical information structures". *Proc. Visualization'91*, San Diego, CA, Oct 1991, IEEE, pp.189-194
- Jolliffe 86 Jolliffe, J. *Principal Component Analysis*. Springer Verlag, 1986
- Keim and Kreigal 94 Keim, D.A., Kreigal, H-P. "VisDB: database exploration using multidimensional visualization". *Computer Graphics and Applications*, Sept 1994, IEEE, pp.40-49
- Keim 97 Keim, D.A. "Visual Techniques for Exploring Databases". Tutorial *KDD'97*, Newport Beach, CA, Sept 1997, AAAI Press
- Keim and Herrmann 98 Keim, D.A., Herrmann, A. "The Gridfit Algorithm: An Efficient and Effective Approach to Visualizing Large Amounts of Spatial Data". *Proc. Visualization'98*, Research Triangle Park, NC, Oct 1998, IEEE, pp. 181-188
- Keim 00 Keim, D.A. "Designing Pixel-Oriented Visualization Techniques: Theory and Applications". *Trans. Visualization and Computer Graphics*, 6(1), Mar 2000, IEEE, pp.1-20
- Keim et al. 01 Keim, D.A., Hao, M.C., Dayal, U., Hsu, M. "Pixel bar charts: a visualisation technique for very large multi-attributes data sets". *Information Visualization*, 1(1), Mar 2001, Palgrave, pp.20-34
- Keim et al. 04 Keim, D.A., North, S.C., Panse, C., Sips, C.P.M. "Pixel Based Visual Mining of Geospatial Data". *Computers and Graphics*, 28(3), June 2004, Elsevier Science, pp. 327-344

References





References

- Lamping and Rao 96      Lamping, J., Rao, R. "Visualizing Large Trees Using the Hyperbolic Browser". *Video proc. CHI'96*, Vancouver, Apr 1996, ACM Press, pp.388-389
- LeBlanc et al. 90      LeBlanc, J., Ward, M.O., Wittels, N. "Exploring n-dimensional databases". *Proc. Visualization'90*, San Francisco, CA., Oct 1990, IEEE, pp.230-237
- Lee et al. 01      Lee, J., Podlaseck, M., Schonberg, E., Hoch, R. "Visualization and Analysis of Clickstream Data of Online Stores for Understanding Web Merchandising". *Data Mining and Knowledge Discovery*, 5(1-2), Jan 2001, Springer, pp.59-84
- Leung and Apperley 94      Leung, Y.K., Apperley, M.D. "A Review and Taxonomy of Distortion-Oriented Presentation Techniques". *Trans. Comput.-Hum. Interact.*, 1(2), June 1994, ACM Press, pp.126-160
- Lin 92      Lin, X. "Visualization for the document space". *Proc. Visualization'92*, Boston, Oct 1992, IEEE, pp.274-281
- Lin 97      Lin, X. "Map displays for information retrieval". *American Society for Information Science*, 48(1), 1997, pp.40-54
- Lundstrom et al. 07      Lundstrom, C., Ljung, P., Persson, A., Ynnerman, A. "Uncertainty Visualization in Medical Volume Rendering Using Probabilistic Animation". *Trans. Visualization and Computer Graphics*, 13(6), Nov 2007, IEEE, pp.1648-1655
- MacEachren et al. 05      MacEachren, A.M., Robinson, A., Hopper, S., Gardner, S., Murray, R., Gahegan, M. "Visualizing Geospatial Information Uncertainty: What we know and what we need to know". *Cartographic and Geographic Information Science*, 32(3), 2005, CAGIS, pp.139-160
- Marks et al. 97      Marks, J., Andalman, B., Beardsley, P. A., Freeman, W., Gibson, S., Hodgins, J., Kang, T., Mirtich, B., Pfister, H., Ruml, W., Ryall, K., Seims, J., Shieber, S. "Design galleries: a general approach to setting parameters for computer graphics and animation". *Proc. SIGGRAPH'97*, Los Angeles, 1997, ACM Press, pp. 389-400
- Mead 92      Mead, A. "Review of the development of multidimensional scaling methods". *The Statistician*, 33, 1992, Blackwell, pp.27-35
- Mihalisin et al. 91      Mihalisin, T., Timlin, J., Schwegler, J. "Visualization and analysis of multi-variate data: a technique for all fields". *Proc. Visualization'91*, San Diego, CA, Oct 1991, IEEE, pp.171-178

References

- Miller and Wegman 91 Miller, J.J., Wegman, E.J. "Construction of Line Densities for Parallel Coordinate Plots". *Computing and Graphics in Statistics*, 1991, Springer, pp.107-123
- Morrison et al. 03 Morrison, A., Ross, G., Chalmers, M. "Fast multidimensional scaling through sampling, springs and interpolation". *Information Visualization*, 2(1), Mar 2003, Palgrave, pp.68-77
- Munzner 00 Munzner, T. "Interactive Visualization of Large Graphs and Networks". *PhD. Dissertation*, June 2000, Stanford University
- Murtagh 02 Murtagh, F. "Clustering in Massive Data Sets". In *Handbook of Massive Data Sets*, 2002, Kluwer, pp. 401-545
- Nakayama and Silverman 86 Nakayama, K., Silverman, G.H. "Serial and Parallel Processing of Visual Feature Conjunctions". *Nature*, 320, 1986, NPG, pp.264-265
- NetMap NetMap. "NetMap Link Analysis: making the invisible, visible". Sept 2006, <http://www.netmapanalytics.com/demo.html>
- North 08 North, C. "Big Displays: New Opportunities for Interactive Visualization". Keynote at *GIANT, Int. Workshop on Giga-Pixel Displays and Visual Analytics*, Leeds, Apr 2008, University of Leeds, <https://www.comp.leeds.ac.uk/vvr/giant/north-giant.pdf>
- Novotny and Hauser 06 Novotny, M., Hauser, H. "Outlier-Preserving Focus +Context Visualization in Parallel Coordinates". *Trans. Visualization and Computer Graphics*, 12(5), Sept 2006, IEEE, pp.893-900
- Olken 93 Olken, F. "Random Sampling from Databases". *Ph.D. thesis*, Apr 1993, UC Berkeley
- Palmer and Faloutsos 00 Palmer, C.R., Faloutsos, C. "Density Biased Sampling: An Improved Method for Data Mining and Clustering". *Proc. SIGMOD'00*, Dallas, Texas, May 2000, ACM Press, pp.82-92
- Pang 08 Pang, A. "Risk Assessment, Modeling and Decision Support". In *Visualizing Uncertainty in Natural Hazards* (eds. Ann Bostrom, Steven French and Sara Gottlieb), 2008, Springer
- Peng et al. 04 Peng, W., Ward, M.O., Rundensteiner, E.A. "Clutter Reduction in Multi-Dimensional Data Visualization Using Dimension Reordering". *Proc. InfoVis'04*, Austin, Texas, Oct 2004, IEEE, pp.89-96

References

- Peng 05 Peng, W. "Clutter-Based Dimension Reordering in Multi-Dimensional Data Visualization". *MSc. thesis*, Jan 2005, Worcester Polytechnic Institute
- Piccolo Piccolo. "Visualisation toolkit". July 2007, <http://www.cs.umd.edu/hcil/piccolo/index.shtml>
- Pirolli et al. 96 Pirolli, P., Schank, P., Hearst, M., Diehl, C. "Scatter/Gather browsing communicates the topic structure of a very large text collection". *Proc. CHI'96*, Vancouver, May 1996, ACM Press, pp.213-220
- Plaisant et al. 96 Plaisant, C., Milash, B., Rose, A., Widoff, S., Shneiderman, B. "LifeLines: Visualizing Personal Histories". *Proc. CHI'96*, Vancouver, May 1996, ACM Press, pp.221-227
- Plaisant 04 Plaisant, C. "The Challenge of Information Visualization Evaluation". *Proc. AVI'04*, Gallipoli, Italy, May 2004, ACM Press, pp.109-116
- Porta 06 Porta, M. "Browsing large collections of images through unconventional visualization techniques". *Proc. AVI'06*, Venice, May 2006, ACM Press, pp. 440-444
- Prefuse Prefuse. "Interactive Information Visualization Toolkit". Sept 2007, <http://prefuse.sourceforge.net>
- Purchase 02 Purchase, H.C. "Metrics for Graph Drawing Aesthetics". *Visual Languages and Computing*, 13, 2002, Elsevier Science, pp.501-516
- Rafiei and Curial 05 Rafiei, D., Curial, S. "Effectively Visualizing Large Networks Through Sampling". *Proc. Visualization'05*, Minneapolis, MN, Oct 2005, IEEE, pp.48-55
- Rao and Card 94 Rao, R., Card, S. "The Table Lens: Merging graphical and symbolic representations in an interactive focus + context visualization for tabular information". *Proc. CHI'94*, Boston, Apr 1994, ACM Press, pp.111-117
- Robertson et al. 89 Robertson, G.G., Card, S.K., Mackinlay, J.D. "The Cognitive Co-processor for Interactive User Interfaces". *Proc. UIST'89*, Williamsburg, Virginia, Nov 1989, ACM Press, pp.10-18
- Robertson et al. 91 Robertson, G.G., Mackinlay, J.D., Card, S.K. "Cone Trees: animated 3D visualizations of hierarchical information". *Proc. CHI'91*, New Orleans, Apr 1991, ACM Press, pp.189-194
- Rosenholtz et al. 05 Rosenholtz, R., Li, Y., Mansfield, J., Jin, Z. "Feature Congestion: A Measure of Display Clutter". *Proc. CHI'05*, Portland, Oregon, Apr 2005, ACM Press, pp. 761-770

References



- Salton 94 Salton, G. "Automatic analysis, theme generation and summarization of machine-readable texts". *Science*, 264(5164), 1994, AAAS, pp.1421-1426
- Sarkar et al. 93 Sarkar, M., Snibbe, S.S., Tversky, O.J., Reiss, S.P. "Stretching the rubber sheet: a metaphor for viewing large layouts on small screens". *Proc. UIST'93*, Atlanta, Georgia, Nov 1993, ACM Press, pp.81-91
- Schussman 04 Schussman, G. "Anisotropic Volume Rendering for Extremely Dense, Thin Line Data". *Proc. Visualization'04*, Austin, Texas, Oct 2004, IEEE, pp. 107-114
- Shah and Ramachandran 04 Shah, B., Ramachandran, K.:Raghavan, V. "Storage estimation of multidimensional aggregates in a data warehouse environment". *Proc. Systemics, Cybernetics and Informatics (Vol. 4)*, Orlando, FL., 2004, IIIS, pp. 283-904
- Shneiderman 92 Shneiderman, B. "Tree Visualization with Tree-Maps: A 2-D Space-Filling Approach". *Trans. Graphics*, 11(1), 1992, ACM Press, pp.92-99
- Shneiderman 96 Shneiderman, B. "The Eyes Have It; A Task by Data Type Taxonomy for Information Visualization". *Tech report ISR-TR-96-66*, Univ. Maryland, 1996
- Shneiderman and Plaisant 06 Shneiderman, B., Plaisant, C. "Strategies for Evaluating Information Visualization tools: Multi-dimensional In-depth Long-term Case studies". *Proc. BELIV'06, AVI Workshop, Beyond Time and Errors: Novel Evaluation Methods for Information Visualization*, Venice, Italy, May 2006, ACM Press, pp.38-43
- Siirtola and Raiha 06 Siirtola, H., Raiha, K. "Discussion: Interacting with Parallel Coordinates". *Interact. Comput.*, 18(6), 2006, Elsevier Science, pp.1278-1309
- Skupin 00 Skupin, A. "From Metaphor to Method: Cartographic Perspectives on Information Visualization". *Proc. InfoVis'00*, Salt Lake City, Utah, Oct 2000, IEEE, pp. 91-98
- Spence 01 Spence, R. *Information Visualisation*. Addison-Wesley, 2001
- Spence 02 Spence, R. "Rapid, Serial and Visual: a presentation technique with potential". *Information Visualization*, 1 (1), Mar 2002, Palgrave, pp.13-19
- Spence 07 Spence, R. *Information Visualisation: Design for Interaction* (2nd Edition). Addison-Wesley, 2007

References

- Stasko and Zhang 00 Stasko, J., Zhang, E. "Focus+Context Display and Navigation Techniques for Enhancing Radial, Space-Filling Hierarchy Visualization". *Proc. InfoVis'00*, Salt Lake City, Utah, Oct 2000, IEEE, pp.57-65
- Stasko and Kraemer 00 Stasko, J., Kraemer, E. "An Evaluation of Space-Filling Information Visualizations for Depicting Hierarchical Structures". *Int. J. Human-Computer Studies*, 53, 2000, Academic Press, pp.663-694
- Stasko 06 Stasko, J. "Evaluating Information Visualizations: Issues and Opportunities". *Proc. BELIV'06, AVI Workshop, Beyond Time and Errors: Novel Evaluation Methods for Information Visualization*, Venice, Italy, May 2006, ACM Press, pp.5-7
- Stone et al. 94 Stone, M., Fishkin, K., Bier, E.A. "The Movable Filter as a User Interface Tool". *Proc. CHI'94*, New York, Apr 1994, ACM Press, pp.306-312
- The Natural Building Network "Comparison of Natural Building Techniques : A Partial List of Wall Systems by Michael G. Smith". July 2008, [http://www.naturalbuildingnetwork.org/compare\\_techniques.htm](http://www.naturalbuildingnetwork.org/compare_techniques.htm)
- Tory and Mšller 05 Tory, M., Moller, T. "Evaluating Visualizations: Do Expert Reviews Work?". *Computer Graphics and Applications*, 25(5), 2005, IEEE, pp.8-11
- Trutschl et al. 03 Trutschl, M., Grinstein, G., Cvek, U. "Intelligently Resolving Point Occlusion". *Proc. InfoVis'03*, Seattle, Oct 2003, IEEE, pp.131-136
- Tufte 83 Tufte, E.R. *The visual display of quantitative information*. Graphics Press, Cheshire, CT, USA, 1983
- Tufte 90 Tufte, E.R. *Envisioning Information*. Graphics Press, Cheshire, CT, USA, 1990
- Tweedie et al. 94 Tweedie, L., Spence, R., Williams, D., Bhogal, R. "The Attribute Explorer". *Video proc. CHI'94*, Boston, 1994, ACM Press, pp.435-436
- Tweedie et al. 95 Tweedie, L., Spence, R., Dawkes, H., Su, H. "The Influence Explorer". *Companion Proc. CHI'95*, Denver, Apr 1995, ACM Press, pp.129-130
- Tweedie et al. 96 Tweedie, L., Spence, R., Dawkes, H., Su, H. "Externalizing abstract mathematical models". *Proc. CHI'96*, Vancouver, May 1996, ACM Press, pp.406-412
- Vinson 99 Vinson, N.G. "Design Guidelines for Landmarks to Support Navigation in Virtual Environments". *Proc. CHI'99*, Pittsburgh, May 1999, ACM Press, pp.278-285

References

- Waldeck and Balfanz 04 Waldeck, C., Balfanz, D. "Mobile liquid 2D scatter space (ML2DSS)". *Video proc. IV'04*, London, July 2004, IEEE, pp.494-498
- Ward 94 Ward, M.O. "XmdvTool: integrating multiple methods for visualizing multivariate data". *Proc. Visualization'94*, Washington, Oct 1994, IEEE, pp. 326-333
- Ward 02 Ward, M.O. "A taxonomy of glyph placement strategies for multidimensional data visualization". *Information Visualization*, 1(3-4), 2002, Palgrave, pp.194-210
- Ware 04 Ware, C. *Information Visualization: Perception for Design* (2nd Edition). Morgan Kaufmann, 2004
- Ware 08 Ware, C. *Visual Thinking for Design*. Morgan Kaufmann, 2008
- Wegman and Luo 96 Wegman, E.J., Luo, Q. "High Dimensional Clustering Using Parallel Coordinates and the Grand Tour". *Computing Science and Statistics*, 28, July 1996, Interface, pp.352-360
- Wilkinson et al. 01 Wilkinson, L., Rubin, M., Rope, D., Norton, A. "nViZn: An Algebra-Based Visualization System". *Proc. Smart Graphics*, Hawthorne, NY, Mar 2001, ACM Press, pp. 76-82
- Williamson and Shneiderman 92 Williamson, C., Shneiderman, B. "The Dynamic HomeFinder: Evaluating dynamic queries in a real-estate information exploration system". *Proc. SIGIR'92*, Copenhagen, June 1992, ACM Press, pp.39-346
- Wittenburg et al. 00 Wittenburg, K., Chiyoda, C., Heinrichs, M., Lanning, T. "Browsing Through Rapid-Fire Imaging: Requirements and Industry Initiatives". *Proc. Electronic Imaging 2000: Internet Imaging*, San Jose, CA, Jan 2000, SPIE, pp.48-56
- Wong et al. 03 Wong, N., Carpendale, S., Greenberg, S. "EdgeLens: An Interactive Method for Managing Edge Congestion in Graphs". *Proc. InfoVis'03*, Seattle, Oct 2003, IEEE, pp.51-58
- Wong and Bergeron 94 Wong, P.C., Bergeron, R.D. "30 Years of Multidimensional Multivariate Visualization". *Proc. Scientific Visualization: Overviews, Methodologies & Techniques*, 1994, IEEE, pp.3-33
- Woodruff et al. 98a Woodruff, A., Landay, J., Stonebraker, M. "Constant Information Density in Zoomable Interfaces". *Proc. AVI'98*, L'Aquila, Italy, May 1998, ACM Press, pp. 57-65

References

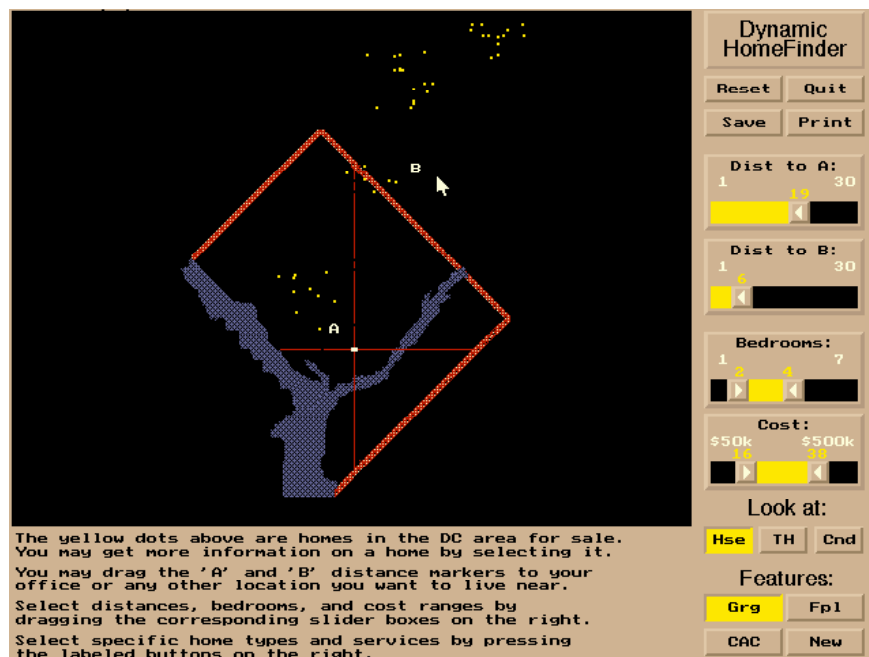
- Woodruff et al. 98b Woodruff, A., Landay, J., Stonebraker, M. "Constant Density Visualizations of Non-Uniform Distributions of Data". *Proc. UIST'98*, San Francisco, Nov 1998, ACM Press, pp.19-28
- Woodruff and Olston 98 Woodruff, A., Olston, C. "Iconification and Omission in Information Exploration". *Proc. CHI'98 Workshop on Innovation and Evaluation in Information Exploration Interfaces*, Los Angeles, Apr 1998, ACM Press
- Woodruff et al. 99 Woodruff, A., Landay, J., Stonebraker, M. "VIDA (Visual Information Density Adjuster)". *CHI'99 Extended Abstracts*, Pittsburgh, Apr 1999, ACM Press, pp.19-20
- Xmdv Xmdv "Xmdv visualization toolkit". July 2008, <http://davis.wpi.edu/~xmdv/>
- Yang et al. 03a Yang, J., Ward, M.O., Rundensteiner, E.A. "Visual hierarchical dimension reduction for exploration of high dimensional datasets". *Proc. Sym. Data visualisation*, Grenoble, France, 2003, Eurographics, pp.19-28
- Yang et al. 03b Yang, J., Ward, M.O., Rundensteiner, E.A., Huang, S. "Interactive hierarchical displays: a general framework for visualization and exploration of large multivariate data sets". *Computers and Graphics*, 27(2), Apr 2003, Elsevier Science, pp.265-283
- Yang et al. 04 Yang, J., Anilkumar, P., Huang, S., Nishant, M., Ward, M.O., Rundensteiner, E.A. "Value and relation display for interactive exploration of high dimensional data sets". *Proc. InfoVis'04*, Austin, Texas, Oct 2004, IEEE, pp.73-80
- Zhang et al. 03 Zhang, L., Tang, C., Shi, Y., Song, Y., Zhang, A., Ramanathan, M. "VizCluster and Its Application on Clustering Gene Expression Data". *Distributed and Parallel Databases*, 13(1), 2003, pp.73-97
- Zuk et al. 06 Zuk, T., Schlesier, L., Neumann, P., Hancock, M.S., Carpendale, S. "Heuristics for Information Visualization Evaluation". *Proc. BELIV'06, AVI Workshop, Beyond Time and Errors: Novel Evaluation Methods for Information Visualization*, Venice, Italy, May 2006, ACM Press, pp.55-60

Table A-1

	clutter reduction technique
appearance	sampling
	filtering
	change point size
	change opacity
	clustering
spatial distortion	point/line displacement
	topological distortion
	space-filling
	pixel-plotting
	dimensional reordering
temporal	animation

Classification of clutter reduction techniques for 2D visualisations used in the Clutter-reduction Taxonomy

Figure A-1



HomeFinder's dynamic query interface [Williamson and Shneiderman 92]



# **Appendix A**

## **Examples of clutter reduction techniques for information visualisation**

The techniques which form the basis of the Clutter-reduction Taxonomy are shown in Table A-1. These have been arranged into three groups, based on how each technique manipulates visual, position, or temporal attributes of the data items in order to reduce display clutter.

Section A.1 describes visualisations which employ one or more of these technique in relation to clutter reduction. These are given in the presentation order of Table A-1. Sampling is not included here as there are many examples in the thesis.

Section A.2 illustrates techniques which are not included in the taxonomy. These include summary statistics and aggregation, dimensional reduction, appearance other than point size and opacity and anisotropic volume rendering.

### **A.1. Clutter-reduction Taxonomy techniques**

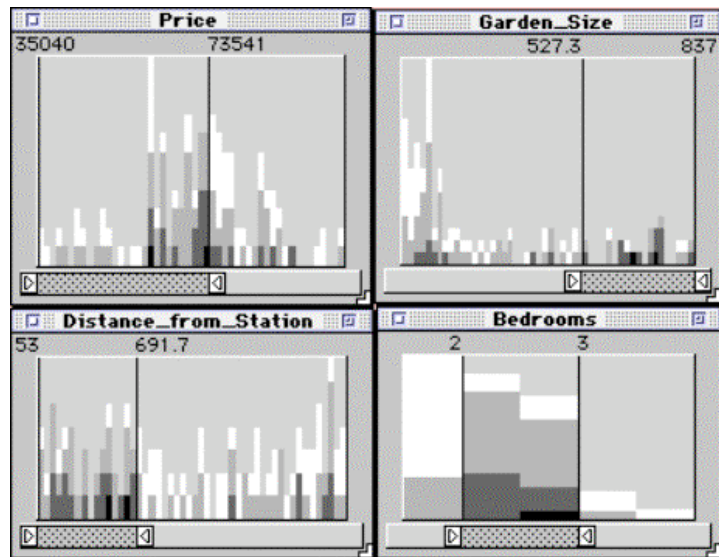
Let us now consider the salient features of each of the clutter reduction techniques listed in Table A-1 through some appropriate example visualisations.

#### **A.1.1. Filtering**

Filtering has been employed widely to deal with cluttered visualisations [Ahlberg et al. 92, Ellis et al. 94, Stone et al. 94] by allowing the user to filter out “uninteresting items” [Shneiderman 96]. Filtering is most effective when the user can change the filter parameters via some interface control and see the effect immediately. This is often referred to as dynamic filtering or dynamic querying.

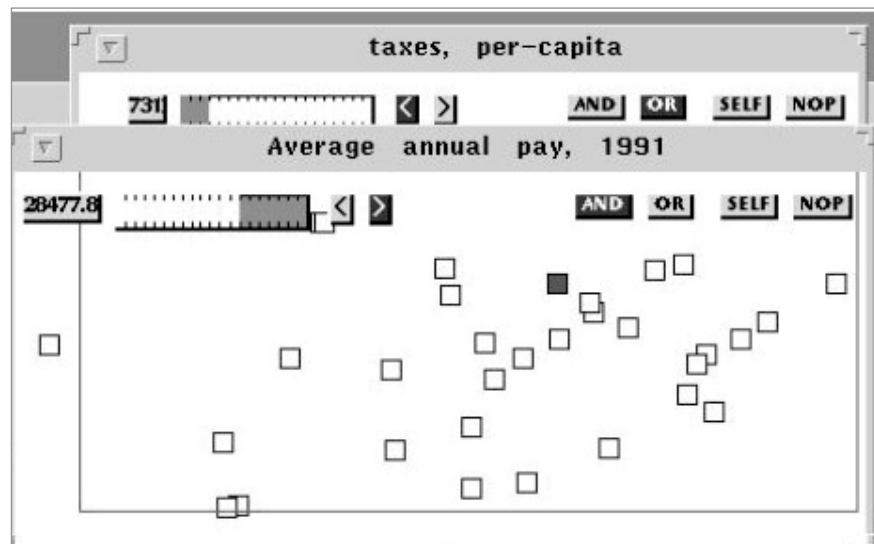
Figure A-1 is a screen shot of Williamson’s HomeFinder [Williamson and Shneiderman 92], one of the earliest dynamic query interfaces. Here, the home buyer or estate agent can set the number of the bedrooms, say 2 to 4, the cost range and other parameter such as the type of dwelling and whether it has a garage or not. The location of houses matching these conditions are shown on the accompanying map. In this example, ‘clutter’ may be due to a very large number of location points on the map, which possibly overlap. Alternatively, clutter may be thought of as those points (homes), which are not of interest to the user and hence are unnecessarily cluttering up the display. In either case, with this method of clutter reduction the user has to decide what data to include or exclude and presupposes that the user actually knows or at

Figure A-2



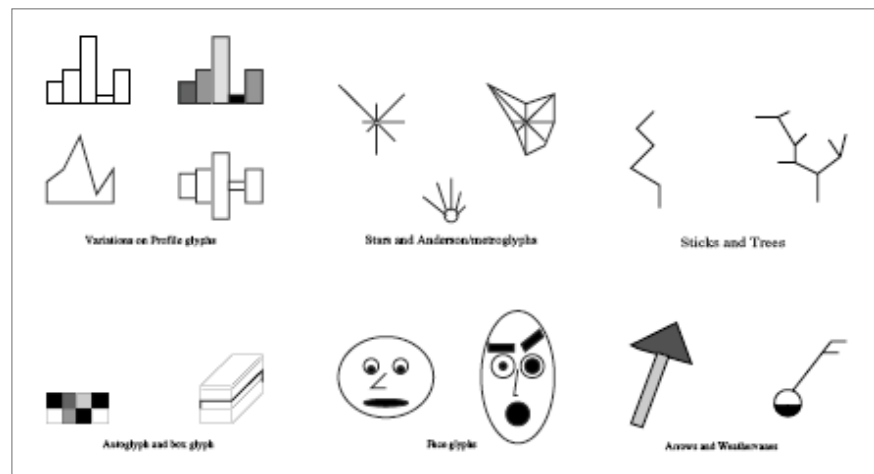
Attribute Explorer [Tweedie et al. 94]

Figure A-3



Enhanced dynamic query filters [Fishkin and Stone 95]

Figure A-4



Examples of multi-attribute glyphs [Ward 2002]

least has some idea of this. Consequently, for an exploratory activity, filtering is not so appropriate since this may unwittingly remove trend or distribution information depending on the filter values. Some systems remove the necessity for the user to decide on what to filtering on, by choosing some form of calculated relevance/importance attribute. For example in Salton [Salton 94], word co-occurrence links between documents are filtered based on the strength of the link so that only the stronger links are shown. This reveals a structure that would be hidden by drawing every single link.

The Attribute Explorer [Tweedie et al. 94], as illustrated in Figure A-2, displays the data as a set of barcharts, one for each attribute. In a similar way to the HomeFinder, the user can filter on these attributes and the homes which match are displayed in black. However, apart from the useful distribution information given by the bar charts (e.g. there are many homes with none or small gardens), the colour coding indicates the degree by which a home misses the selection – dark grey matches all but one attribute range, light grey all but two etc. Hence, in Figure A-2 it is fairly easy to appreciate the amount one would have to change an attribute to increase or in fact decrease the size of the selected set. For example, a small increase in the price would add a fair number whereas doubling the distance to the station only adds a few further homes.

Fekete and Plaisant [Fekete and Plaisant 02] in visualising a million data items, investigated filtering based on the number of overlapping points in a scatterplot. This utilised the graphic processor's stencil buffer to reject data items above or below a specified number of overlaps.

Filtering is often combined with other techniques as part of a visualisation's clutter reduction approach. For example, a combination of filtering and distortion has been employed in the form of dynamic queries via moveable filters [Stone et al. 94, Fishkin and Stone 95] giving the user manual control of the data density on specific areas of the display (see Figure A-3). As a text processing example, the Scatter/Gather browser [Cutting et al. 92, Pirolli et al. 96] uses clustering to group documents, sampling to display representative documents and then filters the document collection based on the users selection.

### **A.1.2. Change point size**

One obvious way to reduce the display space occupied by the data points is to make the points smaller. For instance, the InfoVis Toolkit (IVTK) has a control that changes the point size for scatterplots. This technique may also reduce overlap if the points are not coincident. One disadvantage though is that as the size reduces a data point may loose it ability to convey additional information apart from its location. For example, the colour may not be so obvious [Ware 04] and multi-attribute glyphs, such as those

Figure A-5

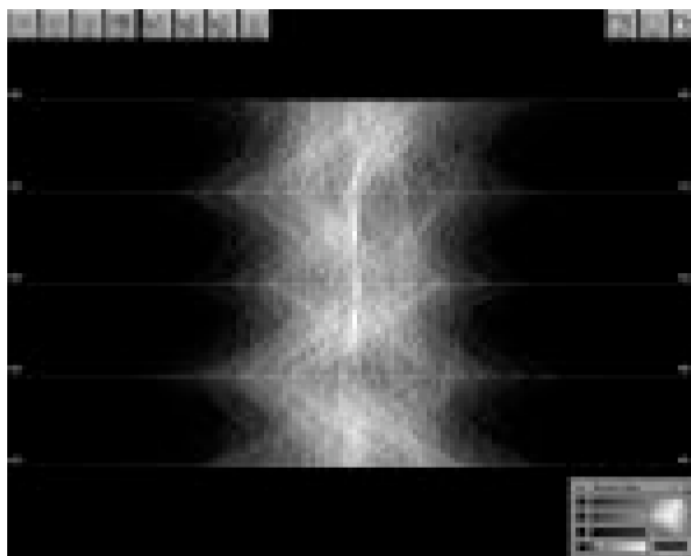


original

VIDA

Visual Information Density Adjuster [Woodruff et al. 99]

Figure A-6



Reducing the opacity of lines in a parallel coordinate plot to produce a density map [Wegman and Luo 96]

shown in Figure A-4, become less feasible.

Semantic zoom makes use of changing the points size, or more precisely the size of the representation, as the user zooms in and out - the higher the zoom factor, the more display space available and hence the larger the point representation (or the more detail which can be added). One of the earliest recorded use of semantic zoom, as noted by Spence [Spence 07] was by Herot [Herot 80] in which he suggests that “a set of semantic data descriptions that could be used to select among alternative graphical interpretations”.

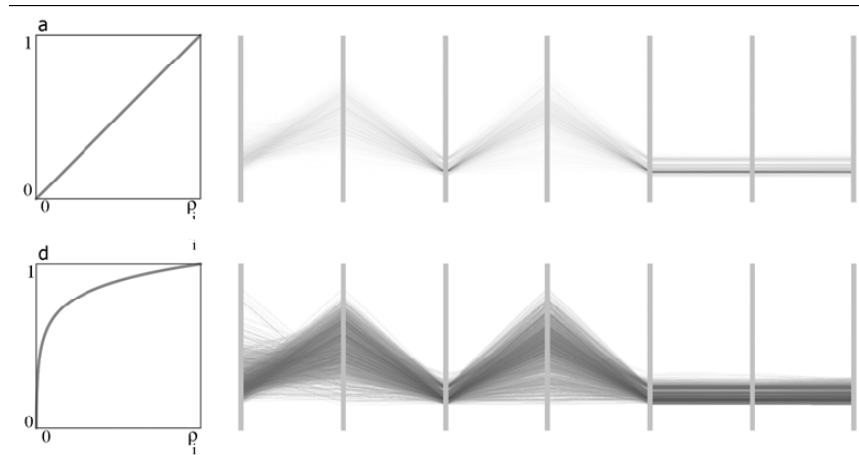
An alternative to altering the point size uniformly across the whole display is *constant density*. This is used by Woodruff et al. [Woodruff et al. 99] with the VIDA (Visual Information Density Adjuster) system and according to the authors is based on the cartographic Principle of Constant Information Density [Skupin 00]. VIDA iconifies or omits each object based on the density of the surrounding region at a given elevation such that objects in more dense areas are drawn with less detail and objects in more sparse areas are drawn with more detail. This is illustrated by Figure A-5 where in the dense regions, the USA states are displayed as dots and in the sparse regions, they are displayed as polygonal outlines.

Care must be taken when interpreting constant density displays as the actual data density is distorted and as in this example, states in less dense areas become more prominent. Other visualisations that make use of constant density displays to some extent are the Digital Library interface (IDVL) [Derthick et al. 03] and LifeLines [Plaisant et al. 96].

### **A.1.3. Change opacity**

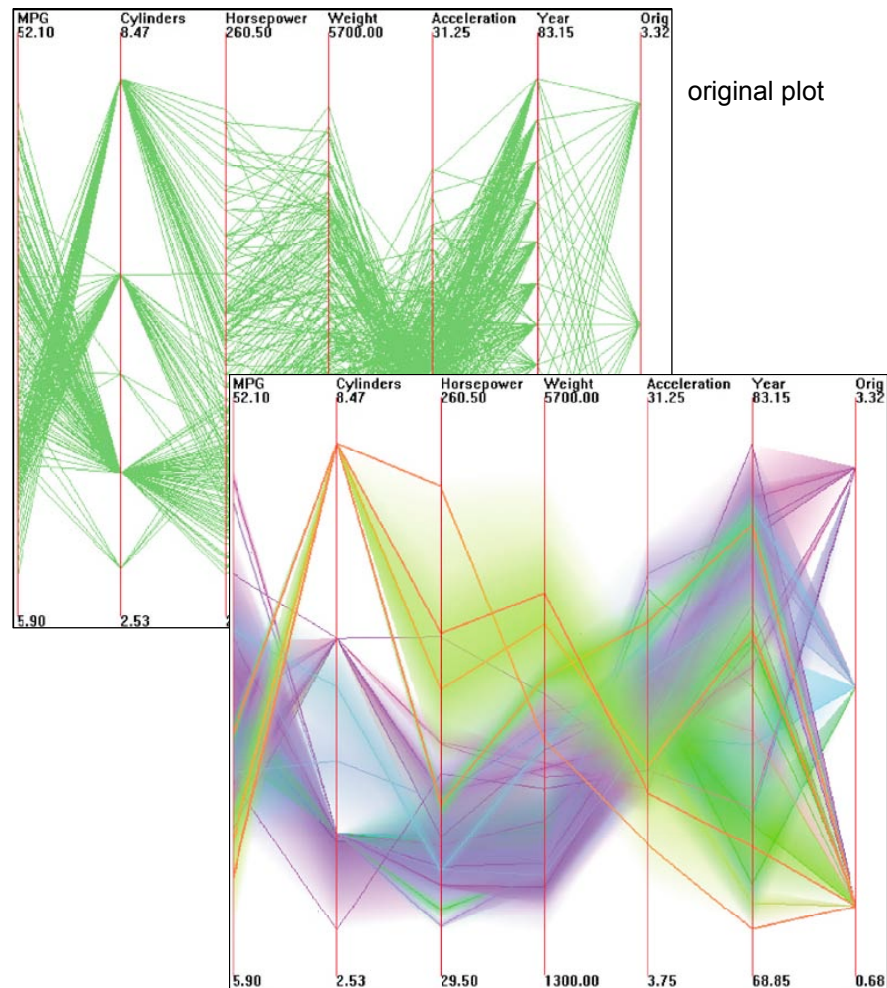
Various other techniques also make use of changes in the appearance of data to reduce screen clutter. Wegman and Luo [Wegman and Luo 96] reduce the opacity (or increase the transparency) of the lines in a parallel coordinate plot to produce a density map and hence identify regions of high overplotting. By itself, opacity does not reduce the number of data items plotted, but can clarify the plot. They suggest that the internal structure of the data is more apparent with a black background (Figure A-6) and that outliers are more prominent when plotting with black on a white background. Fekete and Plaisant [Fekete and Plaisant 02] make good use of the graphics hardware to effectively display up to a million data items. For example, overlap counts from the graphics system can also be mapped to colour or opacity to indicate density of points on a scatterplot. However they point out that opacity is only useful when up to five items overlap as a wide range of shades and blended colours disables pre-attentive processing [Healey et al. 95] and hence the ability to rapidly spot patterns and abnormalities. Fekete and Plaisant experimented with a user control of the range of

Figure A-7



Changing the visibility of structures within a parallel coordinate plot using transfer functions to map line density to opacity [Johansson et al. 06]

Figure A-8



Hierarchical parallel coordinates [Yang et al. 03b] (original plot on the left)

densities displayed and concluded that opacity is only useful when it can be varied interactively to reveal overlaps and is most useful for transient inspections.

Johansson et al. [Johansson et al. 06] map density values to opacity in parallel coordinate data. To achieve rapid interaction they produce high-precision textures of clusters within the plots to represent structures within the dataset, which can be rapidly manipulated with transfer functions (TFs). Figure A-7 shows two parallel coordinate plots of the same data. The upper plot is using a linear TF and just shows the principle structures, whereas the logarithmic TF used in the lower plot enhances low-density regions. A rapid response is accomplished as this manipulation is carried out in the graphics hardware.

#### **A.1.4. Clustering**

Clustering techniques reduce the density of large datasets by grouping the data into clusters of items with some degree of commonality. These clusters can then be treated as individual items to be visualised depending on their average or typical group attributes [Wong and Bergeron 94] or by some description of the cluster (e.g. feature vector). Alternatively, representative members may be selected, as is the case with the Scatter/Gather Browser [Cutting et al. 92, Pirolli et al. 96]. Self Organising Maps [Kohonen 90] are often used in the clustering process. For example, Zhang et al. [Zhang et al. 03] use this technique in VizCluster to reduce the clutter on scatterplots. Artero et al. [Artero et al. 04] also use clustering to reduce visual clutter. Their algorithm uses frequency data, based on counting coincident lines and then smoothes this to produce a density map. This has the effect of grouping lines that are fairly close to each other as well as those that are coincident. The lines are shaded to visually identify those that are in higher density regions.

Another technique is hierarchical clustering [Fua et al. 99, Yang et al. 03b] which constructs a tree of nested clusters of lines, also based on proximity information. One advantage of hierarchical clustering is that the user can decide on the level of detail displayed and with appropriate use of transparency and colouring, the mean (shown as a dark line) and extent (shown as a semi-opaque band) of each cluster can be readily seen (Figure A-8). Note the use of different colours to differentiate the clusters. Another good example of clustering, with user control is hierarchical edge bundling [Holten 06]. The example in Figure A-9 shows how the user can interactively change the bundling strength to go from showing direct node to node information to higher level structures. Colour is used to denote direction and alpha shading helps to differentiate the lines. A noteworthy use of this bundling technique allows the user to select a bundle of particular interest, remove all other lines and then straighten the lines to inspect the detail of the original connections.

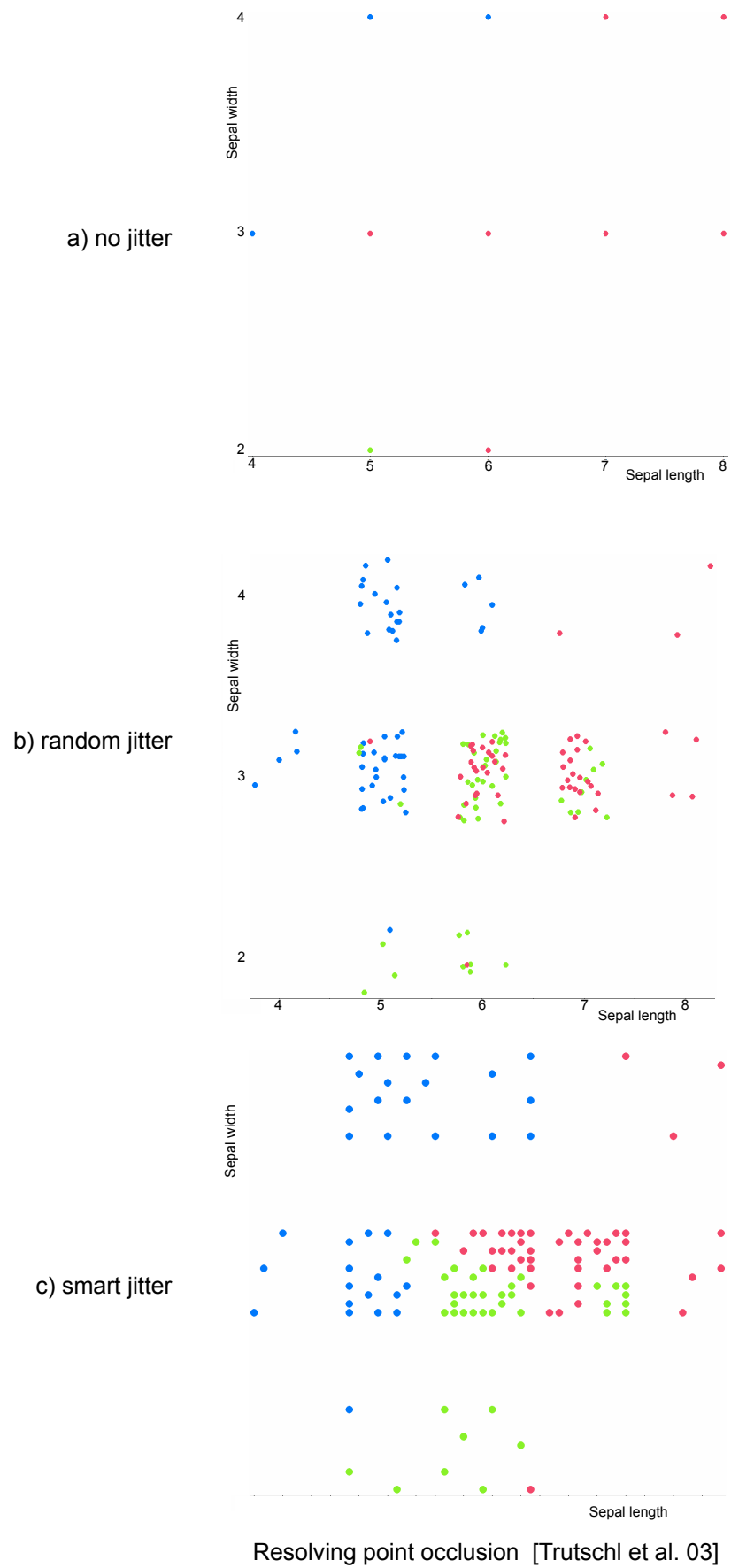
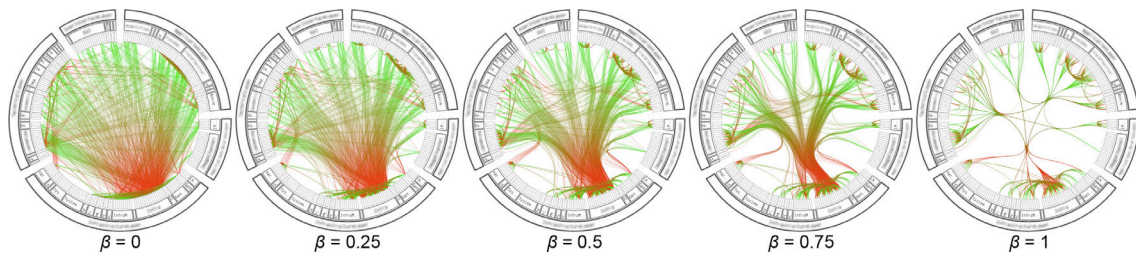


Figure A-10





**Figure A-9** Hierarchical edge bundles [from Holten 06] with bundling strength  $\beta$  increasing from left to right. Low values provide node to node connectivity information whilst higher values reduce clutter as well as providing higher level information. Colour is used to denote direction and alpha shading helps to differentiate the lines.

A cheap and cheerful alternative to hierarchical clustering for parallel coordinates, which has much less processing overhead, is polyline averaging [Siirtola and Raiha 06]. A user selected set of lines is dynamically represented by one average line, with the addition of standard deviation bars on each axis to indicate the extent of the value.

Extensive reviews of data clustering techniques are available [Murtagh 02, Jain and Shneiderman 99] as well as a review of clustering for large databases [Hinneburg and Keim 99].

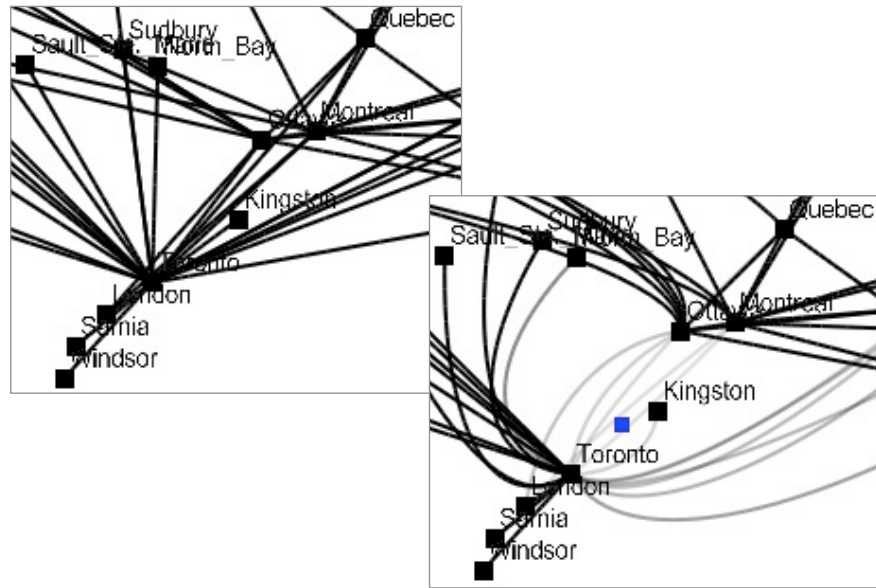
As with all aggregate functions, clustering by its nature hides some of the detail, but has the advantage of reducing the number of data items.

### A.1.5. Displacement

Several applications employ localised point displacement to reduce clutter. The simplest method is random jitter, as used in Spotfire [Ahlberg 96], which displaces overlapping points a random amount (in a random direction). An example of this is given in Figure A-10b. Two other applications, Trutschl's [Trutschl et al. 03] method of intelligently resolving point occlusion and Gridfit [Keim and Herrman 98] use an algorithmic approach to determine where the displaced points are moved to.

Trutschl et al. illustrate their displacement technique with the Fisher Iris Flower dataset [Fisher 36] which has 150 records each of 4 dimensions plus flower type. It is interesting data in that plotting any two dimension does not reveal a distinction between the flower types hence only 11 visible points in Figure A-10a. The next diagram (b) shows the result of applying random jitter whereas diagram (c) is the result of applying this smart jitter and does seem to separate the flower types. They conclude that utilising further dimensions of the data to displace occluded points has the advantage over random jitter of imparting additional information. It does of course rely on their being additional dimensions.

Figure A-11



EdgeLens displaces lines to reveal labels [Wong et al. 03]

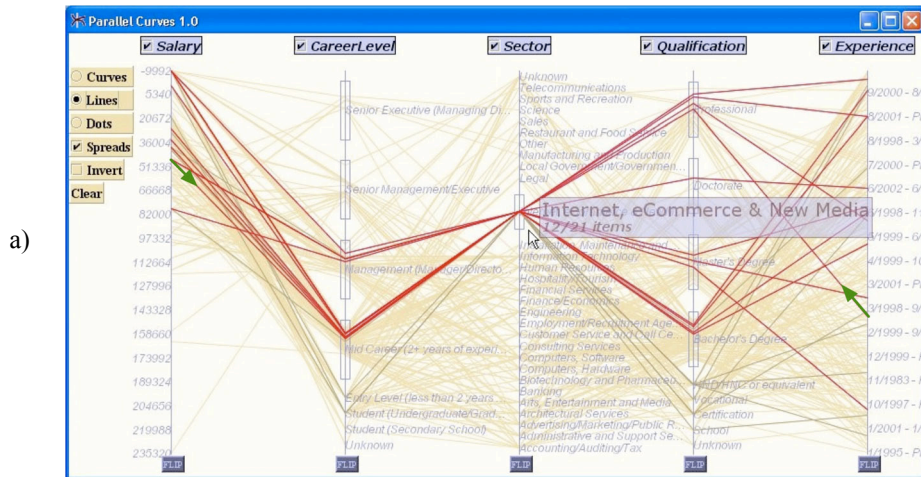
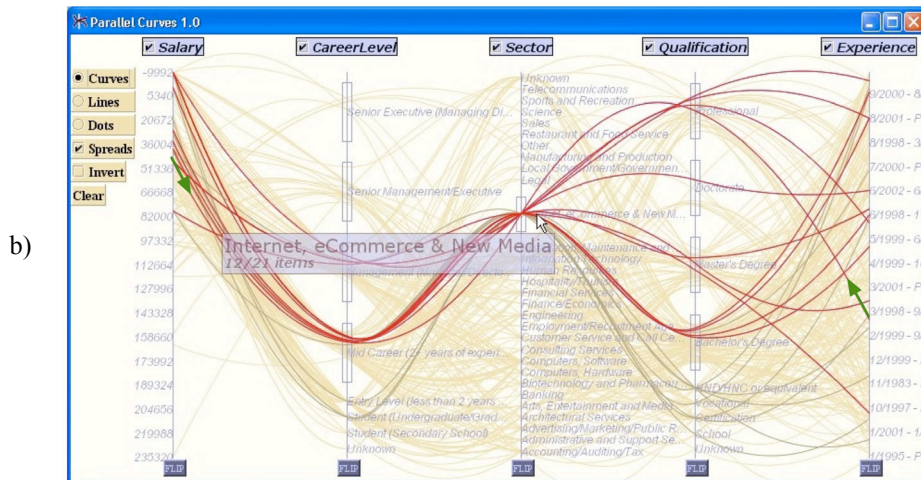


Figure A-12



(a) Parallel coordinate plot. (b) Curving the lines of the plot to help the user follow individual lines [Graham and Kennedy 03]. Note that green arrows mark the ends of the same data record in each diagram.

Gridfit uses a sophisticated approach based on hierarchical partitioning of the available data space to determine where to displace overlapping points. The examples given by Keim and Herrmann [Keim and Herrmann 98] have very large datasets and hence work at a display pixel level. The authors demonstrate that the Gridfit algorithm is efficient and effective, significantly outperforming two other candidate algorithms especially in retaining spatial locality information. However, they do not make a comparison with other clutter reduction techniques and the examples of the Gridfit approach on world map data shows that for high overplotting, the distortion resulting from significant displacement of points may cause a problem for the user.

Mobile Liquid 2D scatter space [Waldeck and Balfanz 04] uses a distance manipulation-based expansion lens, as opposed to a magnification lens, to displace points that are close to each other. The application uses a pressure sensitive pen as the lens and thus the user can press the stylus on a crowded part of the display and nearly overlapping points are separated. This effect is animated, so moving the stylus or changing the pressure results in the points flowing across the screen and hence the “liquid browsing” name coined by the authors. Although the effect is novel and appealing to use, the displacement of points can be fairly large and somewhat random as the points flow around. In addition, this technique does not deal with direct overplotting which is an important consideration.

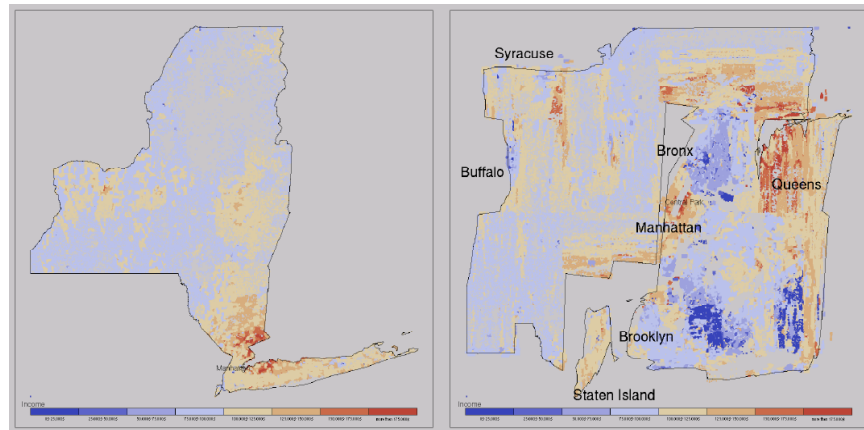
EdgeLens [Wong et al. 03] is a lens technique that interactively curves the lines connecting nodes of a network graph, to reveal information underneath. In the air route example show in Figure A-11 the lines are displaced away from the nodes Kingston and Toronto. The focus of the lens would have to be changed to reveal the airports to the north of Kingston. This example also demonstrates that reducing the opacity of the edges improves the visibility of the text. As well as making more space available for the edges under the lens, curving the lines helps to disambiguate the connected nodes.

A similar technique is used by Graham and Kennedy [Graham and Kennedy 03] to disambiguate the lines in parallel coordinate plots. Where many lines cross at the same point or very close by, as shown in their example in Figure A-12a, it is difficult to follow the path of any one line. However, replacing the polylines with curves makes this task much easier as demonstrated in Figure A-12b. Note that to aid comparison, green arrows have been overlaid to indicate the ends of a particular data record in each plot.

#### **A.1.6. Topological distortion**

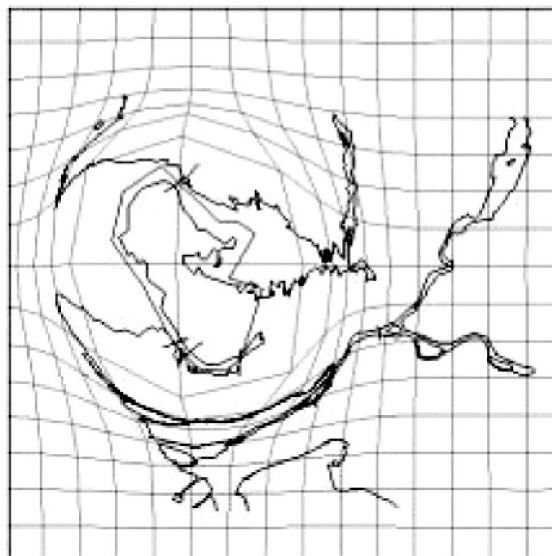
The term topological distortion is used here to represent stretching and the corresponding shrinkage of the underlying 2D space, in as much as topology is

Figure A-13



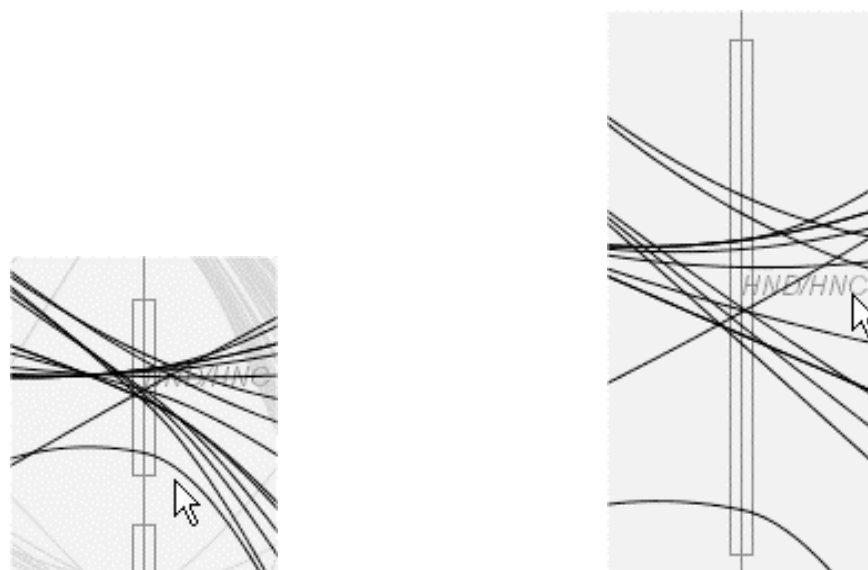
PixelMap distorts the underlying spatial area to avoid overlapping items [Keim et al. 04]

Figure A-14



Carpendale's pliable surfaces indicates degree of distortion

Figure A-15



Topological distortion along axes of a parallel coordinate plot which stretches region with many lines crossing to help disambiguate the paths of the lines. From [Graham and Kennedy 03]

referred to rubber sheet geometry. The relative position of the points are maintained in terms of the direction (i.e. if a point is to the left of another point it will still be to the left after distortion) but the distance between the points may change.

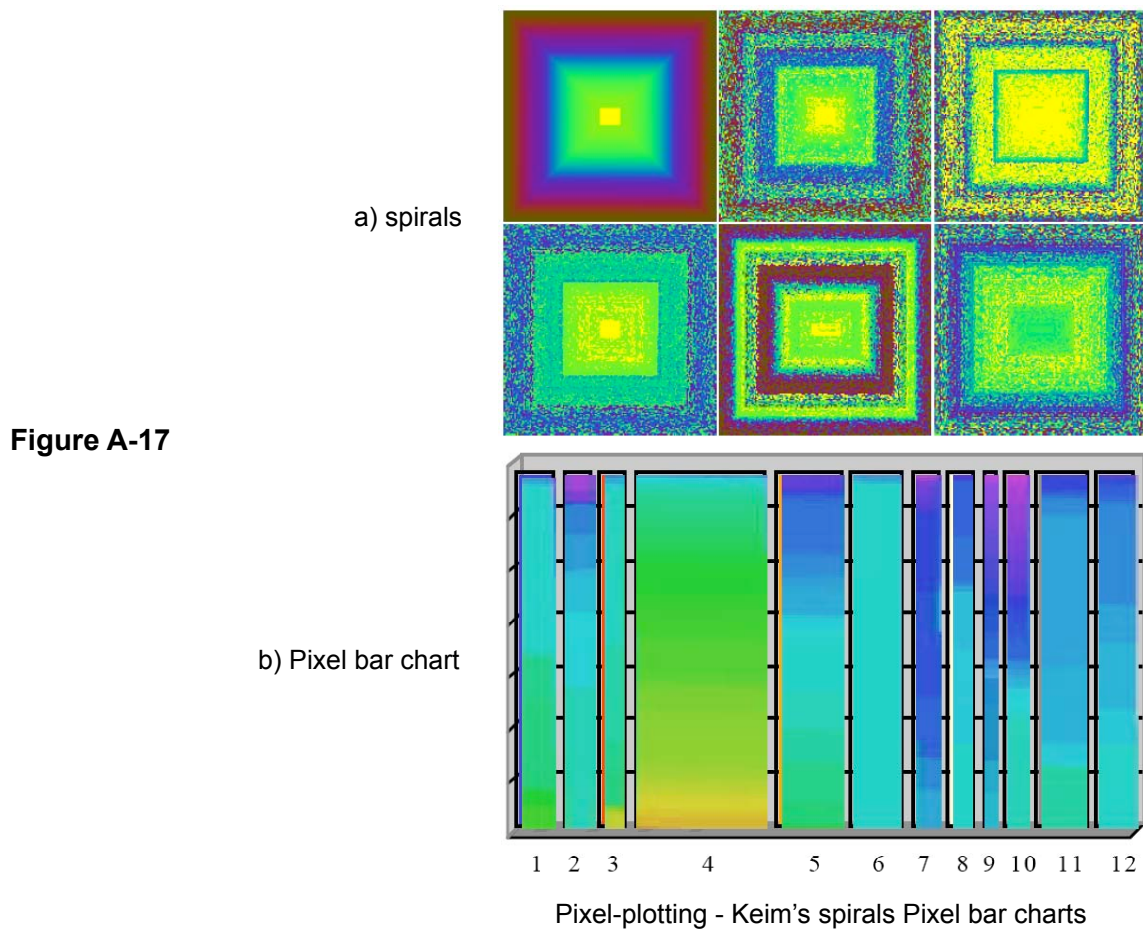
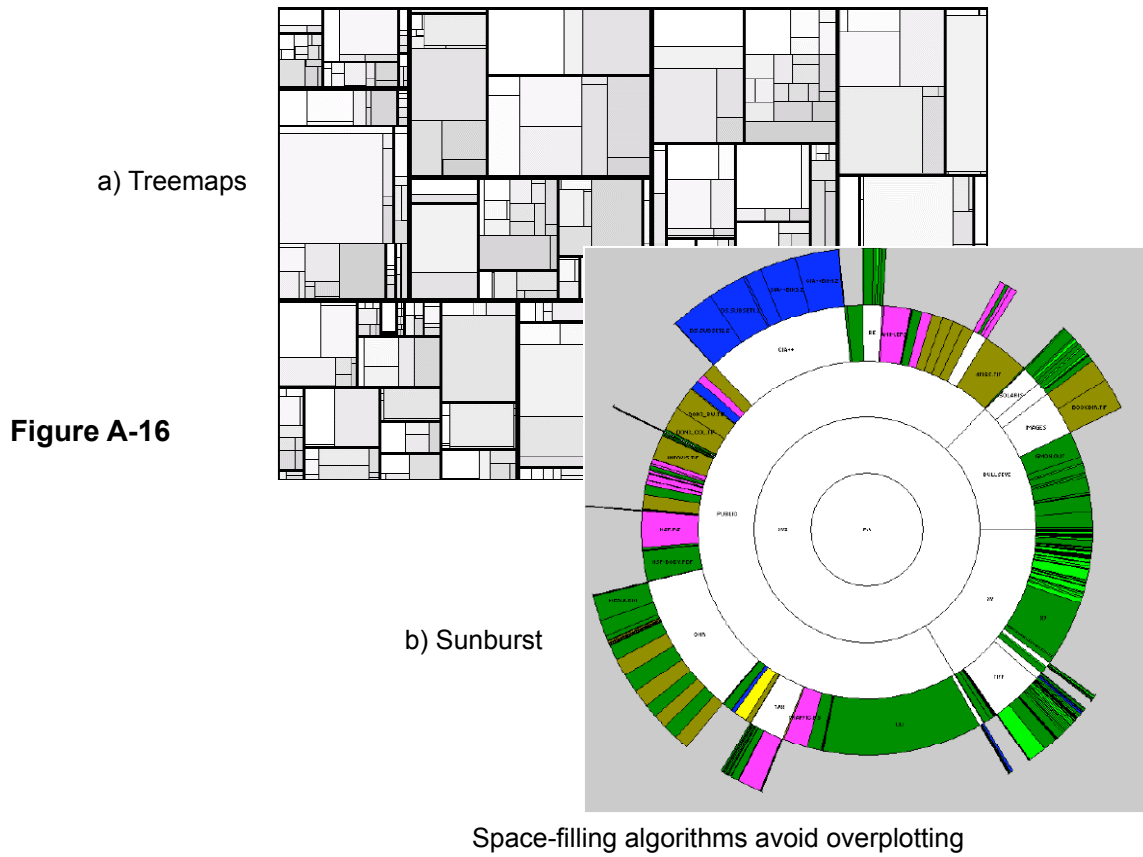
PixelMap [Keim et al. 04] distorts the underlying spatial area to accommodate all the data points and also displaces coincident or overlapping points to avoid overplotting. This technique relies to some extent on users being sufficiently familiar with the map to understand the distortion. Figure A-13 displays household income data for New York state - the left image shows the original map with overlapping data, the PixelMap version is shown on the right. Note that the extra display space given to the very dense regions now allows place names to be added.

Because users often need to see the detail of individual data items, many techniques allow both overview and details to be seen. In focus+context techniques such as Fisheye views [Furnas 86] and the Hyperbolic Browser [Lamping and Rao 96] the display area is distorted so that more screen space is given to a particular area of interest. The density of this area is reduced at the expense of crowding the peripheral display area, however some users may find this type of distortion disorienting. Note that with hierarchical data, tree drawing algorithms using a force-directed layout will spread out the nodes so they occupy free space and hence will not be strictly topological.

As noted, a problem with topological distortion is that the user may have difficulty in realising the extent of the distortion and when plotting spatial data may not even recognise places on the resulting map. Place names and boundary outlines, such as state or country, will provide landmarks which can help the user. Carpendale et al. [Carpendale et al. 95] take another approach with their 3D pliable surfaces. They superimpose grid lines, as illustrated by Figure A-14 or apply shading which emphasises the distortion as a 3D surface.

Zooming in on a plot can be regarded as uniform topological distortion as the 2D surface is stretched and then clipped to the original screen display limits. In terms of clutter reduction, zooming and spatial distortion have the effect of increasing the display space available to a given number of data items, thus spreading out the points. However, this does not affect direct overplotting, but may help with partially overplotted points. Separate overview windows can provide valuable “where are we now” information but they are subject to severe overplotting and some technique to reduce clutter is often necessary.

An example of non-uniform distortion along a particular graph axis is given by Graham and Kennedy [Graham and Kennedy 03] as an addition to their line displacement technique described in the previous section. The idea is to spread out the points on



one or more axes so that crowded crossing points have more space and this helps to disambiguate the lines. As with all topological distortion techniques, coincident crossing points are still coincident<sup>1</sup>. Figure A-15, from their paper, illustrates this technique. Note that regions of the axis with few lines crossing will be shrunk.

Leung and Apperley [Leung and Apperley 94] in their review and taxonomy of distortion-oriented presentation techniques, present a useful comparison of the transformation and magnification functions of some of the techniques, along with the complete mathematical formulas in an appendix. An excellent review and categorisation of focus and context interfaces is presented by Cockburn et al. [Cockburn et al. 06]. In addition to discussing distortion-oriented focus+context techniques such as Fisheye views, overview+detail and cue-based techniques, they interestingly classify zooming as a temporal separation between the focused and contextual views.

### **A.1.7. Space-filling**

Space-filling algorithms avoid overplotting. The commonly used version, Treemap [Shneiderman 92] is designed to present hierarchically structured data using some attribute of each node to determine the size of each rectangle and often setting the colour to represent some other attribute. The early implementations of Treemap tended to be unstable when the data changed, such that the positions of the data items were jumbled. More recent algorithms (e.g. [Bederson et al. 02]) have largely alleviated this drawback by creating ordered layouts. A variation on Treemap is Sunburst [Stasko and Zhang 00] which extends the hierarchy outwards and maintains the order. Regions of the plot can be magnified and displayed outside or inside the circular plot. Example layouts of Treemap and Sunburst are shown in Figure A-16.

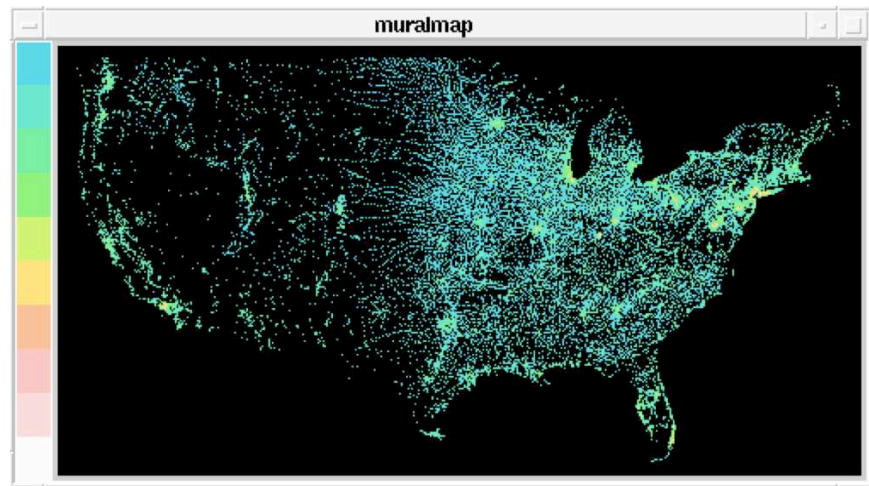
### **A.1.8. Pixel-plotting**

Some techniques use a single coloured pixel to represent the value of each data point, hence reducing the screen real estate needed to display a large dataset to a minimum. Keim [Keim 00] packs the pixels together using various recursive space-filling algorithms to aid the discovery of patterns; separate windows for each dimension allow relationship to be discerned (Figure A-17a), although the user does require some training to appreciate the unusual spatial arrangement. Pixel bar charts [Keim et al. 01] shown in Figure A-17b also use 1 pixel per point but the addition of a spatial mapping, similar to scatterplots, within each bar enables additional attribute values to be effectively represented.

---

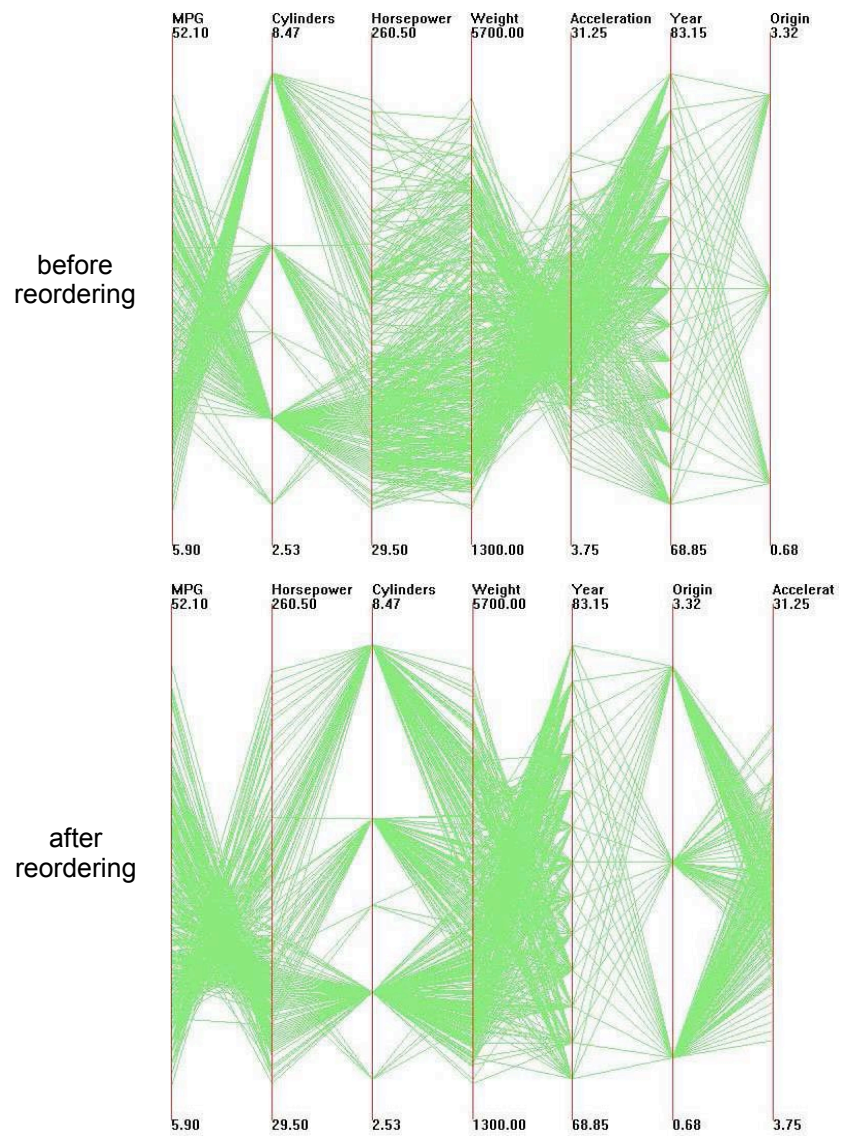
<sup>1</sup> Keim's PixelMap algorithm [Keim et al. 04] does display coincident points to avoid overlap.

Figure A-19



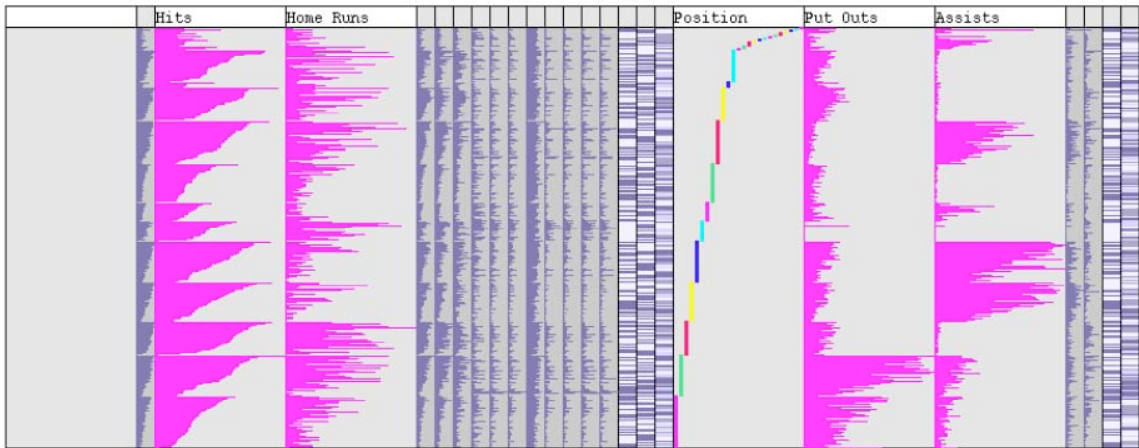
Information Mural utilises the display space by plotting at a pixel level [Jerding and Stasko 98]

Figure A-20



Dimensional reordering in a parallel coordinate plot. [Peng et al. 04]





**Figure A-18** TableLens displays attribute values as pixel bars [Rao and Card 94]

TableLens [Rao and Card 94] (see Figure A-18) takes an adaptive approach by representing numbers as mini-histogram bars when rows in the table are narrowed, although users have to scroll to view more than a few hundred records. Information Mural [Jerding and Stasko 98] adopts pixel-plotting to display very large datasets but if there is more data points than pixels it processes the data using an anti-aliasing compression technique to avoid overplotting. This technique has the advantage of making the overlap density perceptible and can also be used when the colour of data item represents some attribute value. Figure A-19 shows some of the USA 1990 census data plotted as an information mural.

Avoiding overplotting has the obvious benefit of not losing data from view, but the number of data records that can be displayed is limited to the number of pixels that can fit on a screen and the number of attributes per record.

### A.1.9. Dimensional reordering

As an addition to clustering, Peng et al. [Peng et al. 04] utilise dimensional reordering in parallel coordinate plots to minimise the impact from outlier, which they argue obscure any inherent structure from clustered lines. In the example given in Figure A-20, the vertical attribute axes have been re-ordered by the algorithm in an attempt to minimise their clutter measure which is defined as the proportion of outliers against the total number of points. An outlier is defined as one which does not have any neighbours within a threshold distance. The latter has a default value but it can be changed by the user. Peng et al. have also demonstrated this technique to scatterplot matrices, star glyphs and dimensional stacking.

### A.1.10. Animation

There are many techniques which utilise animation. These include rapidly changing images on the display screen to reduce clutter, providing additional information,

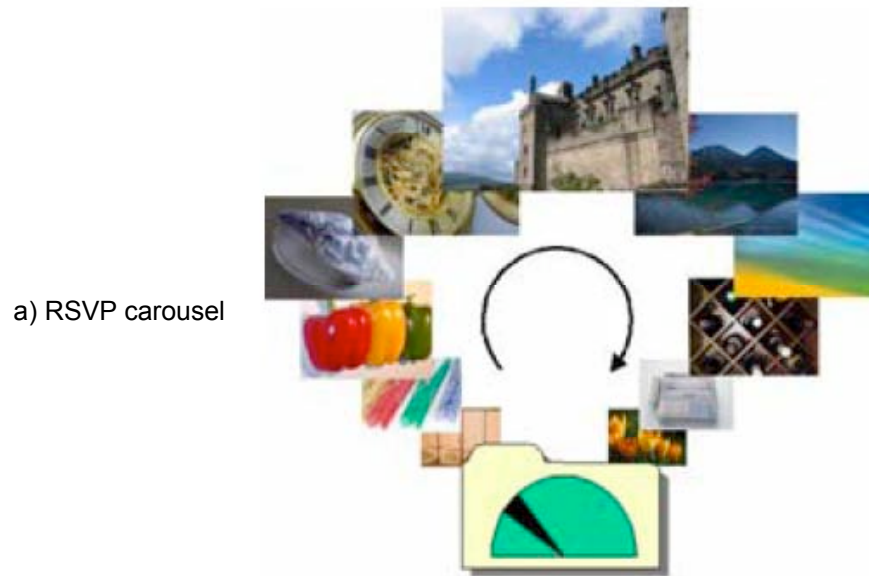
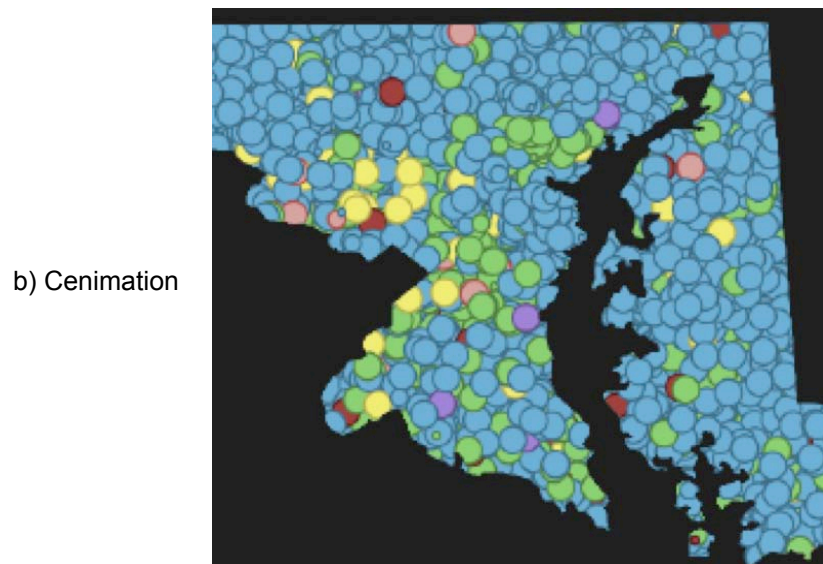
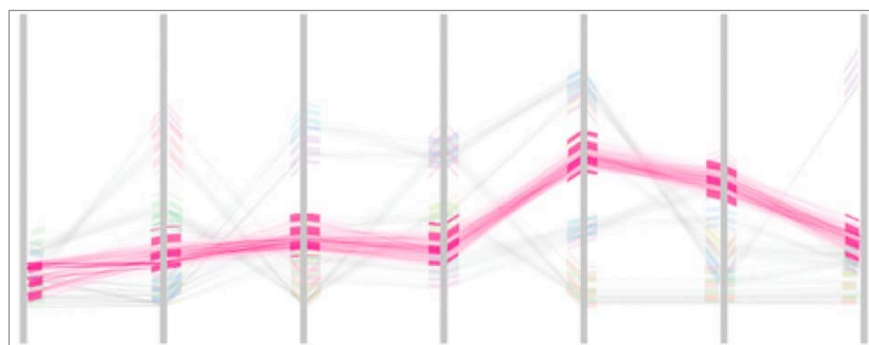


Figure A-21



Animation to avoid overlap a) RSVP carousel [Cooper 06] b) Cenimation [Engle et al. 06]

Figure A-22



Feature animation (pattern of bars where the cluster crosses each attribute axis) imparts information on skewness or variance to a cluster in a parallel coordinate plot. The speed of the pattern movement indicates either the variance of the cluster at that point or the magnitude and direction of the skewness. [Johansson et al. 06]

maintaining a context for the user or as an aid to navigation through the dataset.

The RSVP (Rapid Serial Visual Presentation) image browser [deBruijn and Spence 00, Spence 02, Cooper 06] avoids cluttering the display by showing one or at most a few of the set of images to the user at any one time. Making use of pre-attentive visual processing with which we can recognise particular patterns and colours in a very short time, it is feasible to change the image every 100-200ms and so rapidly view a large number of images in a small screen space. Experiments have been conducted by the RSVP authors comparing different ways of animating the images. The carousel mode, shown in Figure A-21a, moves the images in the selected folder quickly in a circular path, enlarging the image when at the top.

Another visualisation that uses animation is Cenimation [Engle et al. 06] (Figure A-21b shows a screenshot). When animated, data bubbles float to the surface in quick succession - in dense regions this is analogous to a boiling. In both cases, no data item is ever hidden completely although it may take a while for it to appear if there are many images or the plot is very overcrowded.

Some photo viewing applications [e.g. Wittenburg et al. 00] also utilise animation, moving the images along a curve in a 3D the space to give the user a small size preview of the photos about to be shown.

Other animation effects are used to provide additional information. For instance in the Bead visualisation [Brodbeck et al. 97] which uses multidimensional scaling, the layout is animated using *jitter discs* which provide an indication of the stability of the layout using larger coloured discs around those points which have no single low-dimensional configuration. Johansson et al. [Johansson et al. 06] have a technique called *feature animation*, which is used on a parallel coordinate plot to present the skewness or variance of each dimension and cluster by means of a moving pattern of lines, with different phase velocities. The speed of the pattern movement indicates either the variance of the cluster at that point or the magnitude and direction of the skewness. (see Figure A-22).

The so-called Grand Tour, introduced by Asimov [Asimov 85] and a feature of some visualisation toolkits and applications, is a technique for investigating the structure of high-dimensional data using animation. In essence, it is looking at the data from all possible angles and projects the multidimensional data onto a dense set of possible 2D planes and views the resulting smooth movement of the points.

Animation is frequently used to maintain context for the user when the plot is changed. For instance, Ward [Ward 02] recommends smooth animation between original and distorted views and Fekete and Plaisant [Fekete and Plaisant 02] give a good example of animation being used to offer a smooth transition between views.



Note that this often implies animating the movement of landmarks rather than all the data items, in order to speed up the transition. 3D models such as Cone Trees [Robertson et al. 91], one of the early hierarchical data visualisations, allow the user to animate the structure or change the view points in order to see data items which may otherwise be obscured by nearer items. In fact, rotations were animated in Cone Trees in order to allow the human perceptual system to track the changes and avoid the effort to re-assimilate the relationship between substructures. Heer and Robertson [Heer and Robertson 07] propose a taxonomy of transitions between graph types (e.g. bar, pie and scatterplot) and in user studies found that animated transitions between graph types can significantly improve the users understanding.

It has also been found that animation is important for transitions in graph drawing to help users keep a mental model of a changing graph. Friedrich and Eades [Friedrich and Eades 02], in their paper describing the Marey animation module, give a practical overview of how to smoothly transform one graph drawing into another, which includes a set of criteria for animation

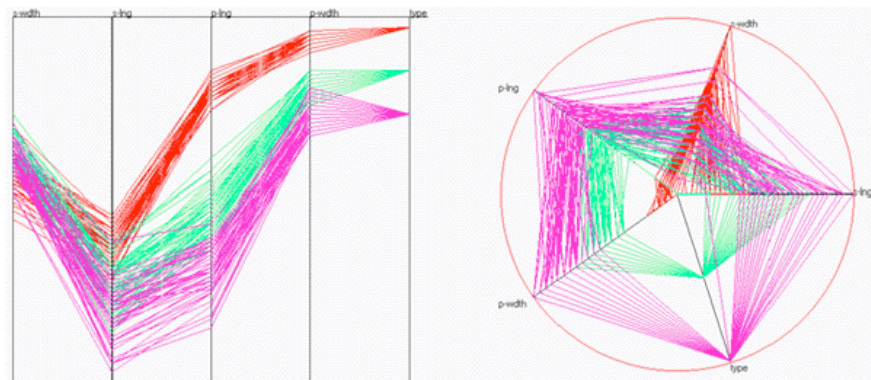
## **A.2. Other techniques**

Let us now look of some techniques that do not feature in Table A-1 as these are more of a pre-processing operation and less suitable for interactive clutter reduction. Nevertheless, these techniques can play a part in reducing display clutter and are described below, using suitable examples.

### **A.2.1. Summary statistics and aggregation**

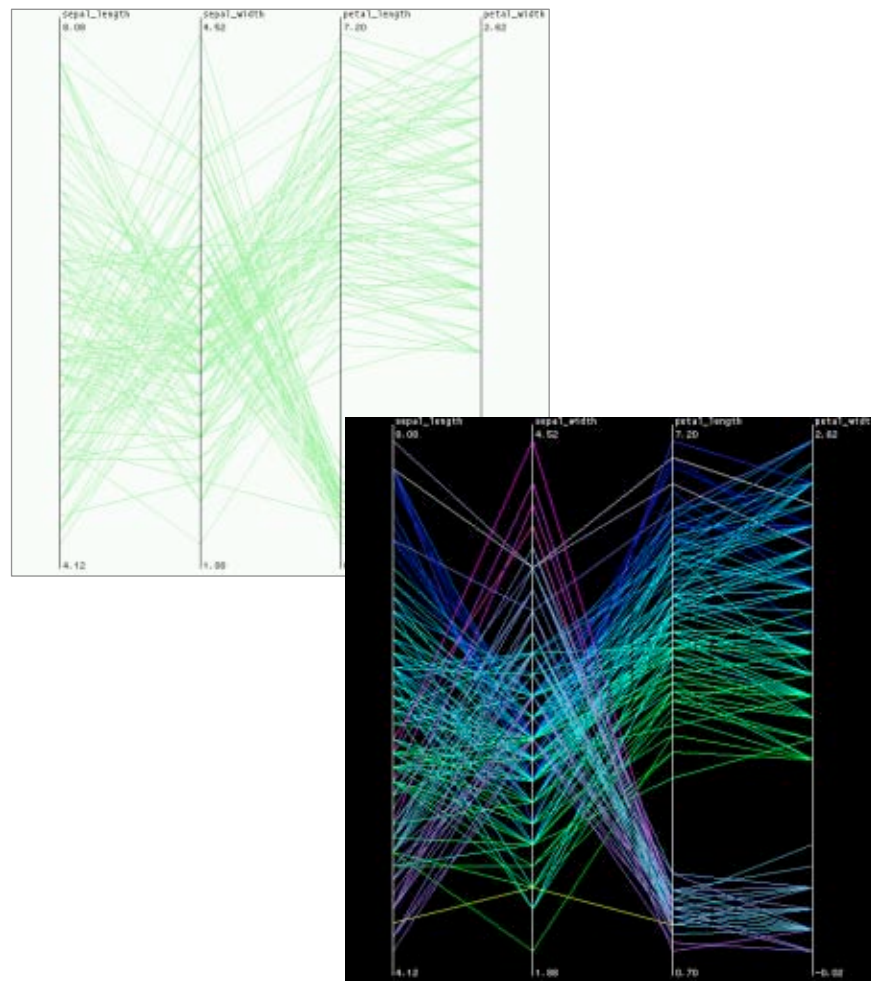
Perhaps the oldest way to deal with large amounts of data is the use of summary statistics, either numerically as tables (e.g. mean, standard deviation) or graphically in the form of histograms and similar graphs. The large number of individual data values is reduced to a few numbers or bars on a chart. Similarly, density, contour plots and maps [Lin 97] reduce data to the number (or some other measure) within a certain area. Attribute Explorer [Tweedie et al. 94] produces histograms so users can see the distribution of data and use brushing to find relationships and utilise the groups (histogram bars) to filter the data. Query previews [Doan et al. 95] also makes use of aggregation of attribute values, but the principle objective in this case is to reduce the data to a manageable amount prior to transmission over a network. The user formulates an approximate query with the help of the aggregate data, details are then downloaded and the query can be refined further. The PDQ-Tree Browser [Kumar et al. 97] provides both an aggregated, hierarchical view of the data set and a tightly coupled detailed view that allows users to prune the trees using various controls. Data judged as irrelevant is therefore removed thus allowing the users to concentrate on the details

Figure A-23



Parallel coordinates and circular parallel coordinates [Zhang et al. 03]. The circular plot can be simplified by zip zooming, which reduces the number of dimensions.

Figure A-24



Proximity-based colouring (right) discriminates lines in the original parallel coordinate plot (left) by automatically assigning different colours to clusters [Fua et al. 99]

of a smaller, hopefully relevant, dataset. nViZn [Wilkinson et al. 01] use aggregation (rectangular or hexagonal binning) to handle datasets with millions of cases.

Although aggregation reduces the amount of data to be displayed, which can help reduce clutter, it is normally a pre-processing operation and as such does not lend itself to be visually interactive. For this reason it has not been included in the set of clutter reduction techniques used in the taxonomy.

### **A.2.2. Dimensional reduction**

Dimensional reduction methods transform high-dimensional data to a dataset with fewer dimensions (which may be as low as two). This reduces the complexity in terms of the number of dimensions to visualise, but it does not reduce the actual number of data items, as with clustering for instance. Common approaches are Principal Component Analysis, PCA (orthogonal linear transform based on variance measure) [Jolliffe 86], Self Organising Maps, SOM (unsupervised learning which produces 2D maps) [Kohonen 90] and Multi-dimensional Scaling (iterative non-linear transform) [Mead 92]. Although these methods are used within some visualisations (e.g. SOM in Trutschl's intelligent jitter (Section A.1.5) and Bead [Brodbeck et al. 97]) they are generally not interactive in terms of clutter reduction and hence have not been included as a major technique.

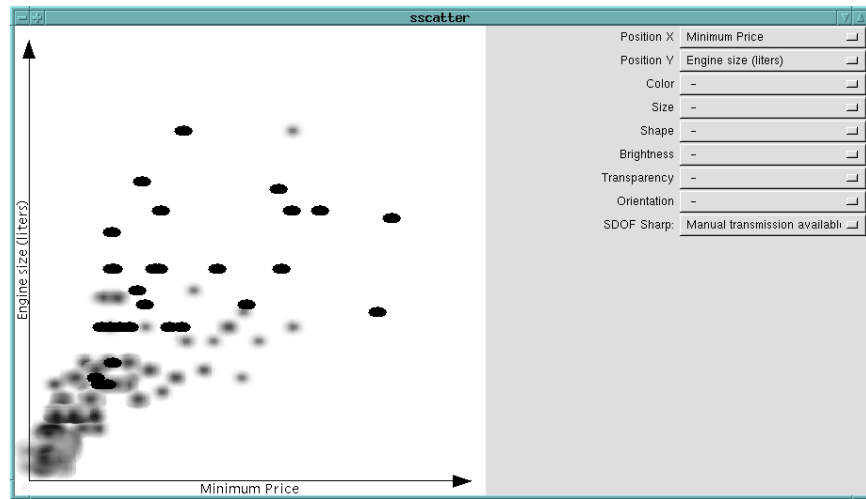
Several other dimensional reduction techniques used with parallel coordinates are worth a mentioning - zip zooming [Zhang et al. 03] and Visual Hierarchical Dimension Reduction, VHDR [Yang et al. 03a]. Zip zooming combines several adjacent dimensions (based on a *coefficient of shape difference*) to reduce the complexity of the plot and displays the data as a circular parallel coordinates (see Figure A-23). On the other hand, VDHR utilises hierarchical clustering, which the authors claim generates more meaningful subspaces than traditional techniques for dimensional reduction. Both of these have an element of interactivity in deciding the degree of dimensional reduction, but not enough to be included.

### **A.2.3. Appearance other than point size and opacity**

Apart from using colour to represent attribute values, some visualisations use colour to highlight particular data items such as those selected by some filtering mechanism or when brushing in multiple view applications. Others apply colour to discriminate between different clusters [Johansson et al. 06, Fua et al. 99]. An example of proximity-based colouring devised for this purpose by Fua et al. is given in Figure A-24.

Kosara et al. [Kosara et al. 02] use blurriness to discriminate between data items. By making some items appear out of focus, those rendered normally will stand out, as can easily be seen in Figure A-25.

Figure A-25



Blurriness can discriminate between points whilst still maintaining context [Kosara et al. 02]



These techniques, although useful for discriminating data items, are fairly specialised and hence have not been included with the more common point/line size and opacity techniques.

#### **A.2.4. Anisotropic Volume Rendering**

Anisotropic Volume Rendering [Schussman 04] reduces the clutter of 3D scientific visualisations consisting of a huge number of lines by converting shaded lines into anisotropic voxels<sup>2</sup>. The authors claim significant speed enhancement, reduction in storage space and good level of detail; this may be a promising approach to try on parallel coordinate plots.

---

<sup>2</sup> Volume elements arranged on a fixed regular grid that define objects in 3D space.

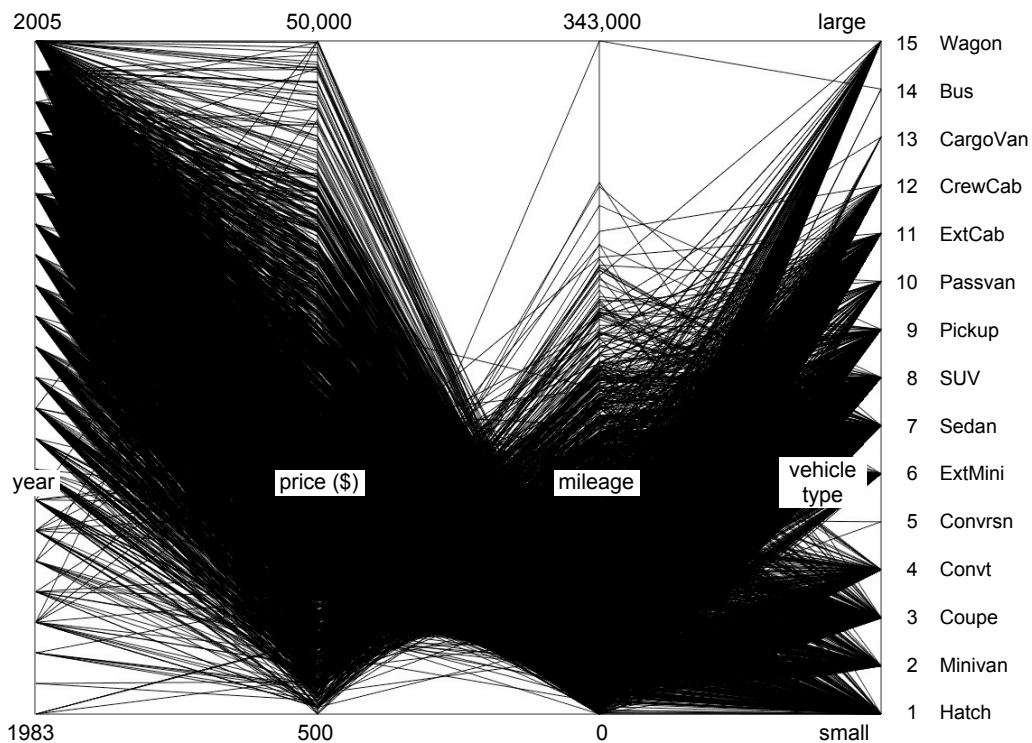


## Appendix B

### Description of datasets used in this work

#### B.1. Portland cars dataset (cars 5k and cars 1k)

The 5850 records contain details of cars for sale within 40 miles of Portland, Oregon, accessed online from <http://www.cars.com> on 31/3/2005. The attributes on the parallel coordinates plots shown in Figure B-1 are, from left to right: year of manufacture, price in US\$, mileage, and vehicle type (given as an integer code). The highest values are at the top of the axes.



**Figure B-1** A parallel coordinate plot of the Portland cars dataset showing the extent of the data.

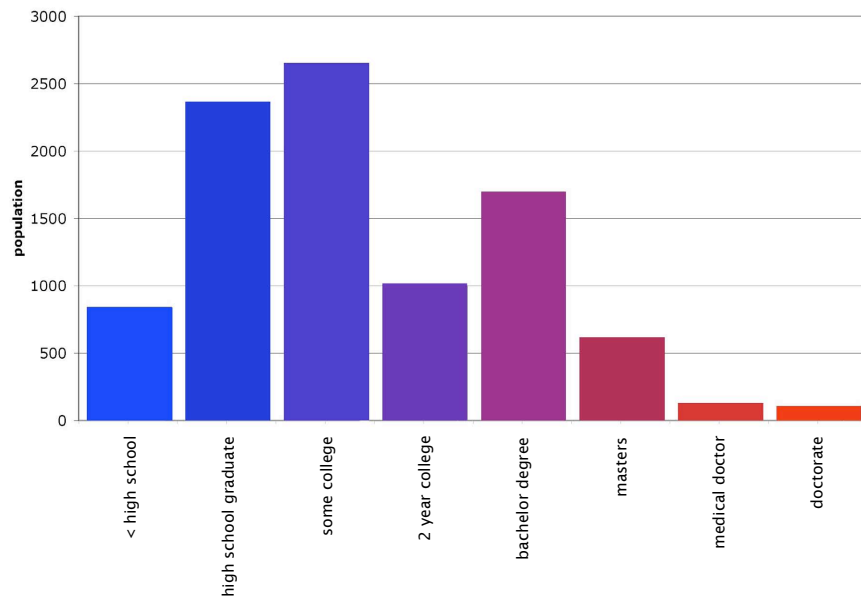
**cars 1k** - A 1000 record randomly selected subset of this data used in many of the experiments

#### B.2. SIPP 2004 dataset

Age, monthly income, gender and educational achievement.

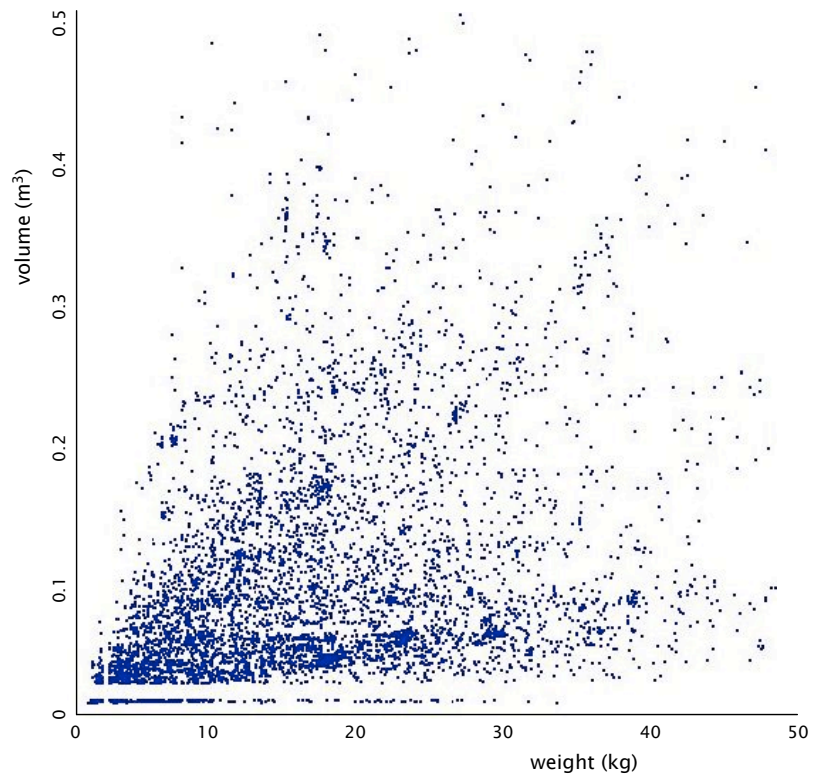
9432 records from the US Census Bureau Survey of Income and Program Participation (SIPP). <http://www.bls.census.gov/sipp/>

Figure B-2



Distribution of educational achievements for SIPP dataset

Figure B-3



Parcel dataset (German post office)

variables:

EEDUCATE - highest education attained

ESEX - gender

TAGE - age last birthday

TPEARN - monthly earned income (\$)

EEDUCATE has been recoded into to 8 classes shown below. Original EEDUCATE values are given in brackets.

0 < high school (31-38)

1 high school graduate (39)

2 some college (40-41)

3 2 year college (43)

4 bachelor degree (44)

5 masters (45)

6 medical doctor (46)

7 doctorate (47)

The distribution of these classes is shown in Figure B-2.

Details of the survey from the SIPP website:

The survey design is a continuous series of national panels, with sample size ranging from approximately 14,000 to 36,700 interviewed households. The duration of each panel ranges from 2 1/2 years to 4 years. SIPP collects source and amount of income, labor force information, program participation and eligibility data, and general demographic characteristics to measure the effectiveness of existing federal, state, and local programs; to estimate future costs and coverage for government programs, such as food stamps; and to provide improved statistics on the distribution of income in the country.

### **B.3. Parcels dataset**

7760 parcels delivered by the German postal service.

Variables: weight (kg), volume (m<sup>3</sup>)

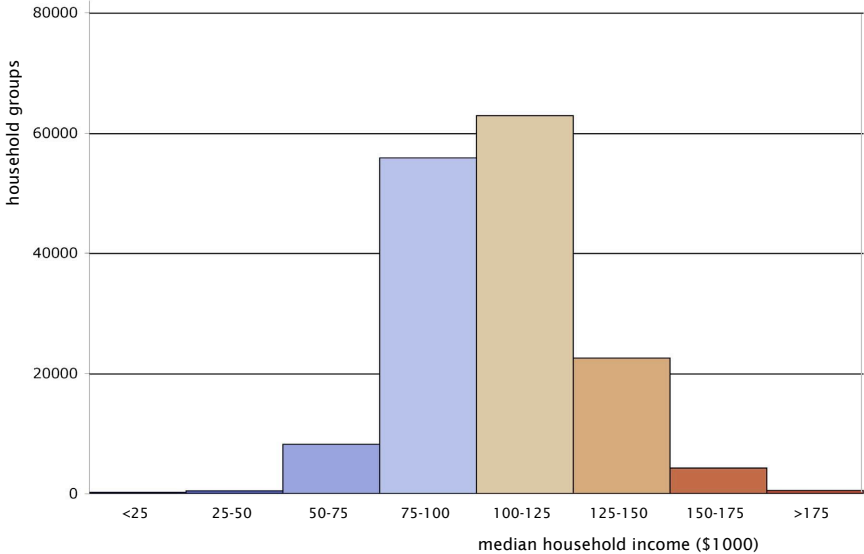
A scatterplot of the data is given in Figure B-3.

### **B.4. Synthetic clustering dataset**

Full dataset contains 17,000 6-dimensional data items with five clusters. The five clusters contain 5,000, 4,000, 3,000, 2,500 and 2,500 data items, respectively, and all have Gaussian distributions. The clusters are deliberately constructed so that they have extensive overlap in several dimensions but can still be separated.

Referenced in [Johansson 06] and obtained from <http://www.itn.liu.se/~jimjo/data/clusterdata.zip>.

Figure B-4



Distribution of median household income for USA census dataset

**Synthetic 5k** is a randomly selected subset of the data.

### **B.5. People dataset**

10,000 records from a US Census. Unknown source.

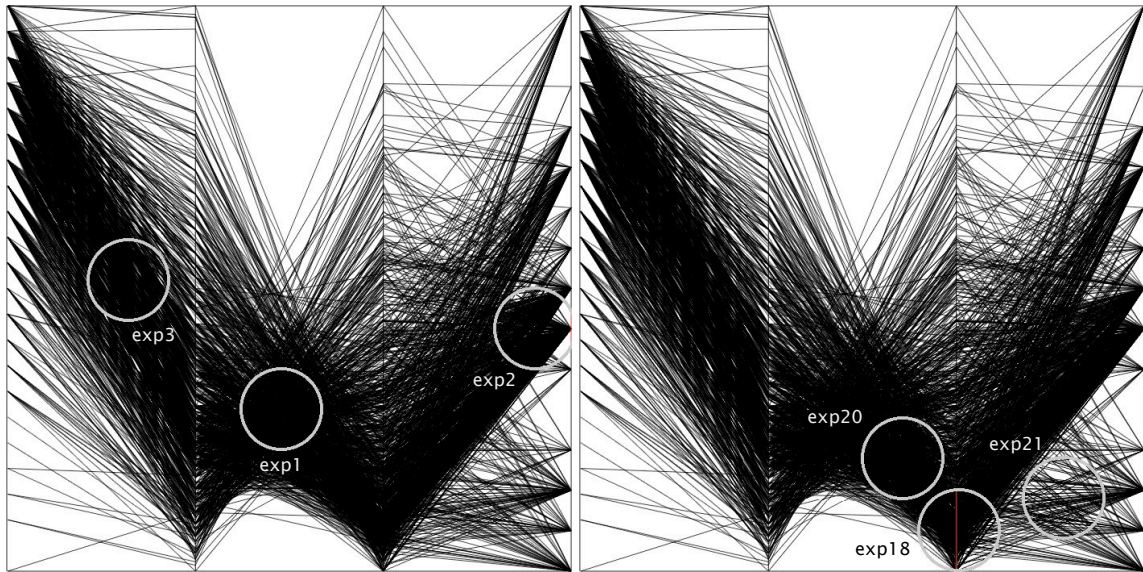
### **B.6. Stockmarket dataset**

Weekly closing share price of the top 1431 stocks in the Dow Jones index for the second half of 2001. Dataset originally packaged with the non-commercial application called TimeSearcher, developed at the University of Maryland.

### **B.7. Household Income dataset**

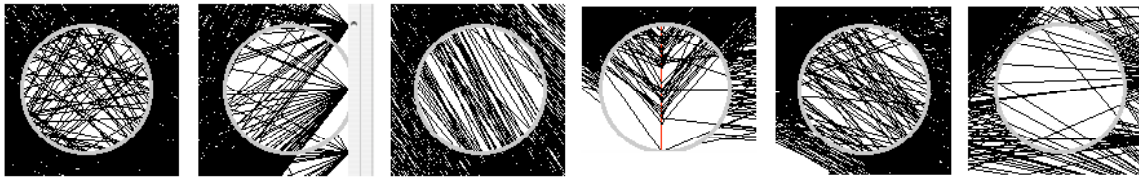
USA Census 2000 data giving the median household income for 155,095 groups of households. The location (latitude, longitude) of each group is specified. <http://www.census.gov>

The distribution of the incomes is given in Figure B-4.



**Figure C-1**

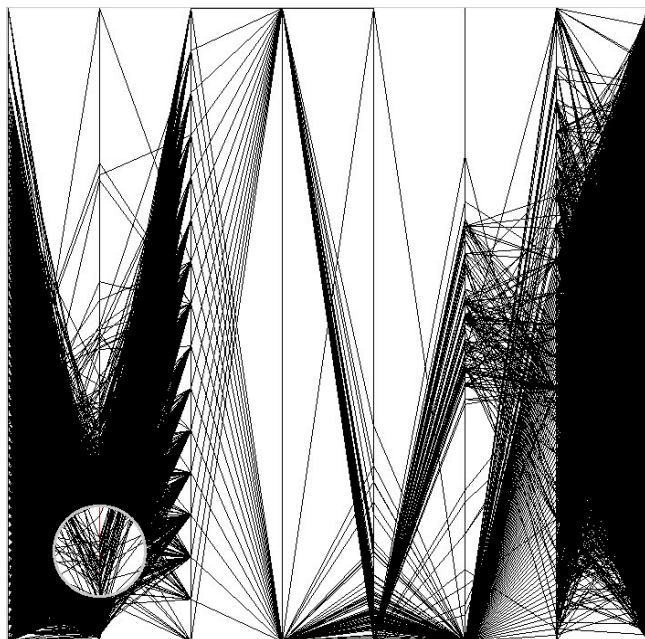
Lens positions for exp1, 2, 3, 18, 20 and 21



**Figure C-2**

Lens at 10% sampling rate for exp1, 2, 3, 18, 20 and 21

**Figure C-3**



Lens position for exp7 to 14 (People 10K dataset)



## Appendix C

### Details of experiments with the parallel coordinate Sampling Lens

This section provides details of experiments conducted to find an effective and efficient way to determine the occlusion of lines with the sampling lens on a parallel coordinate plot. This work is described in Chapter 5.

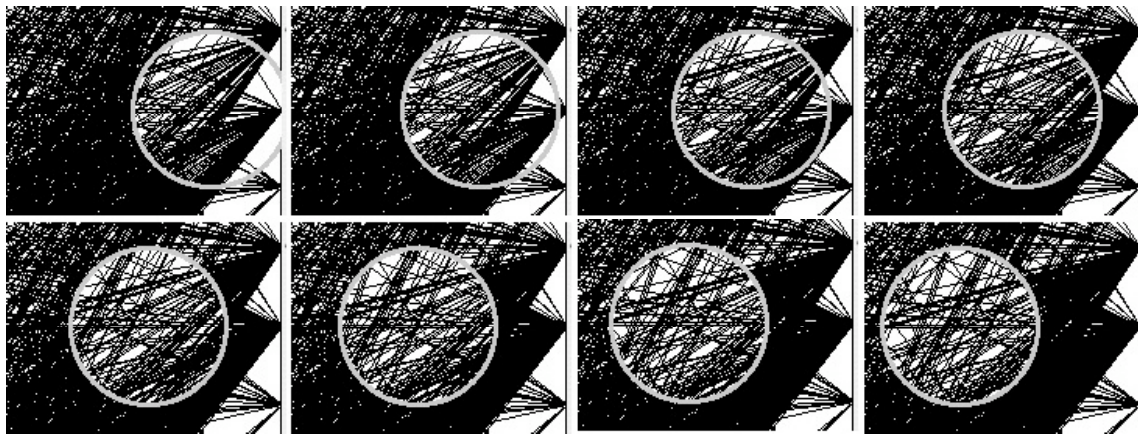
In addition to the tabular data, accompanying figures show the position of the lens on the plots as well as typical patterns of lines within the lens.

#### C.1. exp1 to exp21

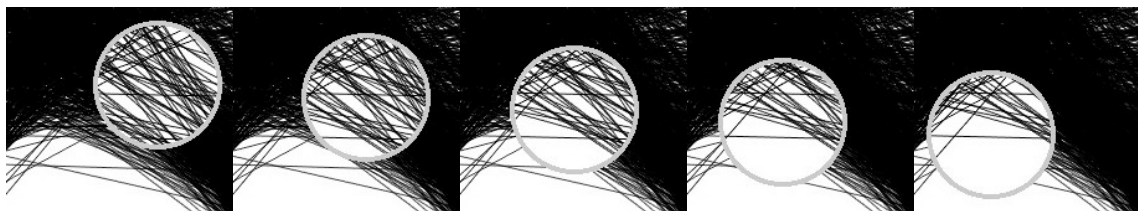
Much of the analysis was performed on exp1, 2 and 3 with exp 18, 20, and 21 providing further data on the behaviour of the occlusion algorithms with different patterns within the lens region. Other experiments, confirmed the findings for larger and different datasets, investigated the effect the non-overlap zone and looked at the Reality Check function. In addition, data was collected on changing the raster cell-width (Section 5.5) in order to speed up the raster calculation.

key: **OSR** - overall sampling rate, **NOZ width** - non overlap zone width (see Section 5.2)

exp	data set	OSR	lens position	screen size	NOZ width	comments
1	cars 1K	100	340,500	701,700	1	lots of criss-crossing
2	cars 1K	100	655,400	701,700	1	to only a few points on zone line
3	cars 1K	100	150,340	701,700	1	low number of overlaps
4	cars 5K	100	340,500	701,700	1	as exp1
5	cars 5K	100	655,400	701,700	1	as exp2
6	cars 5K	100	150,340	701,700	1	as exp3
7	people 10K	100	100,600	700,700	1	very dense region
8	people 10K	50	100,600	700,700	1	as exp7 different OSR
9	people 10K	25	100,600	700,700	1	as exp7 different OSR
10	people 10K	10	100,600	700,700	1	as exp7 different OSR
11	people 10K	10	100,600	700,700	1	as exp10 + Reality Check
12	people 10K	10	100,600	700,700	1	as exp11 + Reality Check
13	people 10K	10	100,600	700,700	1	as exp12 + Reality Check
14	people 10K	10	100,600	700,700	1	as exp13 + Reality Check
15	cars 1K	100	655,400	701,700	0	as exp2 but NOZ=0
16	cars 1K	100	340,500	701,700	0	as exp1 but NOZ=0
17	cars 1K	100	150,340	701,700	0	as exp3 but NOZ=0
18	cars 1K	100	470,650	701,700	1	on zone line, many meeting points



**Figure C-4** Lens positions for exp22, 23, 24, 25, 26, 27, 28, 29 (30% lens sampling rate)



**Figure C-5** Lens positions for exp30 to 34 (10% lens sampling rate)

19	cars 1K	100	470,650	701,700	0	as exp18 but NOZ=0
20	cars 1K	100	400,560	701,700	1	lots of criss-crossing
21	cars 1K	100	600,610	701,700	1	low density, fair amount criss-crossing

Note: For all experiments, cell widths of 20, 16, 12, 10, 8, 6, 5, 4, 3, 2, 1 and for each cell width, lens sampling rates of 100, 95, 90, 85, 80, 75, 70, 65, 60, 55, 50, 45, 40, 35, 30, 28, 26, 24, 22, 20, 19, 18, 17, 16, 15, 14, 13, 12, 11, 10, 9, 8, 7, 6, 5, 4, 3, 2, 1. Lens radius = 50

**Table C-1** Details for experiments 1 to 21

## C.2. exp22 to 29

Investigating the effect of uneven density across the lens. Note that the data from the next set of experiments (exp30 to 34) were found to be more applicable and hence the results of these experiments have not been reported.

exp	data set	OSR	lens position	screen size	NOZ width	comments
22	cars 1K	100	655,400	700,700	0	top left picture
23	cars 1K	100	645,400	700,700	0	
24	cars 1K	100	635,400	700,700	0	
25	cars 1K	100	625,400	700,700	0	top right picture
26	cars 1K	100	615,400	700,700	0	bottom left picture
27	cars 1K	100	605,400	700,700	0	
28	cars 1K	100	595,400	700,700	0	
29	cars 1K	100	585,400	700,700	0	bottom right picture

Note: Lens radius = 50, cell width = 1, max overlap lines = 1000

**Table C-2** Details for experiments 22 to 29

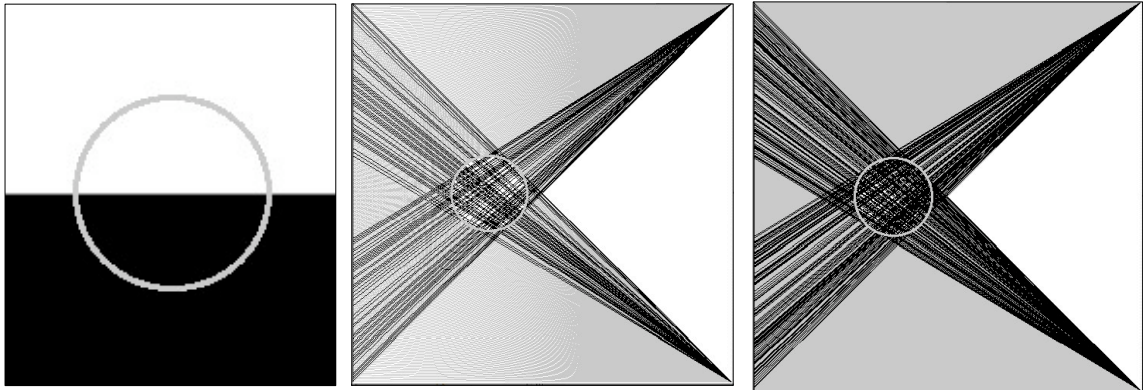
## C.3. exp30 to exp34

Investigating the effect of uneven density across the lens.

exp	data set	OSR	lens position	screen size	NOZ width	comments
30	cars 1K	100	395,565	700,700	0	left end screen shot
31	cars 1K	100	380,575	700,700	0	
32	cars 1K	100	365,585	700,700	0	
33	cars 1K	100	350,595	700,700	0	
34	cars 1K	100	335,605	700,700	0	right end screen shot

Note: Lens radius = 50, cell width = 1, max overlap lines = 1000

**Table C-3** Details for experiments 30 to 34



**Figure C-6** Screen shots for synthetic data experiments 42, 43 and 44

### C.4. exp35 to exp41

Investigation into binning (Section 5.6)

exp	data set	OSR	lens position	screen size	NOZ width	comments
35	cars 1K	100	395,565	700,700	0	as exp30
36	cars 1K	100	380,575	700,700	0	
37	cars 1K	100	365,585	700,700	0	
38	cars 1K	100	350,595	700,700	0	
39	cars 1K	100	335,605	700,700	0	as exp34
40	cars 1K	100	655,400	700,700	0	as exp2
41	cars 1K	100	150,340	700,700	0	as exp3

Note: Lens radius = 50, cell width = 1, max overlap lines = 1000

**Table C-4** Details for experiments 35 to 41

### C.5. exp42 to exp44

Investigation into binning using two very different synthetic datasets. The dense-parallel dataset (exp42) was generated to give distinct change in density. The lens was moved to alter the degree of coverage of the very dense region. The cross-fan dataset was devised to give a many lines crossing at acute angles, and hence mimic the criss-cross character of some of the regions found in 'real' dataset plots.

exp	data set	OSR	lens position	screen size	NOZ width	comments
42	dense_parallel	100	varies	500,500	-	versions 1,2 & 3
43	cross_fan_1	100	180,250	500,500	-	402 lines
44	cross_fan_2	100	180,251	500,500	-	1002 lines

Note: Lens radius = 50, cell width = 1, max overlap lines = 1000 (except exp44 where it was 500)

**Table C-5** Details for experiments 42 to 44

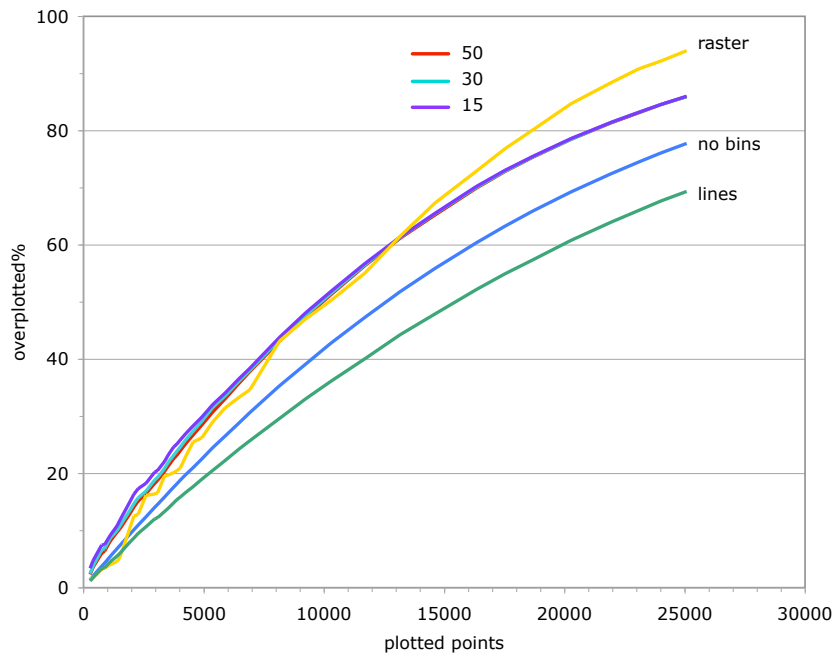
#### exp42

Tests were conducted with different degrees of coverage of the lens with all single points, all double points and alternating single and double. The results were as expected, with the weighted random values effectively corrected by the binning (100 bins).

#### exp43

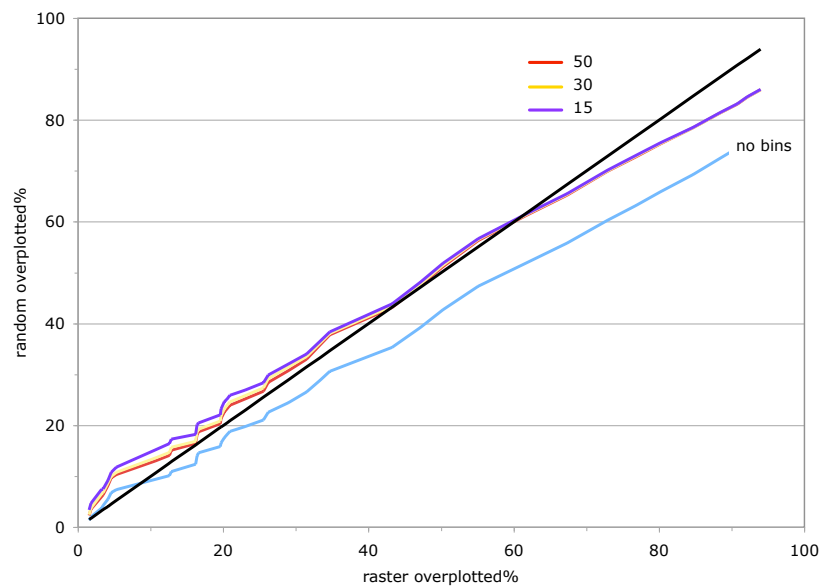
The density for this experiment was insufficient to give high occlusion values and hence exp44 was carried out with many more lines.

Figure C-7a



Investigating binning with a synthetic dataset - *random* overplotted% for different bin widths, together with *raster* and *lines* values

Figure C-7b



As Figure C-7a but with *random* overplotted% normalised against *raster* overplotted%

**exp44**

The behaviour with the synthetic dataset, with a large number of lines crossing at acute angles, is similar to other empirical results, with the *lines* algorithm giving a lower estimate of occlusion. Even though the density is fairly constant across the lens area, binning still has marked effect on the *random* values, bringing these closer to the *raster* values. Results from exp44 are given in Figures B-7a and B-7b

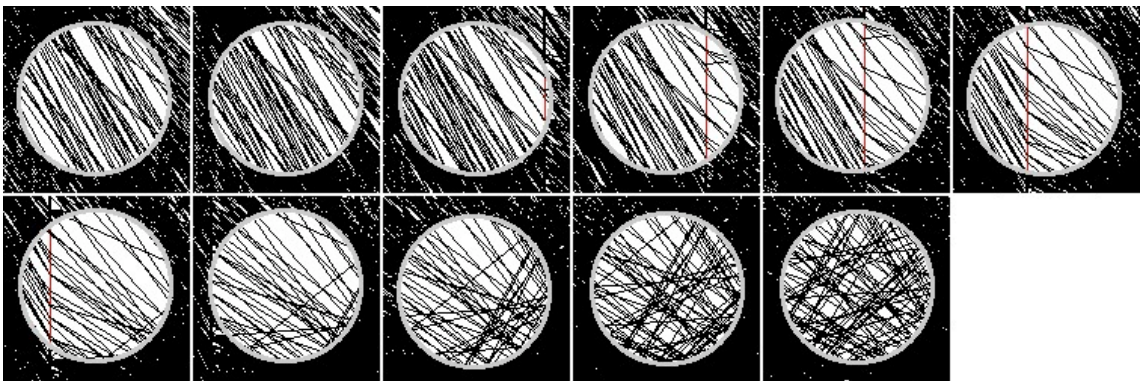
**C.6. exp45 to exp55**

Investigating the effect of the pattern within the lens on the occlusion measures *overplotted%*, *overcrowded%* and *hidden%*.

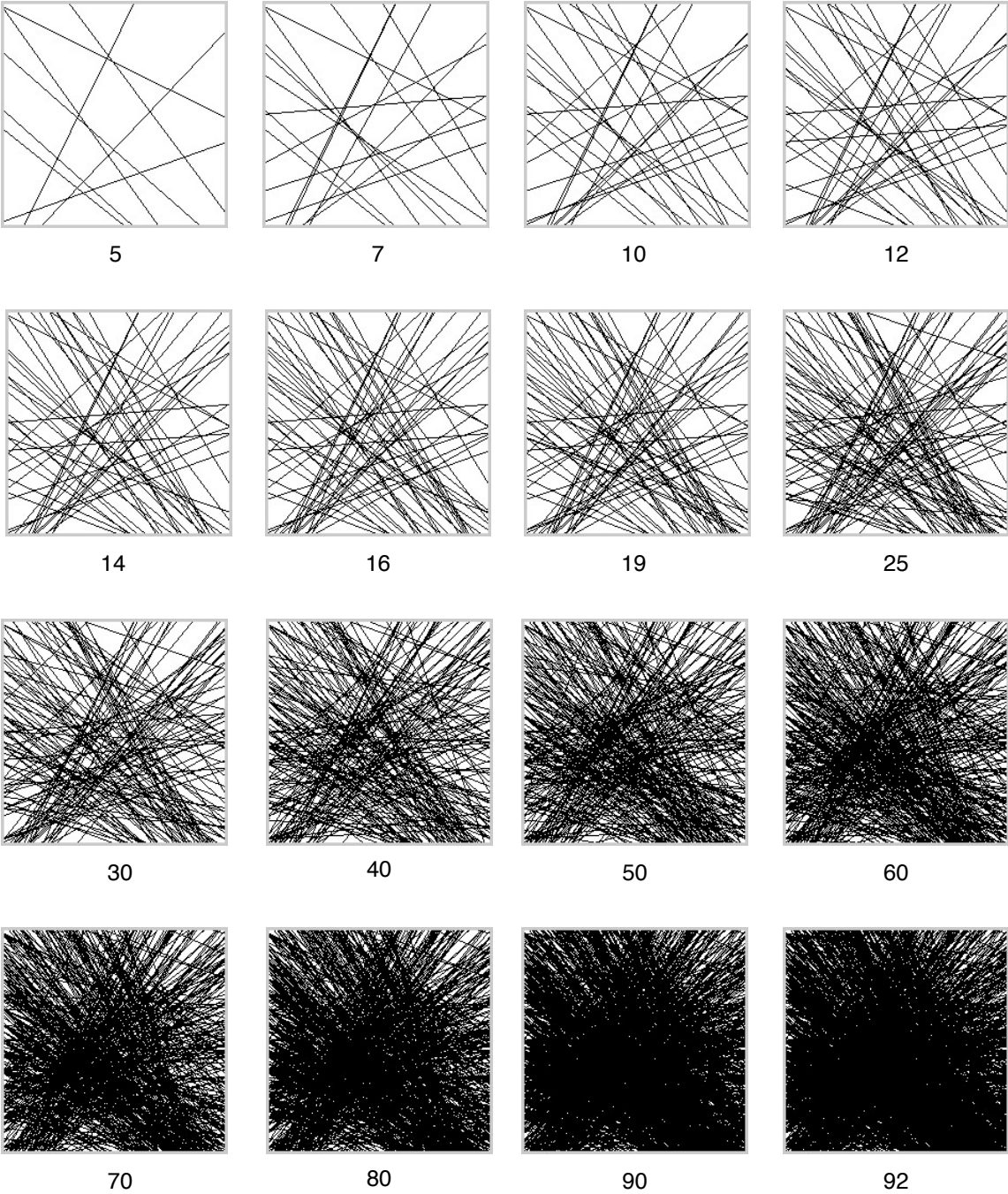
exp	data set	OSR	lens position	screen size	NOZ width	comments
45	cars 1K	100	150,340	701,700	1	
46	cars 1K	100	169,356	701,700	1	
47	cars 1K	100	188,372	701,700	1	
48	cars 1K	100	207,388	701,700	1	
49	cars 1K	100	226,404	701,700	1	
50	cars 1K	100	254,420	701,700	1	
51	cars 1K	100	264,436	701,700	1	
52	cars 1K	100	283,452	701,700	1	
53	cars 1K	100	302,468	701,700	1	
54	cars 1K	100	321,484	701,700	1	
55	cars 1K	100	340,500	701,700	1	

Note: Lens radius = 50, cell width = 1, max overlap lines = 500

**Table C-6** Details for experiments 45 to 55



**Figure C-8** Lens screen shots for exp45 (top left) to 55 (bottom right). Note that lens is successively moved to the right



**Figure C-9** Lens screen shots for a range of occlusion values (using weighted random binning – bin width = 10, lens radius = 105) - exp56



## C.7. exp56

Generating lens sample plots for a range of weighted random occlusion values.

exp	data set	OSR	lens position	screen size	NOZ width	comments
56	cars 1K	100	312,488	624 x 703	-	square lens

Note: Lens radius = 105, cell width = 1, max overlap lines = 500

**Table C-7** Details for experiment 56

## C.8. exp57 to exp59

Investigation into binning - exp57 to 59 (Section 5.6.2)

exp	data set	OSR	lens position	screen size	NOZ width	comments
57	cars 1K	100	340,500	701,700	1	as exp1
58	cars 1K	100	655,400	701,700	1	as exp2
59	cars 1K	100	150,340	701,700	1	as exp3

Note: Lens radius = 50, cell width = 1, max overlap lines = 500

**Table C-8** Details for experiments 57 to 59

## C.9. Data collected for each experiment

Each data item is tab delimited and can be read by a spreadsheet application.

### Included as a header - first line of the file

data file	filename of the data used in the experiment
display size	width and height of display area (pixels)
adjust radius	has lens radius been adjusted? (true or false)
rectangular clipping	rectangular lens clipping rather than circular? (true or false)
max overlap lines	top limit of the number of lines in the lines calculation

### Each run generates one of these lines

lens sampling rate	lens sampling rate (1-100)
lens radius	lens radius (pixels)
lens cell width	raster cell width (pixels)
zone width	non-overlap zone width (pixels)
overall sampling rate	overall sampling rate
lens position	centre position of the lens
lines overplotted%	lines algorithm calculation
raster overplotted%	calculated from frequency data - $S_n/(S_1 + S_n)$
random overplotted%	based on theoretical model (marbles & slots)
P2PM	time to generate pixel map in msec (ParallelCoordinateVis. paintToPixelMap)



	(next values from density visualiser)
columns	number of columns
cells	number of cells
max freq count	maximum number of points in any cell
marbles	number of plotted points
empty cells	empty cells
single cells	cells with 1 point
multiple cells	cells with more than 1 point
OWLCalcTime	time for overlap within lens calculation in msec (LOT)
totalLines	number of lines crossing lens
linesUsed	number of lines used in the calculation
totalChecked	number of line pairs used in the calculation
totalIntersect	number of intersecting lines
% intersect	percentage of lines intersecting
totalCoincident	coincident lines
overallLineLength	total length of the lines
number of zones	number of zones within the lens
overallPixelLineLen	number of pixels crossed by all the lines
	(next set of values are generated by histogram class)
marbles	number of plotted points
slots	number pixels with 1 or more plotted point
empty	empty slots = $\text{slots} * \text{Math.pow}(1-1/\text{slots}, \text{marbles})$
one	slots with 1 plotted point = $\text{marbles} * \text{Math.pow}(1-1/\text{slots}, \text{marbles}-1)$
>1	slots with more than 1 plotted point = $\text{slots} - \text{slots} * \text{Math.pow}(1-1/\text{slots}, \text{marbles}) - \text{marbles} * \text{Math.pow}(1-1/\text{slots}, \text{marbles}-1)$
overplotted%	$= 100 * \text{slotsWithMoreThanOne} / (\text{slotsWithMoreThanOne} + \text{slotsWithOne})$
hidden%	$= 100 * (\text{marbles} - \text{singleMarbles} - \text{slotsWithMoreThanOne}) / \text{marbles}$
avgPPL	average pixels per line
variancePPL	variance of pixels per line
sdevPPL	standard deviation of pixels per line
minPPL	minimum pixels per line
maxPPL	maximum pixels per line
DMU time()	time to update density map (msecs)
	(next 2 values are the same as overplotted% and hidden% above)
overplotted%	$= 100 * \text{slotsWithMoreThanOne} / (\text{slotsWithMoreThanOne} + \text{slotsWithOne})$
hidden%	$= 100 * (\text{marbles} - \text{singleMarbles} - \text{slotsWithMoreThanOne}) / \text{marbles}$
overcrowded%	$= 100 * (\text{marbles} - \text{singleMarbles}) / \text{marbles}$
binWidth	bin width in pixels
bins	number of bins across
raster%	weighted raster%
random%	weighted random%

data file portland\_cars\_1k\_nodist.csv  
 display size 700 x 700  
 adjust radius FALSE  
 rectangular clipping FALSE  
 max overlap lines 500

lens sampling rate	lens radius	lens cell width	zone width	overall sampling rate	lens position	lines overplotted%	raster overplotted%	random overplotted%	P2PM time	columns	cells
100	50	1	1	100	340,500	97.29	98.72	99.16	145	20	7252
95	50	1	1	100	340,500	97.31	98.54	98.95	137	20	7252
90	50	1	1	100	340,500	97.27	98.13	98.59	232	20	7252
85	50	1	1	100	340,500	97.24	97.6	98.16	111	20	7252
80	50	1	1	100	340,500	97.21	97.13	97.57	121	20	7252
75	50	1	1	100	340,500	97.33	96.21	96.77	111	20	7252
70	50	1	1	100	340,500	97.43	94.86	95.82	93	18	7252
65	50	1	1	100	340,500	97.08	93.63	94.61	88	17	7252
60	50	1	1	100	340,500	96.07	92.15	92.99	177	17	7252
55	50	1	1	100	340,500	94.48	89.55	90.66	80	17	7252
50	50	1	1	100	340,500	92.63	86.97	88.26	68	16	7252
45	50	1	1	100	340,500	89.95	84.38	84.91	70	12	7252
40	50	1	1	100	340,500	86.9	80.64	81.05	66	12	7252
35	50	1	1	100	340,500	82.66	75.6	76.07	57	12	7252
30	50	1	1	100	340,500	76.64	69.83	69.33	49	12	7252
28	50	1	1	100	340,500	74.15	67.13	66.74	41	11	7252
26	50	1	1	100	340,500	71.38	64.85	64.42	37	11	7252
24	50	1	1	100	340,500	68.11	61.63	60.71	58	11	7252
22	50	1	1	100	340,500	64.79	57.91	57.43	28	10	7252
20	50	1	1	100	340,500	61.03	54.82	54	24	8	7252
19	50	1	1	100	340,500	58.04	51.77	51.12	20	8	7252
18	50	1	1	100	340,500	56.58	50.39	49.67	25	8	7252
17	50	1	1	100	340,500	54.36	48.79	47.81	24	8	7252
16	50	1	1	100	340,500	51.74	46.71	45.81	16	8	7252
15	50	1	1	100	340,500	49.17	44.05	43.24	17	8	7252
14	50	1	1	100	340,500	47.2	41.86	40.91	24	8	7252
13	50	1	1	100	340,500	44.84	39.57	38.04	13	8	7252
12	50	1	1	100	340,500	41.87	37.84	36.46	19	7	7252
11	50	1	1	100	340,500	39.54	35.51	34.18	12	7	7252
10	50	1	1	100	340,500	36.29	32.17	31.61	11	6	7252
9	50	1	1	100	340,500	33.35	29.11	28.99	10	6	7252
8	50	1	1	100	340,500	30.71	26.77	27.09	12	6	7252
7	50	1	1	100	340,500	26.07	22.51	23.36	13	5	7252
6	50	1	1	100	340,500	22.13	18.31	19.41	5	5	7252
5	50	1	1	100	340,500	18.93	15	15.72	5	5	7252
4	50	1	1	100	340,500	15.23	12.1	12.33	4	5	7252
3	50	1	1	100	340,500	10.19	8.43	8.92	3	4	7252
2	50	1	1	100	340,500	8.23	6.59	5.95	2	4	7252
1	50	1	1	100	340,500	2.07	2.84	1.98	1	3	7252

page 1

number of zones	overallPixelLineLen	marbles	slots	empty	one	>1	overplotted%	hidden%	avgPPL	variancePPL	sdevPPL	minPPL	maxPPL
1	51491	48489	7252	9	60	7182	99.16	85.06	70.6	322.1	17.9	7	100
1	49351	46504	7252	11	76	7163	98.95	84.43	70.9	317.6	17.8	7	100
1	46679	43972	7252	16	102	7132	98.59	83.55	70.7	322.4	18	7	100
1	44251	41679	7252	23	132	7095	98.16	82.66	70.6	325.3	18	7	100
1	41619	39232	7252	32	175	7044	97.57	81.6	70.8	320.9	17.9	7	100
1	38948	36698	7252	45	232	6973	96.77	80.36	70.9	319.6	17.9	7	100
1	36447	34364	7252	63	300	6887	95.82	79.08	71.1	316	17.8	7	100
1	34008	32048	7252	87	385	6778	94.61	77.64	71.2	326	18.1	7	100
1	31392	29597	7252	122	499	6629	92.99	75.91	71.2	328.7	18.1	7	100
1	28500	26873	7252	178	660	6413	90.66	73.68	71.1	331.4	18.2	7	100
1	26123	24658	7252	241	822	6187	88.26	71.57	71.3	316.3	17.8	7	100
1	23471	22165	7252	341	1042	5857	84.91	68.82	71.5	310.3	17.6	7	100
1	21018	19850	7252	469	1285	5497	81.05	65.83	71.3	314.4	17.7	7	100
1	18409	17409	7252	657	1578	5016	76.07	62.12	72	309.9	17.6	7	100
1	15576	14728	7252	951	1932	4367	69.33	57.22	72.2	310.6	17.6	7	100
1	14618	13831	7252	1076	2053	4121	66.74	55.35	72.4	319	17.9	7	100
1	13784	13073	7252	1195	2155	3901	64.42	53.67	72.8	286.3	16.9	10	100
1	12603	11945	7252	1396	2300	3554	60.71	50.98	72.2	286.9	16.9	10	100
1	11597	11018	7252	1587	2411	3253	57.43	48.58	72.6	276.5	16.6	10	100
1	10546	10107	7252	1799	2508	2944	54	46.05	72.4	289.1	17	10	100
1	9896	9385	7252	1987	2572	2691	51.12	43.91	72.1	303.7	17.4	10	100
1	9519	9033	7252	2086	2599	2565	49.67	42.82	72.5	295.4	17.2	10	100
1	9047	8593	7252	2217	2627	2407	47.81	41.41	72.7	285.4	16.9	10	100
1	8565	8133	7252	2362	2649	2239	45.81	39.88	72.8	296	17.2	10	100
1	7974	7563	7252	2555	2665	2030	43.24	37.9	72	296.9	17.2	10	100
1	7440	7062	7252	2738	2667	1846	40.91	36.09	72.3	282.4	16.8	10	100
1	6816	6467	7252	292	2651	1628	38.04	33.83	71.7	295.5	17.2	10	100
1	6481	6147	7252	3106	2633	1511	36.46	32.57	71.9	304.4	17.4	10	100
1	6009	5696	7252	3306	2597	1348	34.18	30.73	71.3	298.9	17.3	10	100
1	5490	5201	7252	3539	2538	1173	31.61	28.62	70.9	306.8	17.5	10	100
1	4974	4712	7252	3786	2460	1004	28.99	26.46	71.5	312.3	17.7	10	100
1	4610	4365	7252	3972	2391	888	27.09	24.86	71.3	323.2	18	10	100
1	3909	3703	7252	4351	2222	677	23.36	21.68	71.3	355.3	18.9	10	100
1	3193	3025	7252	4778	1993	480	19.41	18.23	71	397.1	19.9	10	100
1	2549	2415	7252	5197	1731	322	15.72	14.94	70.4	440.1	21	10	100
1	1977	1870	7252	5603	1445	203	12.33	11.85	69.7	547	23.4	10	100
1	1422	1335	7252	6032	1110	108	8.92	8.66	66.3	626.5	25	10	91
1	938	882	7252	6421	781	49	5.95	5.84	68.6	923.3	30.4	10	91
1	321	290	7252	6967	278	5	1.98	1.97	55	1449	38.1	10	79

page 3

Figure C-10 Example of the data output from an experiment and read into a spreadsheet

### C.10. Example of the data output from an experiment

Figure C-10 is the data from one run of an experiment (57\_10(bin).txt), read into a spreadsheet. Each row is for a particular lens sampling rate.

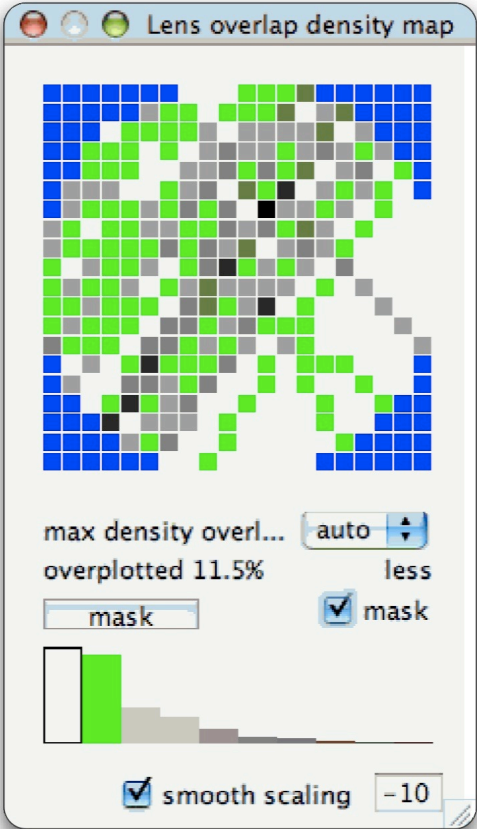
max freq count	marbles	empty cells	single cells	multiple cells	OWLCalcTime	totalLines	linesUsed	totalChecked	totalIntersect	% intersect	totalCoincident	overallLineLength
20	48489	12	93	7147	125	744	500	249500	123768	49	0	41294
20	46504	15	106	7131	136	711	500	249500	124546	49	0	41303
20	43972	23	135	7094	139	674	500	249500	123960	49	0	41171
20	41679	33	173	7046	144	640	500	249500	123460	49	0	41105
20	39232	47	207	6998	130	600	500	249500	124470	49	0	41238
20	36698	53	273	6926	129	561	500	249500	125790	50	0	41521
18	34364	69	369	6814	150	523	500	249500	125506	50	0	41523
17	32048	96	456	6700	122	488	488	237656	119520	50	0	40418
17	29597	144	558	6550	107	450	450	202050	101932	50	0	37289
17	26873	212	736	6304	80	409	409	166872	84644	50	0	33875
16	24658	274	909	6069	74	374	374	139502	71358	51	0	31059
12	22165	389	1072	5791	57	335	335	111890	57284	51	0	27940
12	19850	544	1299	5409	54	301	301	90300	46186	51	0	25034
12	17409	761	1584	4907	34	261	261	67860	35556	52	0	21912
12	14728	1100	1856	4296	26	220	220	48180	25266	52	0	18507
11	13831	1220	1983	4049	24	206	206	42230	22388	53	0	17391
11	13073	1332	2081	3839	19	193	193	37056	19908	53	0	16415
11	11945	1537	2193	3522	24	178	178	31506	16654	52	0	15072
10	11018	1704	2335	3213	13	163	163	26406	14270	54	0	13890
8	10107	1874	2430	2948	11	150	150	22350	12008	53	0	12737
8	9385	2042	2513	2697	10	140	140	19460	10402	53	0	11814
8	9033	2142	2535	2575	9	134	134	17822	9654	54	0	11386
8	8593	2269	2552	2431	8	127	127	16002	8690	54	0	10832
8	8133	2384	2594	2274	8	120	120	14280	7690	53	0	10208
8	7563	2578	2615	2059	7	113	113	12656	6764	53	0	9540
8	7052	2756	2614	1882	5	105	105	10920	5876	53	0	8890
8	6467	3016	2560	1676	7	97	97	9312	4978	53	0	8150
7	6147	3129	2563	1560	4	92	92	8372	4438	53	0	7725
7	5696	3323	2534	1395	6	86	86	7310	3850	52	0	7229
6	5201	3537	2520	1195	7	79	79	6162	3214	52	0	6625
6	4712	3779	2462	1011	3	71	71	4970	2582	51	0	5992
6	4365	3946	2421	885	2	66	66	4290	2202	51	0	5558
5	3703	4316	2275	661	1	56	56	3080	1596	51	0	4729
5	3025	4751	2043	458	2	46	46	2070	1092	52	0	3896
5	2415	5186	1756	310	1	37	37	1332	712	53	0	3108
5	1870	5607	1446	199	0	29	29	812	420	51	0	2402
4	1335	6030	1119	103	1	22	22	462	216	46	0	1724
4	882	6432	766	54	0	14	14	182	96	52	0	1114
3	290	6970	274	8	0	6	6	30	10	33	0	409

page 2

DMU time()	overplotted%	hidden%	overcrowded%	binWidth	bins	weighted raster%	weighted random%
586	99.16	85.06	99.88	10	100	98.94	98.87
106	98.95	84.43	99.84	10	100	98.78	98.64
127	98.59	83.55	99.77	10	100	98.44	98.23
218	98.16	82.66	99.68	10	100	97.98	97.74
124	97.57	81.6	99.55	10	100	97.54	97.15
108	96.77	80.36	99.37	10	100	96.68	96.35
146	95.82	79.08	99.13	10	100	95.51	95.37
163	94.61	77.64	98.8	10	100	94.37	94.2
125	92.99	75.91	98.31	10	100	93.08	92.62
147	90.66	73.68	97.54	10	100	90.86	90.42
151	88.26	71.57	96.66	10	100	88.53	88.21
605	84.91	68.82	95.29	10	100	86	85.11
130	81.05	65.83	93.53	10	100	82.36	81.6
109	76.07	62.12	90.93	10	100	77.82	77.11
114	69.33	57.22	86.88	10	100	71.95	70.86
118	66.74	55.35	85.15	10	100	69.17	68.45
116	64.42	53.67	83.51	10	100	66.7	66.12
181	60.71	50.98	80.74	10	100	63.42	62.61
136	57.43	48.58	78.11	10	100	59.55	59.33
134	54	46.05	75.18	10	100	56.32	55.82
123	51.12	43.91	72.58	10	100	53.46	53.26
87	49.67	42.82	71.22	10	100	52.05	51.91
96	47.81	41.41	69.42	10	100	50.41	49.9
124	45.81	39.88	67.42	10	100	48.29	47.91
110	43.24	37.9	64.75	10	100	45.64	45.49
99	40.91	36.09	62.23	10	100	43.41	43.22
111	38.04	33.83	59	10	100	41.16	40.69
102	36.46	32.57	57.15	10	100	39.28	38.96
108	34.18	30.73	54.4	10	100	36.89	36.51
101	31.61	28.62	51.18	10	100	33.66	33.88
104	28.99	26.46	47.78	10	100	30.59	31.22
125	27.09	24.86	45.22	10	100	28.01	29.01
99	23.36	21.68	39.98	10	100	23.84	25.41
118	19.41	18.23	34.1	10	100	19.38	21.4
112	15.72	14.94	28.32	10	100	16.11	17.94
95	12.33	11.85	22.72	10	100	13.21	15.44
112	8.92	8.66	16.8	10	100	9.51	11.89
126	5.95	5.84	11.44	10	100	7.75	10.05
161	1.98	1.97	3.91	10	100	3.99	4.95

page 4

Figure D-1



Sampling Lens density visualiser

# Appendix D

## Implementation Issues

Section D.1 describes the software instrumentation, interface controls and data collection added for the purpose of conducting the series of experiments as part of the empirical study into auto-sampling (Chapter 5). The data collected for each test is listed and the output format is illustrated.

Section D.2 gives an overview of the architecture of the InfoVis Toolkit used in the development of the Sampling Lens and describes, in broad terms, how the toolkit was extended to provide the required functionality.

Section D.3 describes the development and comparison of Java2D and OpenGL versions of a simple parallel coordinate visualisation. The improved interactive performance motivated the development of an OpenGL version of the Sampling Lens application. The main issues arising from this code migration are discussed.

### D.1. Instrumentation of the Sampling Lens

In order to carry out an empirical study to evaluate metrics and measures (as described in Chapter 5), the Sampling Lens application was instrumented with a density visualiser and two control panels, together with the facility to calculate and collect a wide range of appropriate data. This section describes the experimental controls and the type of data collected. Note that the control panels were not designed for the end user, however, some users showed interest in the density map.

#### D.1.1. Density visualiser

The lens overlap density map or density visualiser provides a rasterised plot of the lines within the lens and proved valuable. The lens cell width, which can be set between 1 and 20 pixels (shown in Figure D-2) adjusts the resolution of the density map.

The example in Figure D-1 is for a 50 pixel radius lens with each coloured square representing a block of 5x5 screen pixels. White represents no plotted points, green means one plotted point and the other cells follow a greyscale up to black. The single point cells (in green) are important in the density metrics being investigated and are therefore highlighted.

Various other controls are featured, including the ability to show the lens mask (in blue). Interestingly, the mask is created by rasterising an off-screen rectangle with

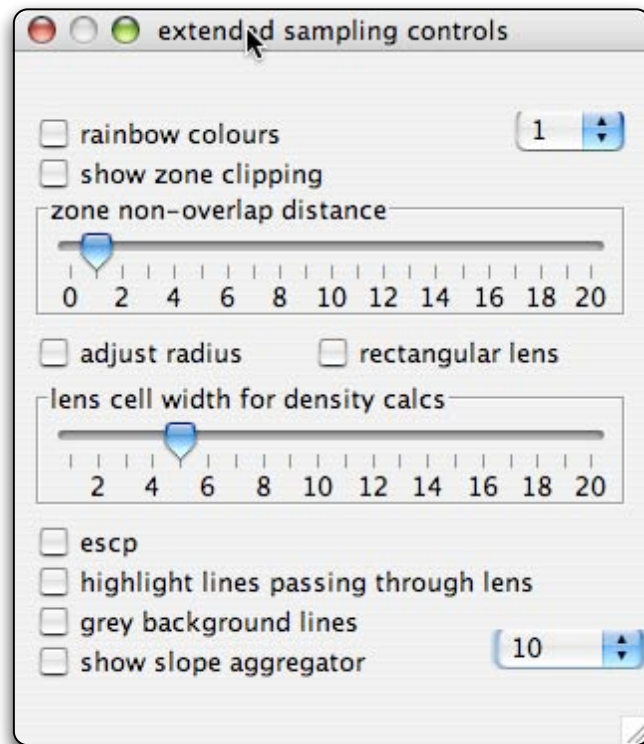


Figure D-2

Extended sampling controls for the Sampling Lens

category	collected data items
General	data filename, display size, radius and clipping options
Basic	lens sampling rate, lens radius, raster cell width, zone width, sampling rate, lens position
Overplotted%	lines, raster, random
Rasteriser	columns, cells, max frequency count, plotted points, empty, single and multiple cells
Lines	total lines, lines used, total checked, total intersections, percentage intersections, coincident lines, total line length, lens zones, pixel line length statistics
Random calc.	plotted points, available pixels, empty, single, multiple, overplotted%, overcrowded%, hidden%
Bins	bin width, bins across, bins down, weighted raster%, weighted random%
Timing	pixel map generation, lines overlap calc, density map generation

**Table D-1** Summary of the data produced for the parallel coordinate experiments. Apart from the general category, the data is output for each of the 39 sampling rates. Experiments into raster cell widths and bin widths, produce separate files for each cell or bin width.



many lines drawn across it, all of which are clipped using the same clipping class as used to draw the on-screen lens. Hence, the cells without plotted points become the mask. Another control is for setting the greyscale for the cells with more than one plotted point. This can either be auto-scaling (as shown on the example panel in Figure D-1) which sets the maximum density to black or the user can set the maximum value from a drop down list. The latter was found useful when comparing density plots. The histogram (optional) at the bottom gives another view of the number of overlapping points in the cells.

### D.1.2. Extended sampling controls

The extended sampling controls in Figure D-2 was devised to investigate various issues that cropped up during this work.

The *zone non-overlap distance* slider was used to investigate a solution to the problem, discussed in Section 5.2, where the calculated occlusion value was greatly overestimated in the situation where a large number of lines met at a single point on an axis. Setting this slider to one or more means that the lines used in the occlusion calculation are clipped this number of pixels from the axis.

The *lens cell width for density calc* slider controls the raster cell width and was used when investigating methods of speeding up the calculation of the *raster* measure (Section 5.5.2). A related issue was that, as the cell width was varied within the raster plot, some cells did not give the expected point counts. Upon investigation, it was discovered that pixels were missed due to an inexact number of whole cells within the lens and a solution was to adjust the lens radius slightly (*adjust radius* checkbox).

Additional controls include *rainbow colouring* (part of RaDAR, Section 4.4.3, *highlight lines passing through the lens*, *grey background lines*, as well as an experimental widget, the slope aggregator, which gives a radial histogram based on the slope of the lines.

### D.1.3. Empirical controls

Data collection for each experiment described in Chapter 5 involved varying the sampling rate (a range of 39 sampling rates was used) and for some experiments a range of raster cell widths (speeding up *raster* calculation – Section 5.5.2) or bin widths (dealing with non-uniform density – Section 5.6.2). In the final version of the software, 47 items in all were collected. A summary is given in Table D-1 (a full annotated list is included in Appendix C.9).

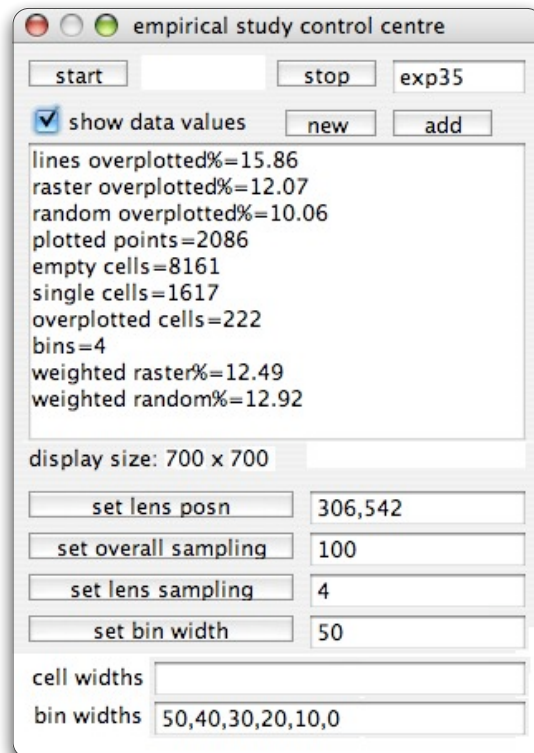
As well as giving the results of occlusion measure calculations (overplotted%, overcrowded% and hidden%) and overplotted% algorithms (*lines*, *raster*, *random*, weighted *raster* and weighted *random*) the data set includes the raw data on which

Figure D-3

raster overplotted%	random overplotted%	PZPM time	columns	cells	max freq count	marbles	empty cells	single cells	multiple cells	OWLCalcTime	totalLines	linesUsed	totalCr
88.66	94.51	127	33	10000	33	43957	2525	948	6627	128	607	607	367
87.41	93.37	104	30	10000	30	41524	2587	933	6480	120	574	574	328
87.2	92.3	94	28	10000	28	39592	2687	936	6377	104	546	546	297
86.44	90.69	105	27	10000	27	37103	2756	982	6262	105	513	513	262
85.55	89.16	98	26	10000	26	35073	2824	1037	6139	83	483	483	232
84.39	87.07	94	24	10000	24	32685	2881	1111	6008	74	449	449	201
82.8	84.67	76	23	10000	23	30351	2987	1206	5807	57	418	418	174
81.49	81.99	79	21	10000	21	28089	3108	1276	5516	51	387	387	149
80.03	79	58	21	10000	21	25902	3220	1354	5426	56	356	356	126
78.18	75.57	63	18	10000	18	23707	3328	1456	5216	39	327	327	106
75.44	71.75	57	16	10000	16	21548	3421	1616	4963	29	299	299	89
74.28	67.24	39	16	10000	16	19306	3984	1650	4766	33	269	269	72
70.2	62.4	46	14	10000	14	17167	3701	1877	4422	21	239	239	56
65.85	57.06	40	13	10000	13	15052	3979	2056	3965	14	212	212	44
61.55	51.3	28	11	10000	11	13002	4353	2171	3476	11	181	181	32
57.66	48.11	32	10	10000	10	11945	4450	2350	3200	10	167	167	27
54.73	44.62	26	10	10000	10	10848	4703	2398	2899	8	151	151	22
52.69	42.6	22	10	10000	10	10238	4838	2442	2720	6	143	143	20
49.88	38.98	18	9	10000	9	9181	5074	2469	2457	6	129	129	16
47.82	36.78	16	9	10000	9	8564	5192	2509	2299	5	120	120	14
47.69	35.77	21	8	10000	8	8287	5326	2445	2229	4	115	115	13
45.15	33.38	27	8	10000	8	7641	5588	2420	1992	3	106	106	11
44.13	31.86	16	8	10000	8	7239	5622	2446	1932	4	101	101	10
41.65	30.34	15	7	10000	7	6842	5770	2468	1762	3	95	95	89
40.36	28.48	11	7	10000	7	6368	5954	2413	1633	3	88	88	76
37.01	26.53	24	7	10000	7	5980	6104	2454	1442	3	81	81	64
37.4	25.01	10	7	10000	7	5505	6281	2328	1391	2	76	76	57
34.17	22.87	9	7	10000	7	4987	6517	2293	1190	2	69	69	46
30.92	20.87	11	7	10000	7	4512	6750	2245	1005	1	61	61	36
29.45	18.73	7	7	10000	7	4015	7039	2089	872	1	54	54	28
26.51	16.53	23	6	10000	6	3512	7314	1974	712	0	48	48	22
19.91	14.12	5	6	10000	6	2972	7579	1939	482	1	41	41	16
14.41	12.36	4	5	10000	5	2383	7766	1912	322	1	36	36	12
11.36	10.78	4	5	10000	5	2241	8001	1772	227	0	32	32	9
9.34	9.06	3	4	10000	4	1872	8298	1543	159	1	27	27	7
6.81	7.05	17	4	10000	4	1445	8649	1259	92	0	21	21	4
5.08	4.78	2	3	10000	3	972	9075	878	47	0	14	14	1
6.08	3.7	1	3	10000	3	750	9293	664	43	0	11	11	1
4.19	1.97	1	3	10000	3	398	9618	366	16	0	6	6	3

Part of a spreadsheet based on the output file produced by way of the empirical control panel. The full listing features 47 columns of raw and calculated data. The 39 rows of data are for a series of sampling rates from 100% down to 1%.

Figure D-4



Control panel for conducting the experiments

these are based. This has been useful for both checking the computation and spotting possible anomalies. The timing data has been equally useful as was seen in Section 5.5.2 when considering the efficiency of the algorithms. Note that times to redraw the display are shown in the empirical study panel (Figure D-4), if required. Other data such as the pixel line length statistics was useful to determine if the number of pixels used in the random calculation could be reliably obtained (as discussed in Section 5.5.1).

The experimental process is automated so that the application steps through the series of sampling rates/cell widths/bin widths as required. Each cell width/bin width produces a formatted text file, with an appropriate name, which can be read into a spreadsheet for further analysis. Part of such a spreadsheet is given in Figure D-3 (a full listing is presented in Appendix C.10). In this format, it is reasonably easy to produce charts to explore the data. I did consider using a database to store the data, either directly from the visualisation or input from an intermediary file, however the table view given by a spreadsheet provides a good visual overview and is easy to manipulate. Cutting and pasting between spreadsheets is again easy, due to producing the results in a standard format.

The experiments are controlled by the empirical study panel shown in Figure D-4. Near the bottom of the panel are the current values of the basic parameters such as sampling rate, lens sampling rate, lens position and raster bin width. When an experiment is not running in automatic mode, the user can enter values into any of the fields and click on the appropriate button to set that value. This is more accurate than using the slider controls on the sampling control panel or dragging the lens to a particular position.

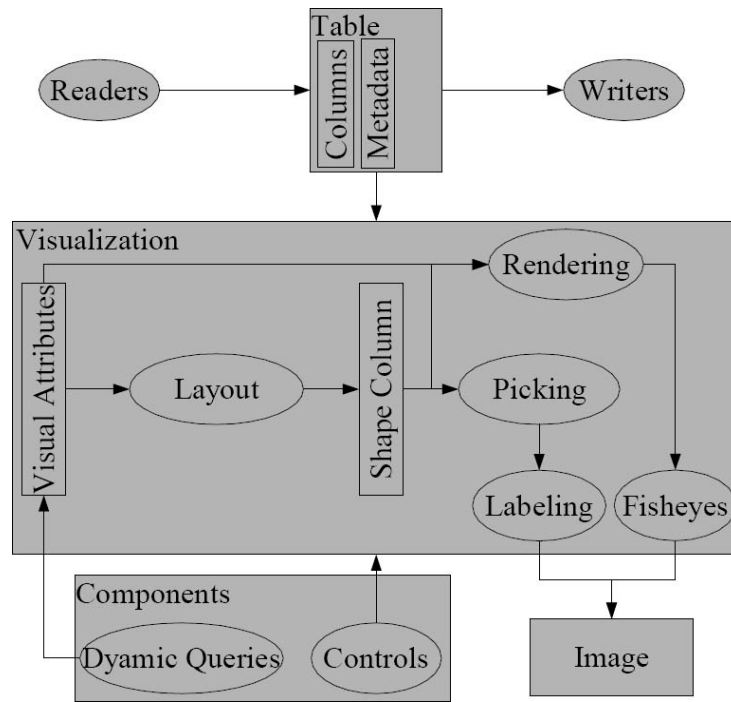
As shown in Figure D-4, the large text area provides feedback on principle raw data and calculated values. However, this feature can be turned off to disable the calculations, hence improving interactivity.

The controls on the top line start and stop automatic data collection; data files are stored in the folder name entered in the field to the right (e.g.. exp35). Manual data collection is also possible; *new* opens a file (name given in text field above) for recording and *add* adds a record to the opened data file. *new* changes to *save* after selection. The text fields at the bottom of the panel are used to set a sequence of either raster cell widths or raster bin widths.

## D.2. Architectural issues

This section provides a high level view of the architecture of the InfoVis Toolkit and an overview of the integration of additional code necessary to implement the experimental version of the Sampling Lens application.

Figure D-5



Internal structure of the InfoVis Toolkit. Squares represent data structures whereas ellipses represent functions. [Figure 2, Fekete 04, The InfoVis Toolkit]

### **D.2.1. InfoVis Toolkit architecture**

The InfoVis Toolkit is an Interactive Graphics toolkit with a Java library which was designed for creating of new visualisation techniques as well as the extension of existing ones. Much effort has gone into the design of the underlying data structure based on tables which achieves a small memory footprint and good performance. It comes with a large set of generic interactive components with which to implement dynamic queries and has some pre-built visualisation and associated control panels.

The internal structure of the toolkit is given in Figure D-5. Data is stored in the columns of tables with other internal columns containing metadata (e.g. to define data type), topological information (so trees and graphs can be represented), visual attributes (e.g. colour, shape, size, label) and indexes. A visualisation component maps the set of semantic attributes stored in the table into visual shapes using its layout algorithm, which are then rendered. As shown in Figure D-5, visual transformations such as a fisheye lens can be applied to the image and picking and labelling are supported natively. Dynamic query controls alter attributes stored in columns and permit rapid selection and filtering. The output is automatically updated when data in any column is changed.

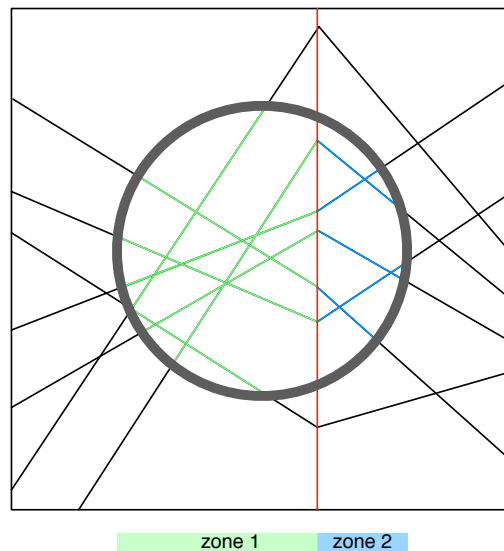
### **D.2.2. Implementing the sampling applications**

Implementing the first sampling visualisations, scatterplot and parallel coordinates, were reasonably straightforward, although hampered to some extent by the limited amount of documentation. A z-index column was created and seeded with random values and methods were written to filter the rows of data based on the position of the sampling rate slider control. A Reality Check function was added, again quite simply by changing the z-index pointers to the start and end of the sample window. The z-index method is described and illustrated in Section 2.2. To accommodate these controls, a sampling control panel (as seen in Section 5.2) was added to the default set of tabbed control panels which make up the dynamic query and control components.

Implementing the interactive sampling lens was more of a challenge, especially when it came to parallel coordinates. Two methods were considered. The first comprised of the following sequence of actions: draw the background lines, clear the circular region of the screen corresponding to the lens position, draw the outline of the lens and then plot the points within the lens. The second method involved maintaining an off-screen buffer duplicating the display but at the sampling rate of the lens. To draw the lens, the part of the off-screen buffer corresponding to the lens region is overlaid on the display and the lens outline is drawn.

The second method was attractive as, provided that neither the overall or lens sampling rates were altered, the lens could be moved about the screen without re-

Figure D-6



Clipping the parallel coordinate lines to the lens outline and any attribute axes within the lens.

packages	classes		additions
<b>io</b>	10	basic readers and writers	
<b>metadata</b>	8	metadata categories	
<b>panel</b>	43	control panels and components	SamplingControlPanel ExtendedSamplingControlPanel EmpiricalStudyControlPanel SamplingDynamicQuery SamplingLensDynamicQuery
<b>column</b>	38	principle data container	SamplingColumn
<b>table</b>	16	visualisation of tables	ParallelCoordinatesVisualization* ScatterPlotVisualization*
<b>tree</b>	45	visualisation of trees	
<b>graph</b>	38	visualisation of graphs	
<b>utils</b>	17	comparators, etc	
<b>visualization</b>	49	basic visualisation and helper classes including renderers, colour management plus interfaces	DefaultVisualization* SamplingLens ScatterPlotSamplingLens ParallelCoordinatesSamplingLens ScatterPlotClutter
<b>utility</b> (new package)	26		ClipIT, Histogram, DensityVisualiser, GridBin, PixelMap, SlopeAggregatorVisualiser, RainbowColours, IntStack, IntSparseArray, LimitPair, LinePlus,

**Table D-2** InfoVis Toolkit package structure together with the principal classes added to implement to Sampling Lens. The classes marked with a \* are extended InfoVis Toolkit classes.

plotting the points<sup>1</sup>. However, Java2D did not feature a circular stencil function and perhaps more importantly, its drawing speed was slow even with a few thousand points and updating the off-screen buffer every time the user changed the lens sampling rate disabled any reasonable interaction. Therefore the first approach was adopted. The filtering function, utilising the z-index determines the set of points that could be displayed within the lens region as a result of the lens sampling, and a further calculation selects those points which are actually within the lens.

The parallel coordinate implementation is similar, although the *shape column* (Figure D-5) contains a polyline representing the set of lines joining points on the attribute axes. This polyline had to be clipped to the circular lens and to the vertical axes if present within the lens. Figure D-6 illustrates the case with one axis passing through the lens (in red); this results in two lists of line coordinates, represented by the green set for zone1 and the blue set for zone2, which are subsequently used for lens drawing and for calculating the occlusion measures discussed in Chapter 5. Note that any lines not in the lens sample (i.e. background lines) have not been shown.

A substantial amount of code (about 8000 lines or 290KB for the Java2D version) has been incorporated into the InfoVis Toolkit to provide the sampling functionality and the experimental infrastructure. In terms of the architecture of the InfoVis Toolkit, the additional code for implementing the Sampling Lens is illustrated in Table D-2.

The sampling controls mentioned earlier (SamplingControlPanel) and associated dynamic query functions (SamplingDynamicQuery and SamplingLensDynamicQuery) for managing the z-index are within the *panel* package. The two experimental control panels mentioned in Section D.1 are also within this package. The code for rendering the lenses was added to the appropriate visualisations in the *table* package (as they both used table-based data) with modifications to the DefaultVisualization class in the *visualization* package for handling such things as buffering the background display and colour management for the rainbow-colour feature (Section D.4).

The ParallelCoordinatesVisualization class is also the place where different occlusion measures are calculated as well as the preparation of the data for the density map visualisation. The SamplingLens class (in the *visualization* package) holds much status (e.g. lens shape, position, density visualiser displayed) and calculated data (e.g. current occlusion value, total pixel line length) and has various listeners for notifying control panels. The InfoVis Toolkit is designed to automatically update the display when column data changes and does not have any feedback mechanism. As a result, it was difficult to notify the control panels when events, such as calculating occlusion values

---

<sup>1</sup> Before overlaying the lens, the rectangular display area containing the lens could be saved and then copied back when the lens was moved or removed.





occurred in the visualisation classes. Fortunately, the `SamplingLens` was known and proved to be a useful class for internal communication.

Other useful classes were developed, from generating a density map to sparse array storage. Some of these are listed in Table D-2 under the *utility* package. Some reflections on the use of the InfoVis Toolkit in developing the Sampling Lens application are given in Section 6.4.1.

### D.3. OpenGL implementation

From the onset, the relatively long time to redraw the display has been problematic. As noted in Chapter 4, the redraw time for a parallel coordinate plot with 1000 records and 4 attributes was almost one second. This obviously makes the adjustment of the overall sampling rate rather sluggish and limits the size of the datasets. The only way of dealing with large datasets is therefore to sample, but displaying the full amount of data, say a 10,000 record dataset could take up to 15 seconds, is clearly unworkable. The slow drawing speed also affects the interaction with the lens, hence the user is limited to small lenses and an acceptable performance is only achieved with a lens sampling window of approximately 50 records.

The InfoVis Toolkit is built for speed, however, the code is deeply nested and fairly complex to follow. It was therefore decided to implement a simple stripped-down version of parallel coordinates in Java through which the processing time components could be measured more precisely. This exercise highlighted the poor performance of Java's drawing primitives. The InfoVis Toolkit was very efficient in its data processing. Other visualisation toolkits [Prefuse, Piccolo] were again considered but none appeared to have sufficient advantage over the InfoVis Toolkit to warrant a change.

#### D.3.1. Java2D vs. OpenGL

After running some OpenGL demonstrations<sup>2</sup> on the Macintosh, this clearly showed the advantage of using the graphics card functionality to boost the performance of graphic intensive applications. It was decided to use Jogl<sup>3</sup>, the Java bindings for OpenGL 2.0 API<sup>4</sup> as the Sampling Lens application was already in Java. Jogl also integrates well with AWT and Swing and there is a native MacOSX version. A Jogl version of the basic parallel coordinates was developed and tested against the Java2D version. A variety of datasets were used to give a wide range of number of lines and line length (see Table D-3) and the results are shown in Figure D-7. The OpenGL

---

<sup>2</sup> <http://jogl-demos.dev.java.net/>

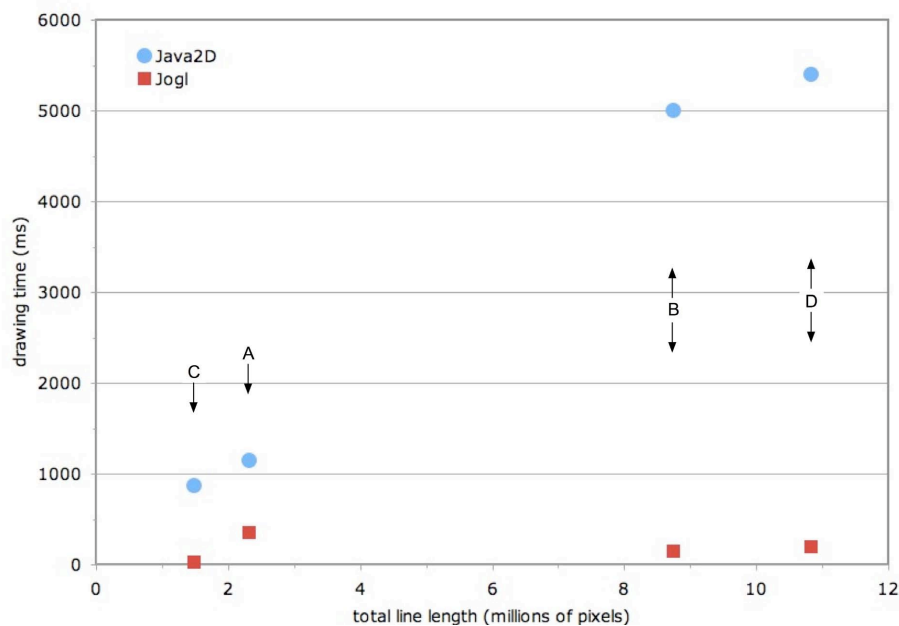
<sup>3</sup> <http://jogl.dev.java.net/>

<sup>4</sup> <http://www.opengl.org/documentation>

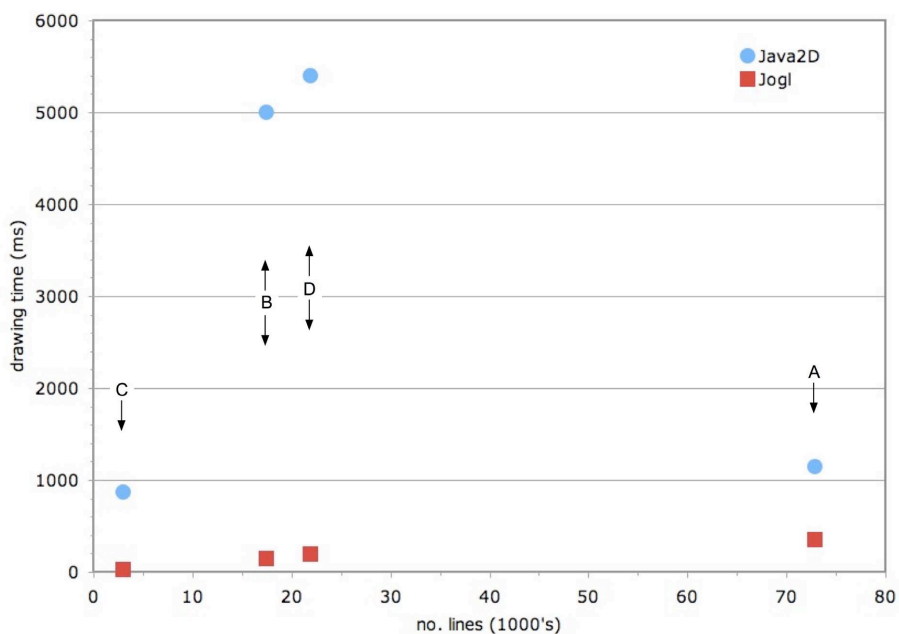
	dataset	records	attributes	no. lines (thousands)	line length (millions pixels)
A	stockmarket	1430	52	72.9	2.3
B	cars 5K	5840	4	17.5	8.7
C	cars 1K	1000	4	3.0	1.5
D	parcels	7321	4	21.9	10.8

**Table D-3** Data sets used to compare the performance of Java2D and OpenGL versions of parallel coordinates

**Figure D-7a**



**Figure D-7b**



Comparison of the drawing time for Java2D and JOGL (OpenGL) version of a simple parallel coordinate applications. A,B,C & D indicate the dataset as given in Table D-3.

version proved to be 30-40 times faster at drawing the parallel coordinate plots. The actual graphic processing is extremely fast as 70-80% of the total time goes in generating the data for the graphic card. It is apparent that the Java2D performance tends to be proportional to the total length of the lines drawn as demonstrated by Figure D-7a. Note that in the accompanying Figure D-7b, dataset A with nearly 73,000 short lines is drawn almost five times as fast as a dataset with a quarter the number of lines, due to the greater total line length.

Looking more closely at the performance of the OpenGL version, Figure D-8b shows that, in contrast to Java2D, the drawing time is more dependent on the number of lines drawn than the line length. The figure also displays the results of using an OpenGL display list, which maintains a lists of drawing commands. As we can see, redrawing from a display list is about three to four times faster as the raw data need not be processed into appropriate OpenGL commands. Display lists are also useful for pre-processing states of the display and are utilised in implementing a Reality Check fade, discussed in Section 4.4.4. Another advantage of OpenGL is its display model which implies that transformations can easily be applied to say re-scale the display area without having to re-plot the data.

### **D.3.2. OpenGL version of the Sampling Lens**

Having proved that using the graphic hardware directly by way of OpenGL can substantially reduce the drawing time and consequently improve the interaction with larger datasets, the Sampling Lens application was modified.

A Jogl drawing panel or canvas can usefully be incorporated into a Java frame and hence the InfoVis Toolkit control panels can still function as before. Jogl has a heavyweight AWT panel, GLCanvas and a lightweight version, GLJPanel. The latter is fully Swing compatible so is advantageous when mixing OpenGL and Swing components, however tests showed that the GLPanel was hardly faster than the Java AWT and therefore a GLCanvas was used. However, discovering where to create the canvas and place the GLEventListener took longer to resolve. Although the InfoVis Toolkit is flexible in developing may different visualisations, the way it builds an application on the fly is complex.

All Java2D drawing primitives were converted to OpenGL commands and to aid debugging, methods which involved drawing, returned display lists which could then be loaded and run in the appropriate order. Various problem were encountered, notably drawing the lens outline, trapping mouse events, converting to the OpenGL colour scheme and viewport scaling inaccuracies. Drawing a thick edged circle was not straightforward as basic graphic object such as rectangles and circles were not

Figure D-8a

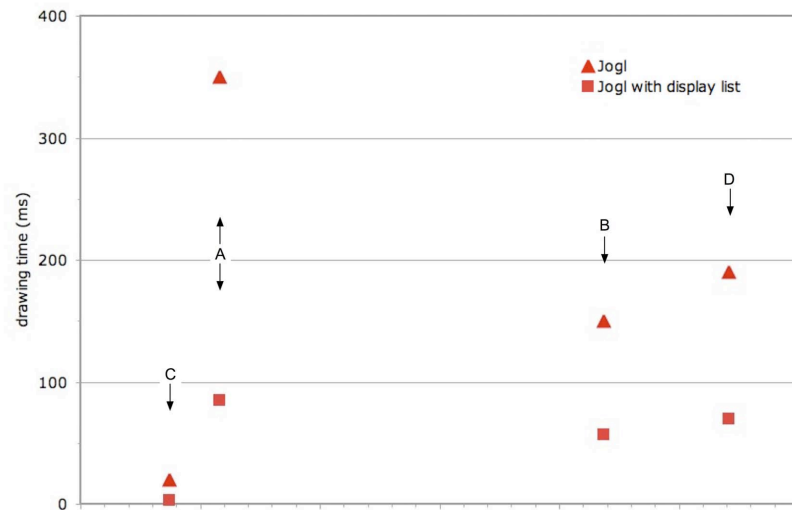
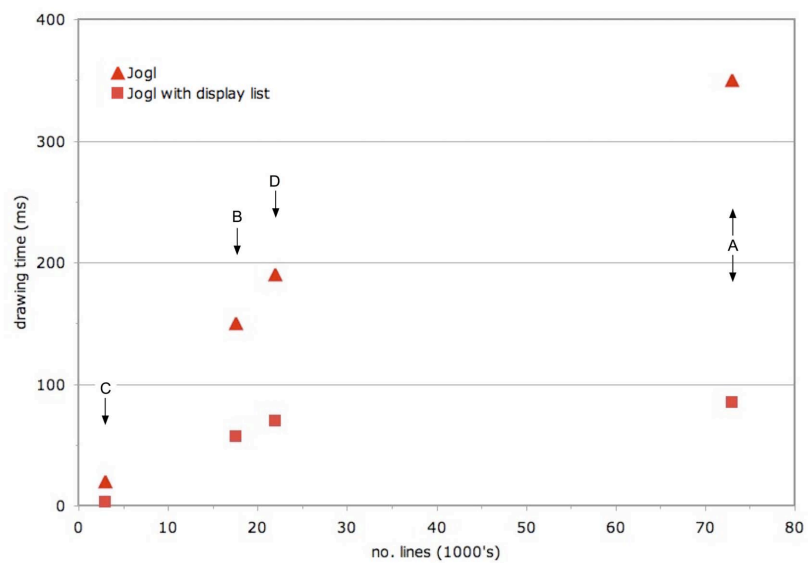


Figure D-8b



Comparison of the drawing time for the JOGL (OpenGL) version of a simple parallel coordinate applications with and without the use of a display list. A,B,C & D indicate the dataset as given in Table D-3.

dataset	lines (thousands)	OpenGL create display list + draw (msec)	OpenGL draw from display list (msec)	Java2D draw (msec)
cars 1K	3	17	4	545
cars 5K	17.5	96	21	2860
synthetic 10K	30	196	57	5090
synthetic 20K	60	446	180	
synthetic 30K	90	692	277	

**Table D-4** Comparison of OpenGL and Java2D for a variety of dataset. Times for different calculation and graphic operations are given in msec. All tests performed on a 640x725 pixel canvas using a PowerBook G4 (867Mz)

## Appendix D Implementation Issues

available, even in the GLU (OpenGL Utility Library)<sup>5</sup>.

The improved performance of the Sampling Lens agreed with the OpenGL prototype. Results for tests using datasets up to 30,000 records (90,000 lines) are given in Table D-4. This shows that the synthetic 10K dataset (30,000 lines) can be drawn within 200msecs which would be acceptable when the user is changing the overall sampling rate.

The times to draw the background from the display list are presented, however redrawing the GLCanvas from a screen save buffer (using OpenGL `glDrawPixels`) was timed at 12msec and is therefore faster than all but the smallest dataset. The actual saving of the GLCanvas was timed at 60msec (`glReadPixels` with `ByteBuffer`) although this only needs to be performed once when the user has finished changing the overall sampling rate.

Drawing the lens is considerably faster than drawing the full set of background lines. For the synthetic 10K dataset this is approximately 30msec for an unsampled lens. Therefore with the addition of 12msec for redrawing the background, movement is smooth which again improves the interaction considerably.

The significant improvement in drawing performance with the OpenGL version of the Sampling Lens implies that much larger datasets can be used, which further increases the benefit of the sampling technique in exploring cluttered displays. In addition, the increase in responsiveness meant that a range of other ideas could be tried out, such as different types of lenses and Reality Check transitions, and these are discussed in the next section. An OpenGL version of the scatterplot visualisation was also implemented, which again increased the responsiveness noticeably and permitted larger datasets to be explored. The Astral Visualiser zooming scatterplot was also implemented in OpenGL.

---

<sup>5</sup> The successful solution involved a *quadric disk*

<b>Technique</b>	<b>Materials needed</b>	<b>Structural properties</b>	<b>Thermal properties</b>
<b>Straw bale</b>	Straw bales, sticks or bamboo for pinning; baling twine or wire	Load-bearing or infill; good earthquake resistance	Excellent insulation; low thermal mass
<b>Cob</b>	Clay soil, sand, straw	Load-bearing or infill; moderate earthquake resistance	Good thermal mass; low insulation
<b>Adobe</b>	Clay soil, sand, straw or other fiber	Load-bearing or infill; poor earthquake resistance	Good thermal mass; low insulation
<b>Rammed earth</b>	Clay soil with high content of sand; often stabilized with cement & reinforced with steel	Load-bearing; good earthquake resistance	Very good thermal mass; low insulation
<b>Earthbags</b>	Woven polypropylene feed sacks filled with clay soil, sandy soil, sand, or gravel	Load-bearing; foundations for other wall systems; good earthquake resistance	Very good thermal mass; low insulation (unless filled with a light fill like pumice or scoria)
<b>Stone</b>	Stones; may be dry-stacked or mortared with a mixture of sand, cement, lime and/or clay	Load-bearing; foundations; poor earthquake resistance unless reinforced	Very good thermal mass; very low insulation
<b>Straw light-clay or slipstraw</b>	Straw (or wood chips, hemp hurds, or other suitable material); clay slip	Infill	Insulation, thermal mass vary with mix; insulation can be high per thickness

## Appendix E

### Comparison of Natural Building Techniques

The Natural Building Network. Comparison of Natural Building Techniques : A Partial List of Wall Systems by Michael G. Smith. [http://www.naturalbuildingnetwork.org/compare\\_techniques.htm](http://www.naturalbuildingnetwork.org/compare_techniques.htm) accessed: July 2008

<b>Best applications</b>	<b>Advantages</b>	<b>Disadvantages</b>
Exterior walls in most climates; quick, temporary structures	Goes up relatively quickly. Fairly easy to permit. Increasing pool of experienced designers/builders.	Very susceptible to moisture damage; bales must be stored carefully; unfinished building must be protected from rain.
Exterior walls in moderate or sunny climates; interior walls; ovens and hearths; benches; garden walls; greenhouses; floors	Highly sculptural; enormous design flexibility. Combines well with other materials.	Labor intensive; goes up slowly, especially in cool, moist conditions. Permitting may take persistence.
Exterior walls in moderate or sunny climates; interior walls; ovens and hearths; domes and vaults in dry, non-earthquake regions.	Adobe blocks can be made in one place and transported. When blocks are made, wall goes up fast. Many pros in SW.	Making and storing adobe blocks takes a lot of space and dry weather. Prone to earthquake damage.
Exterior walls in moderate or sunny climates; benches and garden walls.	Contractors, engineers and permits available in CA and elsewhere.	Very labor or machine intensive. Requires forms. Professionally built RE can be expensive and uses non-natural materials
Foundations for cob, straw bale, etc.; exterior walls in moderate or sunny climates; benches and garden walls; domes and vaults in dry, non-earthquake regions.	Relatively quick earth building technique. Allows for use of wide range of fill materials.	Poly bags very susceptible to UV damage; must be protected from direct sun; long-term durability unknown.
Foundations, basements, retaining walls; fireplaces and hearths; floors and patios; exterior walls in non-earthquake regions with mild climates.	Very durable, even in wet conditions and in contact with ground.	Very labor intensive.
Remodels; exterior and interior walls in many climates.	Walls can be any thickness. Combines well with standard stud framing or timber framing.	Requires forms, so walls generally straight. Prone to water damage. Wood required for frame and forms.

<b>Technique</b>	<b>Materials needed</b>	<b>Structural properties</b>	<b>Thermal properties</b>
<b>Wattle and daub</b>	Straight, flexible sticks (or bamboo); clay soil; chopped straw and/or manure	Infill	Poor insulation; low thermal mass (thick plaster increases mass)
<b>Clay wattle</b>	Clay soil; long straw; sticks	Infill	Poor insulation; low thermal mass unless wall is quite thick
<b>Cordwood masonry</b>	Wood cut into short lengths; mortar may include cement, lime, clay, sand, sawdust; lime/sawdust insulation	Infill or load-bearing (round structures only); poor earthquake resistance	Good insulation; moderate thermal mass
<b>Papercrete, Fibrous cement Fidobe or Hybrid</b>	Recycled paper pulp; sand; cement or clay soil	Infill or load-bearing	Good insulation; thermal mass varies with mix



<b>Best applications</b>	<b>Advantages</b>	<b>Disadvantages</b>
Interior walls; unheated structures such as outdoor showers, sheds, etc.; exterior walls in hot tropics	Walls can be very thin. Uncovered wattle is very decorative.	Requires lots of straight flexible sticks which can be difficult to find. Labor intensive. Prone to water damage.
Interior walls; unheated structures such as outdoor showers, sheds, etc.; exterior walls in hot tropics	Walls can be very thin, curved and sculptural.	New, little-known technique. Requires long straw. Prone to water damage.
Exterior and interior walls	Decorative. Easy to attach wooden framing and furniture.	Wood must be very dry. Tendency for wood to expand and contract, cracking mortar and creating drafts.
Exterior and interior walls; floors; plasters	Very versatile techniques. Walls easily modified.	New technique. Requires a specialized mixer. Questionable water resistance.



# **Appendix F**

## **An Explorative Analysis of User Evaluation Studies in Information Visualisation**

Ellis, G.P., Dix, A. Proceedings 2006 Conference on Beyond Time and Errors: Novel Evaluation Methods for Information Visualization (BELIV'06), Venice, Italy, May 2006, ACM Press.



# An Explorative Analysis of User Evaluation Studies in Information Visualisation

Geoffrey Ellis  
Computing Department  
University of Lancaster  
Lancaster, LA1 4YW, UK  
+44 (0)1524 510340  
g.ellis@comp.lancs.ac.uk

<http://www.hcibook.com/alan/papers/beliv06-evaluation/>

Alan Dix  
Computing Department  
University of Lancaster  
Lancaster, LA1 4YW, UK  
+44 (0)1524 510319  
alan@hcibook.com

## ABSTRACT

This paper presents an analysis of user studies from a review of papers describing new visualisation applications and uses these to highlight various issues related to the evaluation of visualisations. We first consider some of the reasons why the process of evaluating visualisations is so difficult. We then dissect the problem by discussing the importance of recognising the nature of experimental design, datasets and participants as well as the statistical analysis of results. We propose explorative evaluation as a method of discovering new things about visualisation techniques, which may give us a better understanding of the mechanisms of visualisations. Finally we give some practical guidance on how to do evaluation correctly.

## Keywords

Explorative evaluation, Information visualisation, Evaluation, Case study

## 1. INTRODUCTION

How often do we come across a paper describing a new visualisation technique and the future work section at the end states “we intend to undertake a thorough user evaluation” or words to that effect? This is certainly what one of the authors found whilst undertaking a survey of papers in his collection mostly concerned with reducing display clutter in some way. One aim of the survey was to learn from other user evaluation studies to find out about types of participants, experimental details and datasets. He discovered that out of 65 papers describing new visualisation application or techniques, 11 did indeed state that a user evaluation was part of the future work. However a more surprising finding was the fact only 12 out of the 65 papers described any evaluation at all.

So the first question is why do less than 20% of the authors in this

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.

BELIV 2006 Venice, Italy.

Copyright 2006 ACM 1-59593-562-2/06/05 ...\$5.00.

literature sample report user evaluations and over 60% do not even think it is worth mentioning? The second question arising from our review of the user studies presented in these 12 papers addresses the effectiveness of the evaluation. Two of the experiments appear to be flawed; 5 seem to be problematic but at least they gave some useful results; one was an informal study with a single user but was interesting and the result of two were possibly a foregone conclusion. We are therefore left with just two user studies that we considered to be successful!

There are a number of interesting papers in the literature that highlight some of the problems of evaluating visualisation techniques [7,14,15]. We have experienced some of these difficulties ourselves (e.g. finding suitable datasets and participants), but it is hard to believe that there are only 2 user studies from the original set of 65 papers that seem to be particularly useful.

This said, we should note that in the 5 papers that the authors have published together on aspects of visualisation, only 1 includes any user evaluation and that would be classed as unacceptable by the criteria of this paper. So, lest our critique seem over harsh, we include ourselves in it! Note too that where we refer to specific papers below this is not to say that they are particularly bad (often the opposite), but that they illustrate more general issues.

The papers that did not report any user evaluation are not considered here. However, we should point out that most of these do attempt to justify the significance of the application by means of examples, arguing that their particular technique has advantages over existing methods, although direct comparison is not common. Some papers do report empirical studies based on simulations but this is primarily to demonstrate efficiency.

Section 2 looks at some of the aforementioned user studies, highlighting some apparent problems and less frequent successes. Section 3 makes suggestions on why it may be particularly difficult to evaluate visualisations, considering their complexity, dataset, measurement and analysis. Section 4 looks at some of the broader issues of evaluating visualisation and proposes that regarding evaluation as explorative is often a more appropriate standpoint. Finally, in section 5 we offer some more practical advice on how to avoid some of the pitfalls and perform useful and successful evaluations.

# An Explorative Analysis of User Evaluation Studies in Information Visualisation

In summary we find that (a) 'evaluation sucks' [8] (b) it sucks because it is hard and (c) but if you think about it differently it may not be so bad after all and so (d) you can actual do it right.

## 2. CASE STUDIES

This section highlights some possible problems and successes from the user evaluation studies described in the collection of information visualisation papers of one of the authors. Note that this is not intended to be a comprehensive review of the literature as it focussed on a relatively small collection of papers with some relevance to clutter reduction.

### 2.1 Studies with foregone conclusions?

Some user evaluation studies include experiments, which generate results that are possibly a foregone conclusion. For instance, in a study [16], the distortion of edge lines in a dense graph layout is being examined to help users understand which nodes are connected to which lines. One method gently bends the lines, so they separate, but the curve draws the eye towards the end nodes. The other method, also using a lens metaphor, gives a circular distortion, so the end of the distorted line is pointing anywhere but towards the end node! Even without a study, the best method is rather obvious. As is often the case, many participants were involved and a fair amount of data gathering and analysis was performed to 'show' that one method was better than the other. However, the definitive answer to the study lies in the users' comments of the circular distortion method – "edges bend a weird way", "awkward and not useful" and "I don't like this"! Users of the other method instead made comments such as "works great" and "identifies routes very well".

Another example [1] of what might be deemed as unnecessary was an investigation where users were asked to click on a particular number (actually a rectangular box containing the number), when the numbers in the range of 1 to 99 were displayed in a window on the screen. The experiment arranged the numbers in three different ways: un-ordered, partially ordered, and ordered left to right, top to bottom. It is not surprising that the latter task was completed in the least time, while the first took a lot longer. To be fair, the experiment was comparing the users' results against a calculated metric, but was the experiment noted above really necessary?

### 2.2 Wrong sort of experiment?

Other types of user studies that came up in our literature sample appear to be inappropriate in that, instead of evaluating the visualisation per se, they tend to test something else. Some authors have commented that performing a full user evaluation (what ever that is!) is beyond the scope of the project and hence a small trial and/or testing a minor part of the system is all that can be done at that stage. However, it might be the feeling that some reviewers (and conferences) like to see the inclusion of an evaluation, whether or not this is useful, and that drives some of us to include user studies!

For example, one experiment [6] was carried out to discover if users could find information on a map quicker by utilizing popup labels or by zooming in to read the labels. The popup labels were found to be faster despite ignoring the time to zoom in the map. However, comments from the participants revealed that the zooming was often disorientating, so one could argue that it was in fact a focus+context problem that was being tested. If the users of the 'zoom' interface had been given a magnifying lens (electronic version!) that enabled them to read the small text in a

localized area, would the results then have been the other way round? In addition, activating the popup labels required far greater precision in manipulating the mouse pointer than the zoom interface, so was this more a dexterity test? We are not saying that the labelling technique is poor; on the contrary, it appears to be a very good solution for presenting specific information in a crowded space, but the problem lies with an experimental design that is not giving much insight into the benefits of the new technique.

In another experiment [17] users were asked to find patterns using two different types of parallel coordinate plots. Patterns were loosely defined as either clusters or outliers. In the standard version, participants found 8 or 9 out of the 25 patterns that were deemed by an expert to be significant, whereas with the enhanced version users discovered about 16 patterns – a sizeable increase. Should we not question which of these patterns are the important ones? It is certainly useful to discover more relationships in the data, but if we are missing the vital one, then is quantity that important?

Finally, in a study [11] involving a new interface showing web search results, participants spent at least 16 hours on 3 different interface configurations. One relatively small change to the standard interface reduced the query search time by 25%, but this was probably expected based on their previous work. In the third configuration, 7 parameters were changed and sorting out the effects of each of these would be practically impossible. In terms of the timing data, there was little difference between the standard interface and the third configuration, and based on users' preference, its likeability was about the same. Yet, looking at the individual responses, it is evident that some users really liked the third configuration whilst others hated it – a clear example of the dangers of averaging results! Users' comments also revealed that increasing the size of the text was probably the most significant benefit of the third version, something that an analysis of the data would never have uncovered. So, were these experiments really necessary? It could be argued that, observing a few users in an informal evaluation and recording their comments could well have provided as much understanding of the new technique.

### 2.3 Fishing for results?

In one of the studies we looked at [12], the visual interface application was designed to help users browse and understand large document collections. The first experiment compared the effectiveness of the visual interface with a text-based search interface and found it to be not so effective. The authors expected this as the application was designed primarily for browsing. A second experiment was conducted to assess whether the visual interface gave the users a better understanding of the structure of the document collection. Users were asked to draw a tree structure to represent the topics in the collection. These were analysed and as one would expect, the visual interface user group produced diagrams that were more similar to each other than the text searchers. This was used as evidence to show that the visual interface was better in producing a more coherent view of a large document collection. As the authors pointed out, it is difficult to assess how much knowledge a user has gained from a browsing activity. However, presenting one group of users with what amounts to a picture to copy, and using the fact that they mostly ended up with similar pictures to infer that the application is effective in this context is like fishing for results!

## 2.4 What makes a good study?

As mentioned in the introduction, 2 out of 12 studies in the literature set appear to be successful – in that they effectively demonstrate the potential benefits of some application through user evaluation.

For instance, in an evaluation of an interface showing web search results [4], the authors ran subsequent tests by changing an appropriate parameter each time in order to tease out the reasons behind the test results. This method of deciding what to investigate next, based on previous results, seems to be a good approach and probably has a greater chance of understanding the interaction mechanisms than ‘let’s do lots of test runs with any number of independent variables and hope to sort it out in the end’. In order to adopt this iterative approach, one clearly needs participants who can return for a series of experiments, thus increasing the overheads, but then fewer users may be required. The other significant feature of this study is that the authors attempt to generalize their results, something that is sadly lacking in some of the papers.

Another noteworthy example [13] is the evaluation of a new visualisation technique that did not time anything – not a stopwatch or timing device in sight! It involved getting a large group of ‘real’ users from a wide range of jobs but they all had a good knowledge of the domain and the data set. After the prototype was demonstrated to a small group of potential users, they were given the opportunity to use the interface with some typical data and were then asked to comment on this and come up with potential advantages and problems. A wealth of useful data was collected and as others have reported [7], domain experts are often worth their weight in gold.

## 3. PROBLEMS OF EVALUATION

Why is it apparently so hard to evaluate visualisations effectively? In fact there are many reasons, some shared with general user interface evaluation, and some more particular to visualisation.

### 3.1 Complexity

The visualisation process consists of many complex interactions and is thus difficult to be treated as a whole. However, we can attempt to understand the mechanisms that drive it.

#### 3.1.1 Interpretation and credit assignment

In common with other user interfaces, visualisations typically embody many assumptions and theoretical views. Carroll and Rosson's *claim's analysis* [2] seeks to expose the many factors affecting the usability of interfaces, e.g consistency in the layout of navigation buttons or the colour of highlighted data items. These often interlocking ‘claims’ are implicit in software, but this is not a way of thinking that is common amongst those producing novel visualisation techniques. Even, if one has managed to articulate the multiple claims embodied in a visualisation, simple end-to-end timing measures or user satisfaction score gives little indication of which of these have been important in the success (or otherwise!) of a technique.

#### 3.1.2 Mechanism

Even a simple interaction with a visualisation will include multiple stages and steps at both a coarse level (e.g. getting to know a data set and then finding items in the data set) and at a fine level (e.g. visually scanning for some feature, then moving the mouse and selecting a node, then evaluating pop-out detail).

Again end-to-end measures are not the most helpful in working out which steps are making a difference. For example, an experiment might find no difference between the performances of two visualisations, but the task involved included both ‘getting to know’ a data set and finding specific features. It may be that one technique is in fact better at the former and the other at the latter, if one realises this it might suggest ways of creating a hybrid technique, but without this both just appear the same.

Even worse, the aspects of the interaction that cause a difference may be completely irrelevant to the essential qualities of a visualisation technique. In one of the papers we studied, after pages of timing and accuracy data, some users were quoted as saying that one visualisation was preferred to the other because the font was bigger and it was easier to read. While this was in some part related to the nature of the visualisation (a form of fish-eye), it is likely that some small ‘fix’ may well have been able to avoid this problem with the second visualisation – were the differences seen due to a simple detail of the implementation?

### 3.2 Diversity

Another reason that makes evaluation of visualisations harder lies in the diversity of tasks, data sets and participants.

#### 3.2.1 Variety of data sets

Different visualisations deal with different kinds of data. While there has been some attempt to create a standard, (e.g. the FADIVA network a few years ago), we still do not have well-developed and easily available standard datasets in the way that the information retrieval community do (e.g. TREC). This means that visualisation evaluation (and even simple demo-ing), is limited by the availability of data, or compromised by inappropriate, or artificially generated data.

#### 3.2.2 Indeterminacy of tasks

Different tasks are better supported by different visualisations. In a recent evaluation, standard outliner-style TreeViews were compared with PieTrees [10], which have a constant value–area mapping similar to TreeMaps. Not surprisingly, tasks such as ‘find the biggest’ were fastest using the PieTree whereas finding a specific named node, where the area mapping did not act as a heuristic, was fastest with the TreeView.

It would be very easy to have chosen a task that had made one or other look better and think this was definitive – indeed, it is natural to choose tasks (and datasets) that suit a novel technique i.e. ones that it is good at. Furthermore, as a researcher, there is a temptation to deliberately choose the tasks that make one’s method look good – referees are often unforgiving of a truthful paper that says a technique has strengths and weaknesses.

Not only do tasks differ, but the real tasks we want are usually open ended. If the user knows beforehand what is important to see in the visualisation, then there are typically better ways of looking for it: aggregates, searches etc. Visualisations are often at their best for more exploratory tasks, but these are precisely the tasks that are hardest to replicate in an experiment.

#### 3.2.3 Individuality of people

Students may be useful but ... The majority of the studies used students, often computing students as their subjects. Clearly, students are convenient. They are nearby and can be persuaded to give up a few hours either because we convince them that they are doing something worthwhile or there is some monetary incentive.

## An Explorative Analysis of User Evaluation Studies in Information Visualisation

In some cases this is fine, for example, interaction or perception experiments (e.g. check colour or size of objects that are to be selected or manipulated in some way) require little knowledge of the visualisation domain, hence students would be suitable. However, there is a large amount of literature dealing with cognitive issues, which may well guide the designer.

But where a more realistic task (i.e. where the task matches the application domain) is used in the evaluation, participants need a clear understanding of the problem that the visualisation tool is attempting to solve and also, one might argue, an understanding of the data itself. In such cases, the chances of assessing the usefulness of the tool using students will be slim, as we found during the informal testing of our Sampling Lens [5]. Users liked the lens-based tool as it revealed patterns within a parallel coordinate plot in areas where, in the absence of the lens, there were too many overlapping lines. However, when we asked the users what the patterns meant, they did not really understand it; they just thought it was cool!

Other researchers [7,15] have suggested that better information can be obtained by using either a small number of domain experts involved in more qualitative studies, expert visual designers or HCI expert reviewers. Of course, it is more difficult to get access to such a group of people.

The acceptability of a particular kind of experimental subject is dependent on the exact details of the experiment or system. However, we ought to consider these: (i) it is always important to explicitly consider the potential effect of the type of participant on the interpretation of the results; and (ii) understanding the mechanism is again essential so that by considering the details of the interaction one can determine aspects that are capable of evaluation by non experts.

Recognising individuality is also important when analysing results. Different cognitive styles may lead to a particular technique working well for one group and not for another – the overall averages may hide this and simply appear inconclusive. Finding who a technique is really useful for may be more important than making it work pretty well for everyone.

### 3.3 Measurement

Issues of accuracy, precision and significance of statistical data are fundamental when discussing the relevance of experimental results.

#### 3.3.1 125.2 seconds to do what?

Studies often present end-to-end times to do a certain task, and sometimes the average of a set of tasks. What does this tell us about the interaction and more importantly, the understanding of the user? In some cases, time may be of the essence, for example in an in-car information display, but more often 'time' seems to be an easily measurable proxy for 'ease of use' ... and not necessarily the most accurate!

Numbers are powerful when we understand what they mean, but they can also be misleading. In particular, it is often easy to let precision fool one into an impression of accuracy. In many of the experimental results we see the time to do a task or set of tasks given to the nearest tenth of a second. e.g. 125.2. While this may be a true representation of the measured value, if the level of variation is  $\pm 17$  seconds, then would 120 seconds be good enough ... or even, "task completed fairly quickly"?

Advertisers deliberately use apparently precise numbers as a way to suggest validity: "most cats prefer Fishkers" sounds

unconvincing, "applying face cream with RexionolicB++ reduces wrinkles by 37%" has an aura of scientific truth. Whilst academics are not deliberately attempting to deceive their readers, they are perhaps often accidentally deceiving themselves.

Of course if you do not quote exact numbers, it is impossible for a reader to, for example, check your statistics. However, it is possible to be both precise with data and accurate in rhetoric, for example, giving precise numbers in tables, but using the most appropriate number language in the text. Also, where numbers of subjects or trials are small, it is often better to quote the exact numbers (e.g. 7 out of 9 subjects) rather than converting this into an apparently over-precise percentage (52.9%).

#### 3.3.2 Statistics: significance and importance

People find statistics difficult. There are various reasons for this, some to do with education and some to do with the mix of mathematics and real-world understanding. For those with computing background and unlikely to have been exposed to statistics at undergraduate level, this is particularly hard ... and it is not surprising that this is reflected in published work.

One problem lies in the particular meaning of the word "significance" in statistics. It seems that when we have collected our data for a range of dependent and independent variables, we put this into a statistics package and then quote its significance, but we often do not stop to think what it means. If  $p < 0.05$ , we think "yes done it", but in a highly accurate experiment totally unimportant differences may show up as significant – yes visualisation X is faster than visualisation Y ... but only by 3 milliseconds! What (and all) a significance test is saying is that with  $p < 0.05$  the chance of the observed data being a random occurrence is 1 in 20.

Even more worrying is the treatment of non-significant results. Often a graph, which as a visualisation community we know is hard to ignore, appears to show a difference, but the text says this is not significant or 'marginally significant' (whatever that might mean!). In other words, the graph we are seeing could just as well be the effects of random chance ... like a good day at the races, but by being presented to us we are being tempted to believe otherwise ... the advertisers would love us!

Often non-significant results are (erroneously) treated as meaning "no difference". This is a misapprehension that every statistics course highlights, but it is still endemic in the literature. Whilst "not significant" does not mean "no effect", in many cases, a confidence interval can allow you to say "unimportant difference". Sadly, although confidence intervals are not difficult to compute or understand, they seem to be where most statistics courses give out. Furthermore, readers are less familiar with confidence intervals and so explanation is often needed. Indeed one of the authors once received a referee's comment saying he should use proper statistics terminology "significance of  $p < x$ " rather than "confidence" ... not only having limited understanding, but confident enough in his/her statistical ignorance to critique! Again it is not enough to do evaluation correctly, but also reviewers need to be educated to appreciate it.

#### 3.3.3 Points of comparison and control conditions

Any numerical or ordinal measure requires some gold standard, point of comparison or control in order to know what values are good. Choosing a suitable 'control' can be problematic. For example, the paper that introduced the Xerox Butterfly Browser, a 3D visualisation for following references and citations, included an empirical evaluation – against Dialog [9]. Whilst Dialog was



in a sense 'industry standard', it was effectively 1960s technology designed to work over very low bandwidth (10 cps.) phone lines. Perhaps it was not surprising that the users preferred the 3D interactive interface to the command line search, but this hardly tells us much about the new visualisation.

This may seem as an extreme example, but the problem is not so much whether the evaluation in that paper was effective or not, but instead, to determine what a valid point of comparison would have been. Any other 'state of the art' comparison is bound to differ in many respects, causing the credit assignment problem noted above. A suitable alternative would be to change some specific feature and measure the effect of that alteration – more like a traditional psychological experiment. However, if there are any interactions between features (and this is usually the case), then in principle one has to test all possible interactions, which is combinatorially impossible.

**4. THE BIG ISSUES**

Many of the problems discussed in the previous section can be reduced to two issues, the generative nature of visualisation techniques and the lack of clarity over the purpose of evaluation.

**4.1 Evaluating generative artefacts**

Visualisations (like all interfaces) are generative artefacts: that is they are things that are not something of value in and of themselves, but only yield results in some context. In the case of a piece of visualisation software, this is when used by a particular type of user to visualise a particular data set for a particular purpose. To further complicate things, the visualisation software is itself typically an implementation of some more generic visualisation technique. But all we can evaluate is the success of the particular instance. In order to really produce a reliable empirical evaluation of a visualisation technique one would need to have many tasks, many data sets, many users and many implementations of the technique produced by many different designers ... hardly likely in a finite time!

In search for the validation of generative artefacts ...

**empirical evaluation of generative artefacts is methodologically unsound**

... or put in other words, you cannot evaluate a visualisation ... or at least any evaluation cannot tell you, *in itself*, that the visualisation works or doesn't work.

However, whilst you cannot 'prove' a visualisation is good or correct through evaluation; you can perform useful evaluations, and may be able to *validate* the visualisation in other ways.

In some domains, particularly mathematics, it is rare to attempt to perform post hoc evaluation. Instead it is the proof, the process of

reasoning from initial knowledge (or assumptions) through lemmas to theorems that give you confidence in the answer. Sometimes you may put example numbers through a theorem – but this is largely to check for 'silly' mistakes in the proof process, not to prove the theorem through the examples. However, in other domains it is hard to work beyond some point through reasoning. For example, in areas of chemistry, one can deduce that certain classes of compound are likely to have an effect, but the precise form and level of that effect may only be found by exhaustive trial of many potential compounds.

In visualisation we can never have perfect evaluation because of the generative nature of the artefacts we build. Likewise we cannot have perfect justifications because our base knowledge of human perception and cognition is incomplete, and because our ability to reason from these to their implications is flawed. However, if empirical evaluation complements reasoned justification then it can lead to a reliable and strong validation of the visualisation.

Our justification may include (see Fig. 1):

- existing published results of experiments and analysis
- our own empirical data from experiments, studies, etc.
- expert opinion (published or otherwise) and common sense
- arguments based on the above

On the evaluation side we may use:

- empirical evaluation, user studies, timing data, etc.
- peer reviews of our work (other people agree it is a good idea)
- comparison with previous work (do the parts that should behave the same actually do so)

If one is aware of the weaknesses and gaps in the justification, then an evaluation and subsequent analysis can be tuned to verify the questionable aspects of the justification.

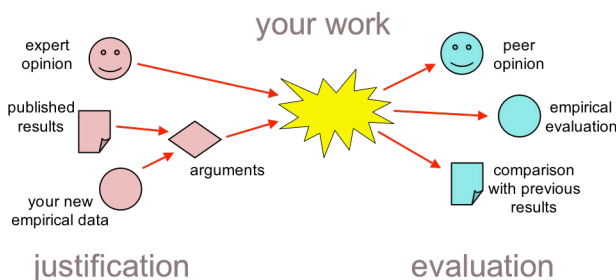
Sadly again, for the researcher, there is a tension between good science and publishability. If you choose to evaluate aspects that are questionable then one is more likely to find problems in the visualisation or to have inconclusive answers. In contrast, if you evaluate the aspects that you are pretty sure are okay from the justification argument, then you are likely to get clear results with nice  $p < 0.05$  significance results ... but learn nothing.

**4.2 Purposes of evaluation: summative, formative and explorative**

The other cross cutting issue is the need to have a clear idea of the purpose. The distinction between summative and formative evaluation is well known, although for usability, the techniques are very similar and hence it is easy to blur the two.

For example, head-to-head comparisons of techniques for dealing with similar data are really attempting a form of *summative* evaluation: "my visualisation is better than yours". Now this is good marketing, but not very useful science, or even design. In fact, when one looks at the discussion of such evaluations those doing it more often than not end up with some level of suggestions for improving their visualisation. In fact, the actual use of the evaluation is *formative*.

This confusion of purpose is also evident in papers that effectively give a record of an iterative interface development process. If this is an iteration of fundamental novel visualisation concepts and techniques, this is good use of evaluation in research, but if (as is often the case in iterative development) it is merely tweaking features that increase usability, but do not hit the heart of the



**Fig1: The two sides of validation: justification and evaluation**

## An Explorative Analysis of User Evaluation Studies in Information Visualisation

novelty (e.g. font size), it is good product development, but not good research. Of course, such development is often needed to get a basic concept into a usable enough package to evaluate ... but that is the forerunner to the evaluation, not the evaluation itself.

So for both summative and formative evaluation, we need to be constantly careful that they are what we really want and that they address the real issues. However, what is really needed in most research contexts is neither. In fact, we require explorative<sup>1</sup> evaluation – evaluations that help us see new things about our ideas and concepts, which are useful to us. Whilst the purpose of summative evaluation is to obtain a seal of approval and the purpose of formative evaluation is to improve a design, the purpose of explorative evaluation is to find out, to provide knowledge. The difference is sometimes just in the way one views results, but can be more fundamental, for example, for explorative purposes one may deliberately use a bad design to uncover user behaviour in extreme circumstances.

As an example of the latter, some years ago one of the authors, working with Stephen Brewster on audio feedback, deliberately created a calculator-style interface that involved large inefficient movements of mouse and eye in order to create mis-clicking errors that only occur infrequently, but problematically, in normal 'good' interfaces. By creating the error we were able to validate our understanding of its causes and thus design appropriate feedback to ameliorate its effects [3]

Indeed, many evaluations that appear weak or problematic, when viewed as summative or formative evaluations, are far more convincing when seen as explorative. For example, the paper [12] cited in section 2.3 seems problematic if seen as the former (a form of fishing), but more appropriate for exploration: "what kind of things is Scatter/Gather good for?"

Because the techniques used for all kinds of evaluation are similar, it is often unclear which kind of evaluation authors intended to undertake. Indeed, Zhai [18] in his response to Lieberman's 'The Tyranny of Evaluation' [8], notes that the value of the best evaluation is often not in the original (summative or formative) purpose, but in accidental understandings and findings – that is the explorative aspects. How much more effective might these evaluations have been if the eventual explorative purpose had been identified explicitly in the first place!

### 4.3 From data to knowledge

To some extent, in even writing (or attending a workshop) on evaluation, we run the danger of subscribing to the phenomenological notion that it is the data (whether qualitative or quantitative) that is in some way the 'real' and 'objective' truth.

In fact it is impossible to generalise data per se; it is always a singular event: whether an ethnography of a particular group at a particular time, or a formal experiment with particular subjects in a particular setting. Any future use of a particular visualisation application, technique or design principle will be different. Knowledge and hence generalisation only comes through the application of reasoning informed by (interpreted) data.

Unfortunately the genre of scientific writing often serves to blur or hide this and many attempts to evaluate by adopting this genre

run the risk of not making the best of their work and at worst misleading their readers.

## 5. DOING IT RIGHT

So effective evaluation of visualisation is hard and fraught with problems, but is also essential in order to rise beyond simply saying "I did this and it's cool". However, bad evaluation is at best useless and at worst can be plain wrong. So, how can we do it right?

**think purpose** – First of all it is important to know what you are hoping to gain from the evaluation. If your aim is to prove that your system is best, go get a job as an advertising executive. If your aim is simply to make your system as good as possible, then sell your product but don't write about its development. If your aim is to make your product as good as possible in order to effectively deploy it and so learn, this is essential, but not a thing to report in detail. However, if your aim is to understand whether, when and under what circumstance a technique or design principle works or is useful – yes now you are doing research.

**think measures and tasks** – Is the thing that you are measuring useful (see also below) and if so what is good. It is easy to measure something just because you can. For example, if you are primarily interested in a user engagement with a visualisation then time to complete a fixed task tells you little. However, in an open-ended task, users' time on task (combined with qualitative data) may tell you how much they were enjoying themselves.

**think success** – Ask yourself: "If this evaluation is as successful as it could be, what will I know at the end that I don't know now?" If the answer is "not much" then why do the experiment? As noted in section 4.1, use the weaknesses in your justification to drive your evaluation, make every subject hour count.

**think failure** – In the case of quantitative experiments or questionnaires where you plan statistical analysis ask: "If this evaluation does not give statistically significant results, will I have learnt anything?" In fact, at very least, you will have learnt enough to do a power analysis and calculate how many more subjects you would need in order to either detect an important difference or conclude (using a confidence interval) that any differences are negligible. However, this turns what you hoped to be your full evaluation into merely a pilot. This may be all you can do and you just have to do more work. However, if you also collect rich data (video, keystroke logs, talk-aloud transcripts, post-task interviews) then you are more likely to have something of value to report. This takes us to ...

**think qualitative and quantitative** – If you speak to one person who has a particular behaviour, is it just that person? If a formal experiment or questionnaire shows that 75% of people have a particular behaviour, then it is clearly prevalent, but is it important and why does it occur? However, if you combine the two, you both know *that* the behaviour occurs and have some idea *why*.

**think mechanism** – If you do not understand a process then you cannot generalise. This often involves qualitative data (as above), but may include quantitative data on parts of an interaction, not just end-to-end measurements.

**think understanding** – How can you manipulate the visualisation itself, the data used or the task you give the user in order to find out most about the most interesting things. And yes, as we noted in section 4.2, this may mean using versions of your visualisation that are not the 'best' ones.

<sup>1</sup> Actually the proper word is exploratory, but explorative rhymes with summative and formative ☺

## 6. SUMMARY AND CONCLUSION

An explorative analysis of the experimental details and results questioned the viability of evaluations in cases where the outcome is probably a foregone conclusion, or where inappropriate experiments are perhaps carried out, or even where the results are possibly unconvincing. It was also apparent that, in many of the studies, comments from the users contributed a great deal to understanding the visualisation and reinforces the belief that ethnographic or observational techniques often provide more useful data.

The fundamental reason behind so few user studies may be due to the fact that information visualisations are very difficult to evaluate. The visualisation process is made up of a complex set of interactions and ideally, we should understand the mechanisms inherent in the process in order to assess the viability of an evaluation. End-to-end time measurements are not particularly useful when attempting to work out the critical components of a visualisation.

We showed that the choice of appropriate tasks, datasets and participants is important when determining how to evaluate a particular visualisation. In addition, when reporting results of experiments, we discussed the importance of understanding the meaning of accuracy, precision and significance of the statistical data and we also highlighted the problem of finding valid point of comparison between visualisations.

We put forward the idea that empirical evaluation of visualisations on its own is methodologically unsound due to the generative nature of visualisation techniques. However, if empirical evaluation is used in conjunction with reasoned justification then this may lead to a reliable and strong validation of the visualisation.

We also emphasise the need to apply formative and summative forms of evaluations in the appropriate context; but in many cases neither may be suitable. We therefore propose explorative evaluation as a method for helping us see new things about our ideas and concepts and revealing those that are useful to us.

In order to balance the more critical stand of the earlier parts of the paper we have tried to give some practical guidance on how to do evaluation correctly. We hope this will be valuable for those who are new to this and a reminder for those more experienced.

As we strive for publishability, experimental designs and reporting of results may be unduly influenced by the expectation of reviewers. Hence it is not enough to do evaluation correctly; reviewers also need to be educated to appreciate it!

## REFERENCES

- [1] Bederson, B.B., Shneiderman, B., Wattenberg, M. Ordered and Quantum Treemaps: Making Effective Use of 2D Space to Display Hierarchies. *ACM Transactions on Graphics*, 21(4), Oct 2002, 833-854
- [2] Carroll, J.M, Rosson, M.B. Getting around the task-artifact cycle: how to make claims and design by scenario. *ACM Transactions on Information Systems*. Vol 10 No 2, April 1992, 181-212
- [3] Dix A., Brewster, S. Causing Trouble with Buttons. Ancillary Proc. HCI'94, Glasgow, Scotland, 1994. <http://www.hcibook.com/alan/papers/buttons94/>
- [4] Dumais, S., Cutrell, E., Chen, H. Optimizing Search by Showing Results In Context. *Proc. CHI'01*, 2001, ACM Press, 277-284
- [5] Ellis, G.P., Bertini, E., Dix, A. The Sampling Lens: Making Sense of Saturated Visualisations, *Proc. CHI'05 Extended Abstracts on Human Factors in Computing Systems*, Portland, USA, 2005, ACM Press, 1351-1354
- [6] Fekete, J-D., Plaisant, C. Excentric Labeling: Dynamic Neighbourhood Labeling for Data Visualization. *Proc. CHI'99*, Pittsburgh, 1999, ACM Press, 512-519
- [7] Kosara, R., Healey, C.G., Interrante, V., Laidlaw, D.H., Ware, C. Thoughts on User Studies: Why, How, and When. *Computer Graphics & Applications*, 23(4), July 2003, 20-25
- [8] Lieberman, H. The Tyranny of Evaluation. (accessed 2006). <http://web.media.mit.edu/~lieber/Misc/Tyranny-Evaluation.html>
- [9] Mackinlay, J. D., Rao, R., Card, S. K. An Organic User Interface For Searching Citation Links, *Proc. CHI'95*, Denver, May 1995, ACM Press, 67-73
- [10] O'Donnell, R., Dix, A., Ball, L. Exploring The PieTree for Representing Numerical Hierarchical Data, *Proc. HCI2006*, London, Sept. 2006, Springer
- [11] Paek, T., Dumais, S., Logan, R. WaveLens: A New View onto Internet Search Results. *Proc. CHI'04*, Vienna, Austria, Apr 2004, ACM Press, 727-733
- [12] Pirolli, P., Schank, P., Hearst, M., Diehl, C. Scatter/Gather browsing communicates the topic structure of a very large text collection. *Proc. CHI'96*, Vancouver, May 1996, ACM Press, 213-220
- [13] Plaisant, C., Milash, B., Rose, A., Widoff, S., Shneiderman, B. LifeLines: Visualizing Personal Histories. *Proc. CHI'96*, 1996, ACM Press, 221-227
- [14] Plaisant, C. The Challenge of Information Visualization Evaluation. *Advanced Visual interfaces*, Italy, 2004, ACM Press
- [15] Tory, M., Möller, T. Evaluating Visualizations: Do Expert Reviews Work? *IEEE Computer Graphics and Applications*, 25(5), 2005, 8-11
- [16] Wong, N., Carpendale, S., Greenberg, S. EdgeLens: An Interactive Method for Managing Edge Congestion in Graphs. *IEEE Symposium on Information Visualization*, Oct 2003, 51-58
- [17] Yang, J., Ward, M.O., Rundensteiner, E.A., Huang, S. Interactive hierarchical displays: a general framework for visualization and exploration of large multivariate data sets. *Computers and Graphics*, 27(2), Apr 2003, 265-283
- [18] Zhai, S. Evaluation is the worst form of HCI research except all those other forms that have been tried. (accessed 2006). <http://www.almaden.ibm.com/u/zhai/papers/EvaluationDemocracy.htm>